



SAINS KOMPUTER

TINGKATAN

5





RUKUN NEGARA

Bahawasanya Negara Kita Malaysia

mendukung cita-cita hendak;

Mencapai perpaduan yang lebih erat dalam kalangan
seluruh masyarakatnya;

Memelihara satu cara hidup demokrasi;

Mencipta satu masyarakat yang adil di mana kemakmuran negara
akan dapat dinikmati bersama secara adil dan saksama;

Menjamin satu cara yang liberal terhadap
tradisi-tradisi kebudayaannya yang kaya dan pelbagai corak;

Membina satu masyarakat progresif yang akan menggunakan
sains dan teknologi moden;

MAKA KAMI, rakyat Malaysia,
berikrar akan menumpukan
seluruh tenaga dan usaha kami untuk mencapai cita-cita tersebut
berdasarkan prinsip-prinsip yang berikut:

**KEPERCAYAAN KEPADA TUHAN
KESETIAAN KEPADA RAJA DAN NEGARA
KELUHURAN PERLEMBAGAAN
KEDAULATAN UNDANG-UNDANG
KESOPANAN DAN KESUSILAAN**

(Sumber: Jabatan Penerangan Kementerian Komunikasi dan Multimedia Malaysia)

KURIKULUM STANDARD SEKOLAH MENENGAH

SAINS KOMPUTER

TINGKATAN **5**

PENULIS

Norlini binti Ramli
Paridah binti Daud
Noor Azma binti Ismail
Noor Lees binti Ismail
Abd. Halim bin Bohadean@Bohari

EDITOR

Intan Farhanna binti Uyob

PEREKA BENTUK

Roshidah Mohamad

ILUSTRATOR

Az-zalzalameiah binti Anas



Oxford Fajar Sdn. Bhd. (008974-T)



KEMENTERIAN PENDIDIKAN MALAYSIA

No. Siri Buku: 0121

KPM2017 ISBN 978-983-47-2375-0

Cetakan Pertama 2017

© Kementerian Pendidikan Malaysia

Hak Cipta Terpelihara. Mana-mana bahan dalam buku ini tidak dibenarkan diterbitkan semula, disimpan dalam cara yang boleh dipergunakan lagi, ataupun dipindahkan dalam sebarang bentuk atau cara, baik dengan cara elektronik, mekanik, penggambaran semula mahupun dengan cara perakaman tanpa kebenaran terlebih dahulu daripada Ketua Pengarah Pelajaran Malaysia, Kementerian Pendidikan Malaysia. Perundingan tertakluk kepada perkiraan royalti atau honorarium.

Diterbitkan untuk Kementerian Pendidikan Malaysia oleh:

Oxford Fajar Sdn. Bhd.

4, Jalan Pemaju U1/15, Seksyen U1,

Hicom-Glenmarie Industrial Park,

40150 Shah Alam,

Selangor Darul Ehsan.

Tel: 03-56294000

Faks: 03-56294009

Laman Web: <http://www.oxfordfajar.com.my>

Reka Letak dan Atur Huruf:

Oxford Fajar Sdn. Bhd.

Muka Taip Teks: Helvetica Neue LT Std 47 Light Condensed

Saiz Muka Taip Teks: 11 pt

Dicetak oleh:

Percetakan Rina Sdn. Bhd.,

Lot 45, Persiaran Mewah,

Bandar Tun Razak,

56000 Kuala Lumpur.

Penghargaan

Penerbitan buku teks ini melibatkan kerjasama banyak pihak. Sekalung penghargaan dan terima kasih ditujukan kepada semua pihak yang terlibat:

- Pegawai-pegawai Bahagian Buku Teks dan Bahagian Pembangunan Kurikulum, Kementerian Pendidikan Malaysia.
- Jawatankuasa Penambahbaikan Prof Muka Surat, Bahagian Buku Teks, Kementerian Pendidikan Malaysia.
- Jawatankuasa Penyemakan Pembedulan Prof Muka Surat, Bahagian Buku Teks, Kementerian Pendidikan Malaysia.
- Jawatankuasa Penyemakan Naskah Sedia Kamera, Bahagian Buku Teks, Kementerian Pendidikan Malaysia.
- Pensyarah daripada Universiti Malaya
- Pensyarah daripada Universiti Teknologi Petronas

Kandungan

Halaman Judul	i
Halaman Hak Cipta dan Penghargaan	ii
Pendahuluan	iv

BAB

1

PENINGKOMPUTERAN	1
1.1 Komputer dan Impak	2
Latihan Pengukuhan 1.1	22
1.2 Seni Bina Komputer	24
Latihan Pengukuhan 1.2	38
1.3 Get Logik	39
Latihan Pengukuhan 1.3	53
Imbas Semula	55
Latihan Pengayaan 1	56

BAB

2

PANGKALAN DATA LANJUTAN	59
2.1 Bahasa Pertanyaan Berstruktur: SQL (<i>Structured Query Language</i>)	60
Latihan Pengukuhan 2.1	117
Imbas Semula	119
Latihan Pengayaan 2	120

BAB

3

PENGATURCARAAN BERASASKAN WEB	123
3.1 Bahasa Penskripan Klien	124
Latihan Pengukuhan 3.1	222
3.2 Bahasa Penskripan Pelayan	224
Latihan Pengukuhan 3.2	273
3.3 Laman Web Interaktif	274
Latihan Pengukuhan 3.3	354
Imbas Semula	355
Latihan Pengayaan 3	356

Glosari	359
Indeks	361
Rujukan	362

Pendahuluan

Buku Teks Sains Komputer Tingkatan 5 ditulis berdasarkan Dokumen Standard Kurikulum dan Pentaksiran (DSKP) Sains Komputer Tingkatan 5 terbitan Kementerian Pendidikan Malaysia. Kurikulum Standard Sekolah Menengah (KSSM) menyediakan dan memperkukuh murid dengan pengetahuan, kemahiran dan nilai Sains Komputer serta pemikiran komputasional yang bersesuaian dengan tahap kebolehan murid berdasarkan Standard Pembelajaran dan Standard Prestasi yang terdapat dalam DSKP. KSSM Sains Komputer memberikan fokus dalam pengukuhan pengetahuan, kemahiran dan nilai dalam empat bidang yang utama, iaitu Pengkomputeran, Pangkalan Data, Pengaturcaraan dan Interaksi antara Manusia dengan Komputer. Fokus ini bertujuan untuk melahirkan murid yang mempunyai kemahiran komputasional dan kritikal melalui aspek mereka bentuk, membangunkan dan mencipta aplikasi baharu.

Buku ini menekankan strategi pengajaran dan pembelajaran yang bertujuan untuk menghasilkan pembelajaran yang berkesan. Aktiviti pembelajaran yang pelbagai telah dicadangkan dalam buku ini. Aktiviti ini merupakan asas untuk melahirkan murid yang bertanggungjawab tentang pembelajaran sendiri, memiliki sifat bekerjasama ketika menjalankan aktiviti berkumpulan, mengaplikasikan pembelajaran dalam kehidupan harian, bersifat ingin tahu, mampu menyelesaikan masalah dalam pembelajaran dan menguasai pelajaran yang dipelajari.

Kemahiran Abad ke-21 telah diterapkan dalam Buku Teks Sains Komputer Tingkatan 5 bagi membentuk murid yang berdaya tahan, mahir berkomunikasi, berfikir secara kritikal, kreatif dan inovatif, mempunyai semangat kerja berpasukan, bersifat ingin tahu, berprinsip, bermaklumat, penyayang dan mempunyai semangat patriotik. Profil ini sangat penting untuk melahirkan murid yang mampu bersaing pada peringkat antarabangsa dan mampu menyumbang kepada manusia sejagat.

Elemen Kemahiran Berfikir Aras Tinggi (KBAT) merupakan satu elemen yang diberikan penekanan dalam buku ini. Hal ini bertujuan untuk menjadikan murid mampu menyelesaikan masalah dan mengaplikasikan ilmu yang dipelajari oleh mereka. Melalui ciri-ciri seperti kemahiran berfikir secara kritis dan kreatif, menaakul, dan strategi berfikir, murid berkeupayaan untuk membuat keputusan, berinovasi dan mereka cipta sesuatu.

Elemen Merentas Kurikulum (EMK) juga ditekankan bagi memperkukuh kemahiran dan keterampilan murid untuk menghadapi cabaran masa hadapan. Elemen yang dipupuk merangkumi penggunaan bahasa pengantar yang betul dan tepat, kesedaran untuk mencintai dan menyayangi alam sekitar, mempunyai nilai-nilai murni, semangat patriotik yang tinggi dan elemen Teknologi Maklumat dan Komunikasi.

Ikoni-ikon dalam Buku Teks dan Fungsinya



APLIKASI HARIAN

APLIKASI HARIAN

Mengaplikasikan pengetahuan sains komputer dalam kehidupan harian



UJI MINDA

UJI MINDA

Menyediakan persoalan menarik untuk menguji tahap kefahaman murid tentang bidang sains komputer



MALAYSIAKU!

MALAYSIAKU!

Menonjolkan kejayaan Malaysia dalam bidang sains komputer



KERJAYA SAINS KOMPUTER

KERJAYA SAINS KOMPUTER

Memperkenalkan jenis kerjaya yang berhubung dengan bidang sains komputer



INOVASI SAINS KOMPUTER

INOVASI SAINS KOMPUTER

Mengetengahkan perkembangan dan inovasi teknologi sains komputer pada masa ini



KBAT

Mendedahkan murid kepada soalan Kemahiran Berfikir Aras Tinggi

Tahukah
Anda



Memberikan info menarik tentang topik yang sedang dipelajari

AKTIVITI

PROJEK

KAJIAN KES

Pelbagai aktiviti yang bercorak penerokaan atau berdasarkan bahan rangsangan untuk menambahkan kefahaman murid tentang standard pembelajaran yang telah dipelajari.



Anda boleh mengakses portal Sains Komputer Tingkatan 5, iaitu sainskomptg5.com untuk mendapatkan latihan tambahan dan jawapan untuk soalan-soalan di dalam buku teks. Selain itu, anda juga boleh memperoleh sumber digital yang menarik dan berinformasi, contohnya animasi daripada portal yang sama.





Fikirkan...

- Bagaimanakah inovasi dalam pengkomputeran memberikan impak kepada manusia dan persekitaran?
- Bagaimanakah aktiviti-aktiviti tidak beretika dalam pengkomputeran dapat dibanteras melalui undang-undang siber?
- Sekiranya anda terdedah kepada *keyboard warrior*, apakah langkah-langkah keselamatan yang perlu anda lakukan?
- Apakah yang boleh disumbangkan oleh seseorang yang berkerjaya dalam bidang pengkomputeran?

BAB 1

PENGGOMPUTERAN

Dalam bab ini, anda akan mempelajari:

- Kesan inovasi dalam pengkomputeran
- Aktiviti tidak beretika dalam pengkomputeran
- Keperluan keselamatan data dalam rangkaian komputer
- Penyulitan dan nyahsulit data menggunakan *Caesar Cipher*
- Undang-undang siber antara negara
- Kerjaya dalam bidang pengkomputeran
- Kitaran Capai-Nyahkod-Laksana
- Konsep asas seni bina komputer
- Pemproses linear dan pemproses semasa
- Penterjemahan pelbagai aras bahasa pengaturcaraan
- Get logik sebagai binaan asas litar bersepadu
- Menghasilkan litar get logik dan jadual kebenaran

Dunia kini semakin bergantung pada Internet dan sistem komputer untuk menjalankan aktiviti-aktiviti harian, sama ada di rumah, di sekolah ataupun di tempat kerja terutamanya yang melibatkan institusi kewangan. Selain menyediakan kemudahan untuk berkomunikasi dan mengurus niaga, Internet juga dijadikan sasaran bagi penjenayah untuk mencari mangsa. Perbuatan tersebut menyalahi undang-undang pemintasan komunikasi dan pendedahan terhadap serangan mengikut keadaan semasa. Kelemahan dalam penguatkuasaan undang-undang dan kekurangan dalam pendidikan dan kesedaran awam tentang bentuk jenayah yang baharu ini menjadikan masalah ini lebih rumit untuk ditangani.



1.1

STANDARD PEMBELAJARAN

Pada akhir pembelajaran murid dapat

1.1.1 Menganalisis kesan inovasi dalam pengkomputeran

1.1.2 Mengenal pasti aktiviti tidak beretika dalam pengkomputeran

1.1.3 Menjelaskan keperluan keselamatan data dalam rangkaian komputer dan langkah keselamatan (*encrypted email, double verification, password*)

1.1.4 Melaksanakan penyulitan (*encryption*) dan nyahsulit (*decryption*) data menggunakan *Caesar Cipher*

1.1.5 Membincangkan implikasi sosial berkaitan penggodaman dan cetak rompak perisian

1.1.6 Mengkaji undang-undang siber antara negara

1.1.7 Mengkaji kerjaya yang berkaitan dengan bidang pengkomputeran masa hadapan



Historical Technological Innovation Examples
goo.gl/glua44



Komputer dan Impak

Teknologi digital dalam bidang pengkomputeran berkembang dengan pantas seiring dengan perkembangan sistem komunikasi rangkaian. Kedua-dua teknologi ini telah mengubah cara belajar, bersosial, berpolitik, bekerja dan menjalankan perniagaan. Walaupun teknologi-teknologi ini suatu perkembangan yang positif, namun terdapat juga agenda yang tidak sihat berlaku seiring dengan perkembangan ini. Istilah dunia tanpa sempadan (*borderless world*) telah melahirkan pelbagai aktiviti yang boleh dieksploitasi oleh penjenayah yang melakukan jenayah dalam ruang siber yang disebut “jenayah siber”. Contoh jenayah ini adalah seperti penipuan, pornografi, ugutan, pelanggaran hak cipta, serangan virus terhadap komputer dan sebagainya. Penjenayah akan sentiasa mencari jalan untuk mengeksploitasi teknologi baharu. Kini, cabaran bagi semua negara adalah untuk memastikan undang-undang jenayah, termasuklah jenayah siber dapat dilaksanakan dengan sempurna bagi mengawal agenda penjenayah siber.



Rajah 1.1.1 Contoh jenayah siber

1.1.1 Kesan Inovasi dalam Pengkomputeran

Cabaran yang terdapat dalam sistem pendidikan abad ke-21 adalah untuk melahirkan murid yang boleh berfikir kritis, kreatif, menyelesaikan masalah dan mempunyai daya kepimpinan untuk bersaing pada peringkat global. Salah satu tunjang sistem pendidikan Kurikulum Standard Sekolah Menengah (KSSM) ialah sains dan teknologi.

Kewujudan Internet telah banyak mengubah landskap pembelajaran masa ini, iaitu dari bilik darjah tradisional ke bilik darjah abad ke-21 yang dikawal selia oleh teknologi. Inovasi dalam teknologi maklumat dan komunikasi (ICT) telah mengubah corak pengajaran dan pembelajaran,

begitu juga dalam persekitaran pekerjaan dan komunikasi dengan dunia luar yang tanpa sempadan. Kesan inovasi dalam bidang pengkomputeran boleh digambarkan seperti dalam Rajah 1.1.2.



Rajah 1.1.2 Kesan inovasi dalam pengkomputeran

Kesan Positif

Kesan positif hasil daripada inovasi dalam pengkomputeran dihuraikan seperti yang berikut:

Inovasi dalam Teknologi Rangkaian

Pada peringkat awal, komputer beroperasi secara sendiri (*stand alone*). Perubahan berlaku dengan mewujudkan hubungan antara komputer dengan komputer atau peranti lain. Kemudian, perubahan ini dilanjutkan kepada konsep rangkaian yang bermula dengan rangkaian kawasan tempatan (LAN), rangkaian kawasan luas (WAN) dan seterusnya terciptanya Internet.

Inovasi dalam Teknologi Storan

IBM memperkenalkan cakera keras (*hard disk*) yang pertama pada tahun 1956 yang berkapasiti 5 MB. Kajian sentiasa dilakukan untuk mencipta jenis dan bentuk storan yang lebih baik bagi memenuhi keperluan penggunaan semasa seperti menambah kapasiti storan, bahan binaan cakera, saiz peranti dan kelajuan yang mempengaruhi inovasi dalam

Tahukah Anda

Salah satu faktor yang memerlukan perubahan bagi menyokong perkembangan IoT (*Internet of Things*) ialah perubahan pada pertambahan bilangan nombor IP bagi peranti elektronik, iaitu IPV6 menggantikan IPV4.



teknologi storan. Selain cakera keras, data juga disimpan dalam cakera liut (*floppy disk*), cakera optik (*optical disk*) atau pemacu pena (*pendrive*).

Seiring dengan perkembangan dalam teknologi Internet, wujudnya storan awan (*cloud storage*), yang membolehkan data disimpan di *cloud*. Data ini juga boleh dicapai pada bila-bila dan di mana-mana melalui pelbagai peranti selagi terdapat liputan perkhidmatan Internet.



Inovasi Sistem Pengoperasian

Komputer beroperasi dengan adanya sistem pengoperasian. Sistem pengoperasian mengawal operasi keseluruhan sistem komputer. Dengan perkembangan pesat dalam teknologi sistem perisian dan teknologi rangkaian, maka terciptanya sistem pengoperasian (*Operating system – OS*) untuk pelayan (*server*) komputer bagi mengawal rangkaian.

Selain komputer, OS juga digunakan untuk peranti lain, seperti alatan mudah alih (*mobile devices*), contohnya *Android*, *iOS* dan *BlackBerry*. Telefon yang dahulunya digunakan untuk membuat panggilan dan menghantar teks sahaja, kini dengan kemajuan teknologi sistem pengoperasian, telefon pintar mempunyai ciri-ciri komputer dan telefon biasa. Hasil inovasi dalam ketiga-tiga bidang teknologi yang dibincangkan telah merangsang perkembangan sistem dalam talian dan jenayah siber.



APLIKASI HARIAN

Media sosial memudahkan pengguna berinteraksi dalam talian. Contoh media sosial ialah *Facebook*, *Twitter*, *WhatsApp* dan *Instagram*.

Perkembangan Sistem dalam Talian

Kesan inovasi dalam teknologi rangkaian, teknologi storan simpanan dan sistem pengoperasian telah merangsang perkembangan sistem dalam talian. Contohnya, mewujudkan sistem e-perbankan, sistem e-dagang, sistem hiburan elektronik, sistem e-pembelajaran, e-kerajaan, perundangan dan jaringan telekomunikasi tanpa batasan.

Kesan Negatif

Kesan negatif hasil daripada inovasi dalam pengkomputeran adalah seperti yang berikut:

Jenayah Siber

Jenayah siber boleh didefinisikan sebagai penyalahgunaan teknologi pengkomputeran dan Internet bagi melaksanakan aktiviti jenayah, iaitu merosakkan, mencuri atau mengubah data dalam sistem komputer bagi tujuan tertentu. Jenayah yang dilakukan akan memberikan kesan negatif atau kerugian kepada mangsa. Namun, jenayah siber sukar membuktikan siapa yang terlibat, cara, tujuan dan tempat berlakunya jenayah. Hal ini berlaku kerana jenayah siber boleh dilakukan dari jarak yang jauh dengan bantuan Internet, tanpa kehadiran secara fizikal penjenayah tersebut.



Many Malaysian Firms Unconcerned About Cyber Crimes and Incidents
goo.gl/Pn4fKa

Jenis-jenis Jenayah Siber

Jenayah siber dapat dikelaskan kepada:

- **Jenayah siber dari perspektif pengguna persendirian dan institusi**

Termasuk penipuan, pembulian siber, penipuan e-mel, kecurian maklumat (*phishing*), perjudian dalam talian, imej penderaan kanak-kanak, pornografi, skim cepat kaya, menyalin perisian secara tidak sah, pengubahsuaian atau merosakkan program dan data komputer, capaian yang tidak dibenarkan, *malware* atau penyebaran virus dan *Distributed Denial-of-Service (DDoS)*, laman web palsu, serta menggodam (*hacking*).

- **Jenayah siber dari perspektif keselamatan negara**

Termasuk serangan penganas siber terhadap infrastruktur kritikal negara seperti sistem kerajaan, sistem kawalan trafik udara, sistem penjana elektrik, infrastruktur rangkaian telekomunikasi, sistem ketenteraan, sistem kawalan industri dan *hacktivism* (gabungan *hacking* dan aktivisme).

Antara contoh situasi jenayah siber adalah seperti berikut:

- 1 Penjenayah siber menyebarkan virus menerusi telefon pintar atau komputer untuk mendapatkan kata laluan perbankan daripada akaun media sosial apabila telefon mangsa dihubungkan menerusi perkhidmatan jalur lebar mudah alih (*Wi-Fi*) di tempat awam, penggodam akan menggunakan peranti khusus bagi memasukkan *malware* ke dalam telefon pintar mangsa untuk mendapatkan maklumat.
- 2 Penjenayah siber akan menghantar e-mel yang dimuatkan dengan virus kepada senarai e-mel mangsa. Apabila e-mel dibuka oleh penerima, virus diaktifkan dan akan melakukan aktiviti mengikut atur caranya seperti mengurangkan ruangan dalam ingatan, menghapuskan data dalam komputer mangsa, mengakses maklumat peribadi dan menghantar maklumat atau virus kepada komputer lain menerusi senarai e-mel mangsa.
- 3 *Phishing* ialah kecurian maklumat, biasanya dibuat melalui e-mel, untuk menipu pengguna bagi mendedahkan butiran pengguna kepada penyerang atau penjenayah siber. E-mel *phishing* biasanya kelihatan seperti dihantar oleh sebuah organisasi yang terkenal dan meminta maklumat peribadi seperti nombor kad kredit, nombor akaun atau nama log masuk (*login name*) dan kata laluan (*password*). Di Malaysia, kebanyakan sasaran serangan *phishing* yang dikesan terdiri daripada pengguna perbankan Internet dengan memberikan helah supaya pengguna mendedahkan maklumat perbankannya.

Tahukah Anda



Apakah *malware*?

Malware ialah singkatan bagi *malicious software* (perisian yang berbahaya). *Malware* merupakan satu program yang direka khas untuk menjangkiti sesebuah sistem komputer tanpa diketahui pemiliknya.

Tahukah Anda



Pada tahun 2015, bilangan kes jenayah siber *spam* yang direkodkan di Malaysia ialah 35 000 manakala untuk *fraud* ialah 2960 kes. Berdasarkan *CyberSecurity Malaysia*, penjenayah mencari sasaran melalui media sosial.

4 Internet ialah “lebu h raya maklumat” kerana Internet membolehkan orang ramai mencari, berkongsi dan menyebarkan maklumat dengan cepat dan mudah. Pengaksesan dan pengagihan salinan bahan digital secara haram terhadap buku yang dilindungi hak cipta digital, filem, muzik, karya seni, perisian, dan kandungan lain oleh penjenayah siber merupakan satu bentuk pelanggaran hak cipta, biasanya dirujuk sebagai “cetak rompak”.

Tahukah Anda

Terdapat tujuh komponen penting bagi etika dalam pengkomputeran, iaitu:

- ✓ Kejujuran
- ✓ Hormat
- ✓ Keyakinan
- ✓ Profesional
- ✓ Tanggungjawab
- ✓ Komunikasi
- ✓ Patuh kepada undang-undang



APLIKASI HARIAN

Anda boleh melayari goo.gl/c7n8fb untuk mengetahui cara untuk mengenal pasti e-mel phishing.

AKTIVITI

1

Kerja Kumpulan

Jenayah Siber 

Pilih jenayah siber mengikut perspektif yang telah anda pelajari. Terangkan jenayah siber tersebut mengikut perspektif yang dipilih dan cara berlakunya jenayah siber tersebut. Cadangkan langkah-langkah untuk mencegah berlakunya serangan tersebut. Sekiranya anda seorang mangsa, cadangkan tindakan yang perlu anda lakukan. Berbincanglah bersama-sama rakan kumpulan anda dan bentangkan hasil perbincangan di dalam kelas, sama ada menggunakan papan tulis atau melalui lakonan.

1.1.2 Etika dalam Pengkomputeran

Etika dalam pengkomputeran merujuk standard panduan bagi mengawal perlakuan pengguna komputer supaya lebih bermoral. Jadual 1.1.1 menunjukkan panduan bagi membantu pengguna untuk menentukan sama ada tindakan berikut beretika ataupun tidak.

Panduan ini telah dicadangkan oleh Institut Etika Komputer (*Computer Ethics Institute*) sebagai rujukan untuk semua pengguna komputer.

Walaupun kod etika disediakan sebagai panduan kepada pengguna komputer, masih terdapat perlakuan yang tidak beretika. Yang berikut ialah contoh aktiviti yang tidak beretika dalam pengkomputeran:

Jadual 1.1.1 Panduan etika untuk pengguna komputer

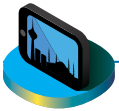
Bil	Etika	Contoh aktiviti yang tidak beretika dalam pengkomputeran
1	Dilarang menggunakan komputer dengan niat untuk membahayakan orang lain	Menggunakan komputer untuk mencipta perisian virus dan menyebarkan virus dengan niat untuk merosakkan atau memanipulasikan data atau fail pengguna lain.



Bil	Etika	Contoh aktiviti yang tidak beretika dalam pengkomputeran
2	Dilarang mengganggu urusan privasi atau tugas komputer orang lain	Mengganggu hak urusan peribadi orang lain dan memuat turun perisian tidak sah ke dalam komputer orang lain sehingga mengganggu fungsi komputer.
3	Dilarang mencero boh ke dalam fail komputer pengguna yang lain	Mendapatkan data daripada fail peribadi orang lain tanpa kebenaran. Mengintip fail orang lain atau membaca mesej peribadi orang lain melanggar hak peribadi.
4	Dilarang menggunakan komputer untuk mencuri	Menggunakan komputer untuk mencuri data peribadi bagi tujuan mendapatkan sumber kewangan.
5	Dilarang menggunakan komputer untuk bertindak sebagai saksi palsu	Menggunakan komputer untuk menyebarkan berita palsu dengan mengubah maklumat tentang seseorang yang boleh merosakkan nama baiknya.
6	Dilarang menggunakan atau menyalin perisian orang lain tanpa kebenaran	Melakukan salinan haram terhadap perisian hak cipta tanpa kebenaran penciptanya.
7	Tidak menggunakan sumber komputer pengguna lain tanpa kebenaran	Tidak meminta izin terlebih dahulu daripada pemilik komputer sebelum menggunakan komputer tersebut.
8	Dilarang menggunakan harta intelek orang lain sebagai hak milik	Membuat salinan atau mengkomersialkan perisian tersebut tanpa kebenaran pemiliknya.
9	Mempertimbangkan kesan sosial daripada program dan sistem yang dibangunkan	Pembangun perisian mencipta permainan video atau perisian pendidikan dengan tidak mempertimbangkan jenis kandungan yang mempunyai unsur-unsur yang tidak baik dan memberi kesan negatif kepada kanak-kanak.
10	Disaran menggunakan komputer dengan menunjukkan pertimbangan dan menghormati pengguna lain	Semasa berkomunikasi melalui Internet, seseorang tidak melayan orang lain dengan hormat, mencero boh ruang peribadi orang lain, menggunakan bahasa kesat, membuat kenyataan palsu mengenai orang lain, tidak berbudi bahasa semasa berkomunikasi melalui Internet dan tidak menghormati masa dan sumber orang lain. Selain itu, seseorang itu tidak bertimbang rasa dengan pengguna komputer baharu.



- 1 Dapatkan maklumat tentang cara untuk mengenal pasti e-mel jenis *phishing*. Bagaimanakah anda dapat melindungi diri daripada menjadi mangsa?
- 2 Pada pandangan anda, apakah peranan Suruhanjaya Komunikasi dan Multimedia Malaysia (SKMM) dan CyberSecurity Malaysia (CSM)?



MALAYSIAKU!

Suruhanjaya Komunikasi dan Multimedia Malaysia (SKMM) ditubuhkan pada tahun 1998 bagi mengawal selia peraturan industri berdasarkan kuasa yang termaktub dalam Akta Suruhanjaya Komunikasi dan Multimedia Malaysia (1998) dan Akta Komunikasi dan Multimedia (1998).

1.1.3

Keperluan Keselamatan Data dalam Rangkaian Komputer dan Langkah Keselamatan

Isu keselamatan data dalam rangkaian komputer perlu diberi perhatian kerana kelemahan ini memudahkan penjenayah siber untuk membuat pintasan terhadap rangkaian komputer dan mencuri maklumat peribadi yang berharga. Menghantar e-mel dan melakukan proses transaksi kewangan dalam talian ialah contoh aktiviti yang dilakukan melalui rangkaian komputer. Proses menghantar dan mengakses data dalam rangkaian komputer perlu melalui beberapa kaedah keselamatan.

Langkah-langkah untuk memastikan keselamatan data boleh dilakukan seperti yang berikut:

Penyulitan e-mel (*Encrypted email*) → Merupakan proses penyulitan maklumat yang akan dihantar melalui e-mel. Biasanya, proses ini melibatkan maklumat yang sensitif atau peribadi. Terdapat beberapa cara untuk menghantar e-mel ini. Contohnya, melalui *SafeGmail* dari *Google Chrome* yang membolehkan anda menghantar e-mel yang disulitkan kepada sesiapa sahaja. *RMail*, *infoencrypt*, *Hushmail*, *iSafeguard*, *safe-mail.net* dan banyak lagi pautan laman web menyediakan penyelesaian bagi penghantaran e-mel secara selamat.

Dua pengesahan (*Double verification*) → Merupakan lapisan pengesahan tambahan yang perlu ditentukan oleh pengguna apabila ingin memasuki akaun atau mengakses data dalam talian. Dengan adanya dua lapisan tersebut, langkah pengesahan untuk mengakses data menjadi lebih sukar. Jika penggodam mampu menggodam lapisan pertama, iaitu kata laluan (*password*), dia tetap memerlukan nombor kunci keselamatan, seterusnya untuk memasuki akaun anda. Semakin banyak lapisan pengesahan, semakin sukar data dapat diceroboh.



Stronger Security For Your Google Account
goo.gl/qNftBz

Rajah 1.1.3 menunjukkan kaedah dua pengesahan untuk menguatkan ciri-ciri keselamatan.



Rajah 1.1.3 Kaedah dua lapisan pengesahan

Kata laluan (*Password*) → Merupakan lapisan pengesahan keselamatan yang utama untuk memasuki akaun pengguna. Dengan hanya satu lapisan keselamatan, iaitu kata laluan, keadaan ini akan memudahkan penceroboh memasuki akaun pengguna.

Bagi menghalang data diceroboh, pemilihan kata laluan amatlah penting. Pengguna mesti mengelakkan pembinaan kata laluan berdasarkan nama, nombor kad pengenalan, tarikh lahir atau sebarang data yang berkaitan dengan maklumat peribadi pengguna.

Kata laluan yang baik dibina daripada gabungan huruf, simbol dan nombor, iaitu antara 8 hingga 12 aksara. Kebarangkalian gabungan aksara ini akan menyukarkan penggodam untuk menemukannya. Contoh kata laluan:

- W#uy4266
- RZ678906

Sebagai murid pada abad ke-21, anda hendaklah lebih peka tentang jenis ancaman yang ada pada masa ini dan mempelajari langkah-langkah yang perlu diambil bagi melindungi hak data peribadi supaya dapat menikmati pengalaman melayari Internet yang baik dan selamat.



UJI MINDA

Mengapakah kecurian maklumat pengguna boleh berlaku apabila pengguna melayari Internet dengan menggunakan *Wi-Fi* di tempat awam?



The Importance of Password Security
goo.gl/z0apWg

AKTIVITI

3

Kerja Individu

Keperluan Keselamatan Data



Dalam proses pengesahan (*verification*), selain penggunaan nombor pin atau kata laluan, kaedah biometrik merupakan cara yang bagus jika digunakan sebagai salah satu lapisan pengesahan.

- 1 Berikan satu contoh kaedah biometrik yang boleh digunakan sebagai pengesahan.
- 2 Nyatakan cara peranti biometrik ini boleh digunakan.

Tahukah Anda

Terdapat empat fungsi utama kriptografi pada masa ini, iaitu:

- 1 Memelihara privasi (*confidentiality*)
- 2 Menjamin ketulenanan (*authentication*)
- 3 Memastikan integriti
- 4 Mengelakkan penafian atau penipuan (*nonrepudiation*)

Sumber:
technet.microsoft.com/en-us/library/cc961634.aspx



APLIKASI HARIAN

Anda boleh layari goo.gl/3BZD02 untuk mencuba proses penyulitan dan nyahsulit. Selamat mencuba!

1.1.4 Penyulitan dan Nyahsulit Data Menggunakan *Caesar Cipher*

Kriptografi (*Cryptography*) merupakan satu cabang dalam bidang Sains Komputer dan Teknologi Maklumat yang amat penting terutamanya untuk menangani masalah dalam bidang keselamatan data agar pengguna selamat daripada dipintas oleh pihak ketiga semasa melalui proses seperti urus niaga perbankan dalam talian, log masuk laman web dan sebagainya. Perkara ini boleh dilakukan dengan proses seperti penyulitan data bagi melindungi privasi data pengguna.

Penyulitan

Penyulitan ialah proses penterjemahan data kepada bentuk kod rahsia yang tidak dapat difahami dan merupakan satu cara yang paling berkesan untuk melindungi data. Data yang belum melalui proses penyulitan disebut sebagai teks biasa (*plaintext*). Selepas melalui penyulitan, data tersebut merupakan teks sifer (*ciphertext*). Maksud bagi istilah penyulitan dan nyahsulit ialah:

Penyulitan (*encryption*) data ialah proses pengekodan teks biasa (*plaintext*) kepada bentuk teks sifer (*ciphertext*) yang tidak dapat difahami.

Nyahsulit (*decryption*) data ialah proses pengekodan teks sifer (*ciphertext*) kepada bentuk teks biasa (*plaintext*).

Jenis Penyulitan

Terdapat dua jenis penyulitan, iaitu penyulitan simetri (*symmetric encryption*) dan penyulitan tidak simetri (*asymmetric encryption*). Rajah 1.1.4 dan 1.1.5 menunjukkan perbezaan bagi kedua-dua jenis penyulitan.

(a) Penyulitan simetri (*Symmetric encryption*)

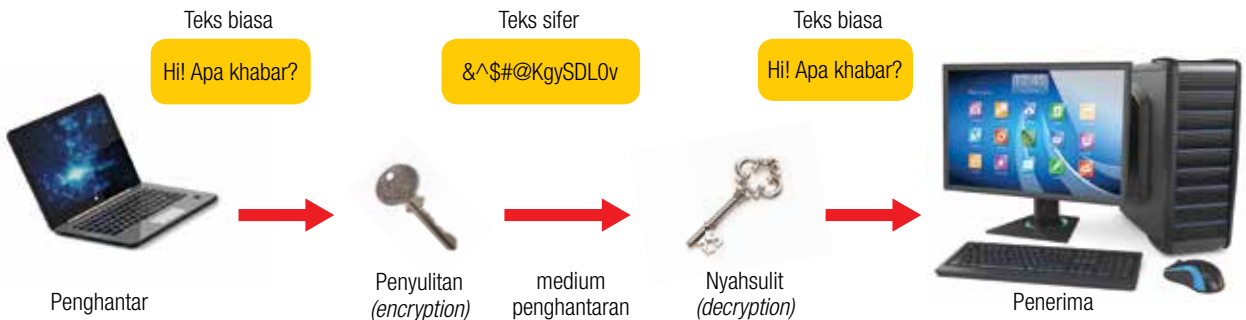
Rajah 1.1.4 menunjukkan penyulitan simetri, iaitu “kunci” yang digunakan bagi penyulitan dan nyahsulit teks adalah daripada jenis yang sama.



Rajah 1.1.4 Penyulitan simetri (*Symmetric encryption*), iaitu penyulitan dan nyahsulit teks menggunakan kunci yang sama.

(b) Penyulitan tidak simetri (*Asymmetric encryption*)

Rajah 1.1.5 menunjukkan penyulitan tidak simetri yang menggunakan konsep pasangan “kunci”. “Kunci” yang berbeza digunakan untuk proses penyulitan dan nyahsulit.



Rajah 1.1.5 Penyulitan tidak simetri (*Asymmetric encryption*), iaitu penyulitan dan nyahsulit teks menggunakan kunci yang berlainan.

Bagi melaksanakan proses penyulitan dan nyahsulit data, terdapat beberapa kaedah yang boleh dipelajari dan digunakan. Salah satu kaedah yang paling ringkas untuk memahami cara proses penyulitan dilaksanakan ialah kaedah *Caesar Cipher*.

Caesar Cipher

Julius Caesar mencipta kaedah *Caesar Cipher* yang juga dikenali sebagai *cipher shift*. *Caesar Cipher* ialah salah satu bentuk penyulitan yang paling mudah. *Caesar Cipher* merupakan *cipher* penggantian (*substitution*), iaitu setiap huruf teks biasa (*plaintext*) digantikan dengan huruf yang sepadan dengan huruf dalam abjad (A–Z). Formula Matematik yang digunakan dalam kaedah *Caesar Cipher* adalah seperti yang berikut:

$$E_n(x) = (x + n) \bmod 26$$

n – merujuk bilangan anjakan (sama ada tiga atau lebih).

$\bmod 26$ – merujuk jumlah huruf A hingga Z, iaitu 26. Sekiranya huruf dianjak sehingga akhir abjad, iaitu Z, hurufnya akan berpusing ke huruf permulaan semula, iaitu bermula dengan A.

Contoh: huruf $E_3(A) = (A + 3) \bmod 26 = D$, iaitu [A + turutan 3 huruf (B, C, D)], iaitu menggunakan peralihan atau anjakan ke kanan sebanyak 3 kali.

Dengan itu, teks biasa huruf A bersamaan dengan teks sifer huruf D.

Proses nyahsulit (*decryption*) terhadap teks sifer (*Ciphertext*) dilakukan dengan cara yang sama tetapi dengan menolak jumlah anjakan.



Caesar cipher
goo.gl/71HnNX



What is encryption and decryption?
goo.gl/UyN5yV

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z → teks biasa
 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z A B → teks sifer

CONTOH 1

Contoh 1 menerangkan kaedah untuk melakukan penyulitan (*encryption*) terhadap teks biasa (*plaintext*) berikut. Diberi nilai anjakan, $n = 3$.

ENCRYPTION

Proses yang dilakukan adalah seperti yang berikut:

Penyelesaian:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z → teks biasa
 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z A B → teks sifer

Teks sifer (*ciphertext*) yang dihasilkan ialah:

H Q F U B S W L R Q

Proses nyahsulit (*decryption*) terhadap teks sifer (*ciphertext*) dilakukan dengan cara yang sama tetapi dengan menolak jumlah anjakan.

CONTOH 2

Contoh 2 menerangkan proses nyahsulit (*decryption*) *ciphertext* yang ditunjukkan, diberi nilai anjakan, $n = 3$.

NRPSXWHU

Penyelesaian:

Daripada huruf *ciphertext*, tolakkan anjakan sebanyak tiga kali.

N	R	P	S	X	W	H	U
K L M N	O P Q R	M N O P	P Q R S	U V W X	T U V W	E F G H	R S T U

Maka, teks biasa (*plaintext*) yang dinyahsulitkan ialah:

K O M P U T E R

Teknik *Caesar Cipher* mudah dilaksanakan tetapi strukturnya yang ringkas mudah dipecahkan jika berbanding dengan teknik yang lain.

AKTIVITI

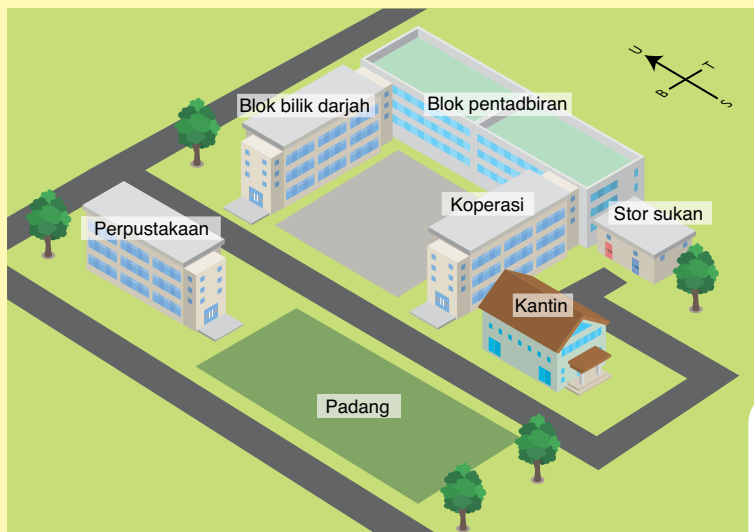
4

Kerja Kumpulan

Caesar Cipher



Cikgu Ramesh telah mengadakan pertandingan mencari harta karun bersempena dengan Sehari Bersama Kelab Sains Komputer di sekolahnya. Cikgu Ramesh telah menyediakan pembayang di lokasi yang terdapat petunjuk untuk ke lokasi akhir harta karun bagi setiap peserta dalam bentuk *ciphertext*. Terdapat lima pembayang lokasi yang perlu dinyahsulitkan oleh peserta, iaitu:



- 1 NDQWLQ
- 2 SHUSXWDDNQ
- 3 WWRU VXNDQ
- 4 NRSHUDVL

Diberi nilai anjakan $n = 3$. Bentuk satu kumpulan yang terdiri daripada beberapa orang ahli. Kemudian, nyahsulitkan *ciphertext* yang diberikan.

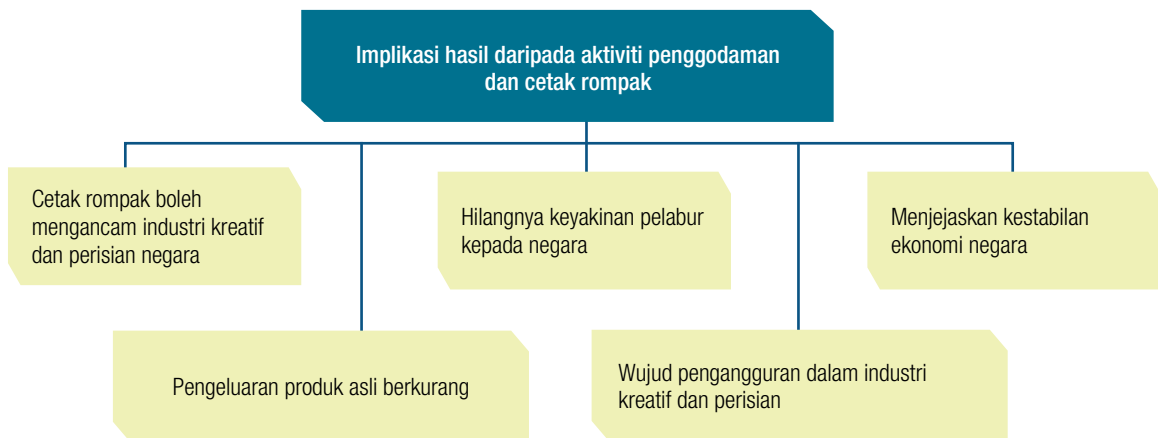
Selamat mencuba!

1.1.5 Implikasi Sosial Hasil Penggodaman dan Cetak Rompak Perisian

Cetak rompak atau pencabulan hak cipta ialah perbuatan menyalin semula, menerbitkan atau menggunakan bahan bercetak atau bahan rakaman digital seperti video dan perisian komputer yang dilakukan tanpa kebenaran pemegang hak cipta bagi harta intelek tersebut dengan matlamat mengaut keuntungan.

Penggodam (*hackers*) ialah penjenayah siber yang bertindak memintas komunikasi rangkaian bagi memperoleh data penting pengguna untuk dieksploitasikan. Kegiatan cetak rompak dan penggodam merupakan satu kesalahan menurut Akta Hak Cipta 1987 (pindaan 1997) dan Akta Jenayah Komputer 1996 yang digariskan oleh kerajaan. Kedua-dua aktiviti yang tidak sah ini mengakibatkan kerugian kepada mangsa, industri dan negara.

Rajah 1.1.6 menunjukkan implikasi hasil daripada aktiviti penggodaman dan cetak rompak kepada negara.



Rajah 1.1.6 Implikasi daripada aktiviti penggodam dan cetak rompak

- **Cetak rompak mengancam industri kreatif dan perisian negara**

Kegiatan cetak rompak akan merugikan pihak pemilik karya asli dan pengeluar yang sah. Impak negatif, terutamanya kepada pengusaha syarikat rakaman video dan lagu, perfileman, penerbitan buku-buku dan juga hasil kreatif yang lain. Individu yang terlibat secara langsung atau tidak langsung dalam industri ini akan terasa kesannya.

- **Pengeluaran produk asli berkurang**

Kualiti produk cetak rompak yang rendah akan memberikan kesan kepada barangan asli dan seterusnya akan menjejaskan pengeluaran produk asli.

- **Hilangnya keyakinan pelabur kepada negara**

Bagi menghasilkan sesuatu produk yang berkualiti, kita memerlukan proses penyelidikan dan pembangunan (R&D). Bagi menampung kelancaran proses penyelidikan dan pembangunan, syarikat memerlukan pelaburan yang tinggi terutamanya daripada syarikat yang bermodal besar dari negara-negara maju. Jika aktiviti cetak rompak dan penggodaman masih berleluasa, sudah pasti pelabur tidak akan melabur di negara kita. Hal ini dikatakan demikian kerana kos pelaburannya sangat tinggi tetapi pulangannya tidak meyakinkan. Para pelabur akan hilang keyakinan kerana mereka berasa bahawa negara ini tidak menyediakan perlindungan yang mencukupi terhadap harta intelek dan juga keselamatan data.

- **Wujud pengangguran dalam industri kreatif dan perisian**

Cetak rompak dan penggodaman juga mendatangkan kesan negatif terhadap industri perisian kerana syarikat masa ini beroperasi secara dalam talian dan menggunakan sistem komputer bagi mengurus operasinya seperti e-perbankan dan e-niaga. Kepercayaan pengguna akan hilang jika keselamatan data peribadi pengguna tidak dapat diurus dengan baik oleh syarikat yang berurusan. Jika pelanggan menarik diri, syarikat akan mengalami kerugian dan sudah pasti akan menghentikan operasinya.



APLIKASI HARIAN

Laporan daripada portal rasmi Kementerian Perdagangan Dalam Negeri, Koperasi dan Kepenggunaan (KPDNKK) mengenai jenayah cetak rompak.



goo.gl/1GBxmV

Keadaan ini meninggikan kadar pengangguran jika syarikat tidak mampu meneruskan operasi dalam bidang tersebut.

- **Menjejaskan kestabilan ekonomi negara**

Jenayah cetak rompak turut menjejaskan ekonomi negara apabila masyarakat mengalih perhatian dengan membeli barangan cetak rompak. Negara akan mengalami kerugian daripada hasil cukai yang merupakan salah satu sumber pendapatan negara.

KAJIAN KES

Artikel yang berikut dipetik daripada akhbar tempatan.

Layari goo.gl/5zxd9M untuk mendapatkan maklumat yang lebih lanjut bersama-sama dengan ahli kumpulan anda.

Berdasarkan maklumat yang diperolehi daripada akhbar tersebut, nyatakan industri utama yang terlibat dalam penggunaan perisian cetak rompak. Kemudian, cadangkan empat faktor yang mengaktifkan kegiatan ini. Kemukakan cara penyelesaian yang munasabah untuk menangani masalah ini. Kemudian, persembahkan hasil kerja kumpulan anda dengan menggunakan perisian seperti *PowerPoint*, *Google Slides* atau peta *i-Think*. Pastikan perisian yang anda gunakan adalah asli!



1.1.6

Undang-undang Siber antara Negara

Internet telah membuka ruang siber kepada perilaku jenayah dengan cara yang tidak pernah ada sebelum ini. Internet telah menawarkan peluang baharu yang lebih canggih untuk penjenayah melakukan jenayah tradisional dengan cara yang moden.

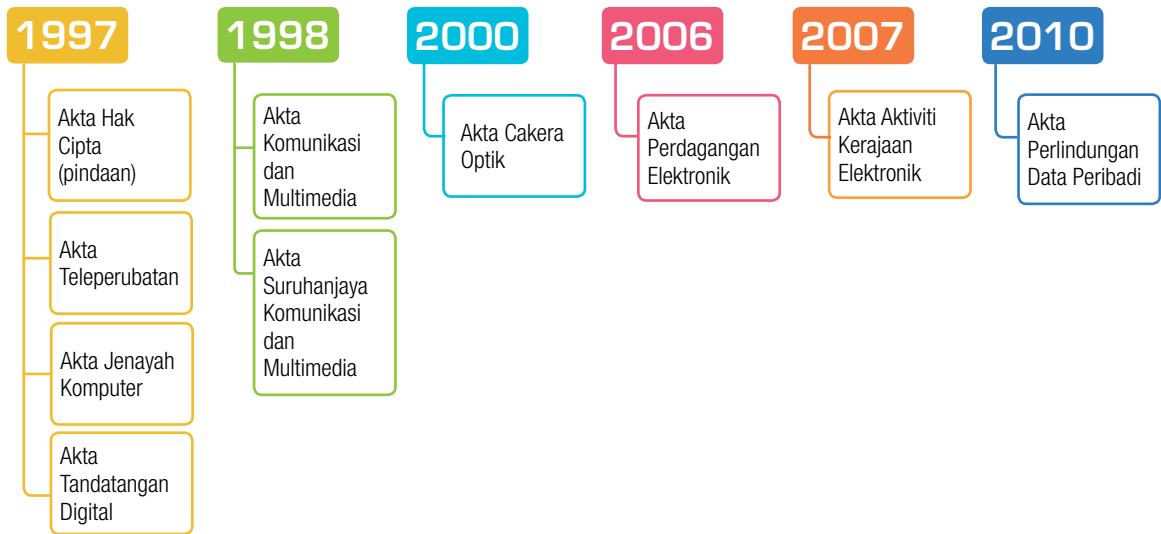
Undang-undang siber merupakan peraturan yang ditetapkan oleh pihak berkuasa di sesebuah negara untuk memberikan hukuman kepada pesalah-pesalah yang melakukan kesalahan dalam talian.

Kerajaan Malaysia telah menggubal undang-undang jenayah siber dengan memperkenalkan beberapa akta dalam undang-undang siber seperti yang ditunjukkan dalam Rajah 1.1.7.



*Cyber Law Acts in
Malaysia*
goo.gl/eB4iYa

Pada peringkat awal, Akta Jenayah Komputer 1997, Akta Tandatangan Digital 1997 dan Akta Teleperubatan 1997 diperkenalkan. Kemudian, Akta Hak Cipta telah dipinda bagi memenuhi isu siber (melalui pindaan 1997). Sehubungan dengan itu, skop hak cipta telah diperluas, iaitu penulis/pencipta juga mempunyai hak eksklusif untuk mengawal ciptaannya menurut *World Intellectual Technology Property Ownership (WIPO) Copyright Treaty 1996*. Seiring dengan perkembangan dunia semasa, lebih banyak akta telah digubal sehingga masa ini.



Rajah 1.1.7 Akta-akta di bawah undang-undang siber

Tahukah Anda

Murid boleh melayari laman web *Department of Homeland Security* dan perhatikan cara agensi ini menangani isu siber di Amerika Syarikat.



Department of Homeland Security-USA
goo.gl/Opmsu3

Pewujudan akta bagi jenayah siber merupakan satu cabaran bagi sesebuah negara. Bagi menangani isu jenayah siber, tiada organisasi, agensi atau negara yang mampu mengekang dan mempertahankan negara daripada serangannya secara persendirian. Negara membangun, seperti Malaysia boleh merujuk pengalaman daripada negara yang lebih maju seperti Australia, Amerika Syarikat dan United Kingdom untuk menganalisis undang-undang bagi pendakwaan kes jenayah siber yang semakin kompleks.

Perbezaan undang-undang siber antara negara tidak ketara kerana wujudnya usaha sama antara agensi kerajaan dengan agensi negara lain untuk menangani jenayah komputer yang merentas sempadan antarabangsa. Pelaksanaan hukuman dan cara penguatkuasaan bergantung pada polisi pemerintahan sesebuah negara. Antara isu yang sering diketengahkan pada masa ini adalah seperti isu-isu kebebasan bersuara, kelucahan dan fitnah.

Undang-undang yang sedia ada memerlukan penambahbaikan terhadap jenayah seperti perjudian dalam talian, pornografi kanak-kanak, kecurian harta intelek, penipuan dan kecurian identiti. Jadual 1.1.2 menunjukkan perbezaan undang-undang siber antara negara secara umum.

Kebebasan dalam penggunaan Internet di Malaysia tertakluk pada inisiatif yang disediakan oleh MSC Malaysia, iaitu perkara ketujuh BoG (*Bill of Guarantees*), untuk memastikan tiada penapisan Internet (*no censorship of the Internet*). Namun begitu, perkara ini masih dikawal selia di bawah bidang kuasa kerajaan yang mengawal selia penggunaan Internet demi memastikan keselamatan dan hak pengguna Internet terlindung. Kerajaan telah mewujudkan agensi-agensi seperti Suruhanjaya Komunikasi dan Multimedia Malaysia (SKMM) dan *CyberSecurity Malaysia* yang diberikan tanggungjawab untuk mengawal selia setiap aktiviti penggunaan Internet yang boleh mengganggu keamanan negara dan perniagaan pelabur-pelabur asing.

Jadual 1.1.2 Perbezaan undang-undang siber antara negara

Negara	Penubuhan Undang-undang Siber	Kawalan terhadap media sosial	Isu kebebasan bersuara	Isu kelucahan	Perjudian
Amerika Syarikat	Akta Jenayah Komputer 1986, <i>Department of Justice</i>	Tiada sekatan terhadap media sosial	Ada kebebasan	Ada kebebasan terhadap isu pornografi kecuali perjudian haram melibatkan penyalahgunaan terhadap kanak-kanak	Tidak menghalang perjudian dalam talian kecuali perjudian haram tertakluk di bawah akta penguatkuasaan perjudian Internet yang tidak sah mulai 2006
China	1997 <i>Cybersecurity Law</i> oleh <i>Public Security Bureau</i> (PSB) yang bertanggungjawab untuk keselamatan dalam negara	Menyekat capaian kepada Internet secara bebas termasuk media sosial, seperti <i>Facebook</i> dengan mengadakan penapisan Internet, iaitu <i>The Great Firewall of China</i>	Tiada kebebasan secara mutlak	Tidak dibenarkan	Tidak dibenarkan
Arab Saudi	<i>Anti-Cyber Crime Law by Royal Decree No. M/17 of March 26, 2007</i>	Menyekat capaian kepada Internet secara bebas, iaitu tidak membenarkan capaian kepada media sosial seperti <i>Facebook</i>	Tiada kebebasan secara mutlak	Tidak dibenarkan	Tidak dibenarkan
Malaysia	Akta Jenayah Komputer 1997 SKMM, <i>Cyber Security Malaysia</i>	Tiada sekatan terhadap media sosial	Tiada kebebasan secara mutlak	Tidak dibenarkan	Tidak dibenarkan

Tahukah Anda

Anugerah Keselamatan Siber Malaysia 2014 merupakan salah satu acara CSM-ACE 2014 yang menyaksikan tujuh kategori yang dikurniakan kepada syarikat dan individu terhadap sumbangan mereka untuk memperkasakan industri keselamatan siber di Malaysia. Anugerah Keselamatan Siber Malaysia merupakan pengiktirafan dan penghargaan tertinggi industri terhadap sumbangan mereka dari segi inovasi, iltizam, kepimpinan industri dan juga sumbangan kepada masyarakat umum melalui penyediaan perkhidmatan pendidikan dan kesedaran keselamatan siber sebagai khidmat masyarakat dengan tidak bertujuan untuk mengaut keuntungan.



Cyber crime Law in China
goo.gl/MzHA7Q

Terdapat beberapa perbezaan takrifan jenayah siber antara negara. Bagi negara-negara Islam, aktiviti seperti judi, pornografi dan pengiklanan seks merupakan suatu jenayah tetapi perkara-perkara ini bukan jenayah di kebanyakan negara. Sebagai contoh, di Amerika Syarikat, hal-hal sebegini tidak dianggap sebagai jenayah. Mereka lebih memberikan tumpuan kepada keselamatan dan perdagangan.

Mengikut laporan dari *U.S. Government Laws* yang berkenaan dengan perlindungan data peribadi, banyak akta yang lebih terperinci telah diperuntukkan bagi melindungi hak peribadi pengguna atau rakyatnya. Jika kita meneliti dokumen daripada *Cybercrime Laws of The United States* (rujuk pautan laman web: https://www.oas.org/juridico/spanish/us_cyb_laws.pdf), kesalahan setiap jenayah telah diperincikan dan hukuman yang setimpal dengan kesalahan tersebut juga dinyatakan. Dalam usaha untuk meningkatkan kawalan terhadap jenayah siber selain pelaksanaan undang-undang, pihak berkuasa Amerika Syarikat juga melakukan usaha sama dengan badan atau agensi swasta bagi mengadakan kempen kesedaran dan menyediakan laman web bagi rujukan rakyatnya.

Dengan itu, kebergantungan rakyatnya tidak hanya pada badan kerajaan, mereka juga boleh membuat aduan melalui mana-mana laman web yang ada program usaha sama dengan badan perundangan dan badan yang menyelenggara urusan perundangan bagi kes jenayah siber. Pangkalan data berpusat diperlukan bagi melayan keperluan akses melalui pelbagai laman web.

Selain agensi dalam negara, Amerika Syarikat juga mengadakan usaha sama dengan negara Eropah. Negara-negara yang berada dalam Kesatuan Eropah juga menggariskan undang-undang bersama bagi menangani masalah jenayah siber di kawasan kesatuan mereka. Usaha sama akan lebih sempurna bagi menangani masalah siber yang melintasi sempadan.

Di Malaysia, badan kerajaan yang memantau, mengekang penyebaran dan menggubal akta jenayah siber ialah Suruhanjaya Komunikasi dan Multimedia Malaysia (SKMM) dan *CyberSecurity* Malaysia. Penguatkuasaan bersama juga dijalankan dengan beberapa agensi kerajaan yang lain seperti Polis Diraja Malaysia (PDRM) (bahagian forensik digital), Kementerian Perdagangan Dalam Negeri, Koperasi dan Kepenggunaan (KPDNKK), Bank Negara Malaysia dan Kementerian Kesihatan Malaysia.

Kerajaan Malaysia juga melaksanakan usaha sama program dengan badan atau organisasi swasta bagi menangani permasalahan dalam jenayah siber. Untuk meningkatkan kesedaran orang ramai mengenai kepentingan menjaga keselamatan maklumat peribadi, *CyberSecurity* Malaysia telah memulakan penerbitan buku bersama Karangkrif, iaitu sebuah buku kisah benar 999, daripada beberapa kes jenayah siber yang telah dikendalikan oleh Cyber999. Cyber999 merupakan agensi di bawah

CyberSecurityMalaysia yang mengurus aduan berkenaan dengan kecurian identiti (*identity theft*), penipuan dan gangguan dalam talian. Program ini merupakan program kesedaran umum.

Malaysia memperkenalkan beberapa akta dalam undang-undang siber seperti yang ditunjukkan dalam Rajah 1.1.7. Setiap akta mengandungi beberapa pecahan seksyen bagi menangani permasalahan secara terperinci. Setiap perincian undang-undang yang dihasilkan seterusnya akan dilakukan penambahbaikan sehingga mencapai piawai yang lebih baik. Sebagai contoh, **Akta Jenayah Komputer 1997**, iaitu mengenai kesalahan yang berkaitan dengan penyalahgunaan komputer untuk tujuan melakukan jenayah telah digubal. Di bawah akta ini, akses secara tidak sah atau mengubah apa-apa program atau data yang terdapat di dalam sesebuah komputer adalah salah dan boleh dikenakan hukuman. Akta ini mengandungi 12 seksyen. Seksyen 9 dalam akta ini menyatakan, “Ianya berkesan kepada apa-apa jenayah yang dilakukan di luar Malaysia walaupun kesalahan tersebut dilakukan oleh seseorang atau dari sesuatu tempat di luar Malaysia selagi komputer, program atau data tersebut berada di Malaysia atau boleh dicapai, dihubungkan atau digunakan dengan komputer di Malaysia”.

Seksyen 9 dalam Akta Jenayah Komputer ini memberikan bidang kuasa yang luas kepada unit penguat kuasa tetapi masalah mungkin timbul bagi penggodam yang tinggal di luar Malaysia kerana akan mengganggu bidang kuasa jenayah negara tersebut. Perkara yang boleh dilakukan adalah seperti mengekstradisi rakyat negara tersebut. Perkara ini boleh dilakukan sekiranya ada kerjasama pada peringkat antarabangsa bagi menyelesaikan masalah jenayah siber.

Tahukah Anda



Hak cipta ialah satu perlindungan undang-undang yang diberikan tempoh masa tertentu kepada karya-karya ekspresi kreatif.

- Percabulan hak cipta merupakan satu kesalahan di bawah Akta Hakcipta 1987.
- Pemilik hak cipta karya mempunyai hak eksklusif terhadap karyanya.

AKTIVITI

5

Kerja Berkumpulan

Undang-undang Siber

Kaji empat sebab kes jenayah siber kini semakin meningkat walaupun pelbagai akta ditubuhkan bagi menghukum pesalah yang melakukan jenayah tersebut. Huraikan jawapan anda dengan contoh-contoh terkini. Bentangkan kajian anda dalam bentuk slaid dengan menggunakan perisian *PowerPoint*.

1.1.7 Kerjaya dalam Bidang Pengkomputeran

Kerjaya dalam pengkomputeran berhubung kait dengan reka bentuk, penciptaan, pengurusan dan penyelenggaraan pelbagai komponen bagi sistem termasuk perisian, perkakasan, integrasi dan sistem multimedia. Pengkomputeran boleh dibahagikan kepada empat bidang utama seperti dalam Rajah 1.1.8.



Rajah 1.1.8 Empat bidang utama dalam pengkomputeran

Jadual 1.1.3 menunjukkan peranan dan kerjaya yang terdapat dalam bidang pengkomputeran.

Jadual 1.1.3 Peranan dan kerjaya dalam bidang pengkomputeran

Bidang	Peranan dan tanggungjawab	Kemahiran dan kerjaya
Sistem rangkaian dan keselamatan rangkaian	Kerjaya dalam bidang ini bertanggungjawab untuk mengendalikan semua aspek keselamatan rangkaian, termasuklah penyulitan, pentadbiran <i>firewall</i> dan protokol rangkaian, dan boleh mengendalikan pelbagai tanggungjawab daripada membangunkan dasar keselamatan bagi menjalankan penilaian risiko dan audit keselamatan, atau mengesyorkan peningkatan sistem keselamatan data untuk menangani pencerobohan terhadap keselamatan data.	Kerjaya: Juruanalisis keselamatan maklumat Kemahiran: Kepakaran dalam bidang rangkaian dan keselamatan komputer
Sokongan maklumat dan perkhidmatan	Kerjaya dalam bidang ini bertanggungjawab untuk mengintegrasikan pengurusan perniagaan dan analisis data, menilai pelaksanaan sistem komputer untuk menyelesaikan masalah perniagaan yang rumit, menyediakan sokongan pengkomputeran dan sokongan teknikal, menyelenggara sistem maklumat untuk perniagaan dan memberikan cadangan tentang perisian dan perkakasan untuk mencapai objektif perniagaan.	Kerjaya: Pentadbir pangkalan data, juruanalisis sistem komputer, juruanalisis data raya (<i>big data</i>), juruanalisis sokongan komputer, pengurus pengkomputeran, pentadbir sistem komputer Kemahiran: Dalam perisian aplikasi perniagaan yang melibatkan pengurusan pangkalan data
Pengaturcaraan dan pembangunan perisian	Kerjaya dalam bidang ini bertanggungjawab untuk merancang, mereka bentuk, mengemaskinikan dan menguruskan perisian komputer dan sistem melalui pengaturcaraan dan pembangunan perisian.	Kerjaya: Juruanalisis sistem, jurutera sistem komputer, pembangun perisian, pengatur cara komputer Kemahiran: <i>Java, C#, C++, .Net, SharePoint, Python</i>

Bidang	Peranan dan tanggungjawab	Kemahiran dan kerjaya
Web dan komunikasi digital	Kerjaya dalam bidang ini bertanggungjawab untuk membangunkan aplikasi berasaskan web, menulis kod, mengintegrasikan sistem dan menyediakan bantuan teknikal. Seseorang itu perlu membangunkan dan menghasilkan media interaktif, termasuklah produk digital dan multimedia.	<p>Kerjaya: Pembangun laman web, pembangun perisian multimedia, pakar media digital, pereka grafik dan multimedia</p> <p>Kemahiran: <i>JavaScript</i>, HTML, AJAX, PHP, <i>.Net</i>, <i>Phyton</i>. Perisian dan perkakasan multimedia seperti <i>Adobe Photoshop</i>.</p>

Elemen Mutakhir dalam Pengkomputeran yang mempengaruhi Pembangunan Bidang Kerjaya Baharu

Transformasi dalam industri pengkomputeran telah membawa kepada pembangunan bidang kerjaya baharu dalam pengkomputeran. Tiga elemen telah dikenal pasti menjadi komponen utama infrastruktur bagi produk dan perkhidmatan dalam sektor perindustrian masa ini termasuklah:

- **Mudah Alih**
 - ✓ Melibatkan penghasilan aplikasi dan kandungan mudah alih (*mobile apps and content*) bagi peranti selain komputer.
- **Keselamatan data**
 - ✓ Apabila wujud pengkomputeran awan (*cloud computing*), pemayaan (*virtualization*), peningkatan dalam keselamatan data turut mendapat perhatian. Keperluan kepakaran dalam bidang penyulitan, pengesahan, pengurusan sijil, infrastruktur *firewall* dan *anti-malware* meningkat. Keperluan terhadap kerjaya penganalisis keselamatan data, pentadbir keselamatan rangkaian dan pentadbir keselamatan sistem turut mendapat perhatian yang tinggi.
- **Data raya (*Big data*) dan kecerdasan perniagaan & analitik (*business intelligence & analytics*)**
 - ✓ Kemampuan menterjemah, menganalisis dan keupayaan untuk mengubah kuantiti data mentah yang besar ke bentuk maklumat menjadi senjata penting dalam ekonomi global. Perniagaan di seluruh dunia memerlukan pakar pengkomputeran terlatih sebagai pengurus data yang mempunyai set kemahiran pentafsir yang memahami cara untuk mendapatkan data, menganalisis dan melaporkan data tersebut.



Understanding Information Technology: A Guide To IT career opportunities
goo.gl/mzLzXO



Tahukah Anda

Apakah forensik digital dan hubungannya dengan penyiasatan jenayah komputer?



goo.gl/wZEmtQ

Jadual 1.1.4 menunjukkan beberapa pautan laman sesawang bagi agensi yang berkaitan dengan bidang pengkomputeran.

Jadual 1.1.4 Pautan laman sesawang bidang pengkomputeran

Bidang	Pautan
Bidang keselamatan data	(i) http://www.mosti.gov.my/faqs/cybersecurity-malaysia/ (ii) http://www.SKMM.gov.my (iii) http://www.cybersecuritycareers.net/MY/#jobresults
Kaji selidik berkenaan pekerjaan dalam bidang IT di Malaysia	Persatuan industri komputer dan multimedia Malaysia (PIKOM – <i>The National ICT Association of Malaysia</i>) http://www.pikom.org.my/
Jabatan Forensik Digital	https://jobs.laimoon.com/malaysia/computer-forensics
Bidang kreatif multimedia	(i) http://MDeC.my (ii) http://cic.jobsmalaysia.gov.my/html/kerjaya-1-1.php?id=21&bidang=7
Rangkaian komputer	http://www.payscale.com/research/MY/People_with_Jobs_in_Computer%2FNetwork_Security/Salary#by_Job
Sains komputer	http://www.careerjet.com.my/computer-science-graduate-jobs.html

LATIHAN PENGUKUHAN 1.1

- _____ ialah satu proses menukar teks biasa kepada teks sifer bagi menghalang capaian yang tidak dibenarkan.
- Penyulitan _____ ialah proses penyulitan yang menggunakan pasangan kunci yang berlainan jenis.
- Inovasi dalam teknologi _____, _____ dan _____ menyumbang kepada perubahan sistem pembelajaran secara tradisional kepada e-pembelajaran atau lebih dikenali sebagai *e-learning*.
- Akta _____ yang telah dipinda pada tahun 1997 bertujuan untuk menangani isu jenayah siber.
- _____ menggunakan dua lapisan pengesahan, iaitu menggunakan kata kunci keselamatan atau biometrik selain menggunakan kata laluan (*password*) bagi menyukarkan pencerobohan data daripada pihak yang tidak dibenarkan.

6 Tandakan “✓” (untuk beretika) atau “✗” (untuk tidak beretika) bagi situasi yang diberikan.

Bil.	Situasi	✓/✗
(i)	Semasa membina laman web, didapati terdapat laman web yang sama seperti yang diingini, kemudian murid menyalin, mengubahsuainya dan menerbitkannya sebagai laman web sendiri.	
(ii)	Memuat turun perisian-perisian percuma daripada Internet ke dalam komputer sekolah.	
(iii)	Menghantar iklan promosi jualan barangan kepada kenalan dan bukan kenalan melalui e-mel.	
(iv)	Menolong menyebarkan risalah memohon bantuan atau derma melalui laman sosial seperti WhatsApp, Facebook yang dihantar oleh kawan-kawan.	
(v)	Majikan memantau dan membaca e-mel pekerja tanpa pengetahuan pekerja tersebut.	

7 Adakah mengubah keadaan semasa lampu isyarat sehingga menyebabkan lalu lintas menjadi kucar-kacir dan mungkin mewujudkan kemalangan dikategorikan sebagai jenayah siber? Bincangkan jawapan dan berikan alasan anda.



8 Akta undang-undang jenayah siber yang manakah yang akan digunakan untuk menghukum pesalah yang menggunakan perisian berlesen tetapi tidak mengikut syarat yang ditetapkan? Sebagai contoh, jumlah lesen yang dibeli tidak sama dengan jumlah perkakasan yang menggunakannya.



1.2

STANDARD PEMBELAJARAN

Pada akhir pembelajaran murid dapat

1.2.1 Menjelaskan kitaran capai-nyahkod-laksana (FDE—Fetch, Decode, Execute Cycle)

1.2.2 Menerangkan konsep asas seni bina (Von Neuman) – Unit kawalan, ALU (Arithmetic Logic Unit), register, clock, address bus dan data bus

1.2.3 Mengkaji perbezaan pemproses (processor) linear dan pemproses semasa

1.2.4 Membincangkan penterjemah pelbagai aras bahasa pengaturcaraan

1.2.5 Merumuskan pelaksanaan atur cara dalam sebuah pemproses



Timeline of Computer History
goo.gl/Wfzxp0q

Seni Bina Komputer

Pengenalan kepada Seni Bina Komputer

Seni bina komputer terbahagi kepada tiga komponen utama seperti yang ditunjukkan dalam Rajah 1.2.1, iaitu Unit Pemprosesan Pusat (UPP) atau dikenali sebagai *Central Processing Unit* (CPU), Unit Ingatan, serta Unit Peranti Input dan Peranti Output.



Rajah 1.2.1 Komponen utama dalam seni bina komputer

Unit Pemprosesan Pusat (CPU)

Unit Pemprosesan Pusat juga dikenali sebagai pemproses atau CPU yang merupakan “otak” bagi sistem komputer. CPU terdiri daripada komponen-komponen yang berikut:

- (a) Unit Aritmetik dan Logik (ALU)—Melakukan operasi aritmetik dan operasi logik.
- (b) Unit Kawalan—Unit yang mengawal operasi ingatan, pemproses dan peranti input dan output.
- (c) Daftar (*Register*)—Befungsi untuk menyimpan data secara sementara, menghantar arahan yang dicapai daripada ingatan utama sebelum diproses di ALU dan menerima hasil daripada ALU sebelum dihantar ke ingatan utama.

Rajah 1.2.2 ialah contoh pemproses masa kini.



Rajah 1.2.2 Pemproses Intel Core i7

Unit Ingatan

Fungsi utama unit ingatan adalah untuk menyimpan data dan arahan semasa komputer digunakan. Ingatan hanya menyimpan atau memegang data dan arahan secara sementara, iaitu selagi arahan yang merujuknya masih beroperasi. Unit ingatan tidak digunakan untuk menyimpan data secara kekal kerana unit ini bersifat meruap. Ingatan bersifat meruap akan hilang kandungannya jika bekalan kuasa elektrik terputus. Hal ini khusus bagi jenis ingatan utama, iaitu RAM (*Random Access Memory*/Ingatan Capaian Rawak). Ingatan utama terbahagi kepada dua jenis, iaitu RAM dan ROM (*Read Only Memory*/Ingatan Baca Sahaja). Set arahan dalam ROM akan kekal untuk selama-lamanya.

Rajah 1.2.3 ialah contoh ingatan utama RAM. Sel lokasi ingatan boleh digambarkan seperti dalam Jadual 1.2.1.



Rajah 1.2.3 *Dynamic Random Access Memory (DRAM)*

Jadual 1.2.1 Sel-sel lokasi ingatan

Alamat ingatan	Kandungan ingatan
1000 0000	1000 1000
1000 0001	0111 0110
1000 0010	1100 1111
1000 0011	0101 0110
↓	↓
1111 1110	1110 1110
1111 1111	

Peranti Input dan Output

Peranti input dan output ialah komponen yang digunakan untuk memasukkan data dan memaparkan hasil daripada sistem komputer. Rajah 1.2.4 menunjukkan beberapa contoh komponen peranti input dan peranti output. Peranti input menukar data luaran kepada satu bentuk yang boleh difahami dan kemudian diproses oleh komputer. Contohnya, papan kekunci, skrin sentuh, pengimbas, perakam suara dan tetikus.

Peranti output pula memaparkan hasil yang diproses oleh komputer kepada bentuk yang boleh difahami oleh manusia. Contohnya, pencetak, monitor dan pembesar suara.



UJI MINDA

Selain peranti input dan output yang telah dibincangkan, nyatakan tiga peranti input dan output lain yang anda tahu.



Pencetak

Monitor, papan kekunci dan tetikus

Pembesar suara

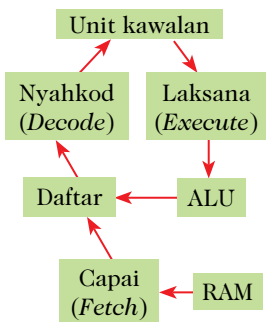
Rajah 1.2.4 Peranti input dan peranti output



The fetch-decode-execute cycle
goo.gl/I9qox8

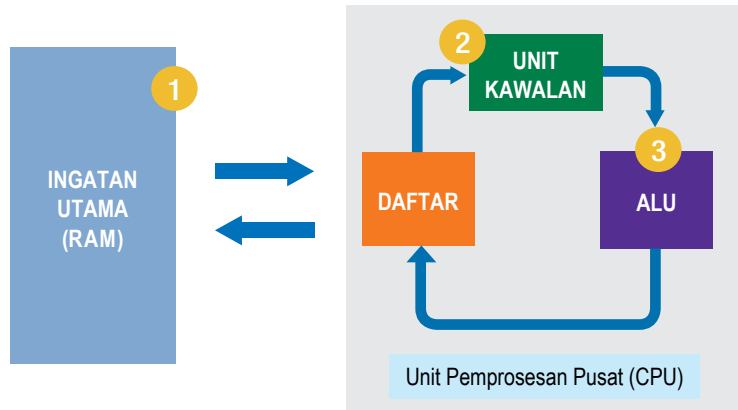
Tahukah Anda

Lihat rajah yang berikut untuk mengingati kitaran capai-nyahkod-laksana dengan lebih mudah.



1.2.1 Kitaran Mesin (Capai-Nyahkod-Laksana)

Tugas utama CPU adalah untuk melaksanakan arahan memproses data. Pemprosesan data dan arahan mengikut kitaran mesin, iaitu kitaran capai-nyahkod-laksana (*FDE: Fetch-Decode-Execute Cycle*). FDE juga dikenali sebagai kitaran arahan (*instruction cycle*). Rajah 1.2.5 menunjukkan proses-proses yang berlaku dalam kitaran mesin di dalam CPU.



Rajah 1.2.5 Proses-proses dalam kitaran mesin (capai-nyahkod-laksana)

Pemproses menghantar isyarat supaya data dan arahan yang dimasukkan melalui peranti input disalin ke dalam ingatan utama. Data dan arahan tersebut diproses melalui tiga proses utama dalam kitaran mesin untuk menghasilkan output. Proses-prosesnya seperti yang berikut:

- 1 Capai (*fetch*)—Unit kawalan memberikan isyarat supaya pemproses mencapai data dan arahan dari ingatan utama dan disimpan dalam daftar sementara menunggu untuk diproses.
- 2 Nyahkod (*decode*)—Unit kawalan menyahkod atau menterjemah arahan bagi menentukan jenis operasi yang akan dilaksanakan oleh ALU.
- 3 Laksana (*execute*)—Pemproses melaksanakan operasi aritmetik dan logik terhadap data seperti yang dinyatakan oleh arahan.

Pemproses akan mengulangi proses capai-nyahkod-laksana sehingga semua data dan arahan selesai diproses. Semasa proses ini diulangi, hasil pemprosesan disimpan sementara di dalam daftar sehingga selesai dan hasil akhir akan dihantar ke ingatan utama untuk dipamerkan melalui peranti output.

Semua perisian komputer dibina daripada set-set arahan, iaitu arahan dikodkan dalam perduaan. Kitar capai-nyahkod-laksana ialah urutan proses-proses yang perlu diikuti oleh pemproses untuk memproses set arahan sehingga selesai.

Tahukah
Anda



AKTIVITI

1

Kerja Kumpulan

Kitaran Capai-Nyahkod-Laksana



Lakukan aktiviti yang berikut.

- 1 Bentuk satu kumpulan yang terdiri daripada beberapa orang ahli.
- 2 Guru akan memberikan sepuluh keping kad yang mengandungi teks sifer untuk tugas yang dilakukan kepada ahli yang pertama. (Nota: Tugas boleh merangkumi kawasan di dalam kelas atau di luar kelas mengikut kreativiti guru.)
- 3 Kemudian, ahli kedua akan mengambil kad pertama daripada ahli yang pertama.
- 4 Ahli yang kedua perlu menyahsulit teks sifer yang diberikan dan kemudian menyerahkannya kepada ahli yang ketiga.
- 5 Ahli yang ketiga perlu melaksanakan tugas daripada teks sifer yang telah dinyahsulitkan.
- 6 Setelah tugas selesai, ulang langkah 3 hingga 5 bagi kad-kad lain.

Berdasarkan aktiviti yang telah dilakukan, jawab soalan-soalan yang berikut.

- 1 Antara langkah-langkah 3 hingga 5, yang manakah menyerupai proses laksana (*execute*) dalam kitar capai-nyahkod-laksana?
- 2 Bina rajah kitar capai-nyahkod-laksana.
- 3 Sekiranya proses nyahsulit tidak dapat dilakukan, adakah kitar capai-nyahkod-laksana masih boleh diteruskan?

Rumusan untuk kitar capai-nyahkod-laksana:

- 1 Pemproses memeriksa kaunter program untuk melihat arahan yang akan dilaksanakan.
- 2 Kaunter program memberikan nilai alamat dalam memori yang terdapat arahan yang seterusnya.
- 3 Pemproses mencapai nilai arahan dari lokasi memori.
- 4 Setelah arahan dicapai, arahan perlu dinyahkod dan dilaksanakan.
- 5 Setelah selesai, pemproses akan kembali semula ke kaunter program untuk mencari arahan seterusnya.
- 6 Kitaran akan berulang sehingga program tamat.

1.2.2 Konsep Asas Seni Bina Komputer (Von Neuman)

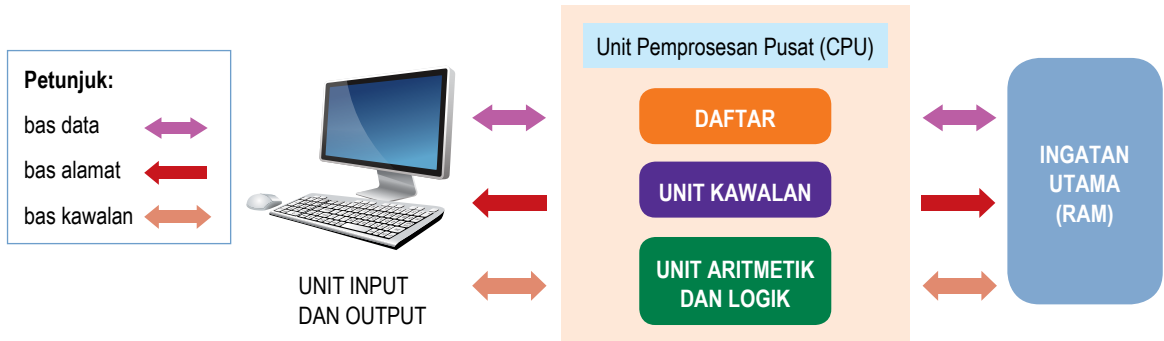
Pada tahun 1945, John Von Neuman mendapat idea mencipta kaedah untuk menyimpan atur cara komputer dalam sistem komputer. Ideanya ialah atur cara dan data disimpan dalam ingatan dan digerakkan dari unit ingatan ke pemproses. Konsep ini dirujuk sebagai seni bina Von Neuman.

Terdapat tiga komponen utama dalam seni bina Von Neuman, iaitu CPU, ingatan, dan peranti input dan output. Komponen-komponen ini dihubungkan dengan sistem bas.



John Von Neuman
goo.gl/JQptJO

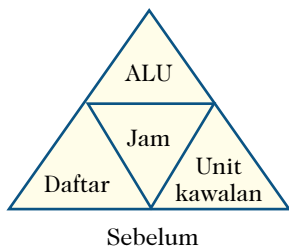
Rajah 1.2.6 menunjukkan seni bina Von Neuman yang komponen-komponennya seperti CPU, ingatan utama dan input/output dihubungkan bersama melalui sistem bas.



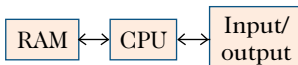
Rajah 1.2.6 Seni bina Von Neuman dan sistem bas

Tahukah Anda

Rajah di bawah menunjukkan empat komponen utama dalam CPU sebelum dan selepas terbinanya CPU Von Neuman.



Sebelum



Selepas

Sistem Bas

Sistem bas terbahagi kepada dua bahagian, iaitu bas dalaman dan bas luaran seperti yang ditunjukkan dalam Rajah 1.2.6.

- Bas dalaman—Menghubungkan komponen-komponen dalam pemproses dan membenarkan komponen tersebut bertukar-tukar maklumat.
- Bas luaran—Menghubungkan pemproses dengan komponen luaran seperti ingatan dan peranti input dan output.

Bas luaran terbahagi kepada tiga jenis, iaitu bas data, bas alamat dan bas kawalan seperti yang ditunjukkan dalam Jadual 1.2.2.

Jadual 1.2.2 Jenis-jenis bas

Bas data	Bas alamat	Bas kawalan
<p>Bas yang digunakan bagi memindahkan data</p> <ul style="list-style-type: none"> antara pemproses, ingatan dan peranti input/output antara daftar, unit kawalan dan ALU 	<p>Bas yang digunakan bagi</p> <ul style="list-style-type: none"> membawa alamat lokasi yang menempatkan data dan arahan dalam ingatan utama mengandungi alamat peranti input/output 	<p>Bas yang mengandungi</p> <ul style="list-style-type: none"> isyarat kawalan yang digunakan bagi mengawal operasi keseluruhan sistem

Unit Pemrosesan Pusat (CPU)

Unit Pemrosesan Pusat (CPU) ialah komponen yang paling penting dalam sistem komputer. Nama lain bagi CPU ialah pemproses. Pemrosesan data dan arahan dilakukan dalam pemproses. Terdapat tiga komponen utama dalam pemproses, iaitu Unit Aritmetik dan Logik (ALU), daftar dan Unit Kawalan.

(a) **Unit Aritmetik dan Logik (ALU)**

Melakukan operasi-operasi aritmetik seperti (tambah, tolak, bahagi, darab, nilai negatif dan sebagainya) dan operasi-operasi logik seperti (DAN, ATAU, TAK, XATAU dan sebagainya). ALU menerima input dari daftar-daftar dalam CPU dan menghasilkan maklumat untuk dihantar ke daftar sebelum disimpan dalam ingatan.

(b) **Daftar**

Komponen yang digunakan bagi menyimpan data dan arahan yang dicapai dari ingatan utama sebelum dihantar ke ALU untuk diproses dan menyimpan hasil pemprosesan sementara sebelum dihantar ke ingatan utama dan dipamerkan melalui peranti output.

Contoh jenis daftar yang terlibat dan fungsinya dalam kitar mesin pemprosesan data.

(i) **PC (Program Counter/Pembilang atur cara)**

Digunakan bagi memegang alamat ingatan utama pada peringkat awal proses kitaran mesin selepas menerima arahan dari CPU. PC akan menyalin alamat tadi ke daftar MAR, nilai PC akan bertambah. Daftar ini akan sentiasa menunjuk atau memegang alamat bagi arahan seterusnya yang akan dicapai.

(ii) **ACC (Accumulator/Daftar Pengumpul)**

Digunakan untuk menyimpan data yang menjadi input dan output semasa operasi ALU.

(iii) **MAR (Memory Address Register/Daftar Alamat Ingatan)**

Daftar ini memegang alamat lokasi maklumat di dalam ingatan utama. Apabila pemproses ingin mencapai data dari ingatan, pemproses akan memberikan isyarat alamat lokasi ingatan yang diperlukan dari PC dan disalin ke dalam MAR.

(iv) **MDR (Memory Data Register/Daftar Data Ingatan)**

Memegang data dan arahan yang dibaca dari ingatan atau yang akan ditulis ke ingatan. Kandungan arahan dalam MDR akan dihantar ke IR, dan kandungan data akan dihantar ke ACC.

(v) **IR (Instruction Register/Daftar Arahan)**

Mengandungi arahan terkini yang akan dilaksanakan. Unit kawalan akan menyahkodkan arahan ini bagi menentukan jenis operasi yang perlu dilaksanakan oleh ALU.

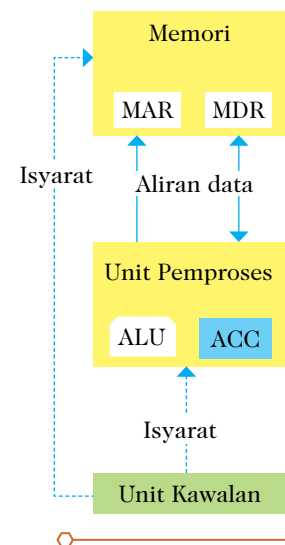
(c) **Unit Kawalan**

Unit kawalan menerima dan menjana isyarat-isyarat bagi mengawal operasi keseluruhan sistem komputer. Input utamanya ialah denyutan isyarat dari sistem jam. Unit kawalan mengandungi penyahkod arahan dan logik pemasaan serta kawalan.

Tahukah Anda



Daftar MAR dan daftar MDR adalah penting untuk komunikasi antara memori dengan unit pemproses. Rajah yang berikut menunjukkan hubungan antara daftar MAR dengan daftar MDR untuk menerangkan komunikasi antara memori dengan unit pemproses.



Sistem Jam (clock system)

Jam merupakan satu peranti cip dalam pemproses yang menjaga denyutan elektronik pada kadar yang tetap bagi menyeragamkan semua operasi komputer. Unit yang digunakan bagi menghitung kadar kelajuan jam dalam pemproses (*clock speed measurement*) ialah hertz. Satu hertz ialah satu denyutan/kitaran per saat.

Jika jam mempunyai kadar 160 MHz, maka ini menunjukkan jam mempunyai frekuensi 160,000,000 denyutan/kitaran sesaat. Pemproses akan menggunakan denyutan ini semasa melaksanakan arahan dalam kitaran mesin. Oleh sebab itu, jika CPU mempunyai kelajuan jam 160 MHz, maka pemproses boleh memproses 160,000,000 kitaran mesin sesaat. Satu arahan memerlukan satu kitaran mesin untuk dilaksanakan. Hal ini bermakna, komputer boleh melaksanakan 160 juta arahan dalam satu saat. Dengan itu, lebih tinggi kelajuan hertz, lebih laju komputer tersebut. Contoh kelajuan CPU adalah seperti yang berikut:

- Intel Core i7 995X 3.6GHz
- Intel Core Duo 2.0GHz
- Intel Core i5 430M 2.27GHz

CONTOH 1

Contoh 1 menunjukkan cara menghitung kelajuan pemproses dalam kiraan kitaran sesaat. Jika diberi 1.3 GHz, berapakah kitaran sesaat?

Penyelesaian:

- 1 MHz → 1,000,000 kitaran sesaat
- 1 GHz = 1,000 MHz
- 1 GHz = 1,000,000,000 kitaran sesaat
- 1.3 GHz = 1,300,000,000 kitaran sesaat



System bus
goo.gl/vg5Z2m



INOVASI SAINS KOMPUTER

The Wharton School of Business telah menjalankan kaji selidik untuk menentukan 30 inovasi komputer dalam tempoh 30 tahun. Imbas QR code yang berikut untuk melihat inovasi tersebut.



goo.gl/Cxxzwp0

AKTIVITI

2

Kerja Kumpulan

Konsep Asas Seni Bina Komputer



- 1 Guru memberikan arahan kepada murid-murid untuk membawa barangan terpakai ke kelas. Contohnya, kertas, botol plastik dan tin minuman.



- 2 Bentuk kumpulan yang terdiri daripada beberapa orang ahli.
- 3 Kemudian, bina sistem komputer Von Neuman yang telah anda pelajari dengan menggunakan barang-barang terpakai tersebut.
- 4 Selepas siap dibina, persembahkan dan bentangkan binaan anda serta sejarah seni bina komputer Von Neuman di dalam kelas.
- 5 Guru perlu memberikan markah kepada setiap kumpulan. Penilaian guru berasaskan aspek pembinaan sistem komputer yang paling kreatif, menarik dan pembentangan yang lancar.

Perbezaan antara Pemproses Linear dengan Pemproses Semasa

Ukuran kelajuan komputer melaksanakan sesuatu tugas bergantung kepada keupayaan pemprosesnya. Komponen pemproses sentiasa dikaji dari segi teknologi pembuatan litar logiknya supaya dapat memproses data dengan kadar kelajuan yang tinggi. Terdapat dua jenis pemprosesan, iaitu pemprosesan linear dan pemprosesan tidak linear yang dilaksanakan dalam pemprosesan semasa.

Pemprosesan Linear

Pemprosesan linear juga dikenali sebagai pemprosesan bersiri, iaitu pemproses memproses data dan arahan secara berjajukan seperti yang digambarkan dalam Rajah 1.2.7. Pemproses memproses data dan arahan mengikut kitaran mesin, iaitu capai(IF)-nyahkod(ID)-laksana(EX) dan langkah ini diulangi sehingga data dan arahan selesai diproses. Pemproses linear hanya sesuai untuk melaksanakan arahan yang ringkas.

Pemprosesan Semasa

Pemprosesan semasa juga dikenali dengan nama pemprosesan tidak linear atau pemprosesan selari. Pemproses ini boleh dikonfigurasi bagi melakukan pelbagai tugas dalam satu masa (*multitasking*). Pemproses semasa mampu melaksanakan beberapa arahan secara serentak dengan menggunakan teknik talian paip (*pipelining*). Teknik ini digunakan sebagai salah satu kaedah untuk mempertingkatkan prestasi pemproses semasa. Teknik ini membenarkan pemprosesan arahan-arahan secara bertindih (*overlapping*) seperti yang digambarkan dalam Rajah 1.2.8.

Perbandingan Pelaksanaan Arahan antara Pemproses Linear dengan Pemproses Semasa

Huraian yang berikut ialah contoh gambaran siri peringkat pemprosesan data dan arahan mengikut kitaran mesin (IF, ID, EX) bagi pemprosesan linear dan pemprosesan semasa. Rajah 1.2.7 menunjukkan pemprosesan arahan secara berjajukan. Diberikan tiga arahan untuk diproses dengan setiap arahan akan diselesaikan mengikut tiga fasa (IF, ID, EX) dalam kitaran mesin. Andaikan setiap fasa diberikan nilai masa pemprosesan yang sama. Dengan itu, bagi menyelesaikan tiga arahan tersebut, jumlah masa yang diperlukan ialah sembilan. Rajah 1.2.8 menunjukkan penggunaan teknik talian paip (*pipelining*) dalam pemprosesan semasa. Teknik ini membahagikan pemprosesan kepada beberapa fasa. Setiap fasa akan menyelesaikan sebahagian arahan dan dilaksanakan secara selari dan serentak. Dalam satu masa, pertandingan fasa boleh berlaku



Ingatan para (*cache*) ialah memori yang kecil dalam CPU. Ingatan para digunakan sebagai tempat simpanan sementara bagi arahan dan data yang CPU akan gunakan semula. Ingatan para mempunyai beberapa tahap, iaitu Tahap 1, Tahap 2 dan Tahap 3.

✓ Tahap 1

Ingatan para merupakan sebahagian daripada cip CPU sendiri. Saiznya terhad antara 8 KB hingga 64 KB.

✓ Tahap 2 dan Tahap 3

Ingatan para yang lebih besar berbanding dengan Tahap 1. Semakin banyak Tahap 2 dan Tahap 3 yang ada, semakin laju arahan komputer boleh dilaksanakan.

Sumber:

<http://www.bbc.co.uk/education/guides/smb9mp3/revision/3>



Pipelining
goo.gl/CL80F3



UJI MINDA

Lihat rajah yang berikut. Berdasarkan pengetahuan anda dalam penggunaan talian paip, warnakan ruang di dalam rajah tersebut bagi pelaksanaan arahan secara talian paip dan secara linear. Selamat mencuba!

Arahan	1	2	3	4
Capai				
Nyahkod				
Laksana				
Jam	1	2	3	4

Arahan	5	6	7	8
Capai				
Nyahkod				
Laksana				
Jam	5	6	7	8

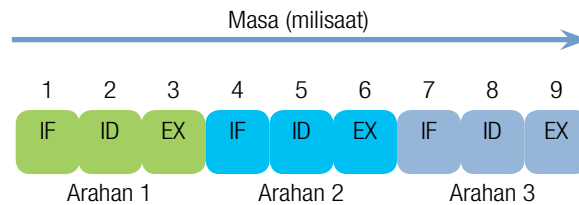
kerana ada fasa yang dilaksanakan secara serentak. Contohnya pada milisaat kedua, dua fasa dilaksanakan secara serentak, iaitu fasa ID bagi arahan pertama dan fasa IF bagi arahan kedua. Dengan itu, masa yang diperlukan untuk memproses tiga arahan sehingga selesai ialah lima milisaat. Masa memproses menjadi lebih singkat. Ini akan meningkatkan prestasi pemrosesan arahan.

Rajah 1.2.7 dan Rajah 1.2.8 menunjukkan perbezaan pelaksanaan arahan bagi pemproses linear dan pemproses semasa.

Petunjuk:

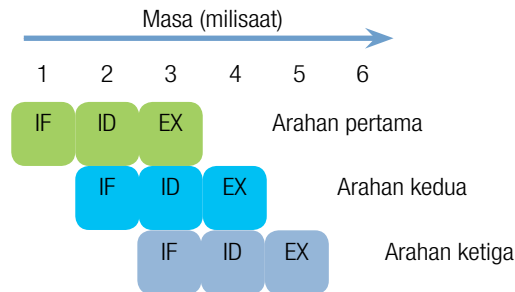
- Kitar 1 untuk arahan pertama
- Kitar 2 untuk arahan kedua
- Kitar 3 untuk arahan ketiga

Bagi menyelesaikan tiga arahan, tiga kitaran diperlukan dengan jumlah masanya ialah sembilan milisaat.



Rajah 1.2.7 Pelaksanaan arahan secara linear

Bagi menyelesaikan tiga arahan, jumlah masanya ialah lima milisaat.



Rajah 1.2.8 Pelaksanaan arahan menggunakan teknik talian paip (*pipelining*)

Teknik ini digunakan dalam pemproses masa ini seperti dalam pemproses Intel Core dan digunakan dalam mikrokomputer atau superkomputer. Jadual 1.2.3 menunjukkan perbezaan antara pemrosesan linear dengan pemrosesan semasa.

Jadual 1.2.3 Perbezaan antara pemprosesan linear dan pemprosesan semasa

Pemprosesan linear	Pemprosesan semasa
<ul style="list-style-type: none"> • Pemproses linear kurang efektif dalam pemprosesan data dan arahan yang panjang dan kompleks. • Pemproses linear hanya boleh melakukan satu tugas pada satu masa, pemproses ini lebih perlahan. 	<ul style="list-style-type: none"> • Lebih efektif bagi pelaksanaan arahan yang panjang dan kompleks. • Lebih efektif kerana arahan-arahan dibahagikan kepada beberapa bahagian dan dilaksanakan secara berasingan. Dengan itu, pemproses boleh melakukan pelbagai tugas dengan cepat pada satu masa.

AKTIVITI

3

Kerja Kumpulan

Teknik Talian Paip



Anda dan rakan anda diberi tugas untuk menggunakan pemprosesan semasa bagi menyelesaikan masalah yang berikut.

Suren, Amar dan Kok Shoon telah membuka sebuah kedai dobi. Setiap hari, mereka akan mengasingkan pakaian yang diterima kepada enam guni. Pakaian di dalam setiap guni perlu dicuci, dikeringkan dan dilipat. Setiap cucian memerlukan masa selama 20 minit, pengeringan selama 30 minit dan lipatan selama 40 minit. Mereka sering kali melakukan tugas ini bermula dari pukul 4 petang sehingga pukul 10 malam. Bagaimanakah Suren, Amar dan Kok Shoon dapat menjimatkan masa mereka untuk melakukan semua tugas tersebut? Pada pukul berapakah semua tugas akan diselesaikan oleh mereka sekiranya alternatif yang lebih bijak digunakan?

1.2.4 Penterjemah Bahasa Pengaturcaraan

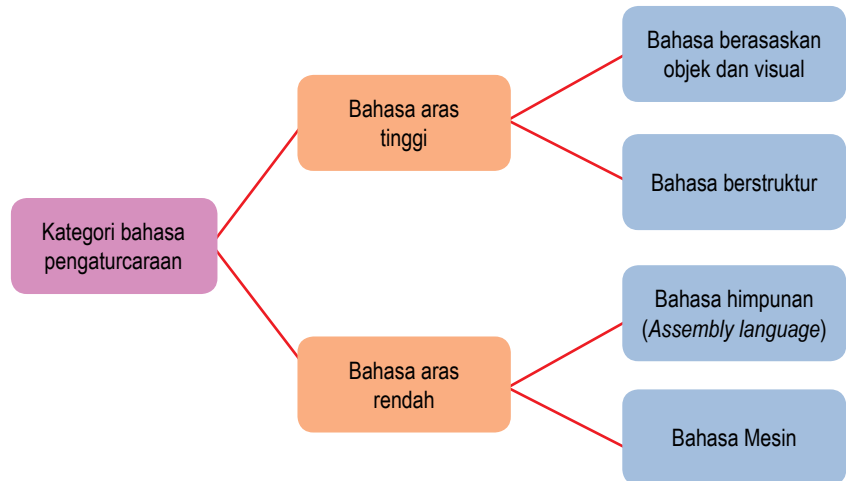
Komputer akan melaksanakan tugas dengan mengikut arahan atau set kumpulan arahan yang ditulis oleh pengatur cara dengan menggunakan pelbagai jenis bahasa pengaturcaraan. Komputer hanya memahami bahasa dalam bentuk kod mesin (digit perduaan 0,1). Rajah 1.2.9 menunjukkan atur cara yang ditulis dalam bentuk bahasa pengaturcaraan yang perlu diterjemahkan kepada bentuk bahasa mesin sebelum difahami oleh komputer.



Rajah 1.2.9 Komunikasi antara manusia dengan komputer

Bahasa Pengaturcaraan

Bahasa pengaturcaraan boleh dikategorikan kepada dua aras seperti yang ditunjukkan dalam Rajah 1.2.10.



Rajah 1.2.10 Kategori bahasa pengaturcaraan

Tahukah Anda

Kebaikan menggunakan bahasa pengaturcaraan aras tinggi adalah seperti yang berikut:

- ✓ Kemudahan (portability) yang lebih baik
- ✓ Mempunyai banyak jenis data dan pengurusan memori
- ✓ Mempunyai sokongan yang baik dari aspek penyelenggaraan perisian
- ✓ Mempunyai sokongan yang lebih baik dari aspek guna semula perisian

Sumber:
<http://mathstat.slu.edu/>

Bahasa Pengaturcaraan Aras Tinggi

Bahasa aras tinggi menggunakan pernyataan yang mudah difahami oleh pengatur cara kerana bahasanya hampir sama dengan bahasa tabii manusia. Bahasa aras tinggi boleh dikategorikan kepada dua jenis, iaitu bahasa pengaturcaraan berasaskan objek dan bahasa berstruktur. Contohnya, *JAVA*, *Visual Basic*, *PHP*, *Python* ialah bahasa berasaskan objek manakala contoh bahasa berstruktur ialah *C*, *PASCAL*, *COBOL* dan *FORTRAN*.

Bahasa pada aras tinggi ini lebih mudah dipelajari dan dengan itu, bahasa ini lebih mudah digunakan untuk membina atur cara. Sebagai pengatur cara, murid tidak perlu mempunyai pengetahuan atau memahami perkakasan dan set arahan dalam bentuk kod perduaan komputer kerana kesemuanya telah dilaksanakan oleh bahasa pengaturcaraan aras tinggi.

CONTOH 2

Contoh 2 menunjukkan arahan dalam atur cara untuk mendapatkan hasil darab bagi dua nombor menggunakan bahasa aras tinggi.

Jumlah = nombor1 \times nombor2

Bahasa Pengaturcaraan Aras Rendah

Bahasa aras rendah ialah pernyataan yang digunakan untuk membina arahan bagi komputer, dalam bentuk simbol yang ringkas, seperti yang ditunjukkan dalam Jadual 1.2.4.

Jadual 1.2.4 Penggunaan simbol dalam bahasa himpunan

Simbol Arahan	Maksud
ADD	Tambah
MUL atau MPY	Darab
SUB	Tolak
DIV	Bahagi
OR, NOT, AND	Fungsi logik dalam <i>Boolean</i>
LOAD	Mencapai data dari ingatan
STORE	Menyimpan hasil ke ingatan

Tidak banyak pengatur cara menulis atur cara dalam bahasa himpunan (*assembly language*) kecuali untuk sebab-sebab tertentu seperti menggunakan perkakasan khas dan melakukan tugas dengan cepat. Pengatur cara perlu memahami seni bina pemproses dan fungsi setiap perkakasan dalam sistem komputer.

CONTOH 3

Contoh 3 menunjukkan arahan mendarab dua nombor yang ditulis menggunakan bahasa himpunan.

Arahan	Maksud
LOAD nombor1	(ACC ← nombor1)
MUL nombor2	(ACC ← ACC × nombor2)
STORE jumlah	(jumlah ← ACC)

Bahasa Mesin

Bahasa mesin ialah bahasa yang sukar difahami dan dimanipulasi. Bentuk pernyataan bahasa mesin bergantung pada jenis komputer. Atur cara dalam bahasa mesin dibina daripada jujukan digit perduaan (01011100).

CONTOH 4

Contoh 4 menunjukkan atur cara yang ditulis dalam arahan kod mesin bagi menambah dua nombor.

		$Y = 12 + 8$
1	0010	00001100
4	0100	00001000
0	0000	00011110
F	1111	00000000 (<i>halt</i>)
Hexadecimal	Kod mesin	(perduaan)

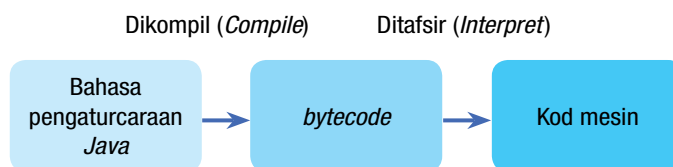
Penterjemah (*translator*) Bahasa Pengaturcaraan

Tugas penterjemah adalah untuk menukarkan arahan dalam bentuk yang difahami manusia kepada bentuk yang difahami mesin, iaitu daripada bahasa pengaturcaraan aras tinggi dan rendah kepada bahasa mesin sebelum atur cara dilaksanakan. Proses penterjemahan bahasa aras tinggi dilakukan sama ada oleh pengkompil (*compilers*) atau pentafsir (*interpreter*). Pengkompil menukarkan seluruh kod sumber atur cara ke kod mesin sebelum dilaksanakan. Pentafsir menukar kod sumber atur cara mengikut baris demi baris dengan satu baris dalam satu masa. Pentafsir adalah lebih interaktif berbanding dengan pengkompil manakala bahasa himpunan diterjemahkan kepada bahasa mesin oleh *Assembler*. Jadual 1.2.5 menunjukkan bahasa pengaturcaraan yang beroperasi di bawah penterjemah masing-masing.

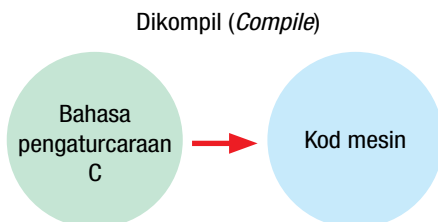
Jadual 1.2.5 Bahasa pengaturcaraan dengan penterjemahnya

Bahasa pengaturcaraan aras tinggi	Pengkompil	Pentafsir
C, C++, COBOL	√	
<i>Python, Perl, PHP, Ruby</i>		√
<i>Java, Visual Basic, C#</i>	√	√

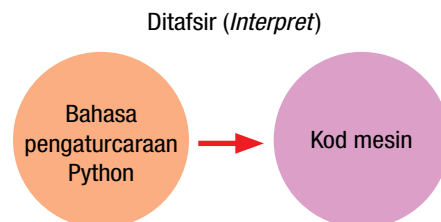
Bahasa pengaturcaraan *Java* ialah bahasa yang melibatkan dua proses semasa terjemahan, iaitu dikompil dan ditafsir sebelum diproses oleh pemproses. Pada peringkat awal atur cara *Java*, pengkompil *javac* menukarkan atur cara ke bentuk yang dipanggil *bytecode*. Proses ini berlaku dalam ingatan JVM. *Bytecode* kemudiannya ditafsirkan pula oleh atur cara lain yang dinamakan *Java*. Atur cara *Java* ini mentafsir *bytecode* baris demi baris dan menukarnya kepada bentuk kod mesin untuk dilaksanakan oleh pemproses. Rajah 1.2.11, Rajah 1.2.12 dan Rajah 1.2.13 menunjukkan proses yang berlaku apabila bahasa pengaturcaraan aras tinggi diterjemahkan kepada kod mesin sebelum diproses.



Rajah 1.2.11 Penggunaan pentafsir (*interpreter*) dan pengkompil (*compiler*)



Rajah 1.2.12 Penggunaan pengkompil (*compiler*)



Rajah 1.2.13 Penggunaan pentafsir (*interpreter*)

CONTOH 5

Contoh 5 menerangkan proses yang berlaku di sebalik setiap arahan yang ditulis dalam bahasa himpunan bagi menyelesaikan persamaan dalam bahasa aras tinggi. Apakah set arahan untuk menyelesaikan persamaan yang berikut?

$$Y = MX + C$$

Penyelesaian:

MOVE Y, M [$Y \leftarrow M$; digerakkan nilai X ke dalam daftar Y]
MUL Y, X [$Y \leftarrow Y * X$; nilai M * nilai X hasil simpan di Y]
ADD Y, C [$Y \leftarrow Y + C$; hasil darab M * nilai X + nilai C]
STORE Y [Simpan hasil akhir Y ke dalam ingatan]

AKTIVITI

4

Kerja Individu

Penterjemahan

Dengan menggunakan arahan dalam bahasa himpunan (*assembly language*), tulis arahan bagi memproses persamaan berikut:

- $Y = A / (B + C)$
- $F = A - B/C$

1.2.5

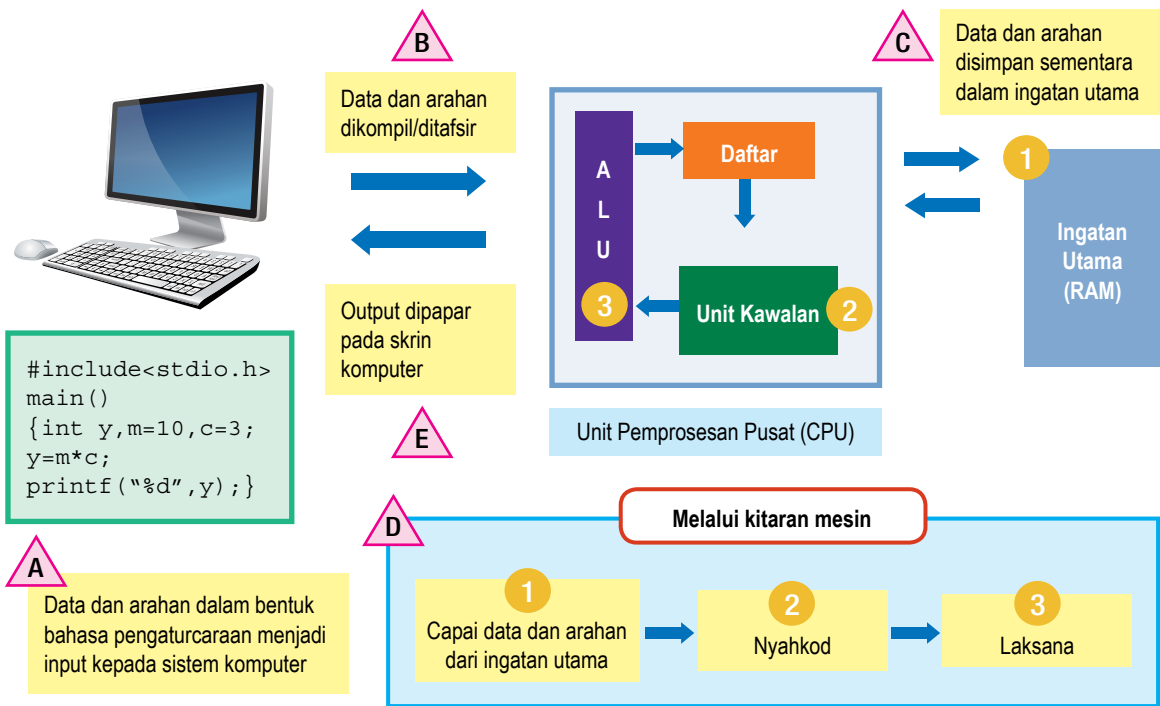
Rumusan Pelaksanaan Atur Cara dalam Sebuah Pemproses

Rajah 1.2.13 merupakan gambaran pelaksanaan pengaturcaraan C yang diinput melalui peranti input dan dikompil dengan penterjemah mengikut jenis pengkompil (*compiler*) yang biasanya dibina bersama dengan bahasa pengaturcaraan itu sendiri. Contohnya, perisian *DEV C++*, *Microsoft Visual C*, dan *Turbo C*. Pengkompil ini menukarkan bahasa pengaturcaraan aras tinggi kepada bahasa mesin dan dihantar ke ingatan untuk disimpan sementara sedang diproses.

Unit kawalan akan memberikan isyarat untuk mencapai data dan memberi arahan dari ingatan utama. Data dan arahan tersebut akan dihantar ke daftar ingatan dalam pemproses melalui bus data. Data dan arahan akan melalui kitaran mesin (*capai-nyahkod-laksana*) di dalam pemproses (CPU) sehingga selesai semua arahan dan seterusnya menyimpan hasil pemprosesan ke ingatan utama melalui bus data untuk dipamerkan di peranti output. Rajah 1.2.14 menunjukkan rumusan pelaksanaan pemprosesan arahan dalam sistem komputer.




Supercomputers
goo.gl/vPDUuA



Rajah 1.2.14 Rumusan pelaksanaan pemprosesan arahan dalam sistem komputer

LATIHAN PENGUKUHAN 1.2

- 1 Nyatakan tiga komponen utama dalam seni bina komputer.
- 2 _____ merupakan komponen ingatan di dalam CPU (*Central Processing Unit*).
- 3 Berikan dua contoh peranti input dan output selain yang dinyatakan dalam buku ini.
- 4 Data yang dicapai (*fetch*) dari ingatan utama dinyahkod di _____.
- 5 Tiga aktiviti asas dalam kitar mesin mengikut turutan ialah _____, _____ dan _____.
- 6 Apakah tujuan penggunaan perisian penterjemah? Berikan contoh perisian penterjemah jenis pengkompil dan pentafsir.
- 7 _____ dan _____ adalah dalam kategori bahasa pengaturcaraan aras rendah.
- 8 Salah satu contoh bahasa pengaturcaraan aras tinggi yang berorientasikan objek (*object-oriented language*) ialah _____.
- 9 Apakah fungsi sistem jam di dalam CPU?
- 10 Berikan perbezaan antara pemproses linear dengan pemproses semasa. 

1.3

STANDARD PEMBELAJARAN

Pada akhir pembelajaran murid dapat

1.3.1 Menerangkan get logik sebagai binaan asas litar bersepadu

1.3.2 Menyenaikan dan melukis simbol get logik berikut:

- (i) Get logik asas (TAK, DAN, ATAU)
- (ii) Kombinasi get logik (TAK DAN, TAK ATAU, XATAU, XTAKATAU)

1.3.3 Membina jadual kebenaran dan ungkapan *Boolean* bagi menerangkan tindakan get logik berikut:

- (i) Get logik asas (TAK, DAN, ATAU)
- (ii) Kombinasi get logik (TAK DAN, TAK ATAU, XATAU, XTAKATAU)

1.3.4 Membina jadual kebenaran bagi satu litar logik gabungan yang mempunyai dua input

1.3.5 Membina ungkapan *Boolean* bagi satu litar logik gabungan yang mempunyai dua input

1.3.6 Menghasilkan litar get logik dan jadual kebenaran berdasarkan pernyataan logik

1.3.7 Menghasilkan litar get logik dengan jadual kebenaran dan ungkapan *Boolean* yang lengkap dalam menyelesaikan masalah

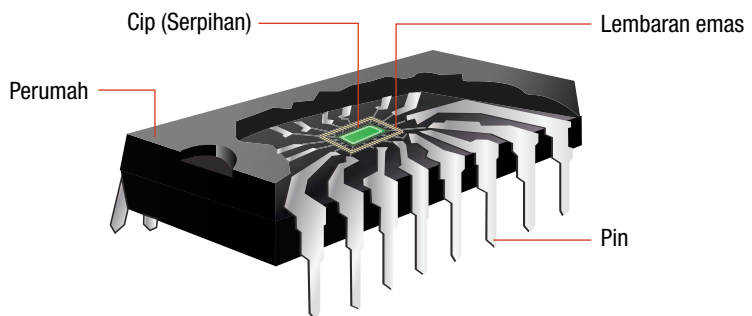
Get Logik

Litar elektronik di dalam komputer dibina dengan beribu-ribu get logik. Get logik merupakan blok binaan asas litar bersepadu. Get logik mempunyai dua input dan satu output. Get logik menerima input dalam bentuk perduaan (0,1) dan juga menghasilkan output dalam bentuk perduaan. Logik perduaan menggunakan dua keadaan, iaitu '0' untuk tiada arus atau 'arus rendah' dan '1' mewakili ada arus atau 'arus tinggi' dalam sistem elektronik digital.

1.3.1 Pengenalan kepada Litar Bersepadu

Peranti elektronik digital kini semakin popular seiring dengan perkembangan dalam teknologi litar bersepadu (*integrated circuit*) atau IC.

Litar bersepadu ialah litar elektronik yang terdiri daripada gabungan komponen elektronik (transistor, perintang, diod dan kapasitor) yang perlu untuk melaksanakan pelbagai fungsi get logik. Get-get ini disambungkan menjadi satu litar yang lengkap dan dikecilkan beribu-ribu kali ganda daripada saiz asal, kemudian ditempatkan pada satu kepingan silikon dan dinamakan cip atau cip mikro (serpihan). Lembaran emas digunakan untuk menyambung serpihan pada pin-pin litar bersepadu. Perumah plastik atau logam biasanya digunakan untuk melindungi serpihan agar tidak rosak. Semakin banyak pin pada litar bersepadu, semakin banyak fungsinya. Rajah 1.3.1 menunjukkan gambaran binaan litar bersepadu.



Rajah 1.3.1 Binaan litar bersepadu

Cip litar bersepadu terbahagi kepada beberapa jenis berdasarkan bilangan get di dalamnya. Jadual 1.3.1 menunjukkan evolusi litar bersepadu.

Tahukah Anda

Selain litar bersepadu (IC), terdapat satu litar lagi yang dikenali sebagai litar diskrit (*discrete circuit*). Kedua-dua litar ini sering kali dibezakan kerana masing-masing terdapat kelebihannya yang tersendiri. Antara kelebihan pada litar bersepadu ialah:

- ✓ Boleh berfungsi pada voltan yang rendah
- ✓ Boleh mengawal kuasa yang terhad
- ✓ Mempunyai saiz yang kecil
- ✓ Murah



Integrated Circuits
goo.gl/07EPvP

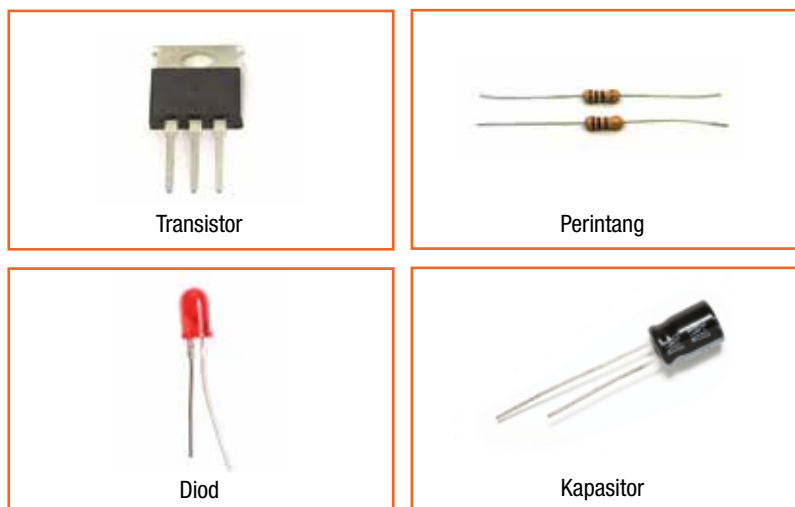
Tahukah Anda

Institute of Electrical and Electronics Engineers (IEEE) / American National Standards Institute (ANSI) merupakan badan piawai yang mencipta simbol get yang digunakan dalam litar logik berdigit.

Jadual 1.3.1 Evolusi litar bersepadu (IC)

Jenis	Tahun	Jumlah get logik per cip	Kegunaan
SSI: Penyepaduan skala kecil	1961 – 1966	<12	Get-get logik (DAN, ATAU, TAK DAN, TAK ATAU)
MSI: Penyepaduan skala pertengahan	1967 – 1971	12 – 99	<i>Flip-flops Adder/counter Multiplexer & Demultiplexer</i>
LSI: Penyepaduan skala besar	1972 – 1980	100 – 9999	Ingatan komputer Pemproses mikro 8 bit
VLSI: Penyepaduan skala sangat besar	1981 – 1990	10,000 – 99,999	Pemproses mikro 16 bit dan 32 bit
ULSI: Penyepaduan skala ultra besar	1991 – 1999	100,000 – 999,999	Pemproses mikro untuk pemprosesan grafik
GSI: Penyepaduan skala giga	2000 – kini	>1,000,000	Pemproses <i>Pentium IV</i>

Penggunaan litar bersepadu ini adalah sangat meluas, misalnya digunakan pada telefon mudah alih, komputer, televisyen dan pelbagai peralatan elektronik. Rajah 1.3.2 menunjukkan jenis-jenis komponen di dalam litar bersepadu.



Rajah 1.3.2 Transistor, perintang, diod dan kapasitor




1.3.2 Simbol Get Logik

Get-get logik ialah elemen asas untuk membentuk suatu litar elektronik bagi sistem digital. IEEE/ANSI yang merupakan badan piawai antarabangsa telah mencipta simbol get logik bagi memudahkan proses pemahaman dan pembelajaran. Simbol get logik boleh dikelaskan kepada dua kumpulan, iaitu get logik asas dan get logik bergabung.

Get Logik Asas (TAK, DAN, ATAU)

Jadual 1.3.2 menunjukkan jenis-jenis get logik asas dan simbol bagi mewakili get logik yang digunakan oleh badan organisasi piawai IEEE/ANSI.


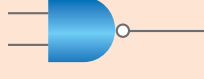


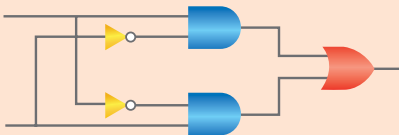

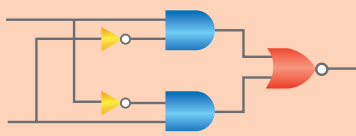

Jadual 1.3.2 Nama dan simbol get logik asas

Nama get logik	Simbol get logik
DAN	
ATAU	
TAK	

Kombinasi Get Logik (TAK DAN, TAK ATAU, XATAU, XTAKATAU)

Jadual 1.3.3 menunjukkan jenis-jenis get logik yang terhasil daripada gabungan-gabungan get logik asas yang dinyatakan dalam Jadual 1.3.2.

Jadual 1.3.3 Nama dan simbol get logik bergabung

Nama get logik	Kombinasi get logik	Simbol get logik
TAK DAN	Get DAN + Get TAK 	
TAK ATAU	Get ATAU + Get TAK 	
XATAU (Disebut sebagai eksklusif ATAU)	Get DAN + Get TAK + Get ATAU 	
XTAKATAU (Disebut sebagai eksklusif TAK-ATAU)	Get DAN + Get TAK + Get TAK ATAU 	

Tahukah Anda

Jumlah input bagi get logik ialah dua dan boleh lebih kecuali get TAK yang mempunyai satu input sahaja.

AKTIVITI

1

Kerja Individu

Get Logik

Lukis simbol get logik yang berikut.

Nama get	Simbol
TAK DAN	
ATAU	
TAK	
TAK ATAU	

1.3.3

Membina Jadual Kebenaran dan Ungkapan *Boolean*

Jadual kebenaran digunakan untuk menyemak output yang dihasilkan daripada get-get logik atau daripada litar logik. Jadual kebenaran juga digunakan untuk membuktikan teorem algebra *Boolean*.

Bagi membina jadual kebenaran, bilangan input perlu dipertimbangkan terlebih dahulu. Input boleh diwakilkan dalam bentuk pemboleh ubah yang boleh dinyatakan dengan huruf-huruf seperti A, B, C, D, X, Y, Z dan sebagainya. Bilangan input akan menentukan kebarangkalian bagi gabungan input yang melalui get. Jadual 1.3.4 menunjukkan kaedah menentukan jumlah kebarangkalian gabungan input yang sebenar kepada litar berdasarkan bilangan pemboleh ubah input yang diberi.



Boolean Algebra Truth Tables
goo.gl/GDvMxO

Jadual 1.3.4 Menentukan jumlah kebarangkalian gabungan input

Bilangan pemboleh ubah input	Jumlah kebarangkalian input
1	$2^1 = 2$
2	$2^2 = 4$
3	$2^3 = 8$
4	$2^4 = 16$

CONTOH 1

Contoh 1 menunjukkan jadual kebenaran bagi SATU pemboleh ubah input (A) dengan dua kebarangkalian nilai perduaan (0 dan 1).

INPUT
A
0
1

CONTOH 2

Contoh 2 menunjukkan jadual kebenaran bagi DUA pemboleh ubah input (A dan B) dengan empat kebarangkalian gabungan nilai input perduaan (0, 1).

INPUT	
A	B
0	0
0	1
1	0
1	1

CONTOH 3

Contoh 3 menunjukkan jadual kebenaran bagi TIGA pemboleh ubah input (A, B dan C) dengan lapan kebarangkalian gabungan nilai input perduaan (0, 1).

INPUT		
A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1


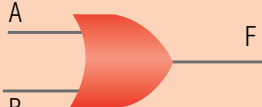
CONTOH 4

Contoh 4 menunjukkan jadual kebenaran bagi EMPAT pemboleh ubah input (A, B, C dan D) dengan 16 kebarangkalian gabungan nilai input perduaan (0, 1).


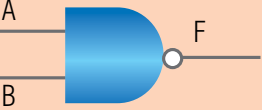
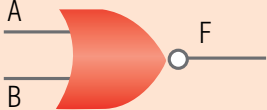
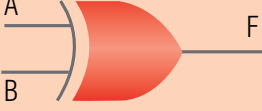

INPUT			
A	B	C	D
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

Jadual 1.3.5 menunjukkan rumusan get-get logik berserta dengan ungkapan *Boolean* dan jadual kebenaran.

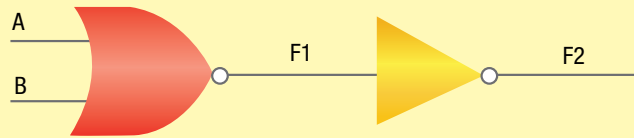
Jadual 1.3.5 Rumusan get-get logik berserta ungkapan *Boolean* dan jadual kebenaran

Nama get	Simbol get	Ungkapan <i>Boolean</i>	Jadual Kebenaran																		
DAN		$F = A \cdot B$ Dibaca: A DAN B $A \times B$	<table border="1"> <thead> <tr> <th colspan="2">INPUT</th> <th>OUTPUT</th> </tr> <tr> <th>A</th> <th>B</th> <th>F</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	INPUT		OUTPUT	A	B	F	0	0	0	0	1	0	1	0	0	1	1	1
INPUT		OUTPUT																			
A	B	F																			
0	0	0																			
0	1	0																			
1	0	0																			
1	1	1																			
ATAU		$F = A + B$ Dibaca: A ATAU B	<table border="1"> <thead> <tr> <th colspan="2">INPUT</th> <th>OUTPUT</th> </tr> <tr> <th>A</th> <th>B</th> <th>F</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	INPUT		OUTPUT	A	B	F	0	0	0	0	1	1	1	0	1	1	1	1
INPUT		OUTPUT																			
A	B	F																			
0	0	0																			
0	1	1																			
1	0	1																			
1	1	1																			



Nama get	Simbol get	Ungkapan <i>Boolean</i>	Jadual Kebenaran																								
TAK (penyongsang)		$F = \bar{A}$ F mempunyai nilai yang bertentangan dengan A	<table border="1"> <thead> <tr> <th colspan="2">INPUT</th> <th colspan="2">OUTPUT</th> </tr> <tr> <th>A</th> <th></th> <th colspan="2">$F = \bar{A}$</th> </tr> </thead> <tbody> <tr> <td>0</td> <td></td> <td colspan="2">1</td> </tr> <tr> <td>1</td> <td></td> <td colspan="2">0</td> </tr> </tbody> </table>	INPUT		OUTPUT		A		$F = \bar{A}$		0		1		1		0									
INPUT		OUTPUT																									
A		$F = \bar{A}$																									
0		1																									
1		0																									
TAK DAN		$F = \overline{A \cdot B}$ Dibaca: A DAN B bar	<table border="1"> <thead> <tr> <th colspan="2">INPUT</th> <th colspan="2">OUTPUT</th> </tr> <tr> <th>A</th> <th>B</th> <th>$A \cdot B$</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	INPUT		OUTPUT		A	B	$A \cdot B$	F	0	0	0	1	0	1	0	1	1	0	0	1	1	1	1	0
INPUT		OUTPUT																									
A	B	$A \cdot B$	F																								
0	0	0	1																								
0	1	0	1																								
1	0	0	1																								
1	1	1	0																								
TAK ATAU		$F = \overline{A + B}$ Dibaca: A ATAU B bar	<table border="1"> <thead> <tr> <th colspan="2">INPUT</th> <th colspan="2">OUTPUT</th> </tr> <tr> <th>A</th> <th>B</th> <th>$A + B$</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	INPUT		OUTPUT		A	B	$A + B$	F	0	0	0	1	0	1	1	0	1	0	1	0	1	1	1	0
INPUT		OUTPUT																									
A	B	$A + B$	F																								
0	0	0	1																								
0	1	1	0																								
1	0	1	0																								
1	1	1	0																								
XATAU Dibaca eksklusif ATAU		$F = A \oplus B$ Jika kedua-dua input bernilai sama, maka akan dihasilkan output '0'	<table border="1"> <thead> <tr> <th colspan="2">INPUT</th> <th colspan="2">OUTPUT</th> </tr> <tr> <th>A</th> <th>B</th> <th colspan="2">$F = A \oplus B$</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td colspan="2">0</td> </tr> <tr> <td>0</td> <td>1</td> <td colspan="2">1</td> </tr> <tr> <td>1</td> <td>0</td> <td colspan="2">1</td> </tr> <tr> <td>1</td> <td>1</td> <td colspan="2">0</td> </tr> </tbody> </table>	INPUT		OUTPUT		A	B	$F = A \oplus B$		0	0	0		0	1	1		1	0	1		1	1	0	
INPUT		OUTPUT																									
A	B	$F = A \oplus B$																									
0	0	0																									
0	1	1																									
1	0	1																									
1	1	0																									
XTAKATAU Dibaca eksklusif TAK ATAU		$F = \overline{A \oplus B}$ Output dari XATAU diterbalikkan kerana tanda bar	<table border="1"> <thead> <tr> <th colspan="2">INPUT</th> <th colspan="2">OUTPUT</th> </tr> <tr> <th>A</th> <th>B</th> <th>$A \oplus B$</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>1</td> </tr> </tbody> </table>	INPUT		OUTPUT		A	B	$A \oplus B$	F	0	0	0	1	0	1	1	0	1	0	1	0	1	1	0	1
INPUT		OUTPUT																									
A	B	$A \oplus B$	F																								
0	0	0	1																								
0	1	1	0																								
1	0	1	0																								
1	1	0	1																								

Rajah yang berikut menunjukkan gabungan beberapa jenis get logik.



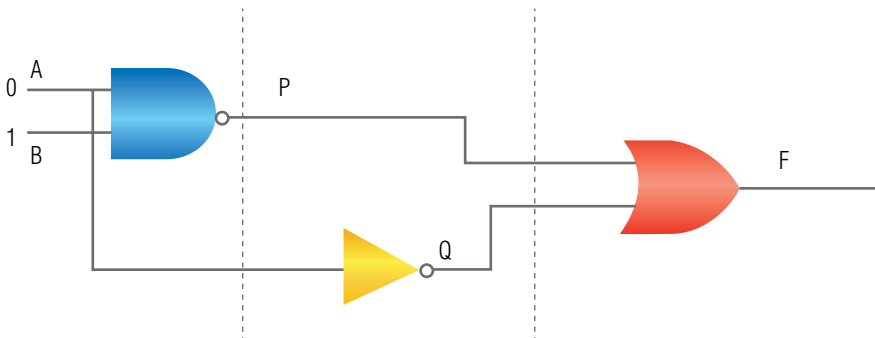
- (a) Tentukan bilangan get asas yang terlibat.
- (b) Lukis semula simbol get asas yang dinyatakan di (a).
- (c) Tulis ungkapan *Boolean* untuk mewakili output F1 dan F2.

1.3.4 Membina Jadual Kebenaran bagi Litar Get Logik Gabungan Dua Input

Dalam standard pembelajaran 1.3.3, murid telah mempelajari cara membina jadual kebenaran bagi get-get logik asas dan get-get logik bergabung. Dalam standard pembelajaran 1.3.4, pembelajaran akan diteruskan dengan cara membina jadual kebenaran bagi sebuah litar logik yang menggabungkan pelbagai jenis get.

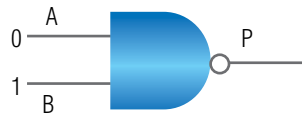
CONTOH 5

Contoh 5 menunjukkan cara membina jadual kebenaran bagi sebuah litar get logik gabungan dua input. Diberikan litar logik dengan dua input, iaitu A dan B seperti dalam rajah yang berikut.



Langkah 1:

Katakan nilai input pada A ialah '0' dan B ialah '1'.



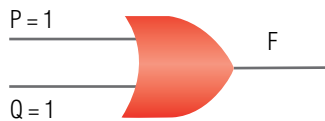
Langkah 2:

Selepas melalui get TAK DAN, iaitu $\overline{1 \times 0} = \overline{0}$ (0 dibarkan) akan menghasilkan 1. Nilai P = 1.



Langkah 3:

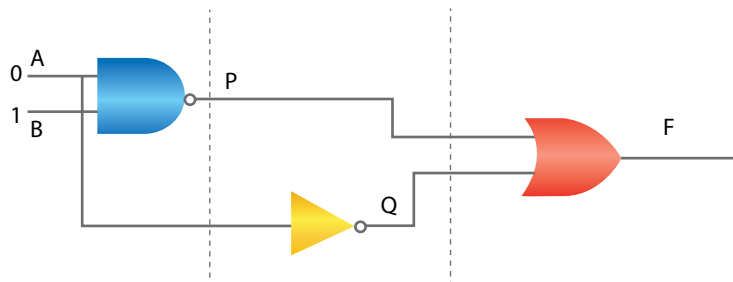
Input A = 0 melalui get TAK akan menghasilkan nilai songsang, iaitu Q = 1.



Langkah 4:

Input daripada P = 1 dan Q = 1 melalui get ATAU menjadikan $F = P + Q = 1 + 1$, sifat get ATAU jika salah satu input melaluinya bernilai 1 atau kedua-dua input bernilai 1, maka output = 1 akan dihasilkan, dengan itu output di F = 1.

Rumusnya, jadual kebenaran bagi litar logik dua pemboleh ubah input A dan B menghasilkan $2^2 = 4$ kebarangkalian gabungan nilai input seperti yang ditunjukkan dalam Jadual 1.3.6, iaitu jadual kebenaran.



Jadual 1.3.6 Jadual kebenaran

INPUT		P Get TAK DAN	Q Get TAK	OUTPUT Get ATAU
A	B	$\overline{A \cdot B}$	\overline{A}	$F = \overline{AB} + \overline{A}$
0	0	1	1	1
0	1	1	1	1
1	0	1	0	1
1	1	0	0	0

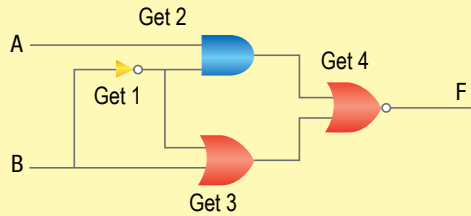
AKTIVITI

3

Kerja Individu

Get Logik

Rajah yang berikut menunjukkan sebuah litar logik yang terdiri daripada get-get logik seperti yang ditunjukkan dalam Jadual 1.3.5.



Jawab soalan-soalan yang berikut berdasarkan litar logik yang ditunjukkan.

- Namakan get-get yang bertanda Get 1, Get 2, Get 3 dan Get 4.
- Tunjukkan langkah-langkah untuk memperoleh output setiap get yang dinyatakan di (a).
- Lengkapkan jadual kebenaran yang berikut dengan nilai yang dihasilkan oleh litar logik yang ditunjukkan.

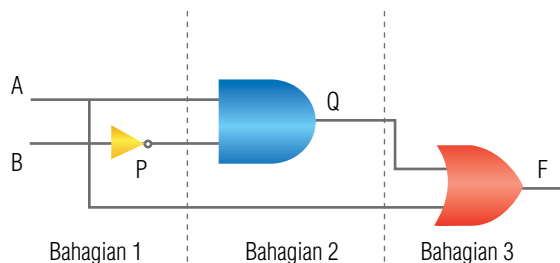
Panduan:

- Namakan Get 1, Get 2, Get 3 dan Get 4.
- Tulis bentuk ungkapan *Boolean* selepas melalui setiap get yang dinyatakan di (i).

INPUT		Get 1	Get 2	Get 3	OUTPUT/Get 4
A	B				F =
0	0				
0	1				
1	0				
1	1				

1.3.5 Membina ungkapan *Boolean* bagi Litar Get Logik Gabungan Dua Input

Topik ini akan menunjukkan cara membina ungkapan *Boolean* bagi litar logik gabungan yang mempunyai dua input seperti yang ditunjukkan pada Rajah 1.3.3.



Rajah 1.3.3 Litar logik dua input

Pemboleh ubah yang mewakili input bagi litar logik ini ialah A dan B. Langkah-langkah yang berikut perlu dilakukan bagi menentukan bentuk output pada setiap keluaran pada P, Q dan akhir sekali output kepada litar, iaitu F.

Murid perlu membahagikan litar kepada tiga bahagian berdasarkan get logik yang terlibat.

Bahagian 1: Melibatkan get logik TAK. Input bagi get ini hanya satu, iaitu B. Output yang dihasilkan oleh get TAK pada P ialah \bar{B} .

Bahagian 2: A dan P merupakan input bagi get DAN, Get ini menghasilkan output, iaitu $Q = A \text{ DAN } P$ atau ditulis sebagai $Q = A.P$, gantikan P dengan \bar{B} , iaitu hasil dari bahagian 1 $\rightarrow Q = A. \bar{B}$, juga boleh disebut A darab B bar.

Bahagian 3: Q dan A merupakan input kepada get ATAU. Get ini akan menghasilkan output, iaitu $F = Q + A$, dibaca Q ATAU A, Q diganti dengan hasil dari bahagian 2 $\rightarrow F = A. \bar{B} + A$. Ungkapan *Boolean* bagi litar logik di atas boleh ditulis seperti berikut:

$$F = A. \bar{B} + A$$

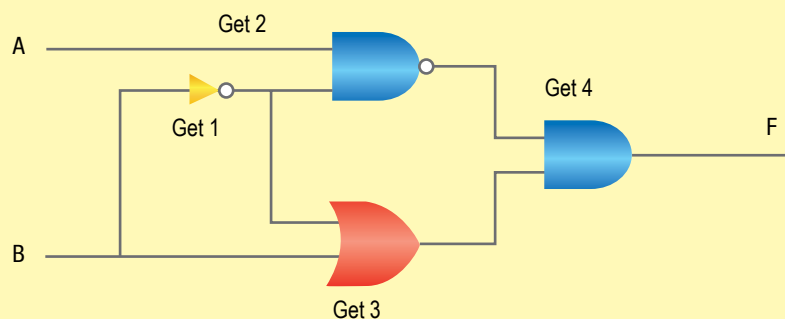
AKTIVITI

4

Kerja Individu

Get Logik

Rajah di bawah menunjukkan satu litar logik, iaitu gabungan dengan dua input, iaitu A dan B.



Tunjukkan cara mendapatkan ungkapan di setiap get yang dilalui. Tulis ungkapan *Boolean* di F, yang merupakan hasil akhir litar logik di atas.

Membina Litar Get Logik dan Jadual Kebenaran

Murid telah mempelajari konsep litar get logik, iaitu cara membina jadual kebenaran dan cara membina ungkapan *Boolean*, kini murid akan mempelajari cara membina litar get logik jika diberi pernyataan logik atau ungkapan *Boolean*. Contoh yang berikut menunjukkan langkah-langkah yang perlu dilakukan sekiranya murid ingin membina litar logik berdasarkan pernyataan logik atau ungkapan *Boolean*.

Diberikan pernyataan logik atau ungkapan *Boolean* seperti yang berikut:

$$F = \bar{A}\bar{B} + AB + A\bar{B} \rightarrow (\text{bentuk ungkapan Boolean}) \text{ atau}$$

Nilai $F = 1$ jika $(A = 0 \text{ DAN } B = 0)$ ATAU $(A = 1 \text{ DAN } B = 1)$ ATAU $(A = 1 \text{ DAN } B = 0)$ \rightarrow (bentuk Pernyataan Logik)

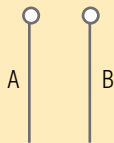
Bagi membina litar get logik, anda perlu membahagikan ungkapan kepada tiga bahagian. Langkah-langkah yang berikut perlu dilakukan:

Langkah 1:

Kira bilangan pemboleh ubah atau input yang terlibat.

Penyelesaian:

A dan B ialah dua pemboleh ubah input kepada litar.

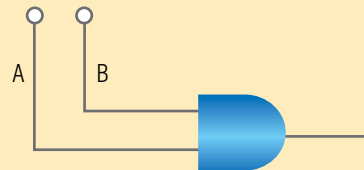


Langkah 3:

Bagi ungkapan AB , tentukan get yang terlibat.

Penyelesaian:

Satu get sahaja terlibat, iaitu get DAN bagi menghubungkan $(A \text{ DAN } B)$.

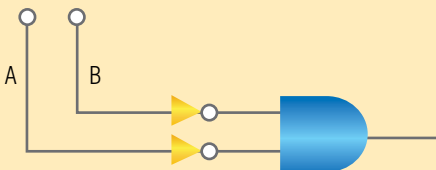


Langkah 2:

Bagi ungkapan $\bar{A}\bar{B}$, tentukan get yang terlibat.

Penyelesaian:

Tiga get yang terlibat, iaitu dua get TAK bagi menghasilkan \bar{A} , \bar{B} dan satu get DAN bagi menghubungkan $(\bar{A} \text{ DAN } \bar{B})$.

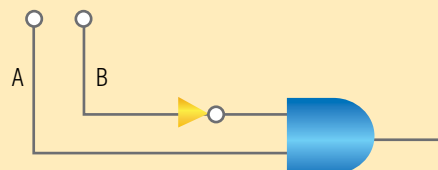


Langkah 4:

Untuk ungkapan $A\bar{B}$, tentukan get yang terlibat.

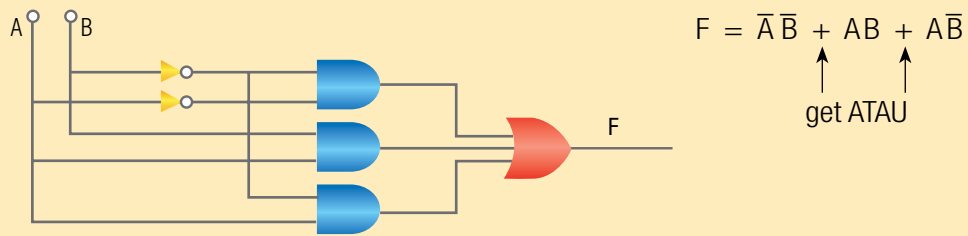
Penyelesaian:

Dua get yang terlibat, iaitu satu get TAK bagi menukar input B ke \bar{B} dan satu get DAN bagi menghubungkan $(A \text{ DAN } \bar{B})$.



Langkah 5:

Ketiga-tiga ungkapan, iaitu dari langkah 2, 3 dan 4 dihubungkan dengan get ATAU. Gabungkan ketiga-tiga rajah. Litar yang berikut terhasil daripada ungkapan *Boolean* yang diberikan.



Seterusnya, bina jadual kebenaran bagi litar yang diperolehi. Jadikan setiap pemboleh ubah input dan setiap pernyataan logik yang terhasil daripada setiap get sebagai nama bagi medan jadual kebenaran. Jadual 1.3.7 menunjukkan kebenaran bagi litar yang dihasilkan.

Jadual 1.3.7 Jadual kebenaran

A	B	\bar{A}	\bar{B}	$\bar{A}\bar{B}$	AB	$A\bar{B}$	F
0	0	1	1	1	0	0	1
0	1	1	0	0	0	0	0
1	0	0	1	0	0	1	1
1	1	0	0	0	1	0	1

Input
Nilai yang berubah selepas melalui get-get
Output

AKTIVITI

5

Kerja Individu

Litar Get Logik dan Jadual Kebenaran

Lukis rajah litar logik dan bina jadual kebenaran bagi mewakili ungkapan *Boolean* atau pernyataan logik yang berikut:

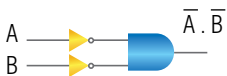
- Nilai output $F = 1$, jika (nilai input $A = 0$ ATAU nilai input bagi $B = 0$) ATAU (nilai input $A = 1$ DAN nilai $B = 0$).
- $Q = (\bar{A} \cdot \bar{B}) + (\bar{A} \cdot B)$

1.3.7 Penghasilan Litar Get Logik, Jadual Kebenaran dan Ungkapan *Boolean* untuk Menyelesaikan Masalah

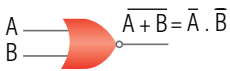
Sebagai rumusnya, murid akan mempelajari cara pakar kejuruteraan komputer membina litar get logik bagi peranti litar bersepadu yang dimuatkan di dalam cip pemproses. Contoh 6 menunjukkan penggunaan litar get logik untuk menyelesaikan masalah.

Tahukah Anda

Persamaan atau ungkapan *Boolean* digunakan bagi memudahkan proses pengubahsuaian saiz dan bentuk litar jika perlu, iaitu dengan menggunakan penggantian get-get. Sebagai contoh, jika litar menggunakan get-get asas DAN, TAK dan ATAU boleh digantikan dengan get-get TAK DAN atau TAK ATAU tetapi menghasilkan output yang sama.



Setara dengan



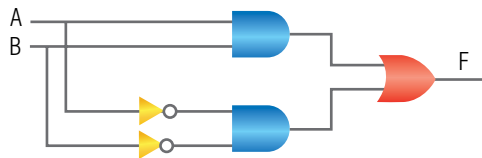
CONTOH 6

Contoh 6 menerangkan langkah-langkah yang perlu dilakukan bagi membina litar logik sistem penggera. Sistem penggera dibina menggunakan litar logik dua input, iaitu A dan B. Sistem akan mengeluarkan isyarat bunyi, F, jika input bagi A mewakili ON dan input B mewakili ON, atau jika input A mewakili OFF dan input B mewakili OFF. Berdasarkan situasi yang diberikan, murid perlu menulis ungkapan logik bagi mewakili senario yang diberikan. Langkah-langkah yang berikut perlu diikuti oleh murid.

Langkah 1 : Tulis ungkapan logik bagi mewakili senario yang diberikan.
 $F = 1$, jika $(A = 1 \text{ DAN } B = 1)$ ATAU $(A = 0 \text{ DAN } B = 0)$

Langkah 2 : Tulis semula dalam bentuk ungkapan *Boolean*.
 $F = A.B + \bar{A}.\bar{B}$

Langkah 3 : Lukis rajah litar get logik bagi mewakili ungkapan *Boolean* di langkah 2.



Langkah 4 : Bina jadual kebenaran bagi dua pemboleh ubah input.

Jadual kebenaran

Input		Peralihan		Output
A	B	$A.B$	$\bar{A}.\bar{B}$	$F = A.B + \bar{A}.\bar{B}$
0	0	0	1	1 (siren berbunyi)
0	1	0	0	0
1	0	0	0	0
1	1	1	0	1 (siren berbunyi)

AKTIVITI 6

Kerja Individu

Litar Get Logik, Jadual Kebenaran dan Ungkapan *Boolean* 

Sistem lampu isyarat menggunakan tiga input pada litar logiknya. Jadual berikut menunjukkan syarat kepada litar bagi menghasilkan output kepada lampu isyarat.

Output Lampu Isyarat	Syarat
$F = 1$	Jika A = ON dan B = OFF dan C = OFF atau Jika A = OFF dan B = ON dan C = OFF atau Jika A = OFF dan B = OFF dan C = ON

Jawab semua soalan yang berikut:

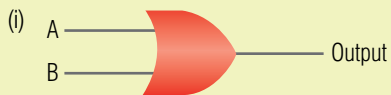
- 1 Tulis ungkapan *Boolean* bagi pernyataan logik yang diberikan dalam jadual di atas.
- 2 Bina litar get logik bagi mewakili ungkapan *Boolean* yang dihasilkan.
- 3 Bina jadual kebenaran bagi menguji kebenaran output litar get logik dalam soalan (2).

LATIHAN PENGUKUHAN 1.3

1 Berdasarkan rajah get logik yang diberikan:

(a) Namakan get-get di (b)(i) hingga (b)(vi).

(b) Lengkapkan jadual kebenaran yang disediakan bagi setiap get yang berikut.



A	B	Output
0	0	
0	1	
1	0	
1	1	



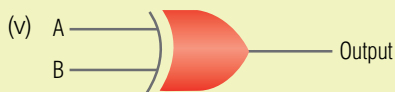
A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	



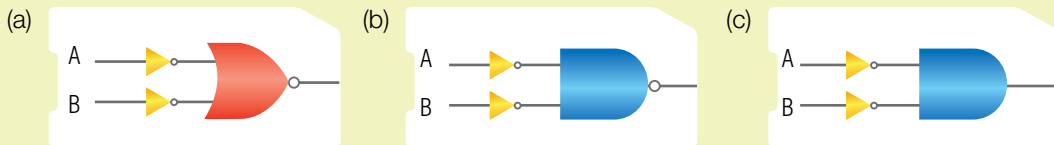
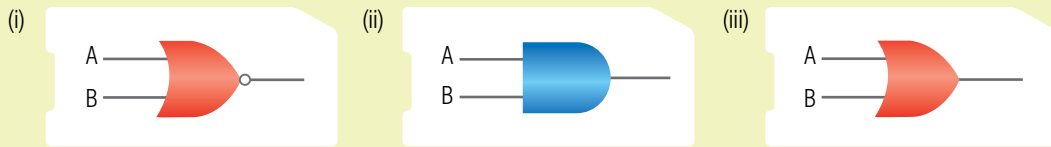
A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	

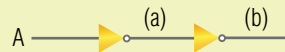
2 Berdasarkan rajah get logik yang diberikan, padankan get logik pada bahagian A yang outputnya setara dengan get logik pada bahagian B.

Bahagian A



Bahagian B

3 Rajah yang berikut menunjukkan sambungan dua get TAK. Nyatakan keluaran di (a) dan (b) jika input ialah A.

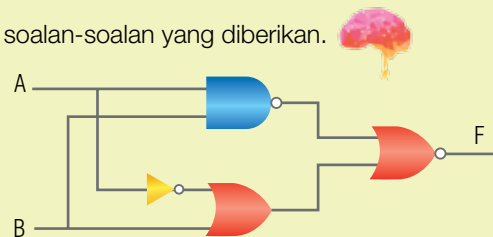


4 Get XATAU (eksklusif ATAU) hanya membenarkan gabungan kedua-dua input _____ bagi mendapatkan keluaran 0 (OFF).

5 Litar bersepadu (IC) ialah litar elektronik yang terdiri daripada gabungan komponen-komponen elektronik _____, _____, _____ dan _____ yang perlu untuk melaksanakan pelbagai fungsi get logik.

6 Berdasarkan rajah litar get logik yang berikut, jawab soalan-soalan yang diberikan.

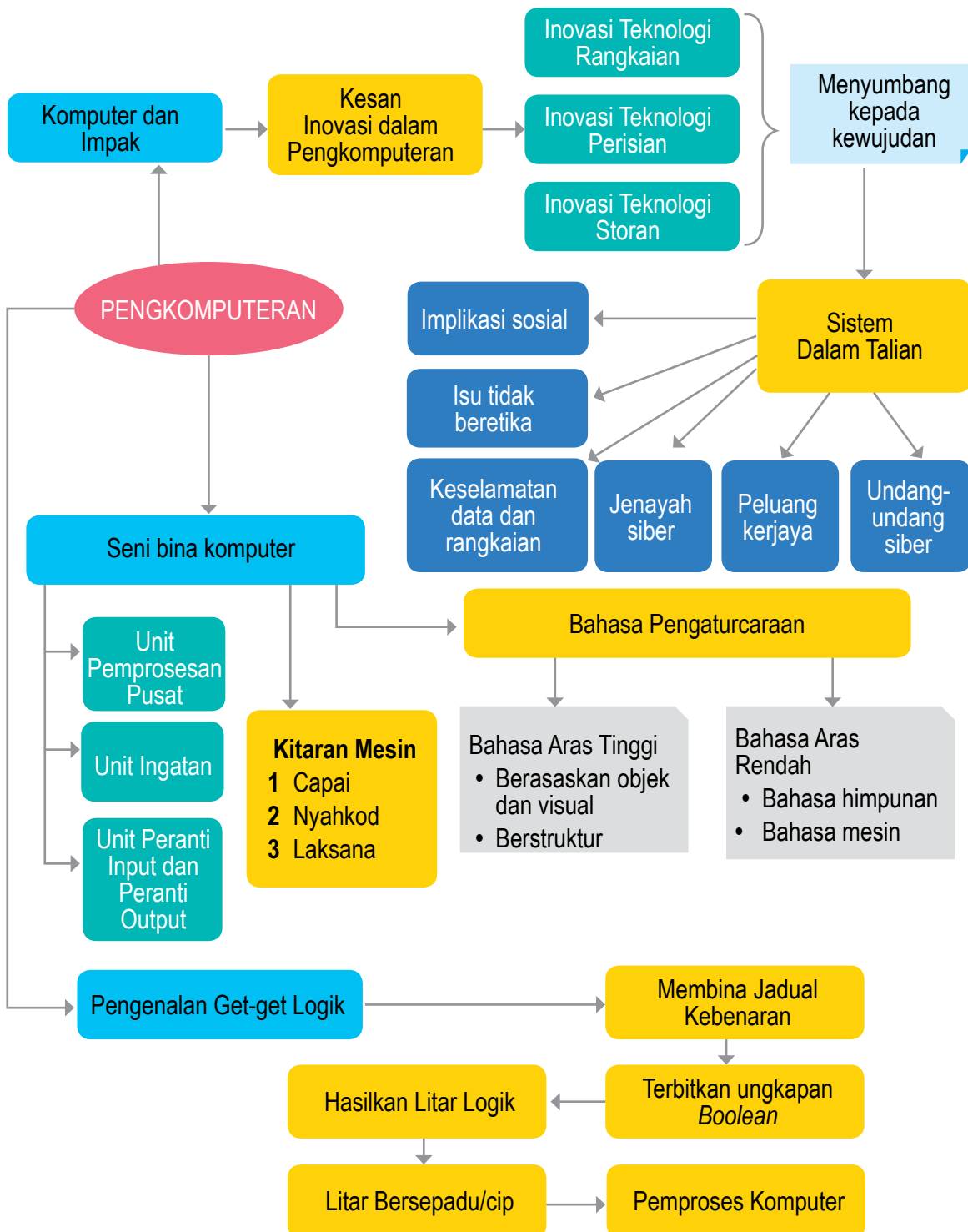
(a) Nyatakan ungkapan yang dihasilkan oleh setiap get.
Tulis jawapan anda pada bahagian (i), (ii), (iii) dan (iv).



(b) Nyatakan nilai sama ada 0 atau 1 bagi setiap output get yang dinyatakan dalam jadual kebenaran yang berikut.

Istilah		Get 1 (TAK)	Get 2 (ATAU)	Get 3 (TAK DAN)	Get 4 (TAK ATAU)
A	B	(i)	(ii)	(iii)	F=(iv)
0	0				
0	1				
1	0				
1	1				

IMBAS SEMULA



LATIHAN PENGAYAAN

1

Bahagian A

- 1 Tujuan perisian penterjemah adalah untuk menukarkan kod atur cara bahasa _____ kepada kod _____ yang difahami oleh komputer sebelum kod atur cara tersebut diproses oleh pemproses.
- 2 _____ ialah ingatan yang paling laju dalam hierarki ingatan yang merupakan salah satu komponen di dalam pemproses (CPU).
- 3 Kegiatan cetak rompak dan penggadam merupakan kesalahan yang boleh didakwa mengikut Akta _____.
- 4 Get logik _____ menghasilkan output 0 apabila kedua-dua inputnya adalah sama.
- 5 _____ merupakan salah satu komponen dalam pemproses (CPU) yang berfungsi untuk menyahkod arahan bagi menentukan jenis operasi yang akan dilaksanakan oleh Unit Aritmetik dan Logik (ALU).
- 6 Arahan-arahan seperti MOVE atau DIV merupakan arahan dalam bahasa _____.
- 7 Di Malaysia, dua badan kerajaan yang memantau, mengekang penyebaran dan menggubal akta jenayah siber ialah _____ dan _____.
- 8 Jadual yang berikut menunjukkan jadual kebenaran bagi get XTAKATAU (eksklusif-TAK ATAU) yang nilai bagi w, x, y dan z ialah _____, _____, _____ dan _____.

Input		Output
0	0	w
0	1	x
1	0	y
1	1	z

- 9 Komponen-komponen di dalam komputer dihubungkan dengan talian elektronik yang disebut sebagai bus. Nyatakan fungsi bagi bus yang berikut:
(i) Bus data
(ii) Bus alamat
(iii) Bus kawalan

Bahagian B

- 1 Pemproses melaksanakan pemprosesan data dan arahan mengikut langkah-langkah dalam kitaran mesin. Terdapat tiga langkah utama dalam kitaran mesin yang akan diulangi oleh pemproses sehingga data dan arahan tersebut selesai diproses dan menghantar output ke ingatan utama. Dengan bantuan rajah kitaran mesin, terangkan cara data dan arahan diproses secara ringkas.

2 Terangkan kesan inovasi dalam pengkomputeran yang mempengaruhi kewujudan *Internet of Things* secara ringkas.



3 Berikan **tiga** aktiviti yang tidak beretika yang mungkin dilakukan oleh murid semasa menggunakan komputer.

4 Mesej yang berikut akan dinyahkodkan:

Plaintext:

- NYAHKOD DATA UNTUK KESELAMATAN

Selepas proses penyulitan (*encryption*) dengan anjakan, ($n = 3$), tulis semula bentuk perwakilan mesej yang dinyahsulitkan (*decryption*).



Teks sifer:

5 Nyatakan **tiga** komponen pemproses dan terangkan fungsi setiap komponen tersebut.

6 Lukis simbol get dan bina jadual kebenaran dengan dua input bagi jenis get seperti yang berikut.

(a) Get TAK ATAU

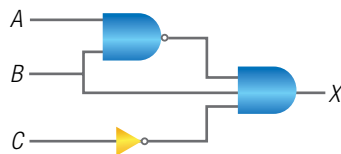
(b) Get eksklusif ATAU

7 Bina jadual kebenaran bagi ungkapan *Boolean* berikut.

$$Q = ABC + (\overline{ABC})$$



8 Tulis ungkapan *Boolean* bagi output yang dihasilkan oleh litar berikut.



9 Lukis rajah litar logik bagi mewakili persamaan *Boolean* berikut.

$$Q = ABC + A(\overline{B} + \overline{C})$$



Anda boleh mengakses kod QR yang ditunjukkan untuk mendapatkan Latih Tubi Bab 1.



Sainskomptg5.com/bab1/LTB1



Fikirkan...

- Jika terdapat ribuan orang murid di sekolah anda, dapatkah anda bayangkan bagaimana untuk mencari rekod seorang murid dengan pantas?
- Apakah cara yang mudah dan pantas untuk menyimpan dan mencapai rekod?

PANGKALAN DATA LANJUTAN

Dalam bab ini, anda akan mempelajari:

- Rajah Hubungan Entiti (*Entity Relationship Diagram - ERD*)
- Model Hubungan Data
- SQL: Membina Pangkalan Data
- SQL: Memanipulasi Data
- SQL: Mencapai Data
- Fungsi Agregat
- Membina Pangkalan Data

Penggunaan teknologi maklumat pada masa ini semakin meluas dan penting dalam kehidupan seharian manusia. Persekitaran seperti sekolah, universiti, perpustakaan, bank, pasar raya dan pelancongan sentiasa berinteraksi dengan data. Tahukah anda di mana data seseorang murid itu disimpan? Dengan kecanggihan teknologi maklumat masa ini, semua data disimpan dalam pangkalan data. Data ialah nadi sesuatu organisasi. Pangkalan data boleh membantu sesuatu organisasi lebih terurus supaya dapat meningkatkan produktiviti. Anda kini boleh menyimpan dan mencapai data dari mana-mana tempat di dunia dengan menggunakan komputer, telefon pintar dan tablet.



2.1

STANDARD PEMBELAJARAN

Pada akhir pembelajaran murid dapat

2.1.1 Melakar ERD (*Entity Relationship Diagram*) bagi permasalahan yang diberikan

2.1.2 Menghasilkan skema hubungan yang ternormal (*normalize*) daripada permasalahan yang diberikan

2.1.3 Membina pangkalan data hubungan menggunakan SQL: *Table*

2.1.4 Menggunakan SQL untuk memanipulasi data seperti sisip/tambah, padam dan kemas kini

2.1.5 Menggunakan SQL *SELECT* untuk mendapatkan data semula dengan menggunakan *GROUP BY* dan *ORDER BY*

2.1.6 Menggunakan fungsi *AVG*, *SUM*, *MAX*, *MIN* dan *COUNT*

2.1.7 Membina pangkalan data hubungan dengan menggunakan SQL bagi menyelesaikan masalah



Bahasa Pertanyaan Berstruktur: SQL (*Structured Query Language*)

2.1.1

Rajah Hubungan Entiti (*Entity Relationship Diagram – ERD*)

Pangkalan data ialah himpunan data berkaitan yang boleh dikongsi bersama oleh pelbagai kategori pengguna. Contoh aplikasi pangkalan data ialah Sistem Pinjaman Buku, Sistem Maklumat Pelajar, Sistem Tempahan Tiket dan sebagainya. Sebelum membangunkan sistem pangkalan data, anda perlu mengenal pasti masalah sistem sedia ada terlebih dahulu. Anda perlu mengetahui dan memahami proses dan aliran data dalam sistem sedia ada. Penyelesaian kepada masalah tersebut dapat digambarkan dengan menggunakan Rajah Hubungan Entiti (ERD). Rajah hubungan entiti ialah model asas dalam membina pangkalan data. ERD merupakan teknik yang digunakan untuk memodelkan data yang diperlukan dalam suatu organisasi. ERD juga menerangkan konsep tentang struktur pangkalan data. Terdapat tiga komponen asas dalam ERD, iaitu entiti, atribut dan perhubungan.

Struktur Rajah Hubungan Entiti

Entiti ialah objek atau konsep yang unik dan dapat dikenal pasti dalam persekitaran tertentu seperti orang, tempat, benda atau peristiwa yang dikutip dan disimpan sebagai data. Contoh entiti ialah MURID, GURU, BUKU, PEPERIKSAAN. Entiti diwakilkan dalam bentuk segi empat tepat seperti berikut:

MURID

Terdapat **dua** jenis entiti, iaitu:

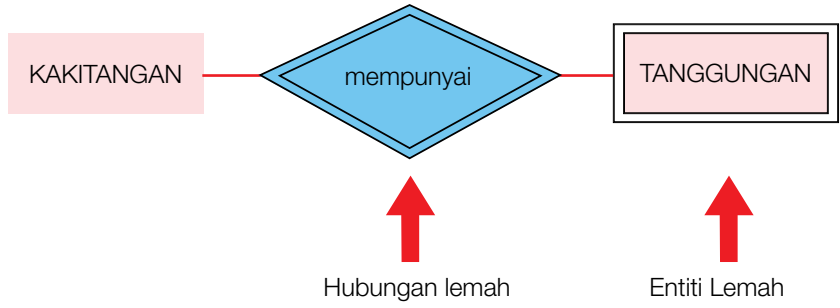
- Entiti kuat
Entiti kuat boleh wujud tanpa bersandar pada entiti lain.
Contoh: MURID, GURU, MATAPELAJARAN.
- Entiti lemah
Kewujudan entiti lemah bersandar pada entiti kuat.



UJI MINDA

Berikan contoh lain bagi aplikasi sistem pangkalan data yang anda ketahui.

Contoh: TANGGUNGAN. Entiti TANGGUNGAN hanya akan wujud jika adanya KAKITANGAN. Entiti lemah diwakilkan dalam bentuk segi empat tepat yang bergaris dua seperti Rajah 2.1.1.



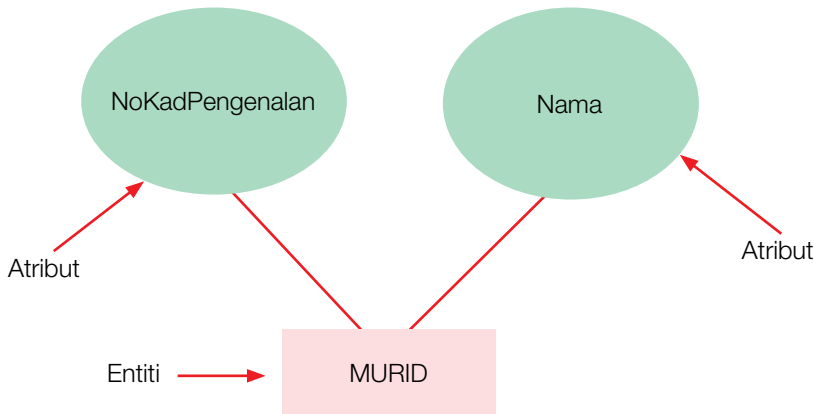
Rajah 2.1.1 Entiti lemah

Hubungan lemah digunakan untuk menyatukan entiti kuat dan entiti lemah. Hubungan lemah diwakilkan dalam bentuk permata yang bergaris dua.

Atribut atau medan ialah ciri-ciri yang menghuraikan sesuatu entiti. Atribut ialah sifat yang memerihalkan sesuatu set entiti. Atribut diwakilkan dalam bentuk bujur. Rajah 2.1.2 menunjukkan perwakilan atribut dalam ERD. Contoh atribut ditunjukkan dalam Jadual 2.1.1.

Jadual 2.1.1 Atribut

Entiti	Atribut
MURID	NoKadPengenalan, Nama, TarikhLahir, Alamat
GURU	NoKakitangan, NamaGuru, Gaji, Kemahiran



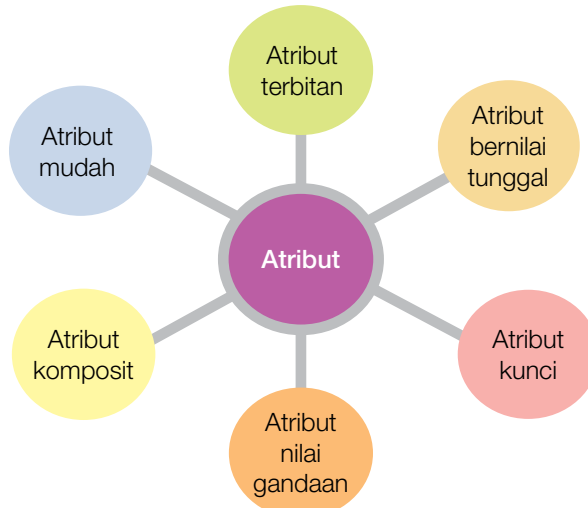
Rajah 2.1.2 Perwakilan atribut dalam ERD

Pada tahun 1999, Anugerah Perdana Teknologi Maklumat (APTMM) telah diperkenalkan. Anugerah ini merupakan pengiktirafan dan penghargaan yang diberi kepada sesuatu organisasi yang menunjukkan kecemerlangan dalam pelaksanaan dan penggunaan berkesan teknologi maklumat dan komunikasi. Anugerah ini dianjurkan oleh MAMPU, Jabatan Perdana Menteri dan PIKOM. Pada tahun 2007, anugerah ini digantikan dengan Anugerah Kualiti Teknologi Maklumat dan Komunikasi (AKICT). Pada tahun 2010 pula, AKICT digantikan dengan Anugerah Inovasi Teknologi Maklumat dan Komunikasi (AIICT).

Tahukah Anda

Siapakah Dr. Peter Chen? Dr. Peter Chen merupakan pengasas *Entity Relationship Diagram* (ERD). Beliau telah mengasaskannya pada tahun 1970-an. ERD masih digunakan sehingga kini dan dikatakan sebagai cara yang hampir terperinci untuk menyatakan hubungan antara entiti.





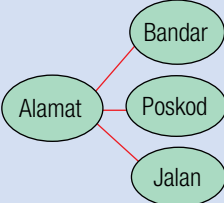
Terdapat **enam** jenis atribut seperti yang ditunjukkan dalam Rajah 2.1.3.



Rajah 2.1.3 Jenis-jenis atribut

Jadual 2.1.2 menerangkan jenis-jenis atribut dan ciri-cirinya.

Jadual 2.1.2 Notasi atribut

Notasi	Keterangan
	Notasi ini digunakan untuk mewakili atribut mudah dan atribut bernilai tunggal <ul style="list-style-type: none"> Atribut mudah tidak boleh dibahagikan kepada beberapa bahagian. Contoh: Nama, NoKadPengenalan. Atribut bernilai tunggal mempunyai satu nilai sahaja. Contoh: NoKadPengenalan.
	Atribut nilai gandaan mempunyai lebih daripada satu nilai. Contoh: Kemahiran Seorang guru mempunyai lebih daripada satu kemahiran.
	Atribut terbitan ini terhasil berasaskan atribut lain. Contoh: GajiTahunan Gaji tahunan seorang guru diperoleh dengan mengira gaji bulanan \times 12 bulan
	Atribut kunci adalah unik. Atribut ini membezakan setiap rekod dalam pangkalan data. Contoh: NoKadPengenalan dan NoKakitangan
	Atribut komposit ialah atribut yang boleh dipecahkan kepada beberapa bahagian. Contoh: Alamat boleh dibahagikan kepada Jalan, Bandar dan Poskod



What is on Entity Relationship
goo.gl/dcGyEC

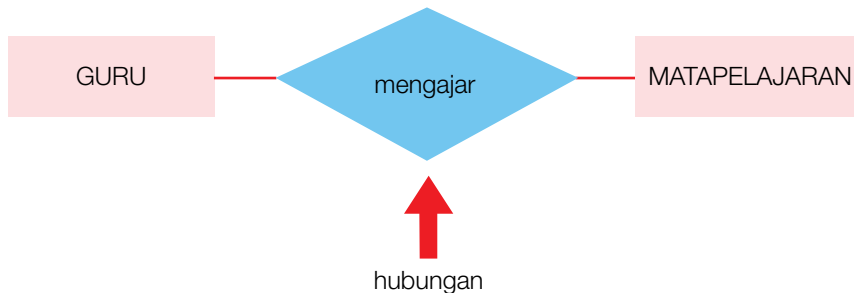


UJI MINDA

Berikan notasi ERD bagi entiti MURID yang mempunyai atribut jantina dan umur.

Hubungan ialah penyatuan antara entiti. Dalam sebuah ERD, terdapat banyak entiti yang mempunyai hubungan antara satu sama lain. Oleh sebab itu, hubungan digunakan untuk menyatukan satu atau lebih entiti yang terdapat dalam ERD. Hubungan menunjukkan bagaimana dua entiti berkongsi maklumat dalam pangkalan data. Hubungan dilabelkan dengan kata kerja. Contoh: mengajar, mendaftar dan menginap.

Seorang GURU **mengajarkan** sekurang-kurangnya satu MATAPELAJARAN. Hubungan diwakilkan dalam bentuk permata seperti dalam Rajah 2.1.4.



Rajah 2.1.4 Hubungan

Kunci primer ialah atribut unik yang membezakan rekod dalam sebuah pangkalan data. Setiap entiti hanya mempunyai satu kunci primer sahaja. Contoh kunci primer untuk entiti MURID ialah NoKadPengenalan dan kunci primer untuk entiti GURU ialah NoKakitangan.

NoKadPengenalan bersifat unik kerana seorang murid hanya mempunyai satu NoKadPengenalan dan nombornya berbeza dengan murid yang lain. Tidak ada murid yang mempunyai NoKadPengenalan yang sama. Dalam ERD, kunci primer hendaklah digariskan.

Kekardinalan menunjukkan hubungan dari segi kuantiti data antara suatu entiti dengan entiti yang lain. Terdapat **dua** jenis kekardinalan, iaitu:

- **Kekardinalan Minimum**
Minimum data suatu entiti berkait dengan entiti lain. Contoh: Katakan terdapat dua entiti, iaitu GURU dan MATAPELAJARAN. Seorang GURU mengajar sekurang-kurangnya satu MATAPELAJARAN. Maka kekardinalan minimum ialah satu kerana seorang guru mesti mengajar sekurang-kurangnya satu mata pelajaran. Tidak ada guru yang tidak akan mengajar.
- **Kekardinalan Maksimum**
Maksimum data suatu entiti berkait dengan entiti lain. Contoh: Satu MATAPELAJARAN boleh diajar oleh seorang GURU atau lebih. Seorang guru boleh mengajar maksimum lebih dari satu mata pelajaran. Maka kekardinalan maksimum adalah banyak.



Pada 26 Oktober 2016, Perdana Menteri Dato' Sri Najib Tun Razak menggesa penggiat sektor digital tempatan agar bekerjasama dengan komuniti teknologi global kerana Malaysia akan meraih manfaat yang besar jika potensi ekonomi digital dieksploitasi dengan sepenuhnya. Kerajaan mengumumkan bahawa 2017 merupakan Tahun Ekonomi Internet Malaysia.



ER Model
goo.gl/9G641p



Cardinality
goo.gl/qaVHPB

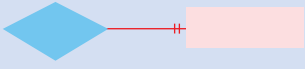
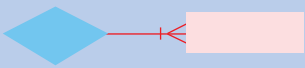
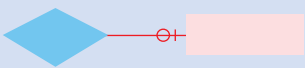
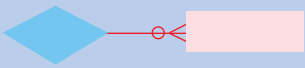
Tahukah Anda

Tip untuk melukis ERD.

- ✓ Kenal pasti semua entiti yang berkaitan dan hubungan antara entiti.
- ✓ Setiap entiti hanya muncul sekali dalam sesuatu rajah.
- ✓ Gunakan nama yang sesuai untuk entiti, atribut dan hubungan.
- ✓ Singkirkan hubungan yang bertindih antara entiti.
- ✓ Gunakan warna-warna yang menarik.

Jadual 2.1.3 menunjukkan jenis-jenis hubungan kekardinalan dan ciri-cirinya.

Jadual 2.1.3 Hubungan kekardinalan

Simbol	Keterangan
Mandatori Satu 	Minimum dan maksimum data adalah satu.
Mandatori Banyak 	Minimum data adalah satu dan maksimum data adalah banyak.
Opsyenal Satu 	Minimum data adalah kosong dan maksimum data adalah satu.
Opsyenal Banyak 	Minimum data adalah kosong dan maksimum data adalah banyak.

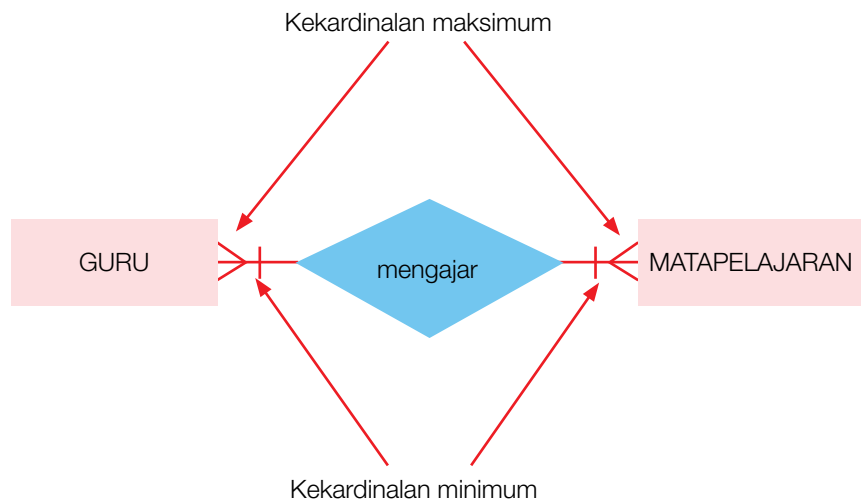
Rajah 2.1.5 menunjukkan hubungan kekardinalan.



UJI MINDA

Berdasarkan Rajah 2.1.5, bina satu ERD yang mewakili situasi seperti di bawah.

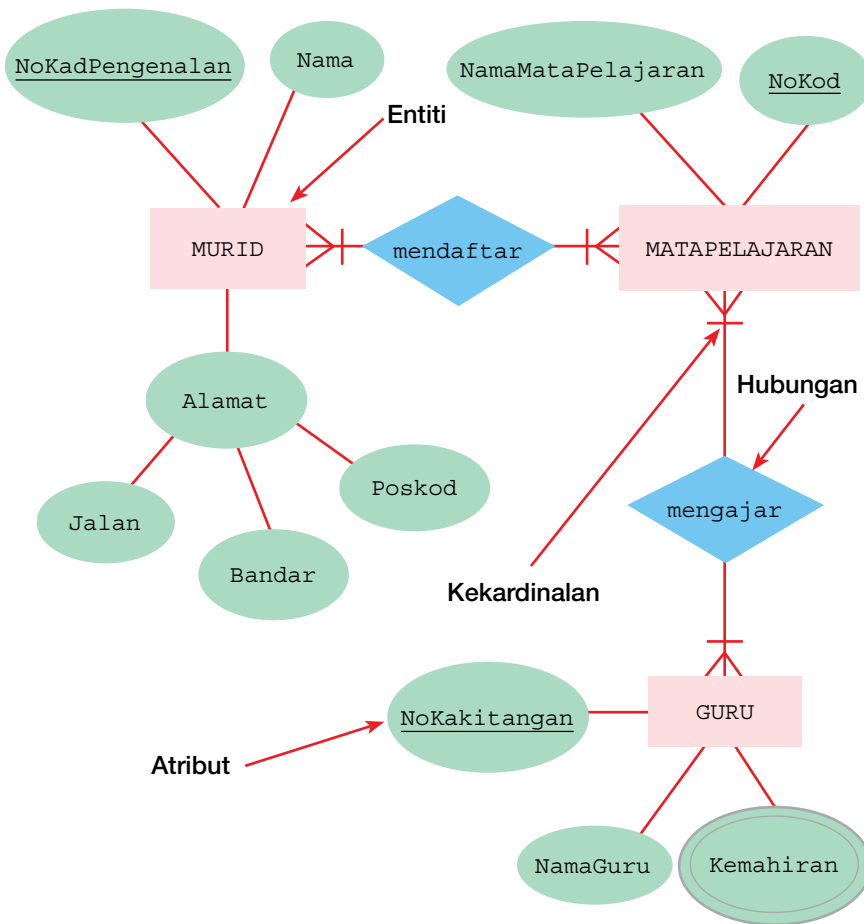
Katakan terdapat MURID dan KELAB. Seorang MURID boleh mendaftar sekurang-kurangnya satu KELAB. Satu KELAB boleh didaftar oleh lebih daripada seorang MURID.



Rajah 2.1.5 Hubungan kekardinalan

ERD dalam Rajah 2.1.5 bermaksud seorang guru harus mengajar sekurang-kurangnya satu mata pelajaran dan maksimum mata pelajaran yang boleh diajar adalah dua atau lebih. Manakala mata pelajaran pula hendaklah diajar oleh sekurang-kurangnya seorang guru dan maksimum guru yang boleh mengajar sesuatu mata pelajaran ialah dua atau lebih.

Rajah 2.1.6 menunjukkan contoh perwakilan entiti, atribut, hubungan dan kekardinalan dalam ERD.



Rajah 2.1.6 Contoh perwakilan entiti, atribut, hubungan dan kekardinalan dalam ERD



UJI MINDA

Berdasarkan Rajah 2.1.6, bina ERD yang menunjukkan perwakilan entiti, atribut, hubungan dan kekardinalan bagi GURU, KELAS dan MATAPELAJARAN Selamat maju jaya!



Microsoft Visio untuk melukis ERD goo.gl/c8eylD

Melakar ERD (Entity Relationship Diagram)

Pengurusan stok dan inventori alatan sukan amat penting dalam kejayaan sesuatu aktiviti sukan dan kokurikulum. Kekurangan peralatan sukan menyebabkan sesuatu sukan sukar untuk dijalankan. Pangkalan data stor sukan mengandungi rekod pelbagai alatan sukan sama ada untuk sukan sekolah ataupun untuk pembelajaran dan pengajaran mata pelajaran Pendidikan Jasmani. Dalam pangkalan data ini juga, maklumat barangan sukan, pembekal, peminjam dan rekod pinjaman yang dibuat oleh semua murid disimpan. Entiti-entiti yang terlibat adalah seperti yang berikut:

- BARANG ialah peralatan sukan yang disimpan. Atribut BARANG ialah NoBarang, NamaBarang, Kuantiti, HargaSeunit.
- MURID ialah peminjam barangan sukan. Atribut MURID ialah IDMurid, NamaMurid dan Kelas.



ER Diagram Tutorial goo.gl/1lJAuA

Tahukah Anda

Apakah kunci primer? Mari kita imbas kembali pembelajaran kita mengenai kunci primer ketika di Tingkatan 4. Kunci primer ialah atribut atau gabungan beberapa atribut dengan nilai yang unik untuk mewakili setiap rekod dalam sesebuah jadual.

- PINJAMAN ialah keadaan berlakunya proses meminjam. Atribut PINJAMAN ialah NoPinjaman, TarikhPinjam dan TarikhPulang.
- PEMBEKAL ialah individu atau syarikat pembekal barangan sukan. Atribut PEMBEKAL ialah NoPembekal, NamaPembekal dan Alamat (terdiri daripada NamaJalan, Bandar, Poskod dan Negeri).

Hubungan dan kekardinalan antara entiti adalah seperti yang berikut:

- Setiap barangan sukan boleh dipinjam oleh seorang murid pada satu masa. Setiap murid boleh meminjam satu atau lebih barangan sukan.
- Seorang murid boleh membuat satu pinjaman atau tidak membuat sebarang pinjaman. Setiap pinjaman merekod data pinjaman yang dibuat oleh seorang murid pada satu masa.
- Setiap pembekal boleh membekalkan satu atau lebih barang dan setiap barang mungkin dibekalkan oleh satu atau lebih pembekal ataupun tidak dibekalkan oleh mana-mana pembekal.

Langkah-langkah untuk melakar ERD adalah seperti yang berikut:

1 Senaraikan entiti dan atribut.

Entiti	Atribut
BARANG	NoBarang, NamaBarang, Kuantiti, HargaSeunit
MURID	IDMurid, NamaMurid, Kelas
PINJAMAN	NoPinjaman, TarikhPinjam, TarikhPulang
PEMBEKAL	NoPembekal, NamaPembekal, Alamat (ialah atribut komposit yang terdiri daripada NamaJalan, Bandar, Poskod, Negeri)

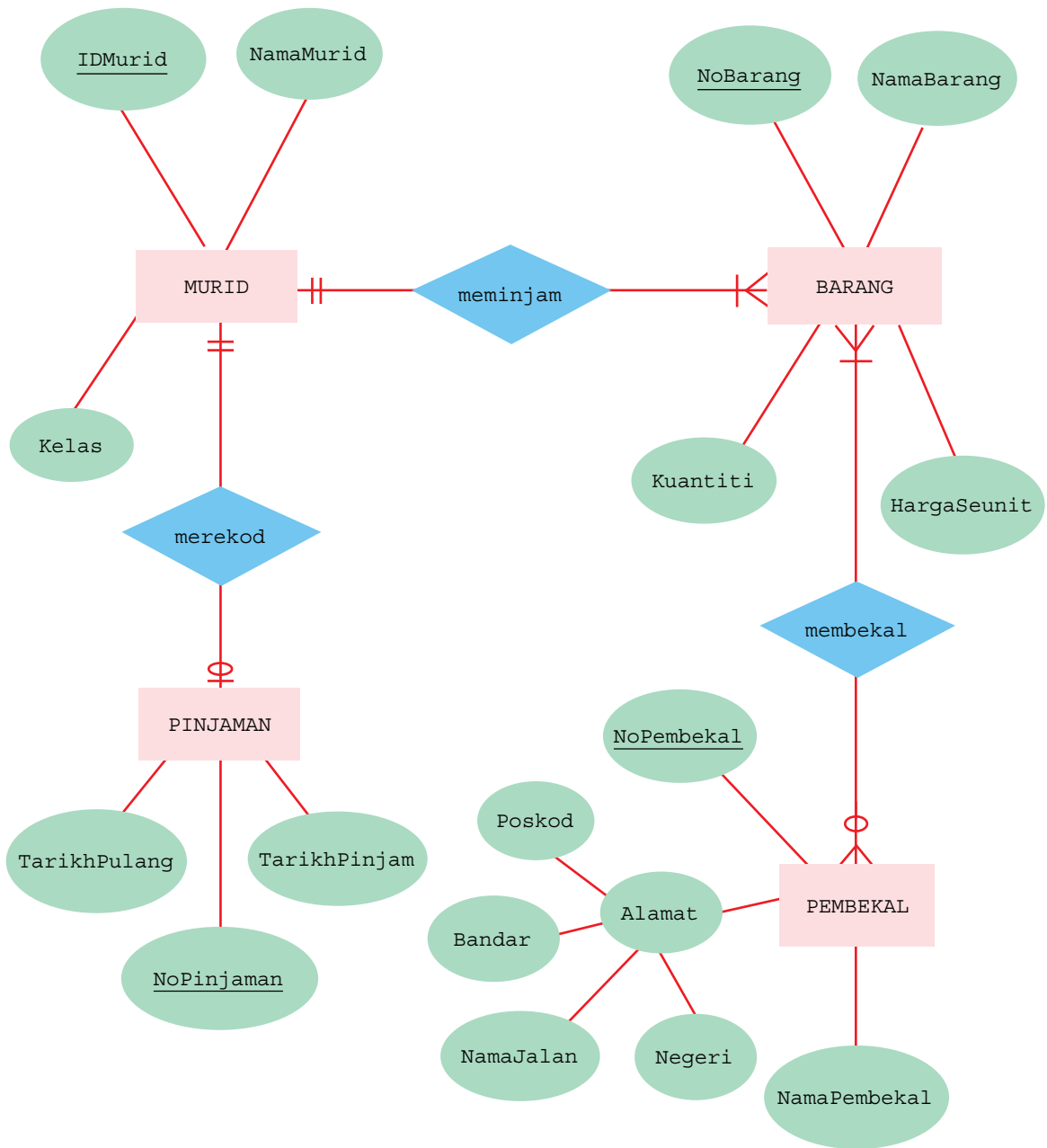
2 Kenal pasti kunci primer.

Entiti	Kunci Primer
BARANG	NoBarang
MURID	IDMurid
PINJAMAN	NoPinjaman
PEMBEKAL	NoPembekal

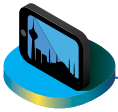
3 Kenal pasti hubungan antara entiti:

- meminjam
- merekod
- membekal

4 Lakarkan ERD dan kenal pasti kekardinalan seperti dalam Rajah 2.1.7.



Rajah 2.1.7 ERD sistem pengurusan inventori



MALYSIAKU!

1BestariNet ialah projek yang diinisiatifkan oleh Kementerian Pendidikan Malaysia (KPM) dan dilaksanakan dengan kerjasama YTL Communications. Projek ini memberikan impak positif dalam jangka masa panjang untuk membantu murid memperoleh kemahiran pembelajaran abad ke-21, merapatkan jurang digital dan merealisasikan konsep teknologi untuk semua sekali gus mengubah kehidupan masyarakat, khususnya mereka yang tinggal jauh di kawasan pinggir bandar. Sebanyak 10,000 buah sekolah rendah dan sekolah menengah di Malaysia akan dilengkapi dengan akses Internet berkelajuan tinggi 4G dan platform pembelajaran maya yang menyediakan sambungan Internet berkelajuan tinggi serta akses kepada Penyelesaian Pembelajaran Bersepadu.

Sumber:
<http://1bestarinet.net/>

AKTIVITI

1

Kerja individu

Kenal pasti Komponen Rajah Hubungan Entiti

Anda telah ditugaskan untuk membina sistem pangkalan data untuk Hospital Sri Inai. Sistem ini bertujuan untuk merekod maklumat pesakit luar yang pernah mendapat rawatan di hospital berkenaan. Sistem ini juga dapat membantu pihak hospital mencapai dan menyimpan data pesakit dengan cepat dan pantas.



Langkah pertama yang perlu anda lakukan ialah mengenal pasti data yang ingin disimpan di dalam pangkalan data.

Berdasarkan hasil kunjungan dan pemerhatian anda ke hospital berkenaan:

- 1 Kenal pasti komponen-komponen rajah hubungan entiti yang berikut:
 - Entiti
 - Atribut
 - Hubungan
- 2 Lukis rajah hubungan entiti (ERD).

2.1.2 Model Hubungan Data

Model hubungan data diperkenalkan oleh E. F. Codd pada tahun 1970. Kebanyakan sistem pengurusan pangkalan data seperti *MS Access*, *Oracle* dan *MySQL* menggunakan model ini. Model ini menyimpan data dalam bentuk jadual hubungan, iaitu data dan hubungan dipersembahkan dalam bentuk jadual. Model ini dibina sebagai langkah seterusnya selepas anda melakar Rajah Hubungan Entiti. Model hubungan data dapat memberikan gambaran kepada anda tentang struktur jadual untuk sistem pangkalan data yang ingin dibangunkan.

Kunci Hubungan

Setiap hubungan perlu ada kunci primer yang membolehkan anda menyimpan dan mencapai data dalam hubungan. **Kunci primer** ialah atribut unik (sama ada terdiri daripada satu ataupun beberapa atribut) yang dipilih untuk mengenal pasti rekod dalam hubungan. Kunci primer hendaklah digariskan seperti contoh yang berikut.

Contoh:

```
MURID (NoKadPengenalan, Nama, Alamat)
```

Kunci asing ialah kunci primer sesuatu hubungan yang muncul dalam hubungan lain dalam pangkalan data yang sama. Kunci ini ditandakan dengan garis putus-putus. Kunci asing digunakan untuk menghubungkan antara dua jadual. Contohnya antara MURID dengan MATAPELAJARAN:

```
MURID (NoKadPengenalan, Nama, Alamat, NoKod)  
MATAPELAJARAN (NoKod, NamaMataPelajaran)
```

Ciri-ciri Model Hubungan Data

Hubungan ialah jadual yang terdiri daripada rekod dan medan. Akan tetapi, tidak semua jadual ialah hubungan. Hubungan mempunyai beberapa ciri tertentu seperti yang berikut:

- Setiap hubungan (jadual) mempunyai nama yang berbeza daripada nama-nama hubungan yang lain.
- Setiap data dalam hubungan mempunyai satu nilai sahaja.
- Setiap atribut (medan) mempunyai nama yang berbeza.
- Setiap rekod adalah berbeza.
- Susunan atribut (medan) tidak mempunyai kepentingan.
- Susunan rekod tidak mempunyai kepentingan.

Hubungan yang mempunyai ciri-ciri di atas dipanggil penormalan atau Bentuk Penormalan Pertama (1NF).

Model hubungan data dalam Rajah 2.1.8 menunjukkan hubungan antara MURID dengan MATAPELAJARAN. Setiap jadual perlu ada kunci primer. Dalam jadual MURID, kunci primer ialah NoKadPengenalan manakala NoKod ialah kunci primer bagi jadual MATAPELAJARAN. Kunci asing diperlukan untuk menghubungkan dua jadual ini. Di sini, kunci asing ialah NoKod yang terdapat dalam jadual MURID. Contoh model hubungan data ditunjukkan seperti dalam Rajah 2.1.8.



APLIKASI HARIAN

Aplikasi “MindAppz” dan “Kingaroo” ialah aplikasi mobil berasaskan pendidikan yang pertama untuk murid di peringkat rendah dan menengah di Malaysia. Aplikasi ini merangkumi Bahasa Melayu, Bahasa Inggeris, Matematik dan Sains untuk sekolah rendah dan Bahasa Melayu, Bahasa Inggeris, Matematik dan Sejarah untuk sekolah menengah.

Tahukah Anda

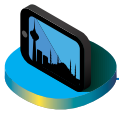


E. F. Codd dikenali sebagai Bapa Sistem Pangkalan Data. Beliau telah mengemukakan 12 prinsip mengenai pangkalan data.



UJI MINDA

Salah satu perbezaan kunci primer dan kunci asing ialah kunci primer tidak boleh menerima nilai *null*, sebaliknya bagi kunci asing. Nyatakan tiga perbezaan yang lain bagi kunci primer dan kunci asing.



MALAYSIAKU!

Pada 25 Ogos 2015, Kementerian Sains, Teknologi dan Inovasi telah melancarkan makmal *Big Data IoT Technology Accelerator (BITX)* bagi menyediakan perkhidmatan dan teknologi berkaitan Industri *Internet of Things (IoT)*. IoT merujuk hasil daripada pemusatan pelbagai teknologi daripada komunikasi wayarles kepada Internet.

MURID

NoKadPengenalan	Nama	Alamat	No_Kod
991002021111	Airis	Shah Alam	TA01
990203042222	Amir	PJ	TA02
990110105555	Naufal	Kelana Jaya	TA03

MATAPELAJARAN

NoKod	Nama
TA01	Bahasa Melayu
TA02	Matematik
TA03	Sejarah

Rajah 2.1.8 Model hubungan data

Skema Hubungan

Model hubungan data mengandungi himpunan hubungan ternormal. Struktur pangkalan data ditunjukkan dalam bentuk skema yang menghuraikan struktur logik secara keseluruhan sesuatu pangkalan data.

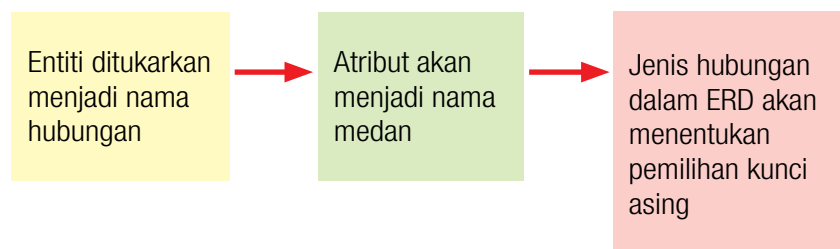
Contoh:

MURID

NoKadPengenalan	Nama	Alamat
-----------------	------	--------

Pemetaan ERD kepada Skema Hubungan

Langkah-langkah memetakan ERD kepada skema hubungan adalah seperti yang ditunjukkan dalam Rajah 2.1.9.



Rajah 2.1.9 Langkah-langkah untuk memetakan ERD kepada skema hubungan

Tahukah Anda

Apakah syarat-syarat yang menepati Bentuk Penormalan Pertama (1NF)?

Syarat-syaratnya ialah:

- ✓ Mempunyai nilai yang tidak boleh dibahagikan lagi
- ✓ Tidak mempunyai kumpulan yang berulang

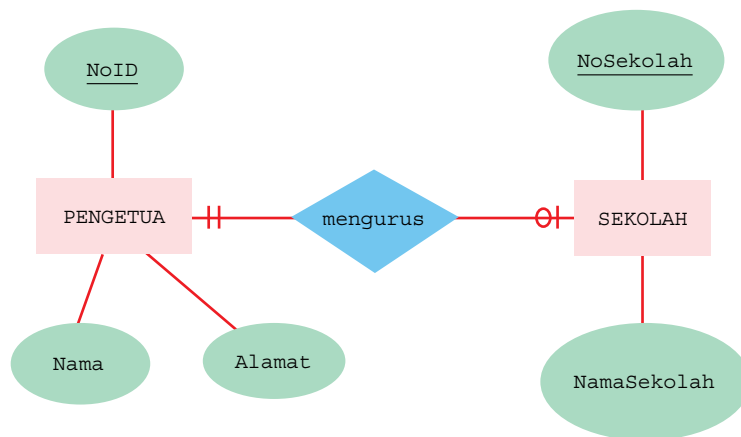
Antara jenis hubungan dalam ERD ialah:

- Hubungan satu-ke-satu
- Hubungan satu-ke-banyak
- Hubungan banyak-ke-banyak

CONTOH 1

Contoh 1 menunjukkan langkah-langkah untuk memetakan ERD kepada skema hubungan. Anda perlu mempertimbangkan jenis hubungan yang terdapat dalam ERD untuk menentukan kunci asing.

1 Hubungan satu-ke-satu seperti Rajah 2.1.10



Rajah 2.1.10 ERD menunjukkan hubungan 1 ke 1

Penyelesaian:

Jadual 2.1.4 Skema hubungan 1 ke 1

PENGETUA

<u>NoID</u>	Nama	Alamat
-------------	------	--------

SEKOLAH

<u>NoSekolah</u>	NamaSekolah	<u>NoID</u>
------------------	-------------	-------------

Kunci asing perlu dipilih daripada pihak entiti yang berada pada bahagian mandatori. Entiti yang berada di pihak mandatori ialah PENGETUA. Oleh yang demikian, NoID akan menjadi kunci asing dalam jadual SEKOLAH seperti Jadual 2.1.4.

Tahukah Anda

ERD merupakan alatan visual yang mengaitkan pangkalan data hubungan dan rangkaian. Sejak ERD diperkenalkan, ERD telah berkembang dan kadangkala digunakan untuk pengurusan perniagaan, pembangunan produk dan dalam pembentukan strategi formula.

Sumber:
<http://creately.com>



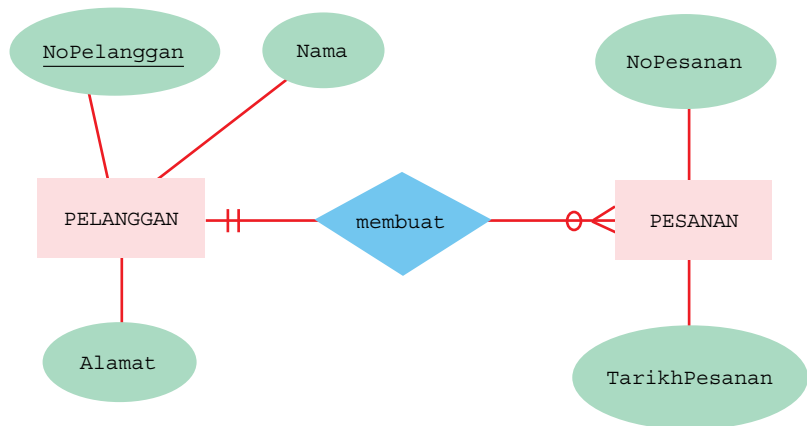
UJI MINDA

Dalam sebuah sistem pangkalan data sesebuah sekolah, pasti ada entiti murid, mata pelajaran dan guru. Murid pula mempunyai atribut seperti nama, nombor kad pengenalan dan alamat. Berdasarkan situasi yang diberikan, bina ERD yang menunjukkan hubungan satu-ke-banyak dan model hubungannya.



Entity-Relationship Model
goo.gl/pPKaFS

2 Hubungan satu-ke-banyak seperti Rajah 2.1.11



Rajah 2.1.11 ERD menunjukkan hubungan 1 ke M

Penyelesaian:

Jadual 2.1.5 Skema hubungan 1 ke M

PELANGGAN

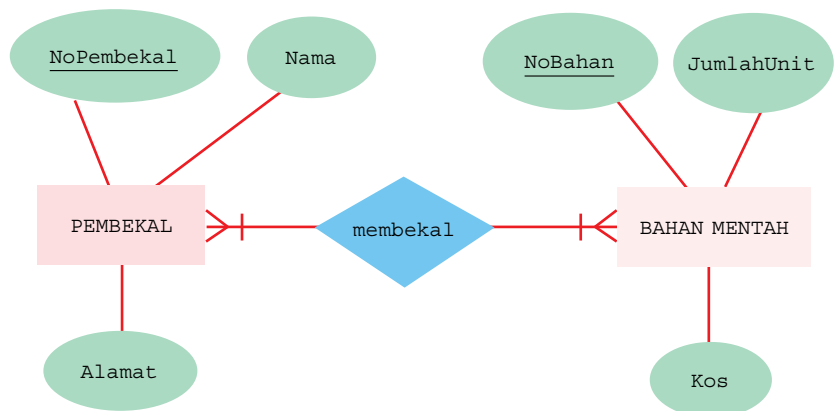
<u>NoPelanggan</u>	Nama	Alamat
--------------------	------	--------

PESANAN

<u>NoPesanan</u>	TarikhPesanan	<u>NoPelanggan</u>
------------------	---------------	--------------------

Dalam hubungan satu-ke-banyak, kunci primer pada bahagian satu akan menjadi kunci asing pada bahagian banyak. Oleh sebab itu, NoPelanggan akan menjadi kunci asing dalam jadual PESANAN seperti Jadual 2.1.5.

3 Hubungan banyak-ke-banyak seperti Rajah 2.1.12



Rajah 2.1.12 ERD menunjukkan hubungan M ke N



Penyelesaian:

Jadual 2.1.6 Skema hubungan M ke N

PEMBEKAL

<u>NoPembekal</u>	Nama	Alamat
-------------------	------	--------

PEMBEKAL_BAHAN_MENTAH

<u>NoPembekal</u>	<u>NoBahan</u>
-------------------	----------------

BAHAN_MENTAH

<u>NoBahan</u>	JumlahUnit	Kos
----------------	------------	-----

Dalam hubungan banyak-ke-banyak, jadual baharu perlu dibentuk dan kunci primer yang menghubungkan kedua-dua entiti akan menjadi kunci asing dalam jadual ini. Oleh sebab itu, NoPembekal dan NoBahan akan menjadi kunci asing dalam jadual PEMBEKAL_BAHAN_MENTAH seperti Jadual 2.1.6.

Skema Hubungan Ternormal

Skema hubungan ternormal boleh dihasilkan berdasarkan lakaran ERD. Berdasarkan Rajah 2.1.7 (sistem pengurusan inventori), petakan kepada skema hubungan ternormal. Telitikan langkah-langkah pemetaan yang berikut:

- 1 Entiti akan menjadi nama jadual seperti MURID, BARANG, PINJAMAN dan PEMBEKAL.
- 2 Atribut akan menjadi nama medan seperti dalam Rajah 2.1.13.
- 3 Pemilihan kunci asing berdasarkan jenis hubungan antara entiti.

MURID

<u>IDMurid</u>	NamaMurid	Kelas
----------------	-----------	-------

PINJAMAN

<u>NoPinjaman</u>	TarikhPinjam	TarikhPulang	<u>IDMurid</u>
-------------------	--------------	--------------	----------------

BARANG

<u>NoBarang</u>	NamaBarang	Kuantiti	HargaSeunit	<u>IDMurid</u>
-----------------	------------	----------	-------------	----------------

PEMBEKAL_BARANG

<u>NoBarang</u>	<u>NoPembekal</u>
-----------------	-------------------

PEMBEKAL

<u>NoPembekal</u>	NamaPembekal	NamaJalan	Bandar	Poskod	Negeri
-------------------	--------------	-----------	--------	--------	--------

Rajah 2.1.13 Skema hubungan ternormal



UJI MINDA



“Satu pangkalan data akan dibangunkan untuk menyimpan maklumat pekerja di sebuah bank. Maklumat peribadi pekerja seperti nama, alamat dan nombor telefon akan direkodkan. Mereka juga diberikan ID pekerja. Kemudian, mereka ditugaskan di bahagian yang berbeza seperti Pembangunan Manusia, Akaun, Operasi dan sebagainya. Dalam setiap bahagian, terdapat beberapa pengurus dan pekerja. Setiap pelanggan bank akan berurusan dengan seorang pekerja bank atau beberapa orang pelanggan akan berurusan dengan seorang pekerja bank pada satu masa.”

Berdasarkan situasi yang diberikan, kenal pasti entiti, atribut, hubungan dan kardinaliti yang boleh didapati.



Data Anomalies
goo.gl/mZlulx

Penormalan

Kadangkala anda mungkin mendapat maklumat berbentuk jadual seperti dalam Jadual 2.1.7. Jika diteliti maklumat dalam jadual, didapati terdapat atribut yang mempunyai kumpulan data yang berulang. Untuk menghasilkan skema hubungan ternormal, proses penormalan perlu dilakukan. Penormalan ialah proses untuk memastikan bahawa model data logikal yang terhasil mempunyai anomali yang paling minimum. Penormalan bertujuan untuk menghasilkan skema hubungan dengan menempatkan atribut ke dalam jadual dengan betul bagi mengelakkan masalah anomali.

Anomali ialah ralat yang mungkin berlaku apabila pengguna ingin mengemas kini jadual yang mengandungi data berulang. Apabila anda ingin mengubah sesuatu data dalam jadual, anda perlu melakukan perubahan terhadap beberapa rekod yang lain. Kegagalan berbuat demikian akan menyebabkan kekurangan integriti dalam pangkalan data. Maka, data yang diperoleh akan menjadi tidak tepat.

Jadual 2.1.7 Jadual Pengurusan Inventori dalam bentuk ONF

NoBarang	NamaBarang	Kuantiti	HargaSeunit	NamaPembekal	Alamat	IDMurid	NamaMurid	Kelas	TarikhPinjam	TarikhPulang
A01	Bola Sepak, Bola Jaring	10 5	30.50 15.30	Angsa Maju Setia Sukan	Alor Setar Shah Alam	M1000	Lim	1 Merah	02/7/2017	02/7/2017
A03	Bola Hoki	10	20.00	Indah Sukan	Kuching	M2000	Elisa	4 Biru	07/7/2017	07/7/2017
A05	Baton	20	10.30	Setia Sukan	Shah Alam	M3000	Devi	3 Ungu	09/8/2017	12/8/2017

Bentuk Penormalan Pertama (1NF)

Sesuatu hubungan data adalah dalam bentuk normal pertama (1NF) jika tiada atribut yang mempunyai kumpulan data yang berulang. Setiap atribut mempunyai satu nilai sahaja. Contoh: Jadual 2.1.7 adalah dalam bentuk yang tidak ternormal kerana mengandungi atribut yang berulang, iaitu IDMurid, NamaMurid dan Kelas. Jadual dikatakan dalam bentuk normal pertama apabila kumpulan data berulang dihapuskan. Ini dilakukan dengan membentuk kunci primer seperti dalam Jadual 2.1.8.

Jadual 2.1.8 Jadual BARANG bentuk pertama (1NF)

BARANG

<u>NoBarang</u>	NamaBarang	Kuantiti	HargaSeunit	NamaPembekal	Alamat	<u>IDMurid</u>	NamaMurid	Kelas	TarikhPinjam	TarikhPulang
A01	Bola Sepak	10	30.50	Angsa Maju	Alor Setar	M1000	Lim	1 Merah	02/7/2017	02/7/2017
A02	Bola Jaring	5	15.30	Setia Sukan	Shah Alam	M1000	Lim	1 Merah	02/7/2017	02/7/2017
A03	Bola Hoki	10	20.00	Indah Sukan	Kuching	M2000	Elisa	4 Biru	07/7/2017	07/7/2017
A05	Baton	20	10.30	Setia Sukan	Shah Alam	M3000	Devi	3 Ungu	09/8/2017	12/8/2017

Bentuk Penormalan Kedua (2NF)

Hubungan data adalah dalam bentuk normal kedua jika kriteria yang berikut telah dipenuhi:

- Jadual adalah dalam bentuk normal pertama
- Semua atribut bukan kunci bergantung sepenuhnya dengan atribut kunci

Jika ada atribut bukan kunci yang hanya bergantung pada salah satu daripada atribut kunci, maka hubungan tersebut dikatakan mempunyai **kebergantungan separa**. Contoh: Jadual BARANG dalam Jadual 2.1.8 mempunyai dua kunci primer (dikenali sebagai **kunci calon**), iaitu NoBarang dan IDMurid. Akan tetapi, atribut NamaBarang, Kuantiti, HargaSeunit, NamaPembekal dan Alamat hanya bergantung pada NoBarang dan atribut NamaMurid dan Kelas hanya bergantung pada IDMurid. Hanya atribut TarikhPinjam dan TarikhPulang bergantung pada kedua-dua kunci, iaitu NoBarang dan IDMurid. Penyelesaiannya, kita perlu cipta jadual baharu untuk mewakili TarikhPinjam dan TarikhPulang.

Jadual 2.1.9 Jadual BARANG bentuk normal kedua (2NF)

BARANG

<u>NoBarang</u>	NamaBarang	Kuantiti	HargaSeunit	NamaPembekal	Alamat
-----------------	------------	----------	-------------	--------------	--------

PINJAMAN

<u>NoBarang</u>	<u>IDMurid</u>	TarikhPinjam	TarikhPulang
-----------------	----------------	--------------	--------------

MURID

<u>IDMurid</u>	NamaMurid	Kelas
----------------	-----------	-------

Tahukah Anda

Langkah-langkah untuk membentuk penormalan kedua:

- ✓ Periksa jadual yang mempunyai kunci calon
- ✓ Bagi setiap kunci bukan atribut, pastikan jika kunci tersebut bahagian pertama atau bahagian kedua
- ✓ Hapuskan kunci separa dan kebergantungannya untuk membentuk jadual yang baharu



Third Normal Form
goo.gl/mUVyTz



Third Normal Form
goo.gl/g1p1GG



UJI MINDA

Nyatakan perbezaan antara kebergantungan separa dengan kebergantungan transitif.



Fifth Normal Form
goo.gl/g1p1GG



Bentuk penormalan tidak terhad sehingga bentuk penormalan ketiga (3NF) sahaja, tetapi boleh merangkumi bentuk penormalan keempat dan juga kelima. Anda boleh mengetahuinya dengan lebih lanjut dengan melayari goo.gl/RfBxnm.

Bentuk Penormalan Ketiga (3NF)

Satu hubungan berada dalam bentuk normal ketiga jika memenuhi kriteria berikut:

- Jadual berada dalam bentuk normal kedua
- Tidak wujud kebergantungan transitif

Kebergantungan transitif bermaksud wujudnya kebergantungan antara atribut bukan kunci. Contoh: Jadual 2.1.9 menunjukkan jadual BARANG yang berada dalam bentuk normal kedua dan terdapat kebergantungan transitif, iaitu atribut Alamat bergantung kepada atribut NamaPembekal dan kedua-duanya adalah atribut bukan kunci. Oleh sebab itu, kedua-dua atribut ini perlu dipisahkan daripada jadual BARANG. Penyelesaian kepada masalah kebergantungan transitif dilakukan dengan mencipta jadual baharu bagi atribut bukan kunci yang saling bergantung seperti dalam Jadual 2.1.10.

Jadual 2.1.10 Jadual BARANG bentuk normal ketiga (3NF)

BARANG

<u>NoBarang</u>	NamaBarang	Kuantiti	HargaSeunit	<u>NamaPembekal</u>
-----------------	------------	----------	-------------	---------------------

PINJAMAN

<u>NoBarang</u>	<u>IDMurid</u>	TarikhPinjam	TarikhPulang
-----------------	----------------	--------------	--------------

MURID

<u>IDMurid</u>	NamaMurid	Kelas
----------------	-----------	-------

PEMBEKAL

<u>NamaPembekal</u>	Alamat
---------------------	--------

AKTIVITI 2

Kerja kumpulan

Menghasilkan Skema Hubungan Ternormal

Lakukan lawatan ke salah satu persekitaran yang berikut:

- Kedai buku
- Pasar raya
- Perpustakaan

Perhatikan bagaimana transaksi berlaku di setiap persekitaran di atas. Dapatkan dokumen asas seperti slip jualan, invoice pelanggan dan slip pinjaman buku. Berdasarkan maklumat yang terdapat dalam dokumen ini, sediakan perkara-perkara yang berikut:

- 1 Senaraikan nama atribut yang terdapat pada setiap dokumen tersebut.
- 2 Senaraikan kebersandaran fungsi antara atribut bagi setiap dokumen.
- 3 Lukis skema hubungan bagi setiap dokumen.

2.1.3 SQL: Membina Pangkalan Data

Structured Query Language atau Bahasa Pertanyaan Berstruktur dengan singkatannya dikenali sebagai SQL digunakan untuk mencapai dan mengemas kini data dari pangkalan data. Berdasarkan ANSI (*American National Standard Institute*), SQL ialah bahasa pengaturcaraan piawai yang digunakan untuk berhubung dengan pangkalan data melalui sistem pengurusan pangkalan data. Piawaian ini tidak bergantung pada jenis mesin yang digunakan sama ada *International Business Machines* (IBM), *Microsoft* atau *Oracle*. Oleh sebab itu, hampir semua sistem pengurusan pangkalan data mengenali pernyataan SQL.

Pengenalan SQL

Arahan SQL dibahagikan kepada **tiga** bahasa, iaitu:

1 *Data Definition Language* atau Bahasa Takrifan Data (DDL)

- Pernyataan SQL yang digunakan untuk mencipta dan mengubah struktur pangkalan data. Contoh:
 - CREATE
 - ALTER
 - DROP

2 *Data Manipulation Language* atau Bahasa Pengolahan Data (DML)

- Pernyataan SQL yang digunakan untuk memanipulasi data dalam jadual. Contoh:
 - SELECT
 - INSERT
 - UPDATE
 - DELETE

3 *Data Control Language* atau Bahasa Kawalan Data (DCL)

- Pernyataan SQL yang mengawal capaian data dalam pangkalan data. Contoh:
 - GRANT
 - REVOKE

Pengenalan kepada Jenis-jenis Data

Apabila mencipta jadual, anda perlu memilih jenis data untuk setiap lajur yang akan dibentuk dalam jadual tersebut. Jenis data yang dipilih menentukan jenis data yang boleh disimpan dalam medan tersebut. Pemilihan jenis data yang betul adalah penting kerana pemilihannya memberikan kesan kepada prestasi keseluruhan pangkalan data. Jenis data juga menentukan jumlah ruang storan yang diperlukan. Jadual 2.1.11 menerangkan jenis data dalam *MySQL*.



APLIKASI HARIAN

Sistem Pengurusan Sekolah (SPS) dibina oleh Kementerian Pendidikan Malaysia untuk mengurus maklumat dan data sekolah, guru, murid, peperiksaan dan kokurikulum. SPS bertujuan untuk membantu mengurangkan beban kerja pentadbir dan guru semasa mengurus data sekolah.

Tahukah Anda



Antara yang boleh dilakukan oleh SQL ialah:

- ✓ melaksanakan pertanyaan kepada pangkalan data
- ✓ mengesan data daripada pangkalan data
- ✓ memasukkan rekod ke dalam pangkalan data
- ✓ mengemas kini rekod dalam pangkalan data
- ✓ memadam rekod daripada pangkalan data dan sebagainya



INOVASI SAINS KOMPUTER

Pada 13 Oktober 2015, Kementerian Sains, Teknologi dan Inovasi (MOSTI) melalui Agensi *Remote Sensing* Malaysia (ARSM) berjaya membangunkan dua unit sistem *Unmanned Aerial Vehicle-Remote Sensing* (UAV-RS). Sistem pesawat udara tanpa pemandu ini akan meningkatkan kecekapan bagi memperoleh maklumat permukaan bumi bagi memenuhi keperluan pemantauan bencana, pengurusan sumber asli dan pembukaan tanah di samping berperanan penting untuk memberikan informasi bagi memelihara kedaulatan negara dan kegunaan awam yang lain di negara ini.

Jadual 2.1.11 Jenis-jenis data dalam MySQL

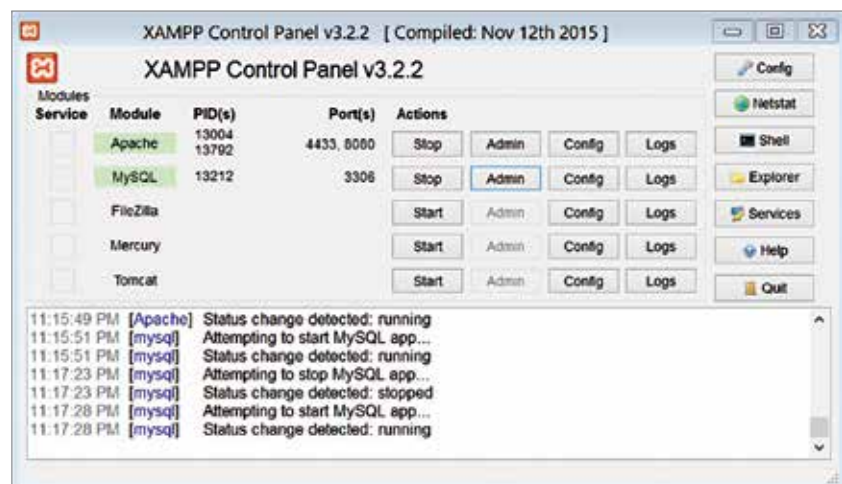
Jenis Data	Keterangan
CHAR	<i>String</i> yang tetap antara 0 hingga 255 patah perkataan. CHAR digunakan untuk lajur yang mengandungi data yang tetap dan seragam. Contoh: Jantina
VARCHAR	<i>String</i> yang boleh diubah antara 0 hingga 255 patah perkataan. VARCHAR digunakan untuk lajur yang mengandungi data yang tidak seragam. Contoh: Nama, Alamat
TINYTEXT	<i>String</i> dengan panjang maksimum 255 patah perkataan.
TEXT	<i>String</i> dengan panjang maksimum 65,535 patah perkataan.
INT	Nombor dalam julat -2,147,483,648 hingga 2,147,483,647 atau nombor positif 0 hingga 4,294,967,295.
FLOAT	Nombor kecil yang mempunyai titik perpuluhan.
DOUBLE	Nombor besar yang mempunyai titik perpuluhan.
DATE	Tarikh dengan format YYYY-MM-DD.
DATETIME	Tarikh dan waktu dengan format YYYY-MM-DD/HH:MM:SS.
TIME	Waktu dengan format HH:MM:SS

Membina Pangkalan Data

Pangkalan data boleh dibangunkan dengan menggunakan Sistem Pengurusan Pangkalan Data seperti *MS Access*, *Oracle*, *MySQL* dan sebagainya. Untuk membangunkan pangkalan data menggunakan *MySQL*, aplikasi ini boleh dilancarkan melalui tettingkap **XAMPP Control Panel** seperti dalam Rajah 2.1.14.

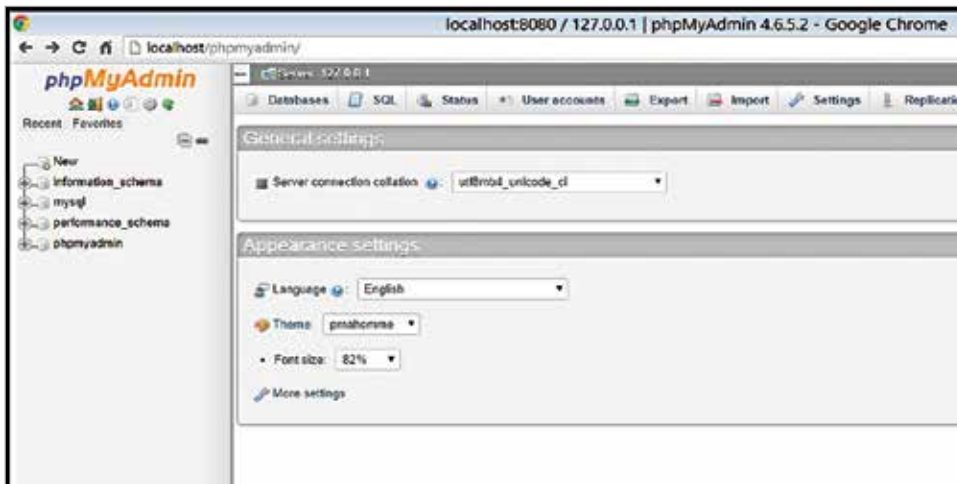
Tahukah Anda

Nama XAMPP merupakan singkatan dari X (iaitu empat sistem operasi seperti *Windows*, *Linux*, *Mac OS* dan *Solaris*), Apache, MySQL, PHP dan Perl. XAMPP adalah perisian sumber terbuka (*open source*) dan berfungsi sebagai pelayan yang berdiri sendiri (*localhost*).



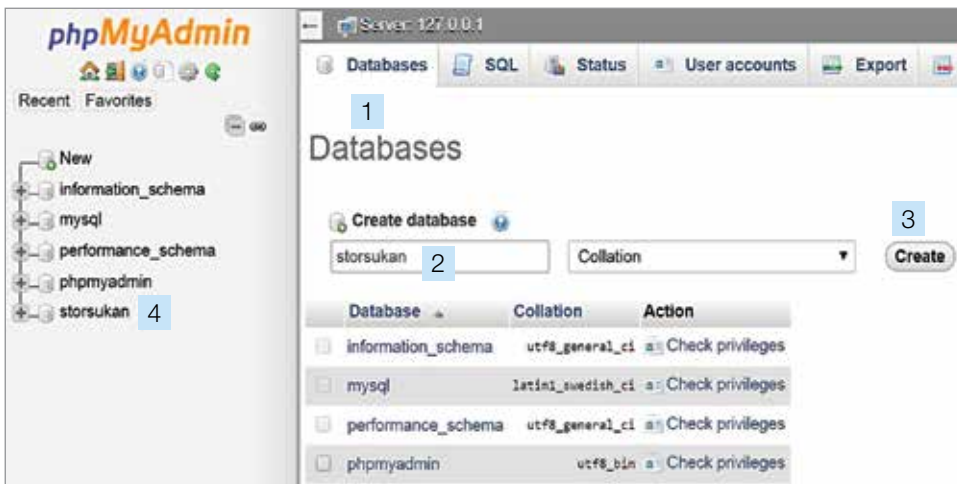
Rajah 2.1.14 XAMPP Control Panel

Apabila tettingkap *XAMPP Control Panel* dipaparkan, klik butang *Start* yang selari dengan modul Apache dan modul *MySQL*. Seterusnya, klik butang *Admin* yang terletak pada modul *MySQL* untuk melancarkan *phpMyAdmin* seperti dalam Rajah 2.1.15.



Rajah 2.1.15 phpMyAdmin

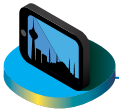
Klik pada tab *Databases* untuk mencipta pangkalan data yang anda inginkan. Contoh: Anda ingin mencipta pangkalan data bernama *storsukan*. Klik pada tab *Databases* dan berikan nama pangkalan data yang anda inginkan dan klik butang *Create*. Pangkalan data yang dicipta boleh dilihat di sebelah kiri skrin anda seperti yang ditunjukkan dalam Rajah 2.1.16.



Rajah 2.1.16 Mencipta pangkalan data

Penerangan:

- 1 Klik pada tab *Databases*.
- 2 Berikan nama pangkalan data yang anda inginkan. Contoh: *storsukan*.
- 3 Klik butang *Create*.
- 4 Pangkalan data telah berjaya dicipta.



MALAYSIAKU!

Di bawah Rancangan Malaysia Kesembilan (2006–2010), SIRIM telah diberikan mandat untuk membangunkan pangkalan data Penilaian Kitaran Hayat (LCA) atau *Life-Cycle Assessment* yang akan membolehkan industri menilai impak kitaran hayat proses pengeluaran dan pengilangan. Perkara ini menggalakkan penggunaan teknologi mesra alam dan pemakaian peraturan sendiri.

Mencipta Jadual (CREATE table)

Apabila pangkalan data telah siap dicipta, klik pada pangkalan data tersebut untuk mencipta jadual yang anda inginkan. Beberapa perkara yang berikut perlu dipertimbangkan sebelum mencipta jadual:

- 1 Kenal pasti jenis data dan saiz maksimum medan untuk setiap atribut.
- 2 Kenal pasti medan-medan yang akan menerima nilai *null*.
- 3 Kenal pasti medan yang akan menjadi kunci utama. Kunci utama digunakan untuk mengenal pasti rekod dalam jadual. Kunci utama perlu unik dan tidak boleh mengandungi nilai *null*.

Format:

```
CREATE TABLE nama_jadual (
  Nama_medan1 jenis_data (saiz) PRIMARY KEY,
  Nama_medan2 jenis_data (saiz) NOT NULL,
  Nama_medan3 jenis_data (saiz),
  .....
)
```

CONTOH 2

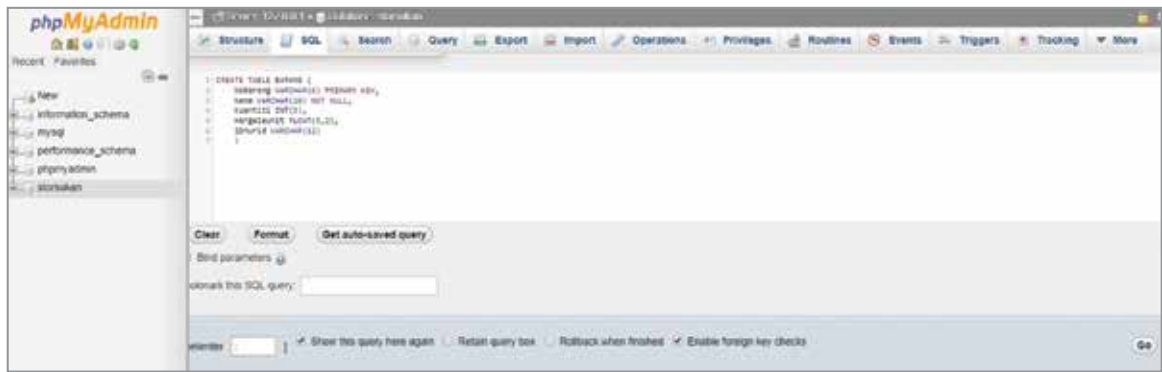
Contoh 2 menunjukkan cara mencipta jadual bernama BARANG yang mempunyai lima medan. Katakan jadual BARANG mengandungi struktur seperti dalam Jadual 2.1.12.

Jadual 2.1.12 Struktur jadual BARANG

BARANG

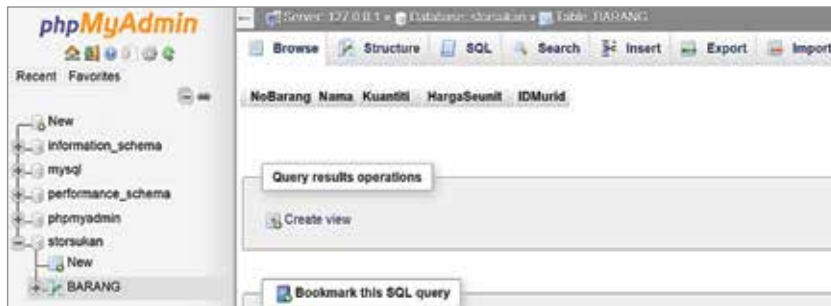
Nama Medan	Jenis Data	Keterangan
NoBarang	VARCHAR (6)	Kunci utama
Nama	VARCHAR (10)	<i>NOT NULL</i>
Kuantiti	INT (3)	
HargaSeunit	FLOAT (5,2)	
IDMurid	VARCHAR (12)	

NoBarang adalah kunci utama bagi jadual ini. Medan Nama ditakrif sebagai NOT NULL, iaitu bermaksud medan tersebut perlu ada nilai dan tidak boleh ditinggalkan kosong. Untuk mencipta jadual dalam pangkalan data storsukan, pastikan anda klik pada storsukan dan kemudian klik SQL untuk menulis pernyataan SQL. Seterusnya, klik GO untuk melaksanakan pernyataan SQL seperti dalam Rajah 2.1.17.



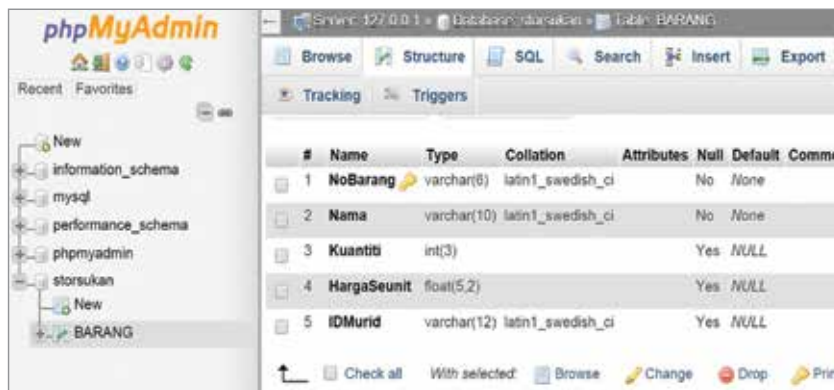
Rajah 2.1.17 Mencipta jadual

Rajah 2.1.18 menunjukkan jadual BARANG yang telah berjaya dicipta dengan menggunakan pernyataan CREATE TABLE. Jadual yang dicipta bernama BARANG akan berada di dalam pangkalan data storsukan yang anda bina sebelum ini.



Rajah 2.1.18 Jadual BARANG

Rajah 2.1.19 menunjukkan struktur jadual BARANG yang telah dicipta. Paparan ini boleh dicapai melalui tab *Structure*. Struktur jadual menerangkan jenis data, lebar medan dan nilai yang boleh dimasukkan ke dalam medan-medan tersebut.



Rajah 2.1.19 Struktur jadual BARANG



ALTER TABLE
Command
goo.gl/nalyNO

Tahukah
Anda 

Kebanyakan sistem pengurusan pangkalan data yang wujud pada masa ini mengimplementasikan bahasa SQL ke dalam versi mereka sendiri. Antara sistem pangkalan data yang menggunakan bahasa SQL ialah:

- ✓ Microsoft SQL Server
- ✓ Oracle
- ✓ MySQL
- ✓ Microsoft Access
- ✓ IBM DB2
- ✓ Sybase

Mengubah Struktur Jadual

Kadangkala anda ingin membuat perubahan pada struktur jadual yang telah dicipta. Antaranya seperti menambah medan, meminda atau mengubah jenis data dan saiz medan dan menghapuskan medan yang tidak diperlukan. Perkara ini boleh dilakukan dengan pernyataan ALTER TABLE.

Format:

```
ALTER TABLE nama_jadual  
pilihan perubahan;
```

Penerangan:

ALTER TABLE – ialah arahan asas untuk mengubah struktur jadual

Nama_jadual – ialah nama jadual yang akan diubah

Pilihan perubahan – contoh: menambah medan, meminda jenis data dan menghapus medan.

Sisip/Tambah Medan

Sesuatu jadual boleh ditambah medan dengan menggunakan pernyataan ADD.

Format:

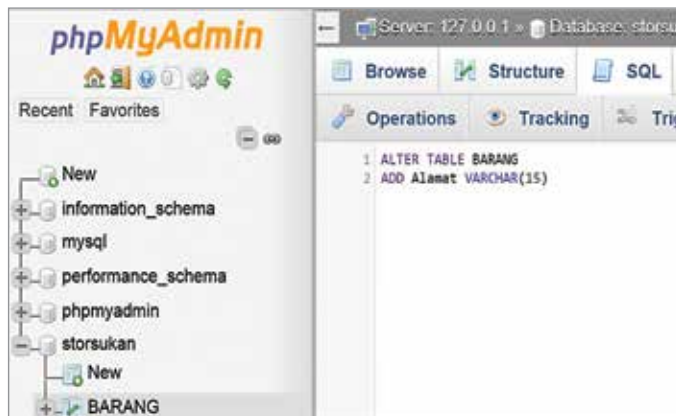
```
ALTER TABLE nama_jadual  
ADD nama_medan jenis data (saiz);
```

CONTOH 3

Contoh 3 menunjukkan pernyataan SQL untuk menambah medan alamat ke dalam jadual barang seperti dalam Rajah 2.1.20. Terdapat dua cara untuk menambah medan:

(a) Menambah medan baharu selepas medan terakhir.

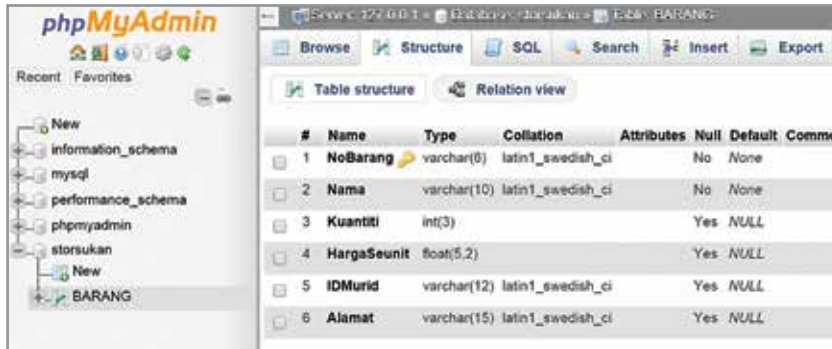
Pernyataan SQL:



Rajah 2.1.20 Menambah medan baharu Alamat



Output:



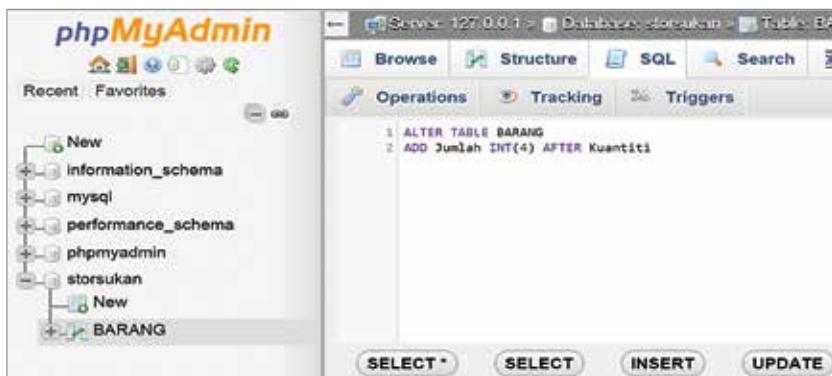
#	Name	Type	Collation	Attributes	Null	Default	Comme
1	NoBarang	varchar(6)	latin1_swedish_ci		No	None	
2	Nama	varchar(10)	latin1_swedish_ci		No	None	
3	Kuantiti	int(3)			Yes	NULL	
4	HargaSeunit	float(5,2)			Yes	NULL	
5	IDMurid	varchar(12)	latin1_swedish_ci		Yes	NULL	
6	Alamat	varchar(15)	latin1_swedish_ci		Yes	NULL	

Rajah 2.1.21 Medan baharu Alamat ditambah

Pernyataan ADD akan menambah medan baharu dalam sesuatu jadual seperti dalam Rajah 2.1.21. Secara amnya, apabila medan baharu ditambah, medan tersebut akan ditambah pada medan yang terakhir.

- (b) Menambah medan baharu di antara dua medan seperti dalam Rajah 2.1.22.

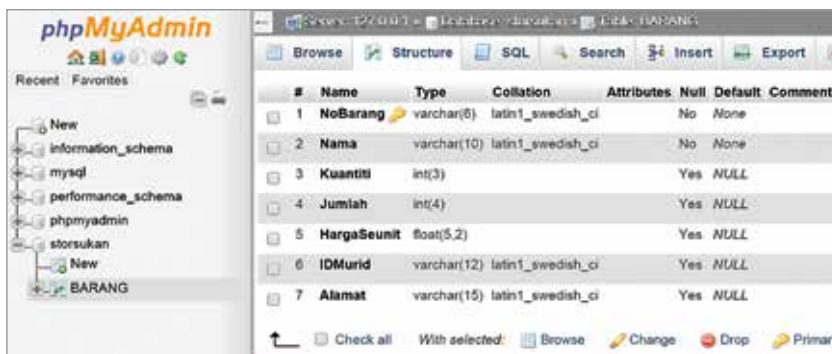
Pernyataan SQL:



```
1 ALTER TABLE BARANG
2 ADD Jumlah INT(4) AFTER Kuantiti
```

Rajah 2.1.22 Menambah medan baharu Jumlah

Output:



#	Name	Type	Collation	Attributes	Null	Default	Comments
1	NoBarang	varchar(6)	latin1_swedish_ci		No	None	
2	Nama	varchar(10)	latin1_swedish_ci		No	None	
3	Kuantiti	int(3)			Yes	NULL	
4	Jumlah	int(4)			Yes	NULL	
5	HargaSeunit	float(5,2)			Yes	NULL	
6	IDMurid	varchar(12)	latin1_swedish_ci		Yes	NULL	
7	Alamat	varchar(15)	latin1_swedish_ci		Yes	NULL	

Rajah 2.1.23 Medan baharu Jumlah ditambah





KERJAYA SAINS KOMPUTER

Kemahiran dalam SQL mewujudkan peluang kerjaya yang berikut:

- Pengatur cara SQL
- Pembangun .NET dan SQL
- Pembangun *Oracle* SQL



Pernyataan AFTER perlu digunakan untuk menambah medan baharu di antara dua medan yang sedia ada. Medan Jumlah ditambah di antara medan Kuantiti dan HargaSeunit seperti dalam Rajah 2.1.23.

Pinda/Ubah Medan

Sesuatu medan yang sedia ada boleh dipinda atau diubah. Untuk menukar jenis data atau saiz data, anda boleh menggunakan pernyataan MODIFY manakala untuk mengubah nama medan, anda perlu menggunakan pernyataan CHANGE.

Format:

```
ALTER TABLE nama_jadual  
MODIFY nama medan jenis data baharu (saiz baharu);
```

Atau

Format:

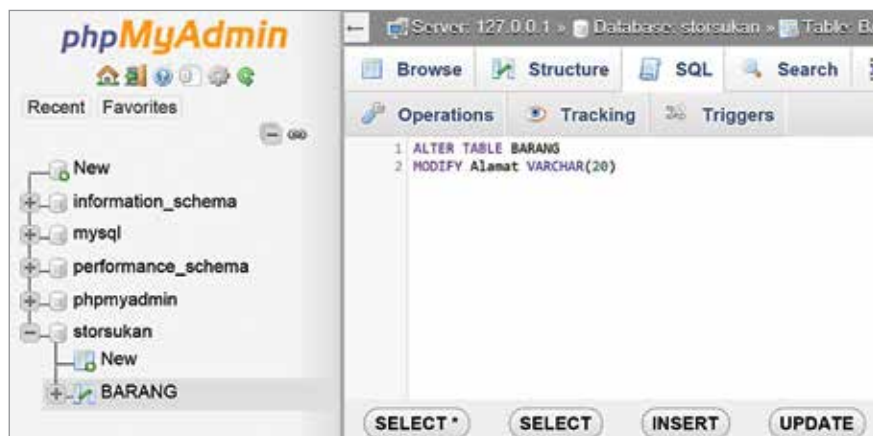
```
ALTER TABLE nama_jadual  
CHANGE nama medan nama medan baharu jenis data baharu (saiz baharu);
```

CONTOH 4

Contoh 4 menunjukkan pernyataan SQL untuk mengubah struktur medan.

(a) Mengubah saiz medan Alamat daripada 15 kepada 20. Rujuk Rajah 2.1.24 dan Rajah 2.1.25.

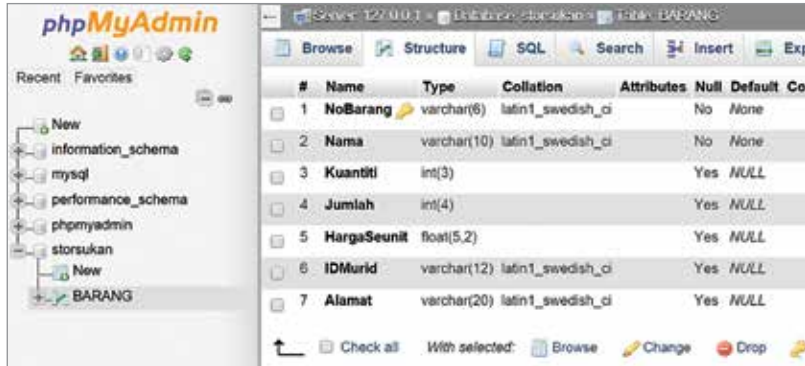
Pernyataan SQL:



Rajah 2.1.24 Mengubah saiz medan



Output:

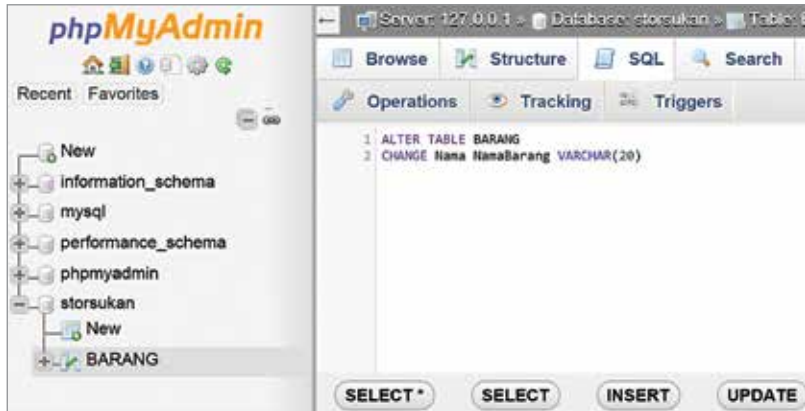


#	Name	Type	Collation	Attributes	Null	Default	Co
1	NoBarang	varchar(6)	latin1_swedish_ci		No	None	
2	Nama	varchar(10)	latin1_swedish_ci		No	None	
3	Kuantiti	int(3)			Yes	NULL	
4	Jumlah	int(4)			Yes	NULL	
5	HargaSeunit	float(5,2)			Yes	NULL	
6	IDMurid	varchar(12)	latin1_swedish_ci		Yes	NULL	
7	Alamat	varchar(20)	latin1_swedish_ci		Yes	NULL	

Rajah 2.1.25 Saiz medan Alamat diubah

- (b) Mengubah medan Nama kepada NamaBarang dan jenis data VARCHAR dengan saiz 20. Rujuk Rajah 2.1.26 dan Rajah 2.1.27.

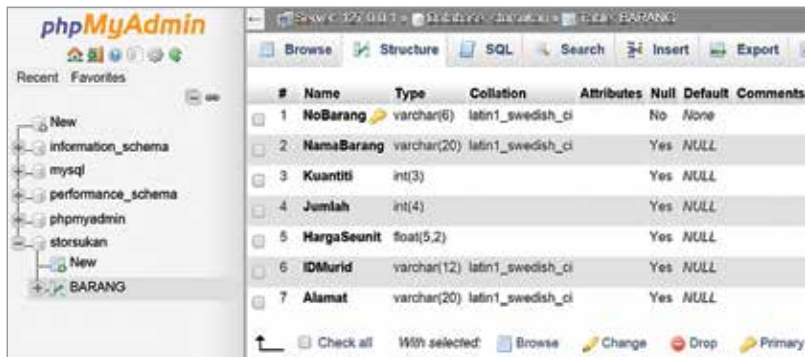
Pernyataan SQL:



```
1 ALTER TABLE BARANG
2 CHANGE Nama NamaBarang VARCHAR(20)
```

Rajah 2.1.26 Mengubah nama medan

Output:



#	Name	Type	Collation	Attributes	Null	Default	Comments
1	NoBarang	varchar(6)	latin1_swedish_ci		No	None	
2	NamaBarang	varchar(20)	latin1_swedish_ci		Yes	NULL	
3	Kuantiti	int(3)			Yes	NULL	
4	Jumlah	int(4)			Yes	NULL	
5	HargaSeunit	float(5,2)			Yes	NULL	
6	IDMurid	varchar(12)	latin1_swedish_ci		Yes	NULL	
7	Alamat	varchar(20)	latin1_swedish_ci		Yes	NULL	

Rajah 2.1.27 Medan NamaBarang dan saiz medan diubah



Hapus Medan

Medan yang telah dicipta boleh dihapuskan dengan menggunakan pernyataan DROP.

Format:

```
ALTER TABLE nama_jadual  
DROP nama medan;
```

CONTOH 5

Contoh 5 menunjukkan cara menghapuskan medan. Anda boleh menghapuskan satu medan atau lebih daripada satu medan dengan satu pernyataan.

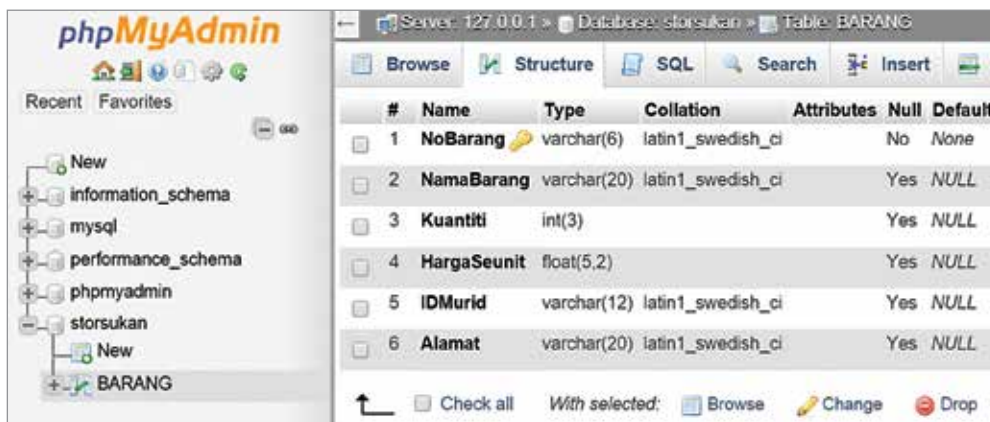
(a) Menghapuskan medan JUMLAH daripada jadual barang. Rujuk Rajah 2.1.28 dan Rajah 2.1.29.

Pernyataan SQL:



Rajah 2.1.28 Menghapus satu medan

Output:



Rajah 2.1.29 Medan Jumlah dihapus



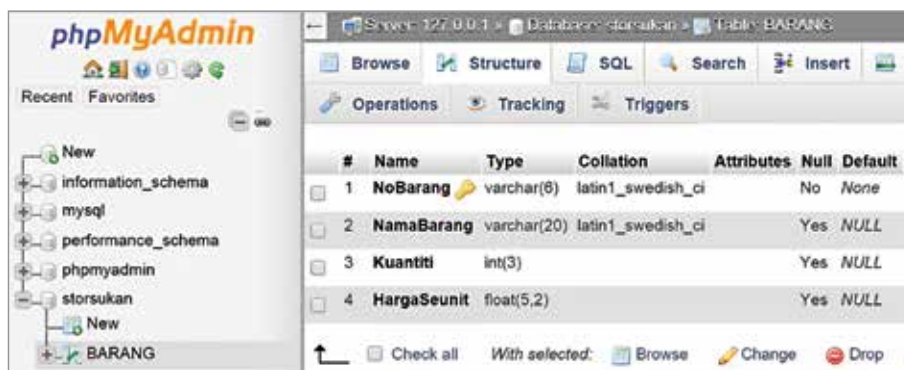
(b) Menghapuskan medan IDMurid dan Alamat. Rujuk Rajah 2.1.30 dan Rajah 2.1.31.

Pernyataan SQL:



Rajah 2.1.30 Menghapuskan dua medan

Output:



Rajah 2.1.31 Medan IDMurid dan Alamat terhapus

Hapus Jadual

Jadual yang tidak diperlukan boleh dihapuskan dengan menggunakan pernyataan DROP TABLE.

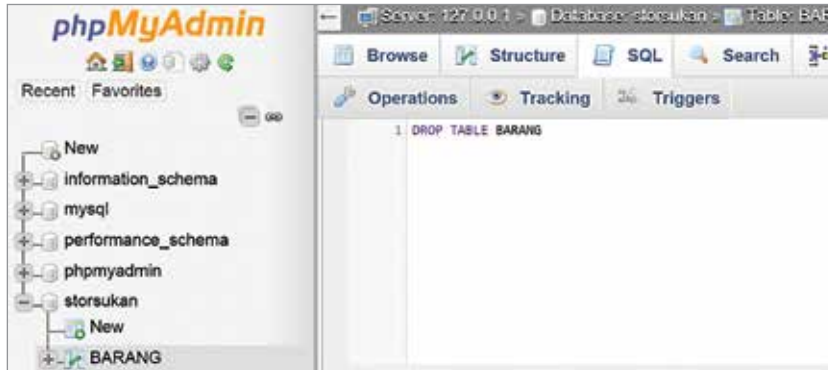
Format:

```
DROP TABLE nama jadual
```

CONTOH 6

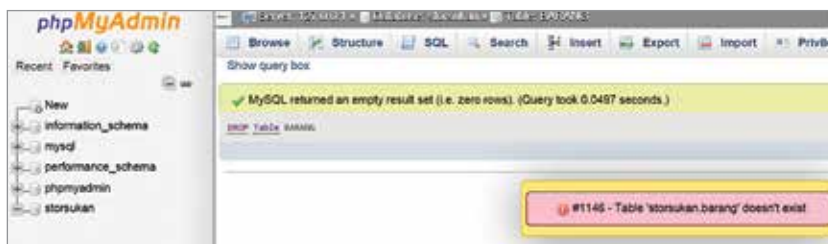
Contoh 6 menunjukkan cara untuk menghapus jadual barang. Rujuk Rajah 2.1.32 dan Rajah 2.1.33.

Pernyataan SQL:



Rajah 2.1.32 Menghapus jadual BARANG

Output:



Rajah 2.1.33 Jadual BARANG terhapus

AKTIVITI

3

Kerja kumpulan

Mencipta dan Mengubah Struktur Jadual

Katakan anda dan beberapa ahli kumpulan telah melawat sebuah hotel yang berhampiran dengan tempat tinggal anda. Anda telah menemui bual pegawai hotel tersebut berkenaan dengan Sistem Tempahan Bilik. Berdasarkan temu bual tersebut, anda dan ahli kumpulan telah mengenal pasti data-data yang diperlukan untuk proses penempahan bilik. Susulan hasil perbincangan anda dan ahli-ahli, jadual-jadual yang berikut telah dikenal pasti.

- BILIK
(NoBilik, Jenis, Harga)
- TEMPAHAN
(NoTetamu, TarikhMasuk, TarikhKeluar, NoBilik)
- TETAMU
(NoTetamu, Nama, Alamat, Pekerjaan)

Berdasarkan maklumat jadual yang diberikan, tulis pernyataan SQL untuk tujuan di bawah:

- 1 Cipta struktur jadual bilik, tempahan dan tetamu.
- 2 Tambah medan untuk merekod nombor plat kenderaan dalam jadual tetamu.
- 3 Hapuskan medan pekerjaan dari jadual tetamu.

2.1.4 SQL: Memanipulasi Data

Data Manipulation Language atau Bahasa Pengolahan Data (DML) berfungsi untuk memanipulasi data dalam pangkalan data. Setelah struktur jadual dicipta, anda boleh memasukkan data ke dalam jadual tersebut. Selain itu, anda juga boleh menghapuskan dan mengubah suai data yang telah dimasukkan.

Sisip/Tambah

Pernyataan INSERT INTO digunakan untuk memasukkan data ke dalam jadual seperti format yang berikut.

Format:

```
INSERT INTO nama_jadual (medan1, medan2, ..... medanN)
VALUES (nilai1, nilai2, .....nilaiN);
```



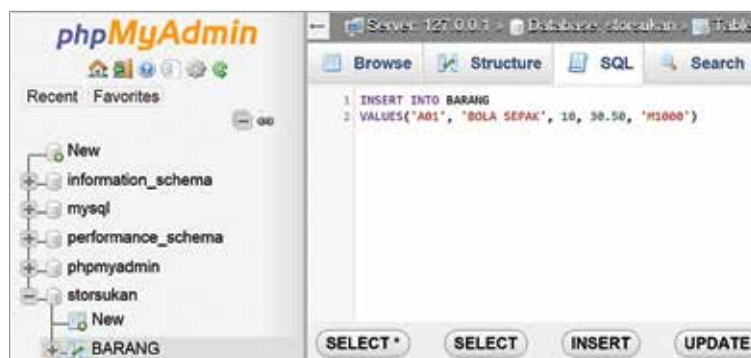
Pada tahun 2011, Malaysia Digital Economy Corporation (MDEC) telah dimandatkan untuk memacu peralihan negara ke arah pembangunan ekonomi digital bagi mencapai visi 2020, melalui inisiatif Digital Malaysia oleh Perdana Menteri Malaysia. Digital Malaysia dilancarkan secara rasminya pada tahun 2012 sebagai program transformasi negara bagi mencapai matlamat tersebut.

CONTOH 7

Contoh 7 menunjukkan cara menyisipkan data ke dalam jadual BARANG. Katakan anda mempunyai jadual BARANG yang terdiri daripada medan NoBarang, NamaBarang, Kuantiti, HargaSeunit dan IDMurid. Sisipan rekod boleh dilakukan dengan memasukkan data ke semua medan atau medan tertentu sahaja.

(a) Menyisipkan satu rekod ke semua medan dalam jadual BARANG.

Pernyataan SQL:



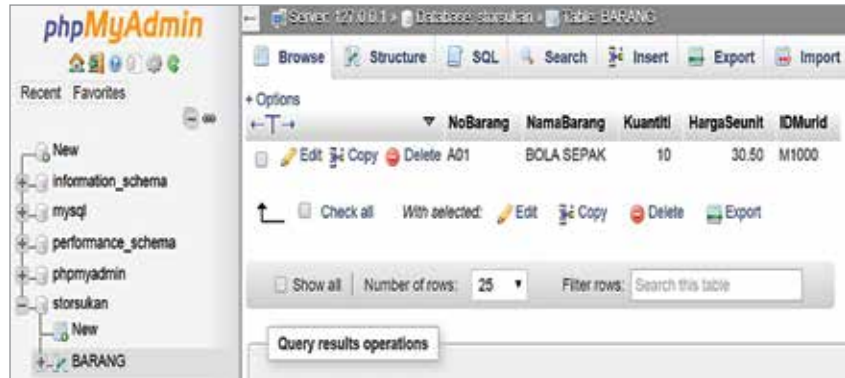
Rajah 2.1.34 Satu rekod disisipkan ke semua medan



APLIKASI HARIAN

Permohonan masuk ke universiti boleh dilakukan dalam talian. Bakal pelajar boleh mengisi borang secara dalam talian dengan menggunakan aplikasi yang dibangunkan oleh kerajaan dan juga badan swasta.

Output:

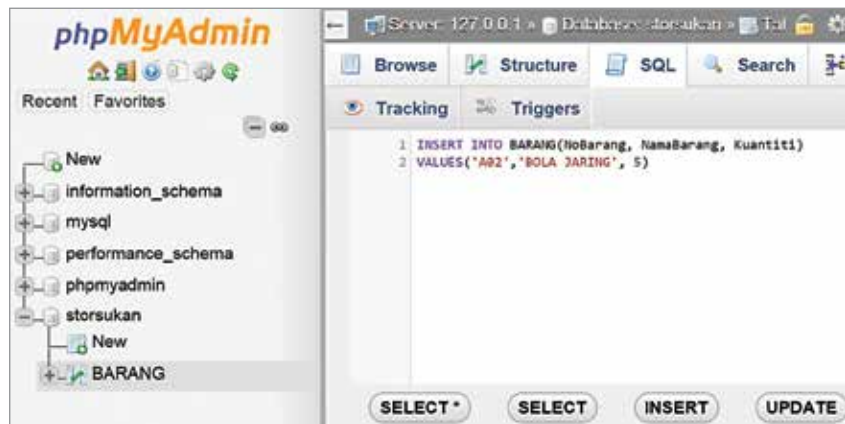


Rajah 2.1.35 Satu rekod disisipkan

Pernyataan dalam Rajah 2.1.34 akan menambah data barang baharu ke dalam semua medan yang terdapat dalam jadual. Data yang ditambah perlu mengikut turutan medan yang terdapat dalam jadual, iaitu bermula dengan NoBarang, NamaBarang, Kuantiti, HargaSeunit dan IDMurid. Setiap data yang ditakrifkan sebagai *string* seperti *char*, *varchar* dan *date* perlu disisipkan dengan menggunakan tanda (''). Output kepada pernyataan ini ditunjukkan dalam Rajah 2.1.35.

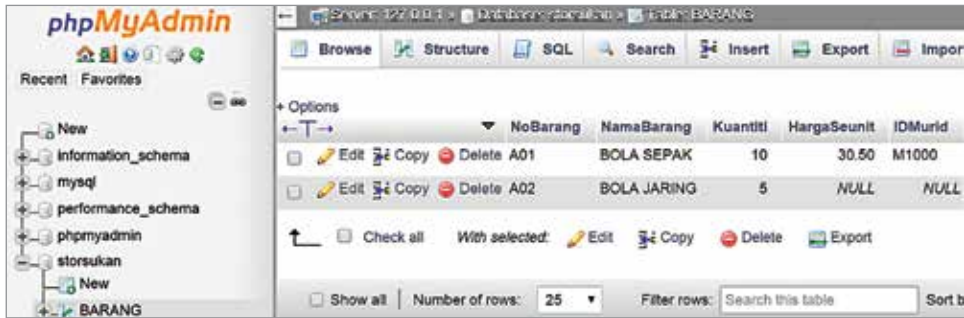
(b) Menyisipkan satu rekod pada medan tertentu sahaja.

Pernyataan SQL:



Rajah 2.1.36 Satu rekod disisipkan pada medan tertentu

Output:



The screenshot shows the phpMyAdmin interface for a database named 'storsukan'. The 'BARANG' table is selected, and its structure is displayed. The table has five columns: NoBarang, NamaBarang, Kuantiti, HargaSeunit, and IDMurid. Two records are visible in the table.

NoBarang	NamaBarang	Kuantiti	HargaSeunit	IDMurid
A01	BOLA SEPAK	10	30.50	M1000
A02	BOLA JARING	5	NULL	NULL

Rajah 2.1.37 Satu rekod disisip

Jika sisipan dilakukan untuk medan tertentu sahaja, anda perlu mengisytiharkan nama-nama medan yang ingin ditambah rekod. Pernyataan dalam Rajah 2.1.36 menunjukkan sisipan data pada medan NoBarang, NamaBarang dan Kuantiti sahaja. Oleh itu, medan yang tiada data akan menerima nilai *null* seperti yang ditunjukkan dalam Rajah 2.1.37.

PADAM

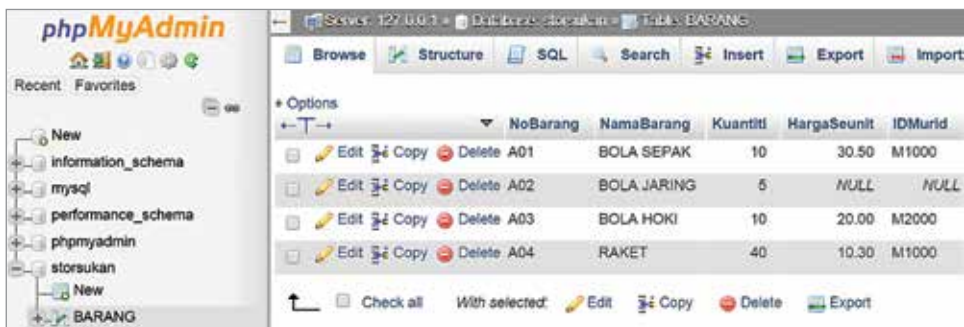
Pernyataan DELETE digunakan untuk memadamkan data dalam jadual. Untuk memadamkan rekod tertentu, pernyataan WHERE perlu digunakan.

Format:

```
DELETE FROM nama_jadual  
WHERE nama_medan = nilai;
```

CONTOH 8

Contoh 8 menunjukkan cara untuk memadamkan rekod dalam jadual. Terdapat dua cara, iaitu memadamkan rekod tertentu sahaja atau memadamkan semua rekod dalam jadual. Katakan terdapat empat rekod dalam jadual BARANG seperti dalam Rajah 2.1.38.



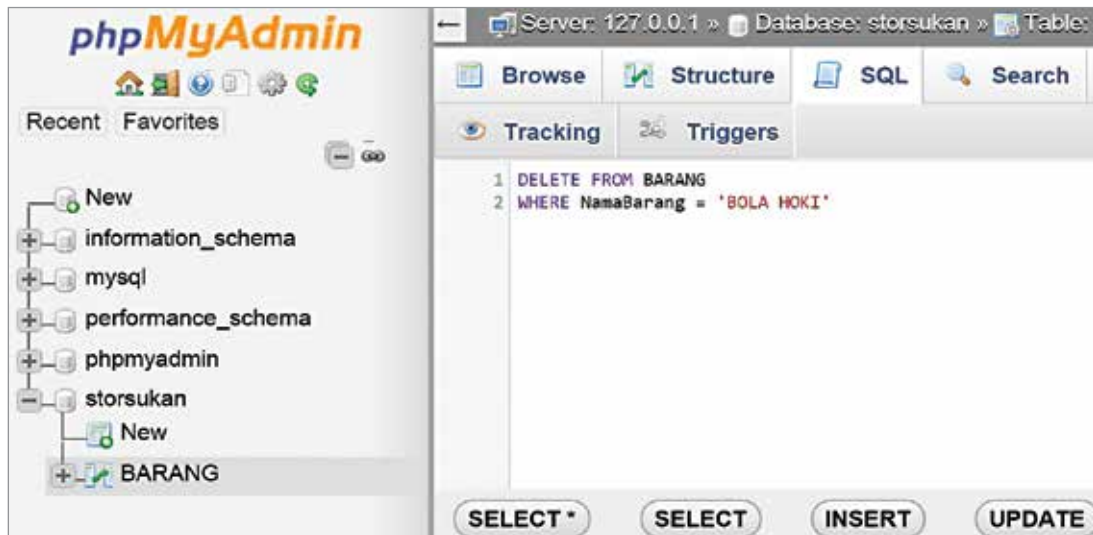
The screenshot shows the phpMyAdmin interface for the 'BARANG' table. The table structure and four records are displayed.

NoBarang	NamaBarang	Kuantiti	HargaSeunit	IDMurid
A01	BOLA SEPAK	10	30.50	M1000
A02	BOLA JARING	5	NULL	NULL
A03	BOLA HOKI	10	20.00	M2000
A04	RAKET	40	10.30	M1000

Rajah 2.1.38 Rekod dalam jadual BARANG

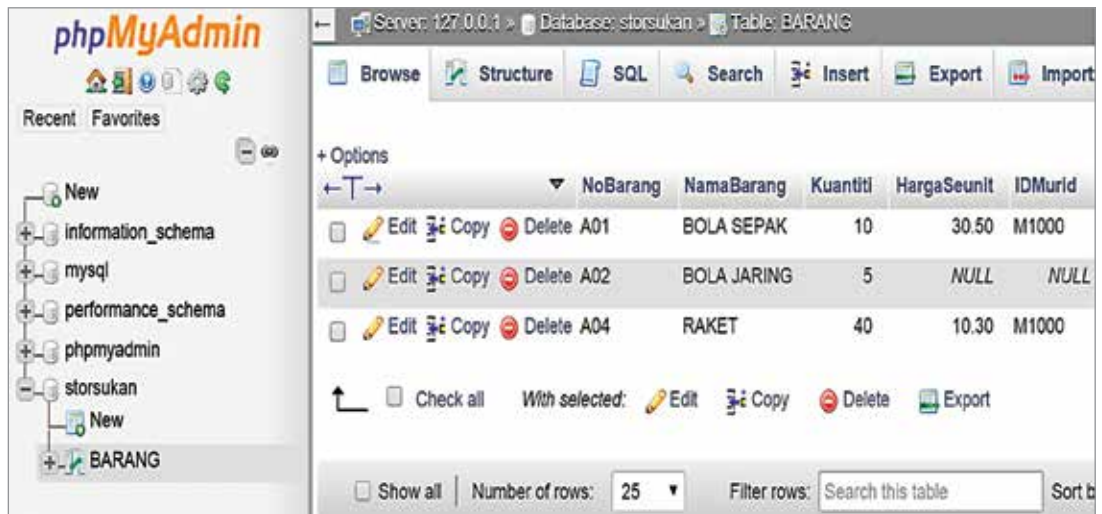
(a) Memadamkan rekod bola hoki dalam jadual BARANG.

Pernyataan SQL:



Rajah 2.1.39 Memadamkan rekod bola hoki

Output:



Rajah 2.1.40 Rekod bola hoki dipadamkan

Pernyataan DELETE dalam Rajah 2.1.39 akan menghapuskan rekod tertentu sahaja berdasarkan syarat yang dinyatakan dalam WHERE. Pernyataan ini akan memadamkan rekod bola hoki sahaja seperti yang ditunjukkan dalam Rajah 2.1.40.



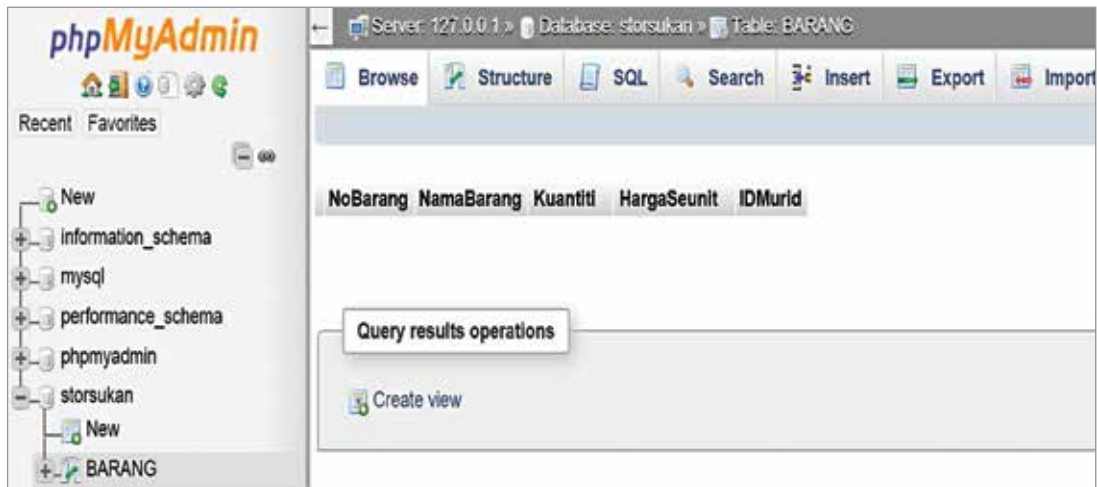
(b) Memadamkan semua rekod dalam jadual BARANG.

Pernyataan SQL:



Rajah 2.1.41 Memadamkan semua rekod

Output:



Rajah 2.1.42 Semua rekod dipadamkan

Pernyataan dalam Rajah 2.1.41 akan memadamkan semua rekod jadual BARANG seperti yang ditunjukkan pada bahagian output dalam Rajah 2.1.42.

Peringatan: Untuk menghapuskan struktur medan, pernyataan DROP perlu digunakan. Namun demikian, pernyataan DELETE boleh digunakan jika ingin menghapuskan data.

KEMASKINI

Untuk mengemaskinikan rekod yang sedia ada dalam jadual, anda boleh menggunakan pernyataan UPDATE.

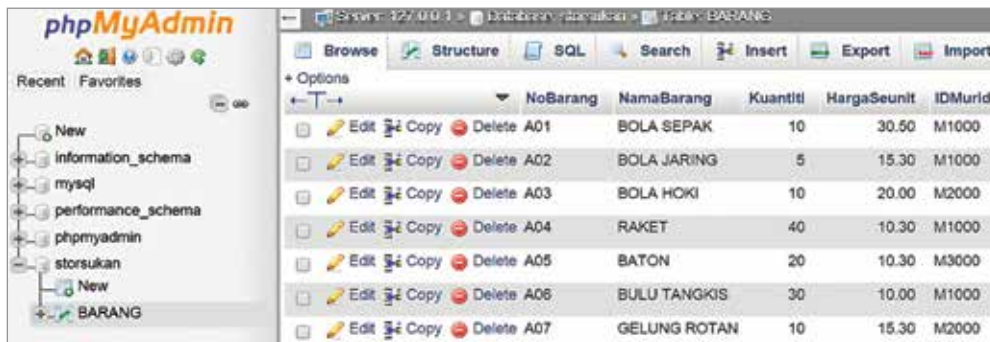
Pernyataan SET menunjukkan data baharu yang hendak dimasukkan dalam jadual manakala pernyataan WHERE digunakan untuk mengemaskinikan rekod tertentu sahaja.

Format:

```
UPDATE nama_jadual  
SET nama_medan = nilai_baharu  
WHERE nama_medan = nilai_tertentu;
```

CONTOH 9

Contoh 9 menunjukkan langkah untuk mengemaskinikan rekod dalam jadual. Anda boleh mengemaskinikan semua rekod atau rekod yang tertentu sahaja. Katakan anda mempunyai jadual BARANG dengan rekod seperti dalam Rajah 2.1.43.



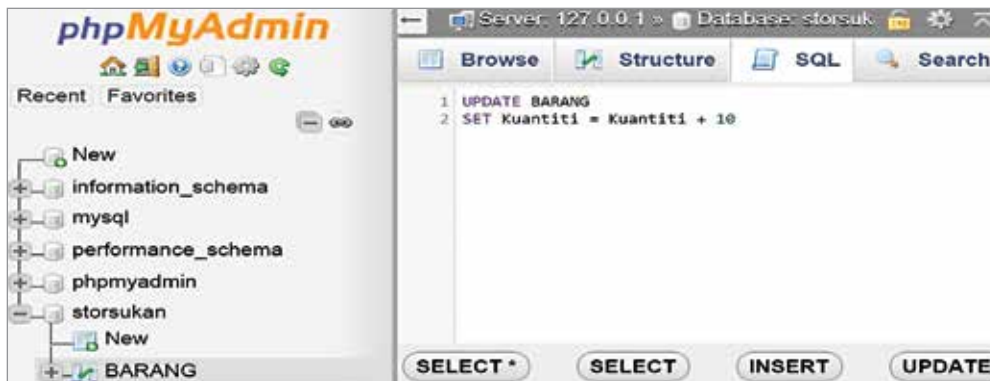
The screenshot shows the phpMyAdmin interface with the 'BARANG' table selected. The table has the following columns: NoBarang, NamaBarang, Kuantiti, HargaSeunit, and IDMurid. The data rows are as follows:

NoBarang	NamaBarang	Kuantiti	HargaSeunit	IDMurid
A01	BOLA SEPAK	10	30.50	M1000
A02	BOLA JARING	5	15.30	M1000
A03	BOLA HOKI	10	20.00	M2000
A04	RAKET	40	10.30	M1000
A05	BATON	20	10.30	M3000
A06	BULU TANGKIS	30	10.00	M1000
A07	GELUNG ROTAN	10	15.30	M2000

Rajah 2.1.43 Jadual BARANG

(a) Menambah stok barang sebanyak 10 unit untuk semua jenis barang.

Pernyataan SQL:



Rajah 2.1.44 Kemaskinikan semua rekod



Output:

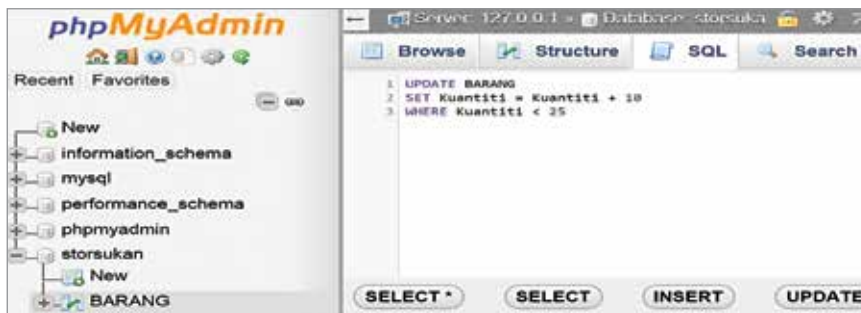
NoBarang	NamaBarang	Kuantiti	HargaSeunit	IDMurid
A01	BOLA SEPAK	20	30.50	M1000
A02	BOLA JARING	15	15.30	M1000
A03	BOLA HOKI	20	20.00	M2000
A04	RAKET	50	10.30	M1000
A05	BATON	30	10.30	M3000
A06	BULU TANGKIS	40	10.00	M1000
A07	GELUNG ROTAN	20	15.30	M2000

Rajah 2.1.45 Paparan rekod selepas dikemaskinikan

Rajah 2.1.44 menunjukkan pernyataan UPDATE untuk mengemaskinikan semua rekod dan outputnya ditunjukkan dalam Rajah 2.1.45.

(b) Menambah stok barang sebanyak 10 unit untuk setiap jenis barang yang bilangannya kurang daripada 25 unit.

Pernyataan SQL:



Rajah 2.1.46 Kemaskinikan rekod tertentu

Output:

NoBarang	NamaBarang	Kuantiti	HargaSeunit	IDMurid
A01	BOLA SEPAK	30	30.50	M1000
A02	BOLA JARING	25	15.30	M1000
A03	BOLA HOKI	30	20.00	M2000
A04	RAKET	50	10.30	M1000
A05	BATON	30	10.30	M3000
A06	BULU TANGKIS	40	10.00	M1000
A07	GELUNG ROTAN	30	15.30	M2000

Rajah 2.1.47 Paparan rekod tertentu dikemaskinikan

Rajah 2.1.46 menunjukkan langkah untuk mengemaskinikan rekod tertentu sahaja dengan menyatakan syarat dalam WHERE, iaitu hanya kuantiti barang yang kurang daripada 25 unit yang akan menerima penambahan 10 unit stok barang. Output pernyataan ini ditunjukkan dalam Rajah 2.1.47.

Berdasarkan Aktiviti 3, anda telah mencipta jadual-jadual yang berikut untuk Sistem Tempahan Bilik di sebuah hotel.

BILIK (NoBilik, Jenis, Harga)
 TEMPAHAN (NoTetamu, TarikhMasuk, TarikhKeluar, NoBilik)
 TETAMU (NoTetamu, Nama, Alamat, Pekerjaan)

Tulis arahan SQL untuk melakukan perkara yang berikut:

- 1 Masukkan maklumat bilik, tetamu dan tempahan yang dibuat oleh pelanggan.
- 2 Pihak hotel ingin menaikkan kadar harga setiap bilik sebanyak 10%. Kemaskinikan harga baharu setiap bilik itu.
- 3 Hapuskan maklumat pelanggan yang telah mendaftar masuk pada 1 November 2016.

2.1.5 SQL: Mendapatkan Semula Data

Sebelum ini, tiga arahan bahasa pengolahan data iaitu UPDATE, INSERT dan DELETE telah dibincangkan. Arahan ini membolehkan anda melakukan perubahan terhadap rekod dalam jadual. Arahan yang seterusnya ialah pernyataan SELECT yang membolehkan anda mencapai rekod daripada jadual.

Pernyataan SELECT

Pernyataan SELECT digunakan untuk mencapai data daripada jadual dalam pangkalan data. Pernyataan SELECT terdiri daripada tiga pernyataan asas seperti dalam Jadual 2.1.13.

Jadual 2.1.13 Struktur asas pernyataan SQL

Pernyataan	Keterangan
SELECT	Senaraikan nama medan yang ingin dipaparkan
FROM	Nama jadual yang hendak dicapai data
WHERE	Menentukan rekod tertentu yang ingin dipaparkan. Tanpa pernyataan ini, semua rekod akan dipaparkan.

Pernyataan SELECT dan FROM perlu dinyatakan apabila mencapai data manakala pernyataan WHERE adalah untuk pilihan. Pernyataan WHERE hanya diperlukan jika anda ingin mencapai data yang tertentu. Sekiranya terdapat lebih daripada satu syarat, pernyataan AND boleh digunakan.

Format:

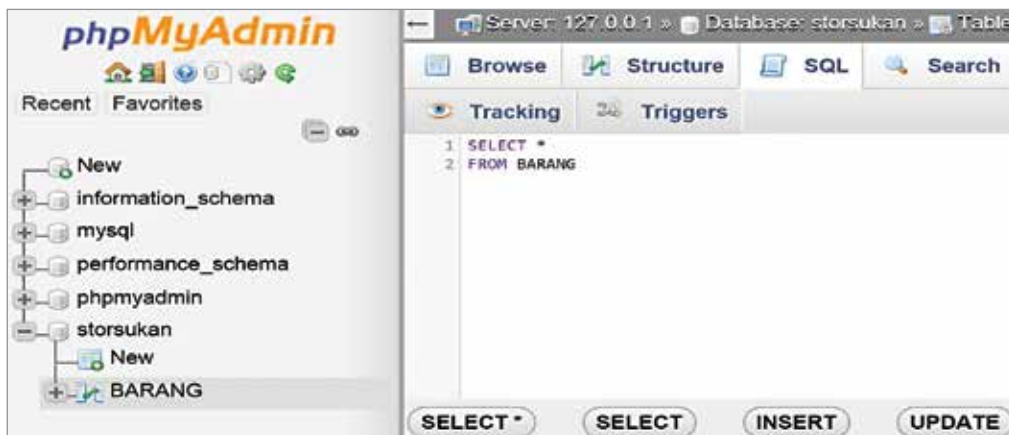
```
SELECT nama_medan  
FROM nama_jadual  
WHERE syarat1  
AND syarat2  
AND syarat...
```

CONTOH 10

Contoh 10 menunjukkan cara untuk mencapai data daripada jadual. Anda akan merujuk jadual BARANG dalam Rajah 2.1.43.

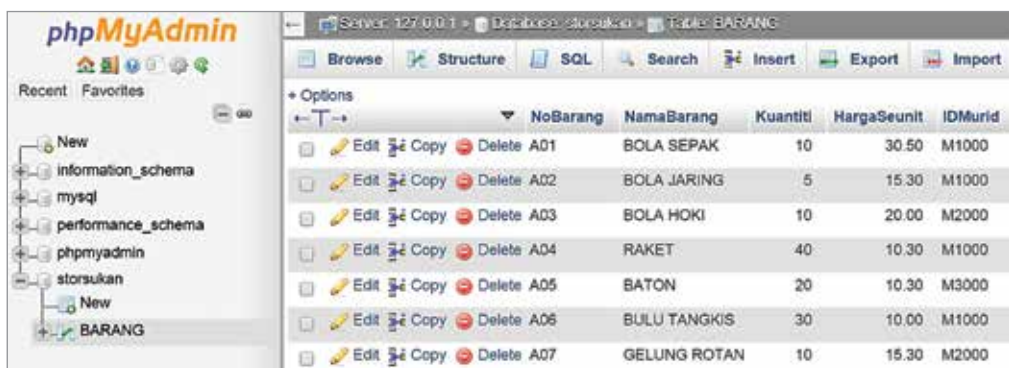
(a) Mencapai semua medan dan rekod dalam jadual BARANG.

Pernyataan SQL:



Rajah 2.1.48 Mencapai semua rekod

Output:



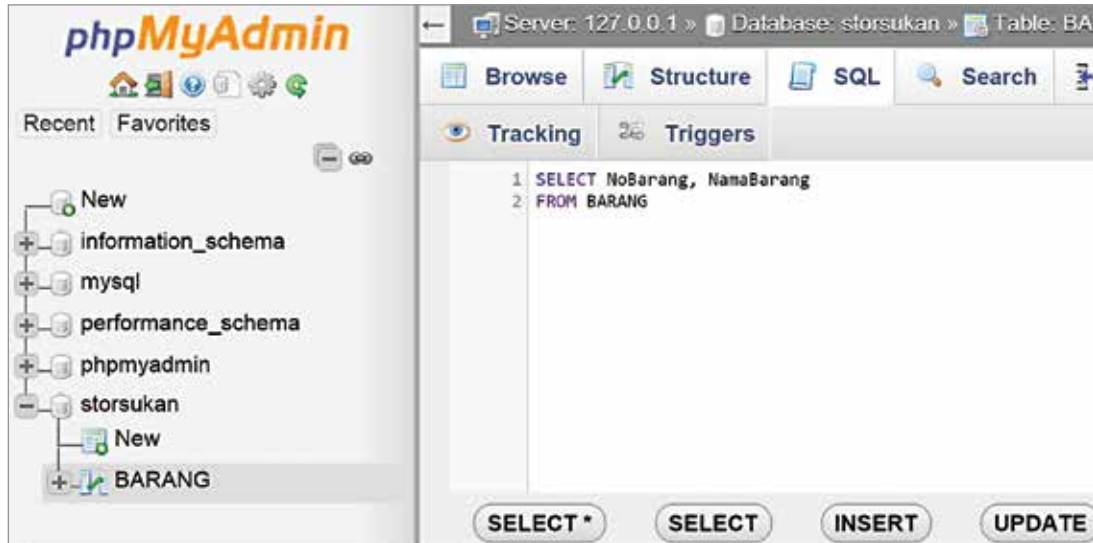
NoBarang	NamaBarang	Kuantiti	HargaSeunit	IDMurid
A01	BOLA SEPAK	10	30.50	M1000
A02	BOLA JARING	5	15.30	M1000
A03	BOLA HOKI	10	20.00	M2000
A04	RAKET	40	10.30	M1000
A05	BATON	20	10.30	M3000
A06	BULU TANGKIS	30	10.00	M1000
A07	GELUNG ROTAN	10	15.30	M2000

Rajah 2.1.49 Semua rekod dalam jadual BARANG

Pernyataan dalam Rajah 2.1.48 memaparkan semua rekod dan medan yang terdapat dalam jadual BARANG. Penggunaan asterisk (*) merupakan singkatan untuk memaparkan semua medan yang terdapat dalam jadual. Output kepada pernyataan ini ditunjukkan dalam Rajah 2.1.49.

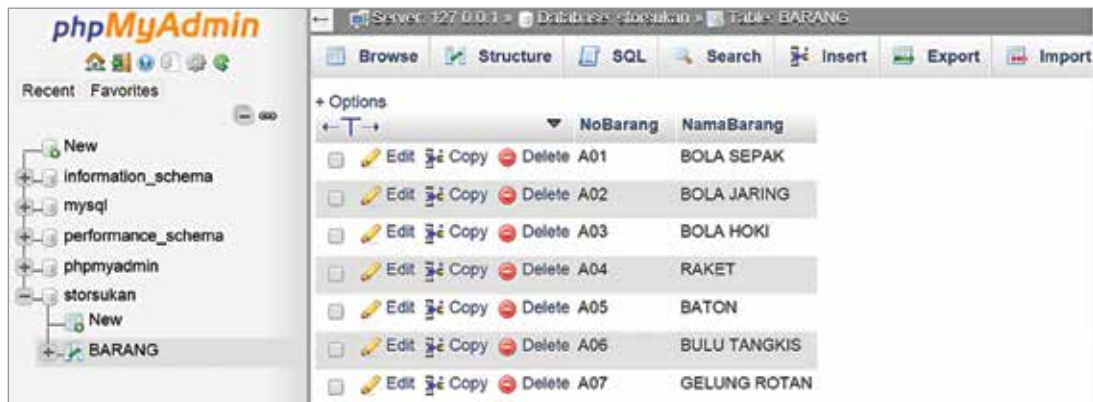
(b) Mencapai semua rekod dan medan tertentu seperti NoBarang dan NamaBarang.

Pernyataan SQL:



Rajah 2.1.50 Mencapai rekod medan tertentu

Output:



Rajah 2.1.51 Paparan rekod medan tertentu

Pernyataan dalam Rajah 2.1.50 memaparkan semua rekod tetapi merujuk medan tertentu sahaja, iaitu NoBarang dan NamaBarang. Untuk mencapai rekod medan tertentu, anda perlu menyenaraikan nama-nama medan yang hendak dicapai. Output kepada pernyataan ini ditunjukkan dalam Rajah 2.1.51.

GROUP BY

Pernyataan GROUP BY digunakan untuk menggabungkan beberapa rekod dalam jadual kepada kumpulan kecil.

Format:

```
SELECT nama_medan  
FROM nama_jadual  
GROUP BY nama_medan;
```



*SQL Aggregate
Functions
goo.gl/1cLZIR*

CONTOH 11

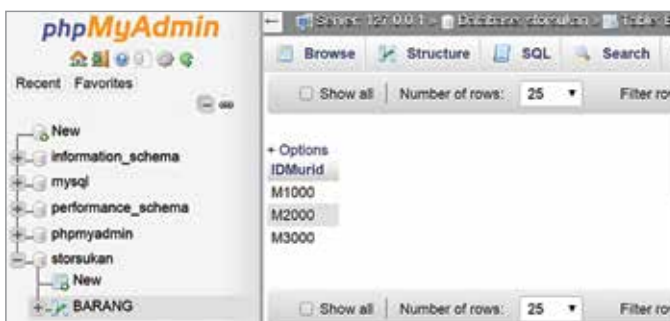
Rajah 2.1.52 menunjukkan cara mencapai data dengan menggunakan GROUP BY. Katakan anda ingin menyenaraikan IDMurid yang meminjam alatan di dalam stor.

Pernyataan SQL:



Rajah 2.1.52 Mencapai data dengan menggunakan GROUP BY

Output:



Rajah 2.1.53 Paparan data yang memenuhi syarat GROUP BY

Pernyataan dalam Rajah 2.1.53 memaparkan IDMurid yang terdapat dalam jadual BARANG dengan menggabungkan rekod yang sama kepada satu. Oleh itu, data yang berulang dipaparkan sekali sahaja.

ORDER BY

Setelah data dimasukkan ke dalam jadual, anda boleh memanggil data dalam bentuk susunan menaik atau menurun dengan menggunakan pernyataan ORDER BY. Pada asasnya, ORDER BY akan mengisih data mengikut susunan menaik. Jika anda ingin mengisih mengikut susunan menurun, anda perlu menambah perkataan DESC.

Format:

```
SELECT nama medan  
FROM nama jadual  
ORDER BY nama medan;
```

CONTOH 12

Rajah 2.1.54 menunjukkan cara untuk mencapai data dalam bentuk susunan menaik dan menurun.

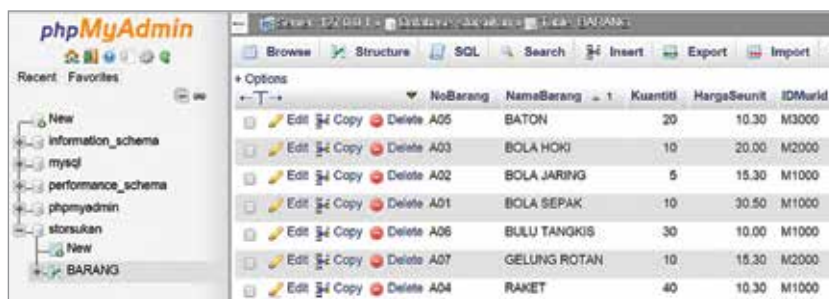
(a) Menyenaraikan maklumat semua barang dalam susunan menaik mengikut nama barang.

Pernyataan SQL:



Rajah 2.1.54 Mencapai data dengan susunan menaik

Output:



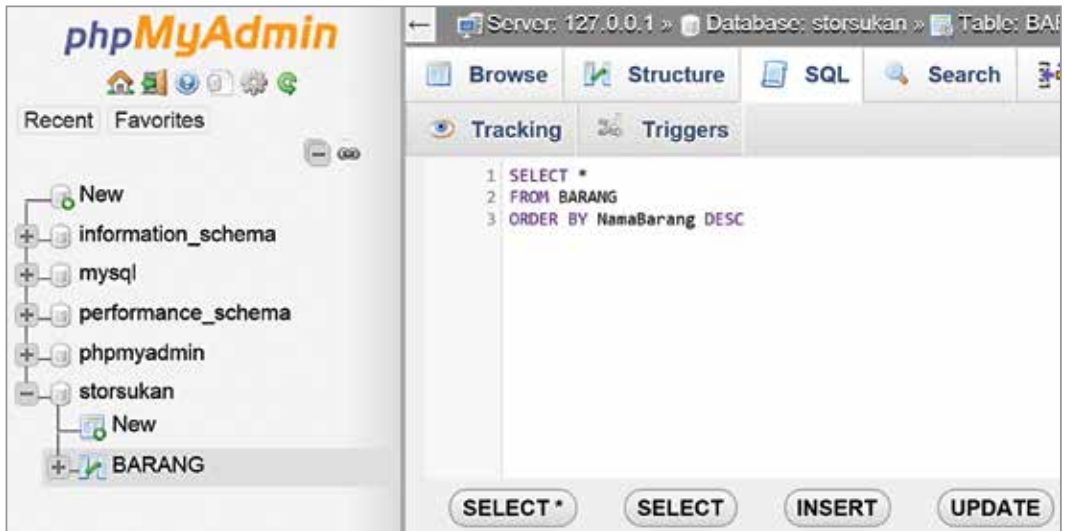
NoBarang	NamaBarang	Kuantiti	HargaSeunit	IDMurid
A05	BATON	20	10.30	M3000
A03	BOLA HOKI	10	20.00	M2000
A02	BOLA JARING	5	15.30	M1000
A01	BOLA SEPAK	10	30.50	M1000
A06	BULU TANGKIS	30	10.00	M1000
A07	GELUNG ROTAN	10	15.30	M2000
A04	RAKET	40	10.30	M1000

Rajah 2.1.55 Paparan data mengikut susunan menaik

Pernyataan dalam Rajah 2.1.54 akan mengisih data jadual BARANG secara menaik mengikut medan NamaBarang dan outputnya ditunjukkan dalam Rajah 2.1.55.

- (b) Menyeneraikan maklumat semua barang dalam susunan menurun mengikut nama barang.

Pernyataan SQL:



Rajah 2.1.56 Mencapai data mengikut susunan menurun

Output:

NoBarang	NamaBarang	Kuantiti	HargaSeunit	IDMurid
A04	RAKET	40	10.30	M1000
A07	GELUNG ROTAN	10	15.30	M2000
A06	BULU TANGKIS	30	10.00	M1000
A01	BOLA SEPAK	10	30.50	M1000
A02	BOLA JARING	5	15.30	M1000
A03	BOLA HOKI	10	20.00	M2000
A05	BATON	20	10.30	M3000

Rajah 2.1.57 Paparan data mengikut susunan menurun

Pernyataan dalam Rajah 2.1.56 akan mengisih data jadual BARANG secara menurun mengikut medan NamaBarang. Perkataan DESC bermaksud susunan data adalah secara menurun seperti dalam Rajah 2.1.57.

Tulis arahan SQL bagi setiap pernyataan yang diberikan.

- 1 Paparkan semua maklumat produk yang terdapat dalam stok.
- 2 Paparkan nama dan kuantiti produk yang terdapat dalam stok.
- 3 Senaraikan nama produk yang diperbuat daripada kayu getah.
- 4 Senaraikan semua maklumat meja yang terdapat dalam stok.
- 5 Senaraikan nama produk yang tidak diperbuat daripada kayu jati.
- 6 Paparkan semua maklumat produk yang diperbuat daripada kayu nyatuh dan jati.
- 7 Senaraikan nama dan harga meja makan yang diperbuat daripada kayu getah.
- 8 Senaraikan nama dan harga produk yang diperbuat daripada kayu getah. Susun output secara menaik mengikut nama.

Paparkan **output** untuk setiap ungkapan. Gunakan maklumat dalam jadual yang berikut.

PRODUK

NoProduk	Nama	Harga	Kuantiti	JenisKayu
P01	MEJA KOPI	200	100	GETAH
P02	MEJA KOMPUTER	375	1000	LAPIS
P03	MEJA MAKAN	800	500	JATI
P04	MEJA KOMPUTER	500	500	NYATUH
P05	MEJA SEKOLAH	100	2000	GETAH
P06	KERUSI	60	1000	GETAH
P07	KERUSI	100	500	JATI
P08	KERUSI	110	500	NYATUH
P09	MEJA MAKAN	300	200	GETAH
P10	MEJA KOMPUTER	120	1000	GETAH

2.1.6 Fungsi Agregat

SQL mempunyai lima fungsi agregat yang boleh dilaksanakan terhadap sesuatu medan dalam jadual untuk menghasilkan satu nilai penyatuan bagi medan berkenaan. Fungsi ini menggabungkan sekumpulan data dan mengembalikan nilai tunggal sebagai hasilnya. Jadual 2.1.14 menunjukkan lima fungsi agregat yang sering digunakan.

Jadual 2.1.14 Fungsi Agregat

Fungsi Agregat	Keterangan
SUM	Mengembalikan jumlah nilai medan tertentu
MIN	Mengembalikan nilai terendah medan tertentu
MAX	Mengembalikan nilai tertinggi medan tertentu
AVG	Mengembalikan nilai purata medan tertentu
COUNT	Mengembalikan jumlah bilangan rekod

SUM

Fungsi SUM digunakan untuk menghitung jumlah nilai dari medan tertentu dalam sesuatu jadual.

Format:

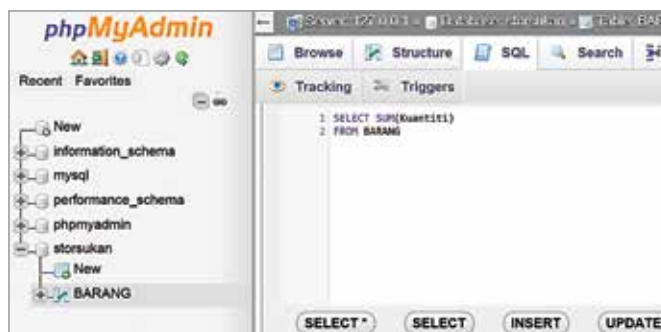
```
SELECT SUM (nama medan)
FROM nama jadual;
```

CONTOH 13

Contoh 13 menunjukkan cara menggunakan fungsi SUM. Katakan anda ingin menghitung jumlah barang dalam stok. Anda boleh menghitung jumlah keseluruhan barang dalam stok ataupun jumlah barang yang memenuhi syarat tertentu dengan menggunakan pernyataan WHERE.

(a) Mendapatkan jumlah barang yang terdapat dalam stok.

Pernyataan SQL:



Rajah 2.1.58 Mencapai data dengan menggunakan fungsi SUM

Output:



Rajah 2.1.59 Paparan data untuk fungsi SUM

Pernyataan dalam Rajah 2.1.58 akan menghitung jumlah keseluruhan barang yang terdapat dalam stok Outputnya ditunjukkan dalam Rajah 2.1.59.

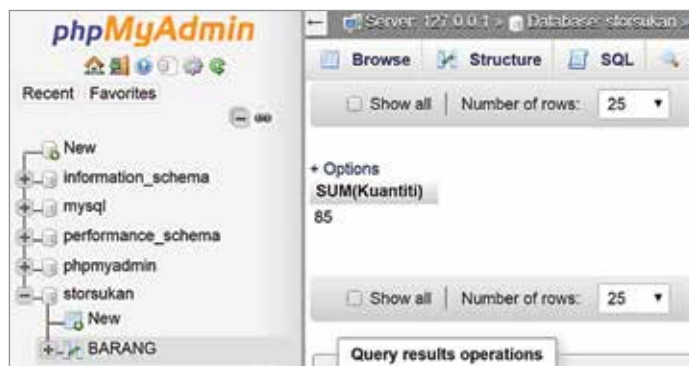
(b) Mendapatkan jumlah barang yang dipinjam oleh IDMurid M1000.

Pernyataan SQL:



Rajah 2.1.60 Mencapai data dengan menggunakan fungsi SUM

Output:



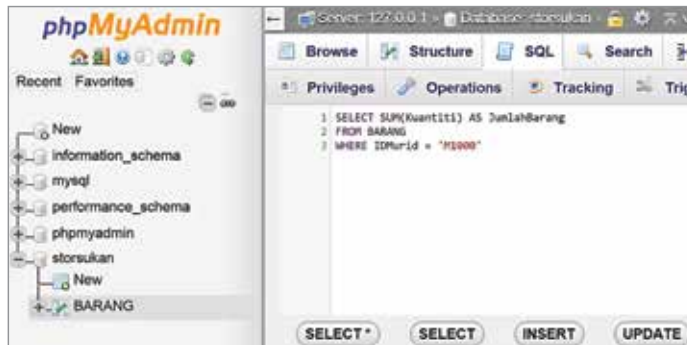
Rajah 2.1.61 Paparan data untuk fungsi SUM



Pernyataan dalam Rajah 2.1.60 akan menghitung jumlah barang yang dipinjam oleh IDMurid M1000 sahaja. Perhitungan ini dapat dilakukan dengan menambah pernyataan pada syarat WHERE yang hanya merujuk IDMurid M1000.

Lihat nama medan yang dipaparkan dalam Rajah 2.1.61, iaitu **SUM(KUANTITI)**. Anda mungkin inginkan nama yang berbeza yang lebih bermakna. Anda boleh berbuat demikian dengan menggunakan ekspresi **AS** seperti dalam Rajah 2.1.62.

Pernyataan SQL:



Rajah 2.1.62 Mencapai data dengan menggunakan fungsi SUM

Output:



Rajah 2.1.63 Paparan data untuk fungsi SUM

Pernyataan dalam Rajah 2.1.62 akan menghitung jumlah barang yang dipinjam oleh murid tertentu sahaja dan menamakan medan **SUM(KUANTITI)** kepada JumlahBarang seperti paparan output dalam Rajah 2.1.63.

MIN

Fungsi MIN, iaitu minimum, akan mengembalikan nilai terendah daripada medan tertentu dalam jadual.

Format:

```
SELECT MIN (nama medan)
FROM nama jadual;
```

CONTOH 14

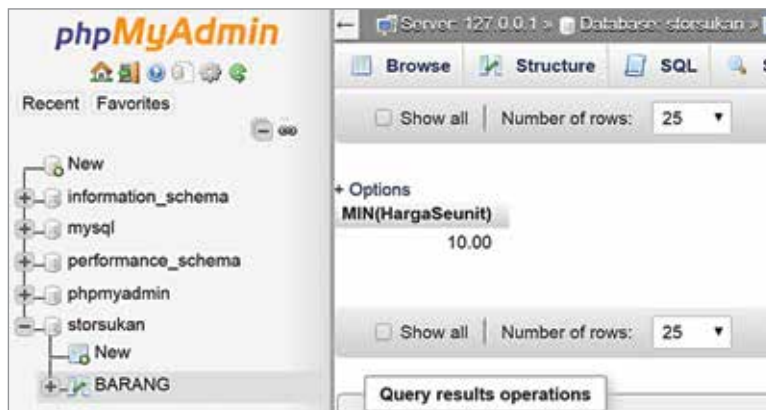
Contoh 14 menunjukkan cara menggunakan fungsi MIN. Katakan anda ingin mendapatkan harga barang yang paling murah yang terdapat dalam jadual BARANG seperti dalam Rajah 2.1.64.

Pernyataan SQL:



Rajah 2.1.64 Mencapai data dengan menggunakan fungsi MIN

Output:



Rajah 2.1.65 Paparan data untuk fungsi MIN

Rajah 2.1.65 memaparkan harga barang yang paling rendah yang terdapat dalam jadual BARANG.

MAX

Fungsi MAX, iaitu maksimum, akan mengembalikan nilai tertinggi daripada medan tertentu.

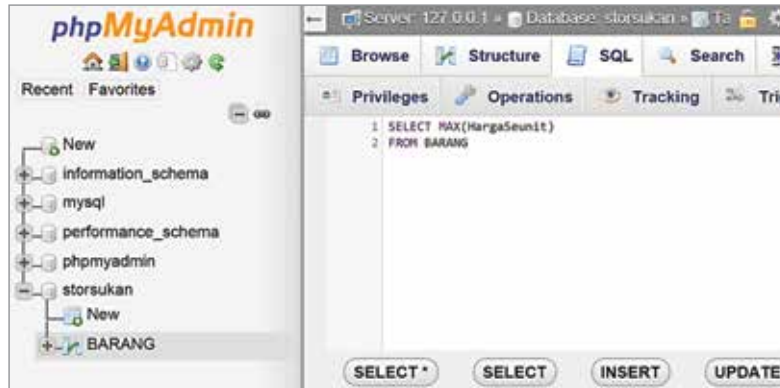
Format:

```
SELECT MAX (nama medan)
FROM nama jadual;
```

CONTOH 15

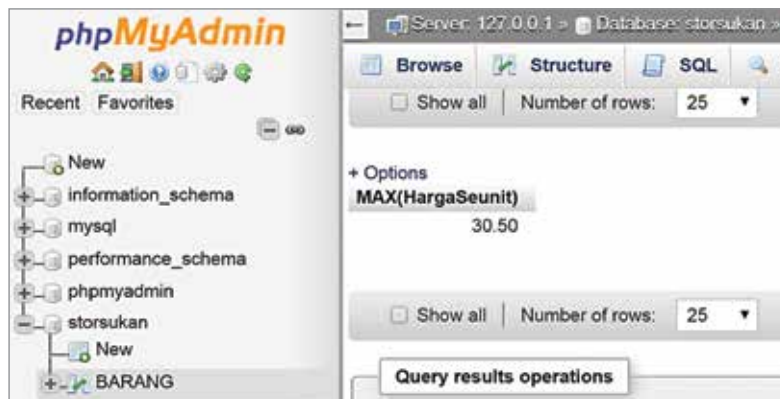
Contoh 15 menunjukkan cara menggunakan fungsi MAX. Katakan anda ingin mendapatkan harga barang yang paling mahal yang terdapat dalam jadual BARANG seperti dalam Rajah 2.1.66.

Pernyataan SQL:



Rajah 2.1.66 Mencapai data dengan menggunakan fungsi MAX

Output:



Rajah 2.1.67 Paparan data untuk fungsi MAX

Rajah 2.1.67 memaparkan harga barang yang paling mahal yang terdapat dalam jadual BARANG.

AVG

Fungsi AVG akan mengembalikan purata nilai sesuatu medan.

Format:

```
SELECT AVG (nama medan)
FROM nama jadual;
```

CONTOH 16

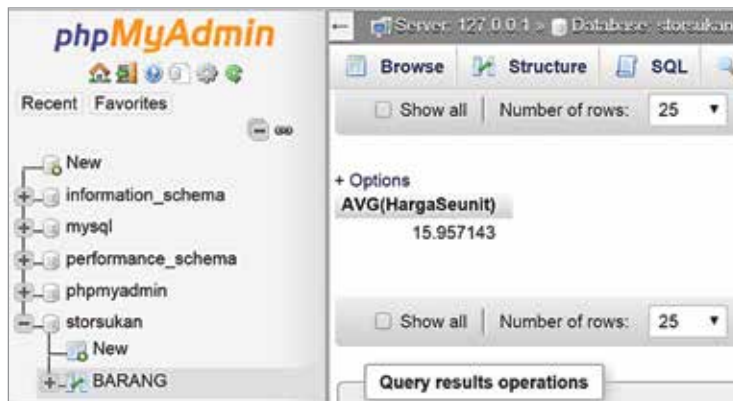
Contoh 16 menunjukkan cara menggunakan fungsi AVG. Katakan anda ingin mendapatkan harga purata barang yang terdapat dalam jadual BARANG.

Pernyataan SQL:



Rajah 2.1.68 Mencapai data dengan menggunakan fungsi AVG

Output:



Rajah 2.1.69 Paparan data untuk fungsi AVG

Rajah 2.1.69 memaparkan harga purata barang yang terdapat dalam jadual BARANG.

COUNT

Fungsi COUNT akan menghitung jumlah rekod tertentu daripada sesuatu jadual.

Format:

```
SELECT COUNT (nama medan)  
FROM nama jadual;
```

CONTOH 17

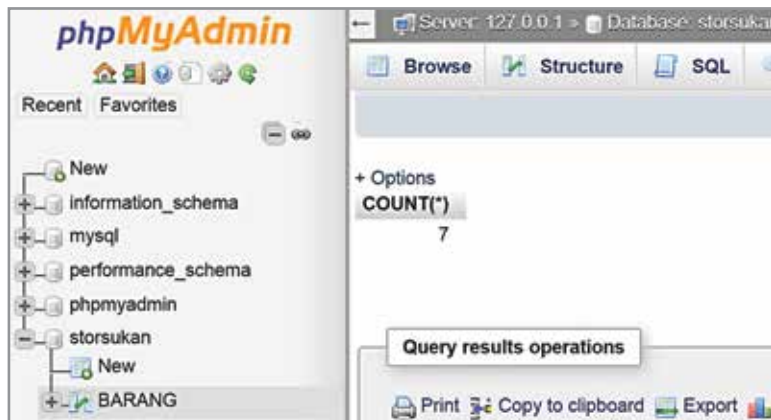
Contoh 17 menunjukkan cara menggunakan fungsi COUNT. Katakan anda ingin mendapatkan jumlah rekod yang terdapat dalam jadual BARANG seperti dalam Rajah 2.1.70.

Pernyataan SQL:



Rajah 2.1.70 Mencapai data dengan menggunakan fungsi COUNT

Output:



Rajah 2.1.71 Paparan data untuk fungsi COUNT

Rajah 2.1.71 memaparkan jumlah rekod yang terdapat dalam jadual BARANG.

Fungsi Agregat dan GROUP BY

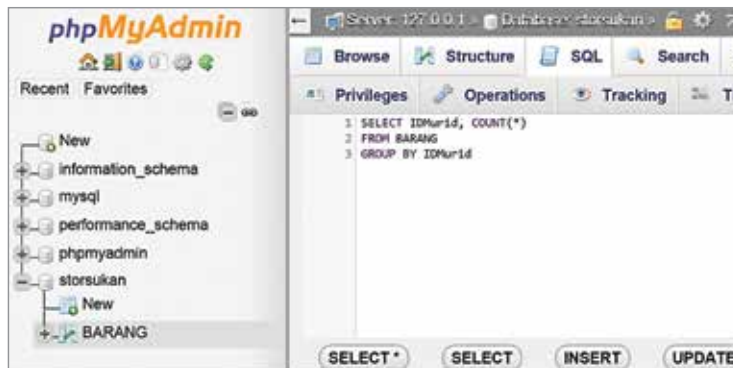
Lazimnya, anda menggunakan fungsi agregat untuk bahagian yang tertentu dalam hubungan. Dalam situasi ini, fungsi agregat boleh digunakan bersama pernyataan GROUP BY.

CONTOH 18

Contoh 18 menunjukkan cara menggunakan pernyataan GROUP BY. Katakan anda ingin mendapatkan jumlah barang yang dipinjam oleh setiap murid. Langkah-langkahnya adalah seperti yang berikut:

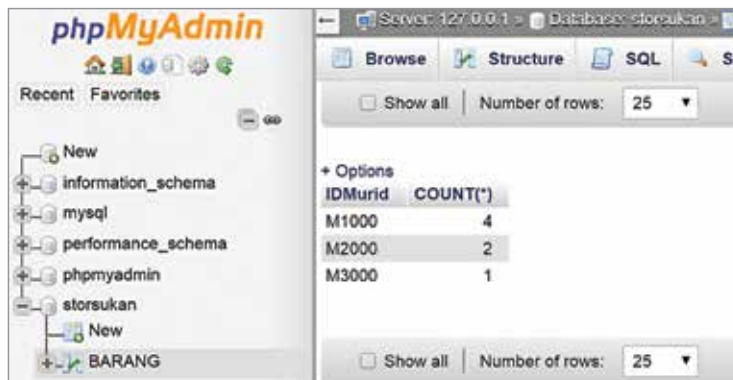
- 1 Kira jumlah rekod dalam jadual (gunakan FROM dan WHERE).
- 2 Kumpulkan rekod mengikut IDMurid (gunakan GROUP BY).
- 3 Pilih satu rekod untuk setiap kumpulan (gunakan fungsi agregat).

Pernyataan SQL:



Rajah 2.1.72 Mencapai data dengan menggunakan GROUP BY

Output:



Rajah 2.1.73 Paparan data dengan menggunakan GROUP BY

Rajah 2.1.72 menghitung jumlah barang dalam jadual BARANG dengan menggunakan ungkapan COUNT(*) dan dikumpulkan mengikut IDMurid dengan menggunakan pernyataan GROUP BY dan outputnya ditunjukkan dalam Rajah 2.1.73.

Penggunaan HAVING

Pernyataan HAVING digunakan untuk menentukan kumpulan data yang hendak dipaparkan. Dalam keadaan ini, pernyataan WHERE tidak boleh digunakan kerana WHERE mengembalikan rekod data tertentu sahaja, manakala HAVING akan mengembalikan kumpulan data tertentu.

Format:

```
SELECT nama medan, fungsi1, fungsi2...fungsi n  
FROM nama jadual  
GROUP BY nama medan  
HAVING (fungsi);
```

CONTOH 19

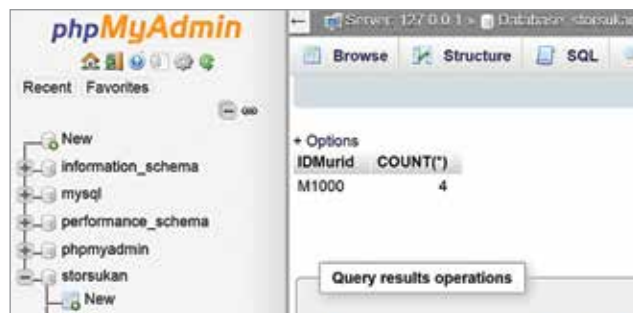
Contoh 19 menunjukkan cara menggunakan HAVING. Katakan anda ingin memaparkan jumlah barang yang dipinjam oleh setiap murid dan anda ingin memaparkan murid yang meminjam barang dengan kuantitinya melebihi 2.

Pernyataan SQL:



Rajah 2.1.74 Mencapai data dengan menggunakan HAVING

Output:



Rajah 2.1.75 Paparan data dengan menggunakan HAVING

Rajah 2.1.74 menghitung jumlah barang dalam jadual barang dan dikumpulkan mengikut IDMurid seperti Contoh 18. Ungkapan HAVING COUNT(*) ditambah untuk memaparkan kumpulan data tertentu sahaja. Output tidak memaparkan jumlah IDMurid M2000 dan M3000 kerana melebihi atau sama dengan 2 seperti yang ditunjukkan dalam Rajah 2.1.75.

Tulis arahan SQL bagi setiap pernyataan yang berikut.

- 1 Dapatkan jumlah produk yang terdapat dalam pangkalan data.
- 2 Berapakah kuantiti meja makan yang terdapat dalam stok?
- 3 Dapatkan nilai minimum, nilai maksimum dan nilai purata harga kerusi.
- 4 Dapatkan jumlah produk yang diperbuat daripada setiap jenis kayu.
- 5 Dapatkan jumlah produk yang diperbuat daripada setiap jenis kayu. Paparkan jumlah produk yang melebihi 1.
- 6 Dapatkan kuantiti kerusi yang terdapat dalam stok.
- 7 Berapakah harga purata meja komputer?
- 8 Berapakah jumlah kuantiti meja komputer?

Gunakan maklumat dalam jadual yang ditunjukkan.

PRODUK

NoProduk	Nama	Harga	Kuantiti	JenisKayu
P01	MEJA KOPI	200	100	GETAH
P02	MEJA KOMPUTER	375	1000	LAPIS
P03	MEJA MAKAN	800	500	JATI
P04	MEJA KOMPUTER	500	500	NYATUH
P05	MEJA SEKOLAH	100	2000	GETAH
P06	KERUSI	60	1000	GETAH
P07	KERUSI	100	500	JATI
P08	KERUSI	110	500	NYATUH
P09	MEJA MAKAN	300	200	GETAH
P10	MEJA KOMPUTER	120	1000	GETAH

2.1.7 Membina Pangkalan Data Hubungan dengan Menggunakan SQL untuk Menyelesaikan Masalah

MySQL ialah perisian sumber terbuka pengurusan pangkalan data yang membolehkan pengguna menyimpan, menyusun dan mendapatkan data semula. Perisian ini amat popular kerana perisian ini mudah digunakan, berprestasi tinggi dan boleh dipercayai. Andaikan anda ingin membina pangkalan data untuk mengurus rekod sukan sekolah, Jadual 2.1.18 ialah borang yang digunakan untuk mendapatkan maklumat berkenaan dengan peserta, acara sukan dan pingat yang dimenangi mengikut kategori masing-masing.

Jadual 2.1.15 Rekod sukan sekolah dalam bentuk ONF

PESERTA

IDPeserta	NamaPeserta	Kelas	Jantina	Kategori	RumahSukan	NoAcara	NamaAcara	Pingat
1000	Komala	5 Cergas	P	Bawah 18	Merah	A01	100 m	Gangsa
1000	Komala	5 Cergas	P	Bawah 18	Merah	A02	400 m	Emas
2000	Nisrina	2 Bistari	P	Bawah 15	Hijau	A02	400 m	Emas
2000	Nisrina	2 Bistari	P	Bawah 15	Hijau	A03	Bola Jaring	Perak
3000	Andy	1 Anjung	L	Bawah 15	Kuning	A04	Bola Sepak	Tiada
4000	Andang	1 Cergas	L	Bawah 18	Kuning	A04	Bola Sepak	Tiada

Jadual 2.1.15 menunjukkan jadual yang tidak ternormal (ONF). Oleh itu, langkah pertama yang perlu diambil ialah menghasilkan skema hubungan ternormal. Berdasarkan jadual, atribut yang berulang-ulang dikenal pasti. Kunci primer diberikan seperti dalam Jadual 2.1.16.

Jadual 2.1.16 Jadual rekod sukan sekolah dalam bentuk 1NF

PESERTA

<u>IDPeserta</u>	NamaPeserta	Kelas	Jantina	Kategori	RumahSukan	<u>NoAcara</u>	NamaAcara	Pingat
1000	Komala	5 Cergas	P	Bawah 18	Merah	A01	100 m	Gangsa
1000	Komala	5 Cergas	P	Bawah 18	Merah	A02	400 m	Emas
2000	Nisrina	2 Bistari	P	Bawah 15	Hijau	A02	400 m	Emas
2000	Nisrina	2 Bistari	P	Bawah 15	Hijau	A03	Bola Jaring	Perak
3000	Andy	1 Anjung	L	Bawah 15	Kuning	A04	Bola Sepak	Tiada
4000	Andang	1 Cergas	L	Bawah 18	Kuning	A04	Bola Sepak	Tiada

Kebergantungan atribut bukan kunci dengan atribut kunci diteliti. Jika ada atribut bukan kunci yang hanya bergantung pada salah satu atribut kunci, maka hubungan dikatakan mempunyai kebergantungan separa. Jadual yang baharu perlu dibentuk untuk mengatasi situasi ini.

Jadual 2.1.17 Jadual rekod sukan sekolah dalam bentuk 2NF

PESERTA

<u>IDPeserta</u>	NamaPeserta	Kelas	Jantina	Kategori	RumahSukan
------------------	-------------	-------	---------	----------	------------

KEDUDUKAN

<u>IDPeserta</u>	<u>NoAcara</u>	Pingat
------------------	----------------	--------

ACARA

<u>NoAcara</u>	NamaAcara
----------------	-----------

Berdasarkan Jadual 2.1.17, kewujudan kebergantungan transitif dikenal pasti. Hasil daripada penelitian tersebut, terdapat kebergantungan antara atribut bukan kunci, iaitu Kategori dan Kelas. Kategori bergantung pada kelas yang menentukan umur murid sama ada di bawah 18 tahun ataupun di bawah 15 tahun. Jadual 2.1.18 menunjukkan jadual dalam bentuk 3NF.

Jadual 2.1.18 Jadual rekod sukan sekolah dalam bentuk 3NF

PESERTA

<u>IDPeserta</u>	NamaPeserta	Jantina	RumahSukan	<u>Kategori</u>
------------------	-------------	---------	------------	-----------------

KEDUDUKAN

<u>IDPeserta</u>	<u>NoAcara</u>	Pingat
------------------	----------------	--------

ACARA

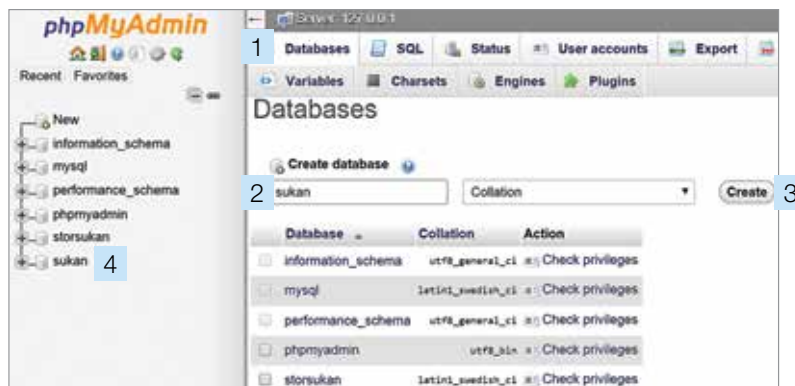
<u>NoAcara</u>	NamaAcara
----------------	-----------

KATEGORI_PESERTA

<u>Kategori</u>	Kelas
-----------------	-------

Setelah skema hubungan ternormal terbentuk, anda boleh mula membina pangkalan data dengan menggunakan MySQL yang terdapat dalam phpMyAdmin. Langkah-langkahnya telah disenaraikan seperti yang berikut:

- 1 Cipta pangkalan data seperti dalam Rajah 2.1.76.

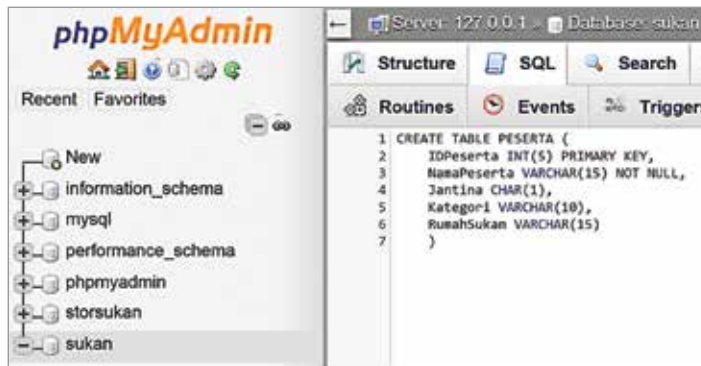


Rajah 2.1.76 Mencipta pangkalan data sukan

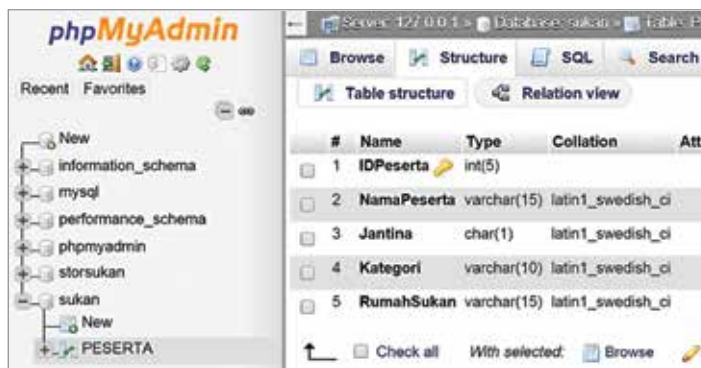
Penerangan:

- 1 Klik pada tab *Databases*.
- 2 Berikan nama pangkalan data yang anda inginkan. Contoh: sukan.
- 3 Klik butang *Create*.
- 4 Pangkalan data sukan telah berjaya dicipta.

- 2 Cipta jadual seperti dalam Rajah 2.1.77. Klik pada tab SQL untuk menulis pernyataan SQL bagi mencipta jadual. Pastikan anda klik pada nama pangkalan data yang dicipta tadi untuk memastikan jadual dicipta dalam pangkalan data yang betul.



Rajah 2.1.77 Mencipta jadual



Rajah 2.1.78 Struktur jadual PESERTA

Paparan output dalam Rajah 2.1.78 menunjukkan struktur jadual PESERTA yang telah berjaya dicipta.

- 3 Masukkan kesemua maklumat peserta dengan menggunakan pernyataan INSERT INTO seperti dalam Rajah 2.1.79.



Rajah 2.1.79 Memasukkan maklumat peserta

IDPeserta	NamaPeserta	Jantina	Kategori	RumahSukan
1000	Komala	P	Bawah 18	Merah
2000	Narina	P	Bawah 15	Hijau
3000	Andy	L	Bawah 15	Kuning
4000	Andang	L	Bawah 18	Kuning

Rajah 2.1.80 Rekod jadual PESERTA



APLIKASI HARIAN

MySQL digunakan oleh organisasi terbesar antarabangsa seperti *Facebook*, *Google* dan *Adobe*.

Sumber:
<https://www.mysql.com>

Paparan output dalam Rajah 2.1.80 menunjukkan kesemua rekod yang telah berjaya dimasukkan ke dalam jadual PESERTA.

Setelah berjaya mencipta pangkalan data rekod sukan yang mengandungi maklumat peserta, acara dan pingat, anda boleh memanipulasikan data dengan menggunakan pernyataan SQL yang telah dibincangkan.

AKTIVITI

7

Kerja kumpulan

Membina Pangkalan Data Menggunakan MySQL

Buat pemerhatian di kawasan persekitaran sekolah anda. Berbincanglah bersama-sama dengan ahli kumpulan anda untuk menentukan sistem pangkalan data yang dapat dibangunkan daripada persekitaran sekolah anda.

Anda boleh mempertimbangkan situasi yang berikut:


- 1 Makmal – mengurus maklumat makmal
- 2 Hari Sukan – mengurus maklumat hari sukan
- 3 Perpustakaan – mengurus maklumat pinjaman buku
- 4 Koperasi sekolah – mengurus maklumat stok koperasi
- 5 Badan beruniform – mengurus maklumat murid dalam setiap badan beruniform

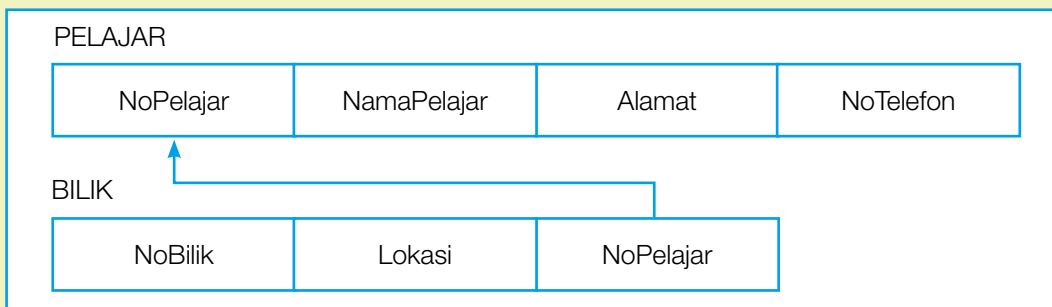
Tugas anda dan ahli kumpulan anda adalah untuk membina pangkalan data dengan menggunakan MySQL bagi salah satu situasi daripada 1 hingga 5 atau situasi yang telah anda kenal pasti.

LATIHAN PENGUKUHAN 2.1

- 1 Anda ingin membina pangkalan data buku alamat untuk menyimpan maklumat rakan-rakan anda. Nyatakan atribut yang terdapat dalam situasi ini.

Atribut	Contoh
Atribut mudah	
Atribut komposit	
Atribut nilai gandaan	
Atribut kunci	


- 2 Lukis Rajah Hubungan Entiti (ERD) bagi setiap pernyataan yang berikut: 
- (a) Setiap pelanggan bank boleh memiliki beberapa jenis akaun dan setiap jenis akaun hanya boleh dimiliki oleh seorang pelanggan.
- (b) Di hospital, seorang pesakit boleh dirawat oleh beberapa orang doktor dan seorang doktor boleh merawat lebih daripada seorang pesakit.
- 3 Rajah 1 menunjukkan hubungan antara dua jadual dalam satu pangkalan data.



Rajah 1

Lengkapkan pernyataan yang berikut:

Berdasarkan Rajah 1, NoPelajar ialah kunci _____ dalam jadual PELAJAR dan _____ dalam jadual BILIK. NoBilik ialah kunci _____. Hubungan yang ditunjukkan ialah _____.

- 4 Jelaskan kegunaan arahan SELECT dalam Bahasa Pertanyaan Berstruktur. Berikan contoh.
- 5 Pertimbangkan jadual PELANGGAN yang berikut: 

NoPelanggan	NamaPelanggan	AlamatPelanggan
1000	Janet	Sandakan, Sabah
1100	Zhi Enn	Kuching, Sarawak

- (a) Tulis pernyataan SQL untuk mencipta jadual pelanggan.
- (b) Tulis pernyataan SQL untuk memasukkan data 1101, Haliza Hassan, 123 Petaling Jaya, ke dalam jadual pelanggan.
- (c) Tulis pernyataan SQL untuk menambah medan baharu, iaitu no telefon.

6 Berdasarkan jadual MURID yang berikut, jawab soalan yang diberikan.

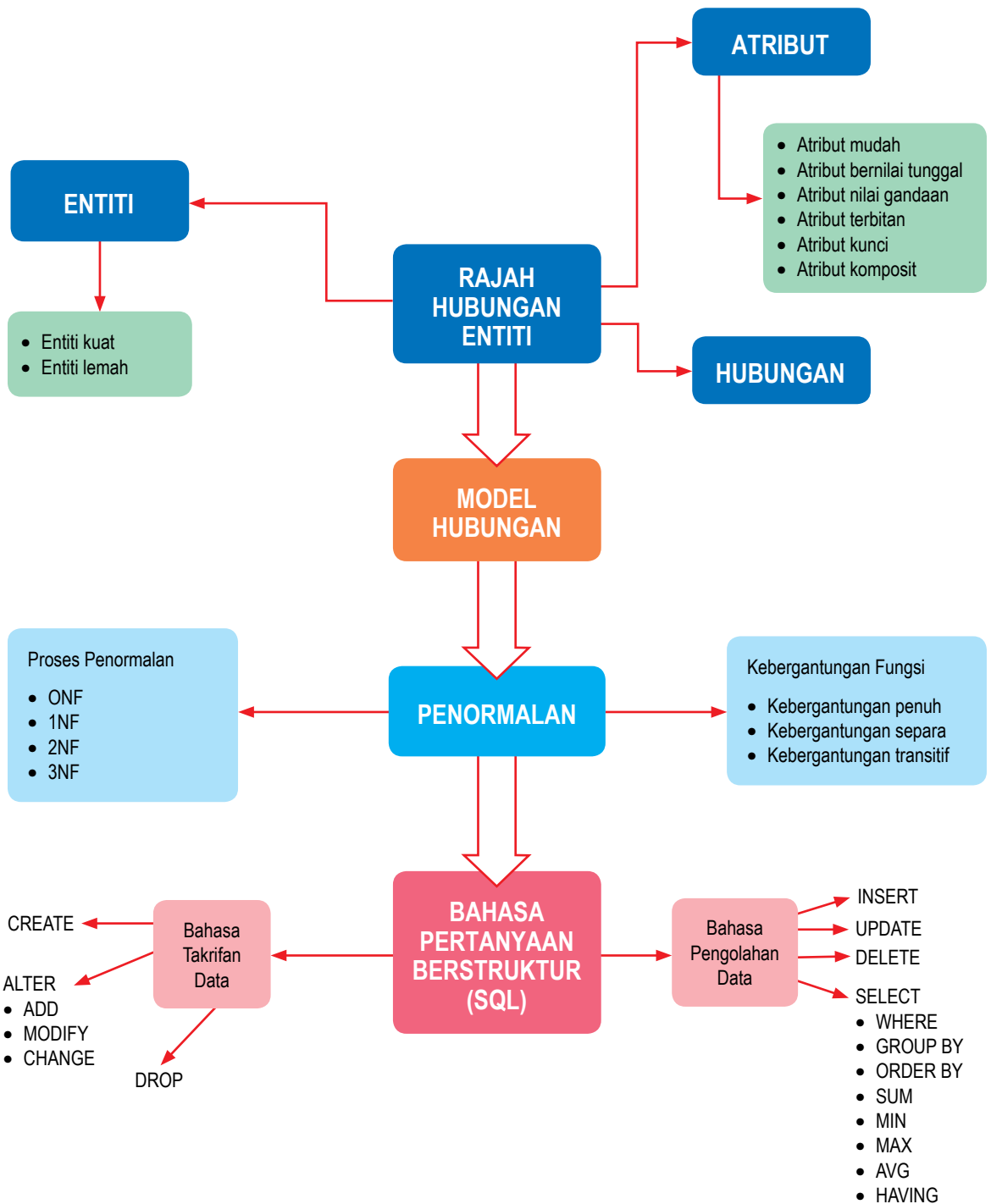


MURID

No_Murid	Nama	Umur	Jantina	Alamat
02052133	Ana Azman	18	P	213, Temerloh, Pahang
02052130	Lee Jun Kit	20	L	10, Masjid Tanah, Melaka
02052099	Aneeta Devi	19	P	9, Puchong, Selangor
02051033	Yurni Yusof	21	P	89, Sungai Petani, Kedah
02052111	Felix Agus	21	L	11, Sandakan, Sabah

- (a) Tulis pernyataan SQL untuk menukarkan nama “Ana Azman” kepada “Anusha”.
- (b) Tulis pernyataan SQL untuk menyenaraikan semua medan daripada jadual MURID mengikut urutan menurun dengan berpandukan nombor murid.
- (c) Tulis pernyataan SQL untuk menghapuskan rekod murid yang bernama Yurni Yusof.

IMBAS SEMULA



LATIHAN PENGAYAAN

2


Untuk soalan-soalan 1–7, isi tempat kosong dengan memilih perkataan atau frasa yang paling sesuai.

Penormalan	Entiti	Kebergantungan transitif
DELETE	Atribut	Kebergantungan fungsi separa
DROP	Kunci primer	CREATE TABLE
SELECT	Anomali	Kunci asing

- _____ ialah kunci yang digunakan untuk mengelakkan pertindihan data dalam pangkalan data.
- Pernyataan _____ table dalam SQL digunakan untuk menghapuskan jadual daripada pangkalan data.
- Pelanggan, Kereta dan Hotel ialah contoh-contoh _____.
- Kebergantungan fungsi antara dua atau lebih atribut bukan kunci dikenali sebagai _____.
- “Ralat yang mungkin berlaku apabila pengguna mengemaskinikan jadual yang mengandungi data yang berulang.” Pernyataan ini merupakan definisi _____.
- Pernyataan _____ digunakan untuk mencapai data daripada jadual dalam pangkalan data.
- _____ ialah ciri-ciri yang menghuraikan sesuatu entiti.
- Berdasarkan situasi yang diberikan, jawab soalan-soalan yang berikut:


Kolej Cemerlang mempunyai 30 buah asrama. Setiap pelajar yang mendaftar akan ditempatkan di salah sebuah asrama. Setiap asrama boleh menampung beberapa orang pelajar. Untuk setiap asrama, pihak pengurusan telah memberikan nombor asrama, nama dan lokasi. Setiap asrama mempunyai beberapa buah bilik yang ditetapkan dengan nombor bilik, nama bilik dan nama blok. Setiap bilik akan dikongsi oleh dua orang pelajar. Maklumat pelajar seperti nombor pelajar, nama dan alamat akan diambil untuk simpanan pihak pengurusan asrama. Setiap pelajar mesti ditempatkan di dalam bilik asrama tetapi mungkin terdapat bilik asrama yang tidak didiami oleh pelajar.

- Kenal pasti entiti, atribut dan hubungan.
- Lukis rajah hubungan entiti (ERD).
- Petakan rajah hubungan entiti kepada skema hubungan.

- 9 Rajah 1 ialah contoh keputusan peperiksaan sesuatu kursus di Kolej Gemilang. Berdasarkan maklumat tersebut, lakukan kaedah penormalan sehingga bentuk normal ketiga. 

KOLEJ GEMILANG KEPUTUSAN PEPERIKSAAN SEMESTER JANUARI 2016		
KOD KURSUS : IS460		
NAMA KURSUS : PANGKALAN DATA		
NAMA PENSYARAH : NORMA		
NO BILIK : B 104		
NO PELAJAR	NAMA PELAJAR	GRED
38214	MIRA	A
40875	CHONG	B
51893	BALANAPAN	A

Rajah 1

- 10 Jadual PELANGGAN yang berikut menunjukkan maklumat pembelian barangan bagi setiap pelanggan di beberapa kawasan. Pertimbangkan skema pangkalan data yang berikut: 

PELANGGAN

Kod	Nama	KodKawasan	NoTelefon	Jumlah
10010	ALAN	615	0108442573	0.00
10011	LEONA	713	0198941238	0.00
10012	KATHY	615	0108942285	345.86
10013	PAUL	615	0128942180	536.75
10014	MIRA	615	0122221672	0.00
10015	AMY	713	0104423381	0.00
10016	JAMES	615	0102971228	221.19
10017	SALLEH	615	0192902556	768.93
10018	SURIA	713	0103827185	216.55
10019	ADAM	615	0192973809	0.00

- Tulis pernyataan SQL untuk memaparkan maklumat pelanggan yang berada di kawasan 615.
- Tulis pernyataan SQL untuk menyenaraikan jumlah kos jualan untuk semua pelanggan bagi setiap kawasan.
- Tulis pernyataan SQL untuk mendapatkan nilai minimum, nilai maksimum dan nilai purata jualan untuk semua pelanggan.
- Tulis pernyataan SQL untuk menghapuskan maklumat pelanggan daripada kod kawasan 713.
- Tulis pernyataan SQL untuk mengemaskinikan kod kawasan daripada 615 kepada 610.



Fikirkan...

- Bagaimanakah suatu permintaan daripada pengguna Internet dapat diproses secara dalam talian?
- Cuba anda bayangkan situasi tanpa laman web, bagaimanakah kita akan berinteraksi dengan sistem dalam rangkaian?
- Bagaimanakah teknologi berasaskan web mampu menyokong pengguna untuk menyelesaikan masalah yang lebih rumit pada masa akan datang?

PENGATURCARAAN BERASASKAN WEB

Dalam bab ini, anda akan mempelajari:

- Keperluan bahasa penskripan klien dalam laman web.
- Keperluan bahasa penskripan pelayan dalam laman web.
- Strategi penyelesaian masalah berasaskan web
- Pembinaan atur cara yang dapat memberikan tindak balas yang sesuai berdasarkan input yang dimasukkan oleh pengguna melalui web.
- Pembangunan laman web mengikut kesesuaian pengguna.
- Pembangunan laman web yang interaktif dalam menyelesaikan masalah

Penemuan konsep web oleh Tim Berners-Lee pada tahun 1989 telah membuka revolusi baharu dalam komunikasi dan perkongsian maklumat manusia. Bertitik tolak dari penggunaan web yang bersifat statik bagi menyokong perkongsian maklumat dalam kalangan akademik, kini web melebarkan sayapnya dengan penciptaan pelbagai aplikasi bagi menyokong setiap individu dan juga organisasi dalam pelbagai aspek termasuk pengurusan masa, kewangan dan maklumat. Oleh sebab itulah, penggunaan komputer, telefon pintar dan tablet menjadi satu keperluan bagi setiap individu untuk pelbagai tujuan. Namun, sejauh manakah pembangunan web interaktif mampu menyelesaikan masalah anda?



3.1

STANDARD PEMBELAJARAN

Pada akhir pembelajaran murid dapat

3.1.1 Menerangkan keperluan Bahasa Penskripan Klien dalam laman web

3.1.2 Menterjemah atur cara mudah yang diberi daripada Bahasa Penskripan Klien kepada carta alir (*bubble sort, selection sort, binary search, max/min/mean, count, queue*)

3.1.3 Menggunakan pemalar, pemboleh ubah dan jenis data yang berlainan dalam atur cara untuk melaksanakan *bubble sort, selection sort, binary search, max/min/mean, count, queue*

3.1.4 Menggunakan struktur kawalan (jujukan, pilihan, ulangan) dalam atur cara

3.1.5 Menerangkan kelebihan *standard library* kepada pengatur cara

3.1.6 Menggunakan *standard library* dalam atur cara

3.1.7 Meringkaskan atur cara dengan menggunakan:
(i) *procedure*
(ii) *function*

3.1.8 Membina atur cara yang melibatkan penggunaan tatasusunan (*array*)

3.1.9 Menghasilkan atur cara interaktif yang mengandungi pengisytiharan data, struktur kawalan, *procedure* dan *function* serta tatasusunan

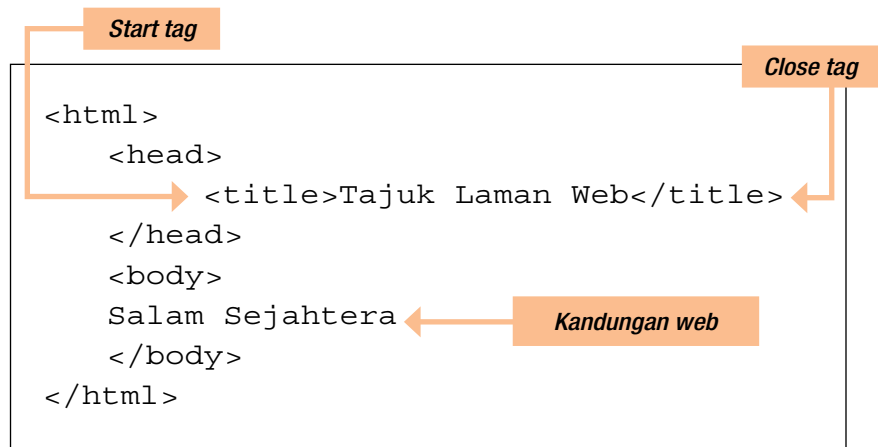
Bahasa Penskripan Klien

3.1.1 Keperluan Bahasa Penskripan Klien dalam Laman Web

Pengenalan kepada HTML dan CSS

HTML atau *Hypertext Markup Language* ialah bahasa *markup standard* yang digunakan oleh pereka web profesional bagi mereka bentuk laman web. Dewasa ini, bahasa ini diperlukan oleh sesiapa jua yang ingin menerbitkan dokumen dalam talian. Contohnya, guru yang ingin memberikan tugas kumpulan kepada pelajarinya dalam talian, pengurus yang ingin mencari pekerja baharu dengan mengiklankan kekosongan dalam laman web korporat mereka dan pencari kerja yang ingin menyiarkan profil mereka memerlukan HTML untuk memaparkan dokumen mereka.

Dokumen HTML adalah mengenai konsep bagaimana dokumen dipaparkan. Dokumen HTML juga ialah konsep bagaimana elemen teks, audio, video serta grafik disusun agar mudah dibaca dan difahami oleh pelawat laman web. Untuk menerangkan struktur bagi setiap laman web, kod HTML atau elemen HTML digunakan. Elemen biasanya terdiri daripada dua tag yang berpasangan, iaitu: *start tag* dan *close tag*. Kandungan yang akan dipaparkan pada web akan berada di antara *start tag* dengan *close tag*. Rajah 3.1.1(a) menunjukkan contoh elemen HTML yang terdiri daripada *start tag* dan *close tag* secara berpasangan dan Rajah 3.1.1 (b) adalah outputnya.



Rajah 3.1.1(a) Contoh tag HTML berpasangan

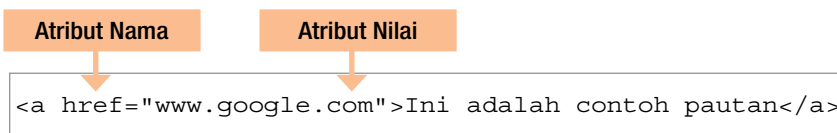


Rajah 3.1.1(b) Contoh output berdasarkan Rajah 3.1.1(a)

Semua elemen HTML mengandungi atribut. Atribut mengandungi maklumat tambahan bagi setiap elemen HTML dan biasanya ditempatkan di *start tag*. Terdapat dua bahagian atribut:

- Atribut nama
- Atribut nilai

Rajah 3.1.2 menunjukkan contoh atribut nama dan atribut nilai.



Rajah 3.1.2 Contoh atribut nama dan atribut nilai

Pengenalan kepada CSS

Cascading Style Sheet (CSS) ialah salah satu bahasa pengaturcaraan yang merupakan tambahan (*extension*) kepada HTML. CSS membenarkan perubahan kepada gaya sesuatu halaman web dan menerangkan persembahan bagi sesuatu dokumen HTML. Rajah 3.1.3 menunjukkan contoh penggunaan HTML dan tanpa CSS. Rajah 3.1.4 pula menunjukkan contoh penggunaan HTML dengan CSS.



Malaysia Website Awards 2016 (MWA) ialah satu program anugerah berprestij yang mengiktiraf kecemerlangan persembahan, bakat dan usaha para pereka web, pembangun web serta agensi yang berkaitan dengan web di dalam negara ini.

Sumber:
www.mwa.my/2016/



Bahasa penskripan klien
goo.gl/0xKutg

Contoh tanpa CSS dalam HTML

```

<html>
  <head>
    <title>Penggunaan CSS</title>
  </head>

  <body>
    <h1>Laman Web Sekolah</h1>
  </body>
</html>
  
```



Rajah 3.1.3 Contoh HTML tanpa CSS dan output

Contoh CSS dalam HTML

```
<html>
  <head>
    <title>Penggunaan CSS</title>
    <style>
      h1
      {
        color:blue;
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1>
  </body>
</html>
```



Rajah 3.1.4 Contoh HTML dengan CSS dan output

Pengenalan kepada JavaScript

Adakah anda tahu bahasa yang digunakan bagi klien untuk berinteraksi dengan laman web? Interaksi merujuk permohonan untuk mendapatkan sesuatu maklumat daripada laman web. Bahasa pengaturcaraan akan memproses sebarang permohonan (*request*) klien melalui suatu laman web.

Bahasa yang digunakan dikenali sebagai bahasa penskripan. Antara bahasa penskripan yang selalu digunakan ialah *JavaScript*, PHP dan ASP.

JavaScript menyediakan keupayaan pengaturcaraan untuk memproses sesuatu tugas berdasarkan permohonan yang diterima daripada klien.

Kod *JavaScript* akan ditulis bersama-sama dalam kod HTML. Jika HTML tidak mempunyai sebarang skrip yang ditulis di dalamnya, *browser* akan menghasilkan laman web berdasarkan kod HTML tersebut.

Kod *JavaScript* ditulis di antara tag '`<script>`' dengan '`</script>`' seperti yang terdapat pada tag dalam HTML pada Rajah 3.1.5.

Tahukah Anda

Java ialah bahasa pengaturcaraan berorientasikan objek manakala *JavaScript* ialah bahasa penskripan berorientasikan objek.

Sumber:
https://www.java.com/en/download/faq/java_javascript.xml

```
<script>
  document.write("Selamat Datang");
</script>
```


Rajah 3.1.5 Tag JavaScript

Terdapat dua cara kod *JavaScript* ditulis dalam HTML:

- 1 *Explicit embedding*
- 2 *Implicit embedding*

CONTOH 1

Contoh 1 menggunakan cara *explicit embedding*, yaitu kod *JavaScript* diletakkan dalam atau bersama-sama kod HTML. Contoh kandungan HTML (tanpa skrip) dan kandungan HTML (berserta skrip) ditunjukkan pada Rajah 3.1.6.

HTML (tanpa skrip)	HTML (berserta skrip)
<pre><html> <body> <p>Nombor: </p> 5 </body> </html></pre>	<pre><html> <body> <p>Nombor: </p> <script> document.write("5"); </script> </body> </html></pre>
	

Kod *JavaScript* untuk mencetak "5" dalam dokumen HTML (laman web)

Rajah 3.1.6 Atur cara HTML, *JavaScript* dan contoh output (*Explicit Embedding*)

CONTOH 2

Contoh 2 menggunakan cara *implicit embedding*, yaitu meletakkan kod *JavaScript* dalam satu fail sendiri yang mempunyai *extension* ".js" seperti contoh `myScript.js` dan akan dipanggil jika ingin digunakan. Fail mengandungi kod *JavaScript* dalam subatur cara `f1()`.

Sekiranya ingin menggunakan fail tersebut, rujuk nama fail tersebut dalam atribut *source* dan panggil `f1()` untuk pelaksanaan kod, iaitu `src="myScript.js"`.

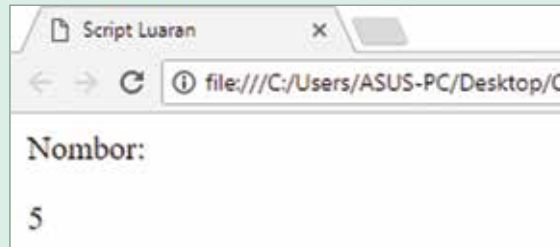
Contoh kandungan fail ".js" berserta skrip dan kandungan HTML (yang merujuk fail ".js") ditunjukkan pada Rajah 3.1.7.

Fail "myScript.js" (skrip)	HTML
<pre>function f1() { document.write("5"); }</pre>	<pre><html> <head> <title>Script Luaran</title> <script src="myScript.js"> </script> </head> <body> <p>Nombor: </p> <script type="text/javascript"> f1(); </script> </body> </html></pre>

Rujuk kod *JavaScript* dalam fail `myScript.js`

Panggil dan laksana kod subatur cara `f1()` dalam fail `myScript.js`

Contoh output

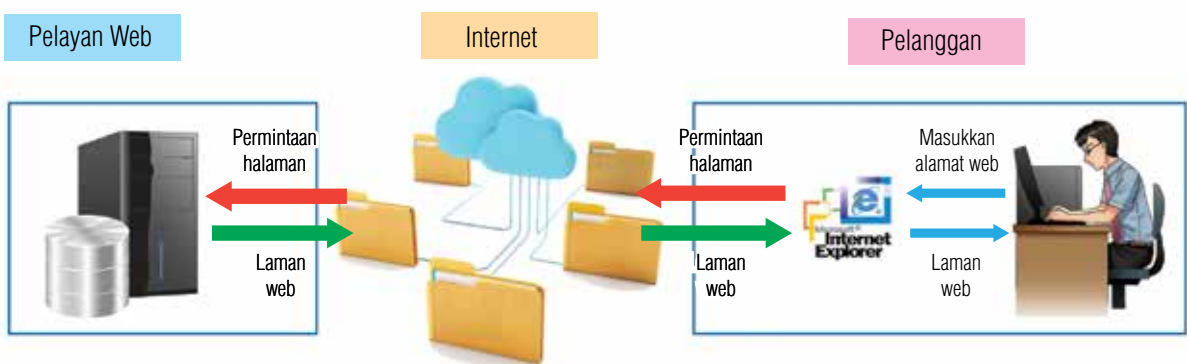


Rajah 3.1.7 Atur cara HTML, JavaScript dan contoh output (*Implicit Embedding*)

Konsep Aplikasi Pelanggan/Pelayan (*Client/Server*)

Reka bentuk web dibangunkan berteraskan pelanggan/pelayan. Pelanggan atau klien biasanya adalah pelayar (*browser*) yang digunakan oleh pengguna bagi melayari web dengan menggunakan sama ada peranti mudah alih, komputer riba atau komputer meja.

Aplikasi pelayan (*server*) pula boleh berada di mana-mana jua di seluruh dunia yang bertindak untuk memproses permintaan atau menghantar laman web yang diminta oleh klien. Satu lagi komponen laman web ialah pangkalan data yang digunakan untuk menyimpan data. Rajah 3.1.8 menunjukkan konsep aplikasi pelanggan/pelayan yang digunakan dalam reka bentuk web.



Rajah 3.1.8 Konsep aplikasi Pelanggan/Pelayan (*Client/Server*)

Bahasa Penskripan

Secara umumnya terdapat dua jenis laman web, iaitu laman web statik dan laman web dinamik. Laman web yang direka menggunakan *Hypertext Markup Language* (HTML) sahaja ialah laman web yang bersifat statik iaitu kandungan paparan pada setiap halaman mengikut tag HTML yang digunakan. Laman web dinamik pula ialah laman web yang direka menggunakan penskripan pelayan seperti ASP atau PHP yang memberi keupayaan kepada pemilik laman web untuk mengemas kini kandungan laman web dengan mudah. Jadual 3.1.1 menunjukkan perbezaan antara laman web statik dengan laman web dinamik.

Jadual 3.1.1 Perbezaan antara laman web statik dengan laman web dinamik

Laman web statik	Laman web dinamik
Kandungan paparan mengikut tag HTML yang digunakan	Kandungan paparan boleh dipanggil dari pangkalan data atau mengikut tindakan pengguna
Dibina menggunakan HTML sahaja	Dibina menggunakan penskripan pelayan
Contoh: laman web yang memaparkan latar belakang organisasi	Contoh: Aktiviti akan datang yang dipaparkan di laman utama laman web yang berubah mengikut tarikh kalendar

Konsep penskripan web digunakan secara meluas dalam pembangunan web. Penskripan ialah sebahagian kecil program yang dilaksanakan berdasarkan interaksi pengguna dengan laman web. Penskripan ini ditempatkan di dua bahagian iaitu penskripan klien di sebelah pelanggan dan penskripan pelayan di sebelah pelayan. Penskripan pelayan bekerja di bahagian belakang pembangunan web dan dilaksanakan oleh pelayan web, manakala penskripan klien pula bekerja di bahagian hadapan pembangunan web dan dilaksanakan oleh pelayan web. Rajah 3.1.9 menunjukkan pelaksanaan penskripan klien dan penskripan pelayan.

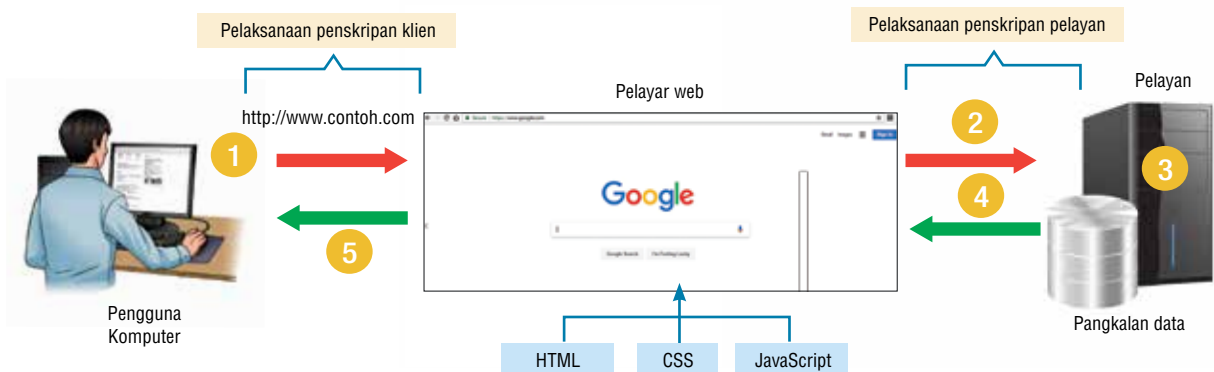
Tahukah Anda

Tugas-tugas lain yang boleh dilaksanakan oleh bahasa penskripan klien:

- Memohon pelayan memaparkan maklumat.
- Mengganti atau mengemaskini sebahagian daripada laman web.
- Mengubah stail dan kedudukan elemen yang dipaparkan secara dinamik.
- Memeriksa input daripada pengguna.



Penggunaan bahasa penskripan klien dan penskripan pelayan goo.gl/Tu3pXe



Rajah 3.1.9 Pelaksanaan penskripan klien dan penskripan pelayan

Penskripan pelayan bekerja di bahagian belakang pembangunan web manakala penskripan klien bekerja di bahagian hadapan pembangunan web. Rajah 3.1.9 menunjukkan contoh pemprosesan oleh penskripan pada klien.

Rajah 3.1.9 menunjukkan pelaksanaan bahasa penskripan klien dan penskripan pelayan dengan penerangan seperti yang berikut:

- 1 Pengguna komputer menaip URL di pelayar web.
- 2 Pelayar web memajukan URL tersebut kepada pelayan web untuk diproses.
- 3 Pelayan web akan memproses permintaan pengguna komputer dengan melaksanakan bahasa penskripan pelayan.
- 4 Pelayan akan mengembalikan hasil pelaksanaan penskripan pelayan kepada pelayar web dengan menghantar laman web yang spesifik berdasarkan URL yang ditaip oleh pengguna komputer.
- 5 Pelayar web akan menterjemahkan dan melaksanakan kandungan laman HTML termasuk mana-mana penskripan klien yang ada pada web tersebut untuk dibaca oleh pengguna komputer.

Bahasa Penskripan Klien

Bahasa penskripan klien secara umumnya merujuk program komputer dalam web yang dilaksanakan pada komputer klien. Program ini direka bagi memanipulasikan dan memaparkan kandungan laman web yang telah dimuat turun dalam komputer klien.

Penggunaan penskripan klien dalam web telah memberikan keupayaan kepada laman web untuk menukarkan atau memberikan respon kepada kandungan web mengikut input yang diberikan oleh pengguna dengan lebih pantas.

Penskripan klien biasanya digunakan untuk proses pengesahan input yang dimasukkan oleh pengguna apabila mengisi borang dalam talian. Selain itu, penskripan klien juga digunakan untuk melaksanakan tugas-tugas yang mudah seperti menukar elemen. Contohnya, jenis *font* atau warna latar web tanpa melibatkan pelayan.

Contoh lain bagi penggunaan penskripan klien adalah seperti penggunaan *shopping cart* dalam laman web yang menawarkan pembelian secara dalam talian kepada pengguna. Pada kebiasaan kita akan menggunakan *shopping cart* untuk menyenaraikan barangan pilihan. Setiap kali kita menambah atau mengurangkan item, jumlah kos akan berubah secara automatik. Perubahan yang berlaku sebenarnya diproses oleh penskripan klien. Rajah 3.1.10 menunjukkan contoh proses yang dilaksanakan oleh penskripan klien melalui penggunaan *shopping cart*.



Kemajuan dalam Internet telah mewujudkan profesion yang berikut:

- Pembangun web
- Pereka web
- Pengurus web
- Pereka grafik
- Penulis atau editor web

Thank you for shopping at MPHONLINE. Please follow the steps below to place your order.

STEP 1 : Order Status

Our shopping cart contains the following items. If you wish to order more than one copy of a particular item, simply change the quantity to the number you want. If you wish to remove an item from your shopping cart, simply tick the check box in **Delete Item**. After making the changes, click on the **Re-calculate** Button to update the shopping cart.

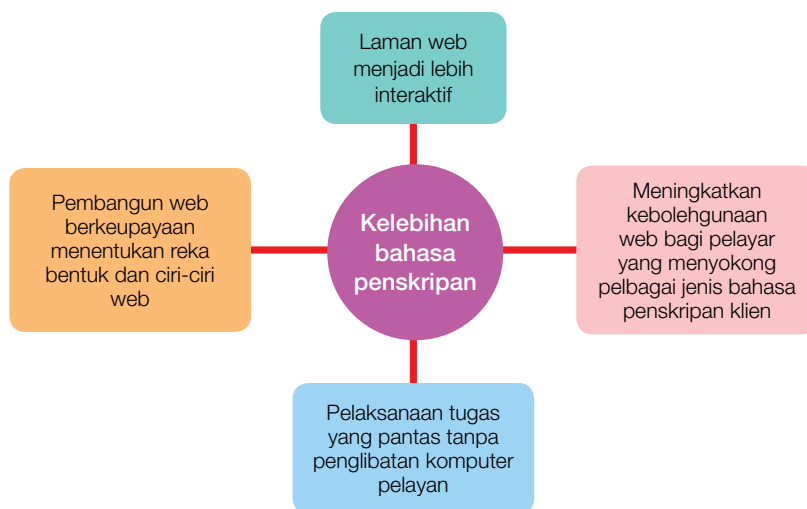
Item no	Delete Item	Quantity	Description	Shipping duration	Our Price (RM)
1	<input type="checkbox"/>	1	Networking For Dummies, 11th Ed. (10% off)	3 to 5 business days	97.00
2	<input type="checkbox"/>	1	Raspberry Pi User Guide 4ed (10% off)	3 to 5 business days	101.05
Sub-total					198.05
[Check Shipping Information] **Shipping Costs					0.00
**Handling Costs					0.00
**Surcharge					0
Grand Total					198.05

Continue Shopping | Delete Ticked Items | Re-calculate

Rajah 3.1.10 Contoh proses yang dilaksanakan oleh penskripan klien melalui penggunaan *shopping cart* dari sumber *mphonline.com*

Jumlah pembelian akan berubah setiap kali kuantiti item pembeli berubah

Namun begitu bukan semua tugas dapat dilaksanakan oleh bahasa penskripan klien. Tugas-tugas yang melibatkan pangkalan data pada sebelah pelayan web atau capaian ke atas fail sistem yang terdapat pada pelayan web ialah contoh tugas yang tidak mampu dilaksanakan oleh bahasa penskripan klien. Antara kelebihan bahasa penskripan klien adalah seperti yang ditunjukkan pada Rajah 3.1.11.



Rajah 3.1.11 Kelebihan bahasa penskripan klien

Tahukah Anda

Terdapat beberapa jenis lain dalam proses pengisihan seperti:

- Isihan cepat (*Quick sort*)
- Isihan selitan (*Insertion sort*)
- Isihan cantum (*Merge sort*)

Sumber:

<http://www.w3schools.in/data-structures-tutorial/sorting-techniques>



Isihan buih (*bubble sort*)
goo.gl/GxsxL6

Tahukah Anda

Terdapat dua jenis isihan dalam lapangan Sains Komputer iaitu luaran (*external*) dan dalaman (*internal*). Sekiranya data muat dimasukkan ke dalam ruang memori, isihan tersebut ialah isihan *internal* manakala jika data terlalu besar, isihan *external* digunakan. Kaedah isihan yang berbeza mempunyai kos serta masa yang berbeza untuk memprosesnya.

Sumber:

<https://www.kidscodecs.com/bubble-sorts/>

AKTIVITI


1

Kerja Individu

Pemerhatian tentang Teknologi berkaitan dengan Bahasa Penskripan Klien

Objektif: Mengenal pasti teknologi yang berkaitan dengan bahasa penskripan klien.

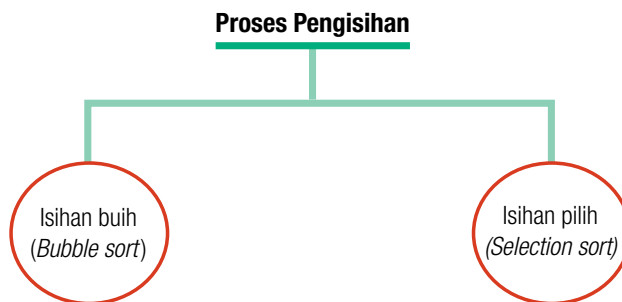
Dengan menggunakan enjin carian seperti *Google* atau *Yahoo!*:

- 1 Senaraikan **EMPAT (4)** bahasa penskripan klien yang biasa digunakan.
- 2 Berdasarkan jawapan anda bagi soalan 1, yang manakah antara empat bahasa penskripan klien tersebut yang paling popular? 

3.1.2 Atur Cara dan Carta Alir bagi Bahasa Penskripan Klien

Semasa membangunkan atur cara penskripan klien, murid boleh melaksanakan proses pengisihan, pencarian, mencari nilai maksimum, minimum dan mengira purata serta bilangan unsur dalam sesuatu atur cara.

Suatu senarai unsur adalah lebih baik diisih untuk memudahkan pencarian agar paparan akan menjadi lebih kemas. Suatu atur cara komputer perlu mengikut turutan arahan untuk melaksanakan proses pengisihan. Proses pengisihan terdiri daripada beberapa jenis. Namun begitu, hanya dua jenis proses pengisihan yang dibincangkan dalam bab ini, iaitu isihan buih (*bubble sort*) dan isihan pilih (*selection sort*) seperti yang ditunjukkan dalam Rajah 3.1.12.



Rajah 3.1.12 Jenis-jenis proses pengisihan

Pengisihan boleh dilakukan secara:

- 1 Turutan menaik
Contoh: 1, 2, 3
- 2 Turutan menurun
Contoh: 3, 2, 1

Isihan buih (*Bubble sort*)

Bubble sort ialah satu algoritma mudah dalam pengisihan suatu senarai yang prosesnya dilaksanakan satu persatu secara berturutan dengan membandingkan unsur-unsur bersebelahan dan akan menukar kedudukan (*swap*) sekiranya berlaku turutan yang salah. Rajah 3.1.13 menunjukkan proses isihan buih.



Rajah 3.1.13 Proses isihan buih

Terdapat dua proses perbandingan yang boleh dilaksanakan untuk menukar kedudukan, iaitu:

- 1 Banding unsur pertama sama ada **lebih kecil** daripada unsur kedua (untuk turutan menurun)
- 2 Banding unsur pertama sama ada **lebih besar** daripada unsur kedua (untuk turutan menaik)

CONTOH 3

Contoh 3 menunjukkan satu senarai nombor diisih secara menaik dengan menggunakan kaedah isihan buih. Atur cara *JavaScript* dan output untuk isihan buih ditunjukkan pada Rajah 3.1.14.

```
<html>
  <body>
    <script>
      var no = [5,1,4,3,2];
      var bilNombor = no.length, i, j;
      var sementara = no[0];

      for(i=0; i<bilNombor-1; i++)
      {
        for(j=0; j<bilNombor-i-1; j++)
        {
          if (no[j] > no[j+1])
          {
            sementara = no[j];
            no[j] = no[j+1];
            no[j+1] = sementara;
          }
        }
      }

      document.write (no);
    </script>
  </body>
</html>
```

no.length untuk mengira bilangan nombor dalam senarai

Dapatkan senarai yang belum diisih

Bandingkan unsur

Tukar kedudukan

The screenshot shows a browser window titled 'contoh3.html' with the address bar displaying 'file:///C:/Users/ASUS-PC/Desktop/contoh3.html'. The main content area of the browser shows the output '1,2,3,4,5'.

Rajah 3.1.14 Atur cara dan output bagi isihan buih (Turutan menaik)



UJI MINDA



Seperti yang telah anda pelajari, isihan buih dilakukan dari kiri ke kanan. Pada pendapat anda, bolehkah isihan buih dilakukan dari kanan ke kiri? Fikirkan dan bincangkannya bersama-sama dengan rakan anda.

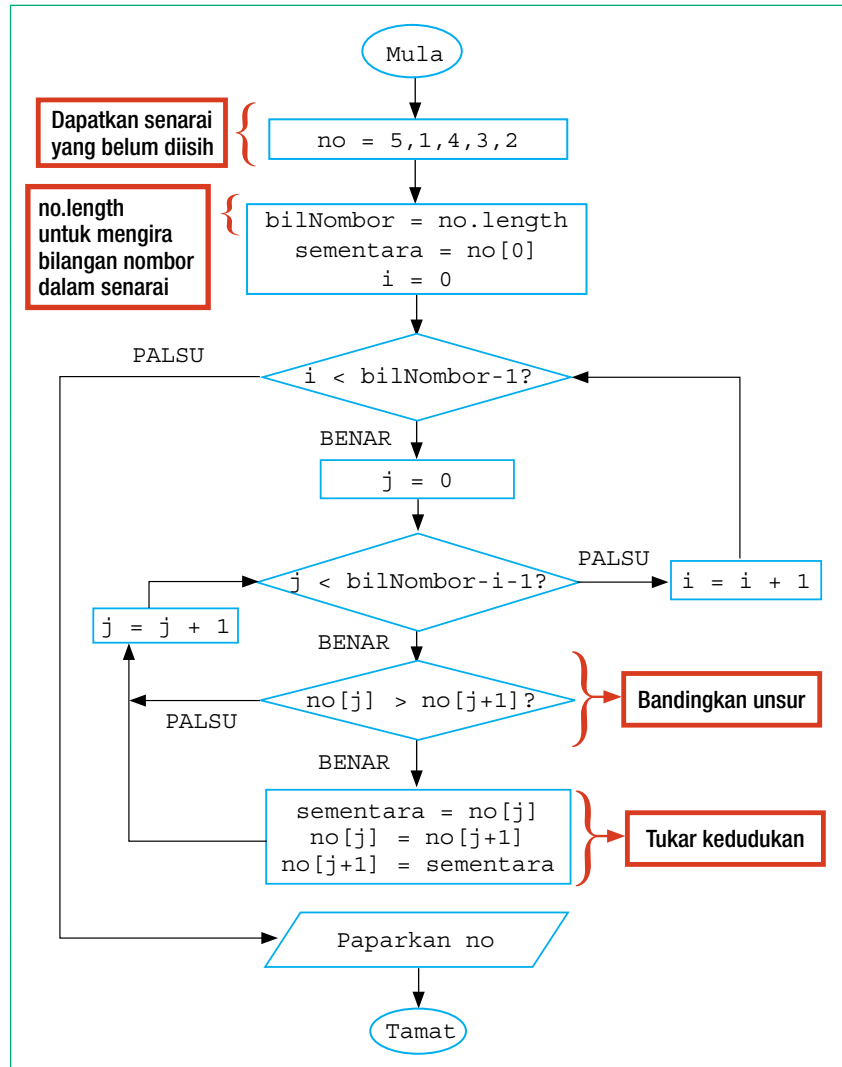


UJI MINDA

9 8 7 5 1 4

Cuba anda lakukan proses pengisihan *bubble sort* dengan menggunakan data di atas. Selamat mencuba!

Carta alir untuk Contoh 3 ditunjukkan pada Rajah 3.1.15.



Rajah 3.1.15 Carta alir bagi isihan buih (turutan menaik)



Struktur Data –
Algoritma Isihan Buih
goo.gl/BNYc7M

Kelebihan Isihan Buih

Antara kelebihan menggunakan algoritma isihan buih (*bubble sort*) untuk membuat pengisihan ialah isihan ini lebih mudah dibina dan lebih mudah difahami.

Pengisihan secara *bubble sort* adalah sesuai digunakan apabila bilangan di dalam unsur adalah kecil memandangkan perbandingan perlu dibuat satu persatu sehingga akhir senarai. Oleh itu, isihan buih adalah sesuai digunakan untuk tujuan pembelajaran dan bukan untuk aplikasi sebenar. Walau bagaimanapun, isihan ini masih boleh digunakan apabila ingin menyusun buku-buku di dalam rak buku yang kecil.

CONTOH 4

Contoh 4 menunjukkan satu senarai kod buku diisih secara menurun menggunakan kaedah isihan buih. Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.16.

```
<html>
  <body>
    <script>
      var kodBuku = [110, 102, 105, 101, 100];
      var bilKod = kodBuku.length, i, j;
      var sementara = kodBuku[0];

      document.write("Senarai kod buku sebelum isih: " + kodBuku);

      for(i=0; i<bilKod-1; i++)
      {
        for (j=0; j<bilKod-i-1; j++)
        {
          if (kodBuku[j] < kodBuku[j+1])
          {
            sementara = kodBuku[j];
            kodBuku[j] = kodBuku[j+1];
            kodBuku[j+1] = sementara;
          }
        }
      }

      document.write ("<br>Senarai kod buku selepas isih secara
      menurun: " + kodBuku);
    </script>
  </body>
</html>
```

kodBuku.length
untuk mengira
bilangan
kodBuku dalam
senarai

Dapatkan senarai
yang belum diisih

Bandingkan unsur

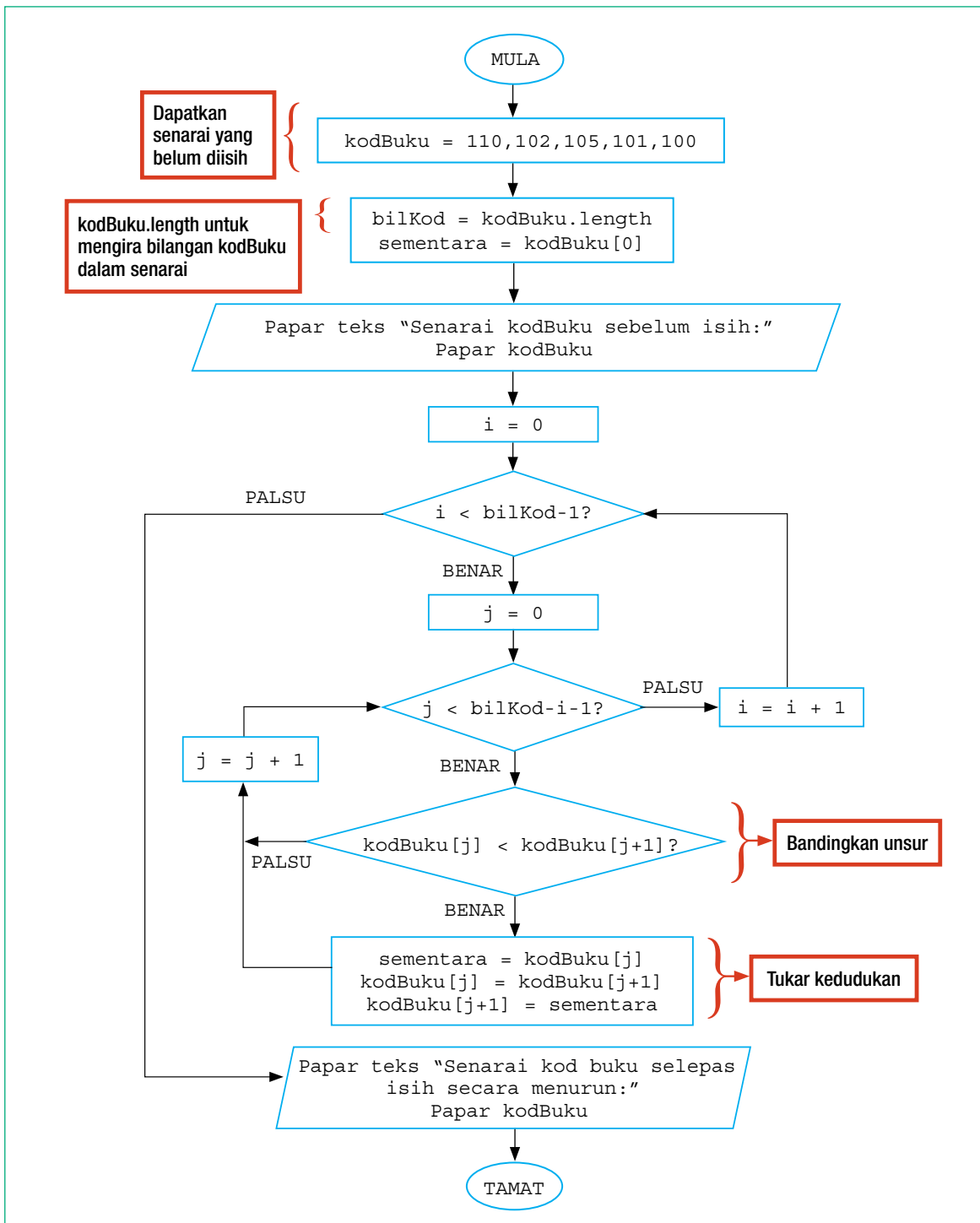
Tukar kedudukan



Rajah 3.1.16 Atur cara *JavaScript* dan output bagi isihan buih (turutan menurun)



Carta alir untuk Contoh 4 ditunjukkan pada Rajah 3.1.17.



Rajah 3.1.17 Carta alir bagi isihan buih (turutan menurun)

Objektif: Menunjukkan langkah-langkah untuk mengisih satu senarai nilai secara menaik dengan menggunakan kaedah isihan buih (*bubble sort*).

Lakukan aktiviti ini di dalam makmal komputer.

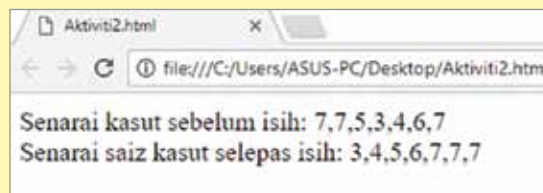
Bahagian 1:

- 1 Buka perisian *Notepad*.
- 2 Taip atur cara HTML dan *JavaScript* yang diberikan:

```
<html>
  <body>
    <script>
      var saizKasut = [7,7,5,3,4,6,7];
      var bilKasut = saizKasut.length, i, j;
      var sementara = saizKasut[0];
      document.write("Senarai kasut sebelum isih: " + saizKasut);

      for(i=0; i<bilKasut-1; i++)
      {
        for (j=0; j<bilKasut-i-1; j++)
        {
          if (saizKasut[j] > saizKasut[j+1])
          {
            sementara = saizKasut[j];
            saizKasut[j] = saizKasut[j+1];
            saizKasut[j+1] = sementara;
          }
        }
      }
      document.write("<br>Senarai saiz kasut selepas isih: " +
        saizKasut);
    </script>
  </body>
</html>
```

- 3 Simpan (*save*) sebagai *Aktiviti2.html*
- 4 Buka fail *Aktiviti2.html* melalui pelayar (*browser*) seperti *Internet Explorer* atau *Google Chrome*.
- 5 Pastikan paparan seperti berikut dihasilkan.

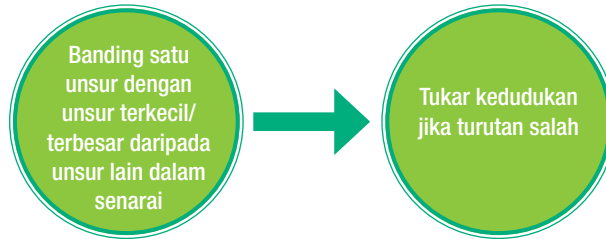


Bahagian 2:

- 1 Ambil sekeping kertas yang baharu.
- 2 Lukis carta alir bagi atur cara pada Bahagian 1, iaitu atur cara bagi mengisih saiz kasut murid-murid daripada yang paling kecil sehingga yang paling besar (turutan menaik).

Isihan pilih (*Selection sort*)

Isihan pilih atau *selection sort* ialah satu algoritma dalam pengisihan suatu senarai yang boleh melibatkan proses seperti dalam Rajah 3.1.18.



Rajah 3.1.18 Proses isihan pilih

Terdapat dua proses perbandingan yang boleh dilaksanakan untuk menukar kedudukan, iaitu:

- 1 Banding unsur dirujuk sama ada **lebih besar** daripada unsur terkecil (untuk turutan menaik)
- 2 Banding unsur dirujuk sama ada **lebih kecil** daripada unsur terbesar (untuk turutan menurun)

CONTOH 5

Contoh 5 menunjukkan satu senarai nombor diisih secara menaik dengan menggunakan kaedah isihan pilih. Atur cara *JavaScript* dan output ditunjukkan dalam Rajah 3.1.19.

```
<html>
  <body>
    <script>
      var no = [5,1,4,3,2], i, j, min, sementara;
      var bilNombor = no.length;
      for (i = 0; i < bilNombor-1; i++)
      {
        min = i;
        for (j = i + 1; j<bilNombor; j++)
        {
          if (no[j] < no[min])
          {
            min = j;
          }
        }
        if (min != i)
        {
          sementara = no[i];
          no[i] = no[min];
          no[min] = sementara;
        }
      }
      document.write(no);
    </script>
  </body>
</html>
```

no.length untuk mengira bilangan nombor dalam senarai

Dapatkan senarai yang belum diisih

Menetapkan nilai terkecil

Bandingkan unsur

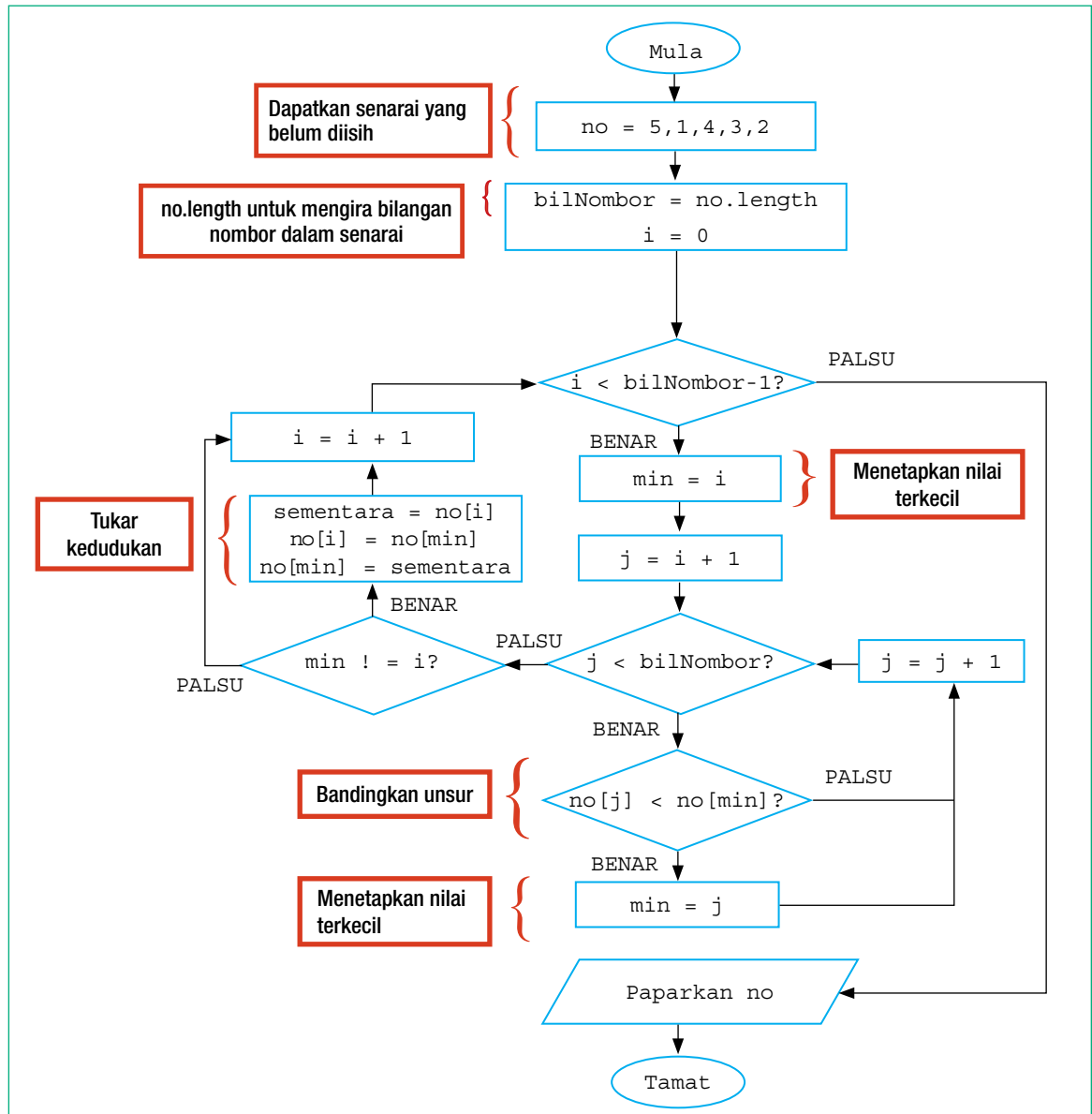
Tukar kedudukan





Rajah 3.1.19 Atur cara JavaScript dan output bagi isihan pilih (turut menaik)

Carta alir untuk Contoh 5 ditunjukkan pada Rajah 3.1.20.



Rajah 3.1.20 Carta alir isihan pilih (turut menaik)



Program: Implement
Selection Sorting in
Java
goo.gl/BWLJfT

Kelebihan Isihan Pilih

Algoritma isihan pilih atau *selection sort* sesuai digunakan dalam membuat pengisihan sekiranya bilangan unsur adalah kecil. Walau bagaimanapun, isihan pilih masih boleh digunakan apabila melibatkan bilangan yang kecil seperti menyusun pelajar mengikut ketinggian sewaktu beratur dalam kelas masing-masing.

CONTOH 6

Contoh 6 menunjukkan satu senarai ketinggian pelajar diisih secara menaik menggunakan kaedah isihan pilih. Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.21.

```
<html>
  <body>
    <script>
      var tinggi = [6,7,7,6,5,6], i, j, min, sementara;
      var bilKetinggian = tinggi.length;
      document.write("Senarai ketinggian pelajar sebelum isih: "
        + tinggi);
      for(i = 0; i<bilKetinggian-1; i++)
      {
        min = i;
        for (j = i + 1; j < bilKetinggian; j++)
        {
          if (tinggi[j] < tinggi[min])
          {
            min = j;
          }
        }

        if (min != i)
        {
          sementara = tinggi[i];
          tinggi[i] = tinggi[min];
          tinggi[min] = sementara;
        }
      }
      document.write("<br>Senarai ketinggian pelajar selepas
        isihan pilih secara menaik: " + tinggi);
    </script>
  </body>
</html>
```

Dapatkan senarai yang belum diisih

tinggi.length untuk mengira bilangan ketinggian dalam senarai

Menetapkan nilai terkecil

Bandingkan unsur

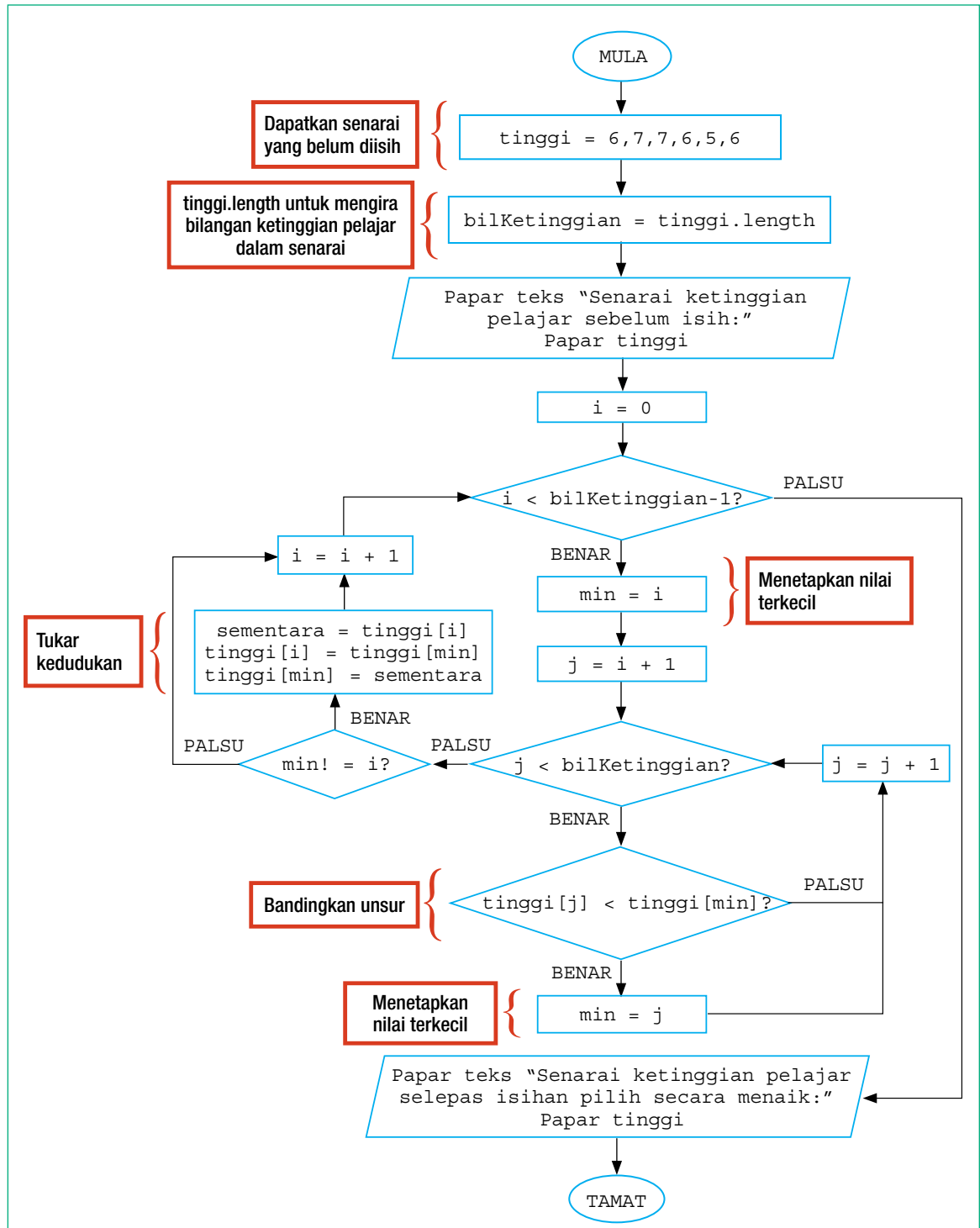
Menetapkan nilai terkecil

Tukar kedudukan

Senarai ketinggian pelajar sebelum isih: 6,7,7,6,5,6
Senarai ketinggian pelajar selepas isihan pilih secara menaik: 5,6,6,6,7,7

Rajah 3.1.21 Atur cara *JavaScript* dan output bagi isihan pilih (turut menaik)

Carta alir untuk Contoh 6 ditunjukkan pada Rajah 3.1.22.



Rajah 3.1.22 Carta alir isihan pilih (turut menaik)



UJI MINDA

Selepas mempelajari kedua-dua jenis isihan *bubble sort* dan *selection sort*, yang manakah menjadi pilihan anda untuk melakukan isihan data? Mengapa? Nyatakan perbezaan yang boleh anda kenal pasti antara kedua-dua jenis isihan tersebut.

AKTIVITI

3

Kerja Individu

Carta Alir *Selection Sort*

Objektif: Menunjukkan langkah-langkah untuk mengisih satu senarai nilai secara menaik menggunakan Isihan Pilih (*Selection Sort*).

Lakukan aktiviti ini di dalam makmal komputer.

Bahagian 1:

- 1 Buka perisian *Notepad*.
- 2 Taip atur cara HTML dan *JavaScript* yang diberikan:

```
<html>
  <body>
    <script>
      var pensil = [2,1,3,3,1], i, j, min, sementara;
      var bilPensil = pensil.length;

      document.write("Senarai bilangan pensil
        daripada 5 orang murid sebelum isih: " +
          pensil);

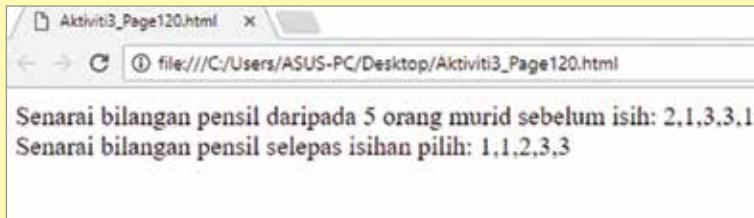
      for(i = 0; i < bilPensil-1; i++)
      {
        min = i;
        for (j = i + 1; j < bilPensil; j++)
        {
          if (pensil[j] < pensil[min])
          {
            min = j;
          }
        }
        if (min != j)
        {
          sementara = pensil[i];
          pensil[i] = pensil[min];
          pensil[min] = sementara;
        }
      }
      document.write("<br>Senarai bilangan pensil
        selepas isihan pilih: " + pensil);

    </script>
  </body>
</html>
```

- 3 Simpan (*save*) sebagai *Aktiviti3.html*
- 4 Buka fail *Aktiviti3.html* melalui pelayar web (*web browser*) seperti *Internet Explorer* atau *Google Chrome*.



5 Pastikan paparan seperti yang berikut dihasilkan:

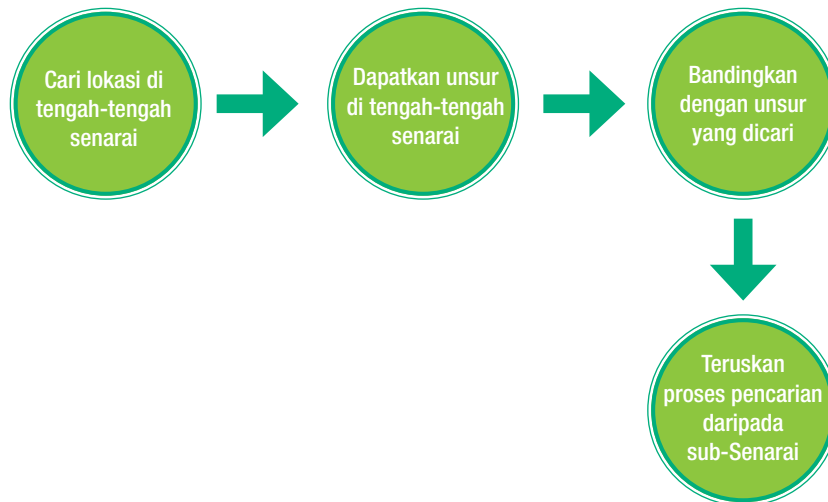


Bahagian 2:

- 1 Ambil sekeping kertas yang baharu.
- 2 Lukis carta alir bagi atur cara yang telah dibina di Bahagian 1, iaitu atur cara bagi mengisih bilangan pensil yang terdapat pada lima orang murid dari yang paling sedikit sehingga paling banyak (turutan menaik).

Carian perduaan (*Binary search*)

Carian perduaan (*Binary search*) sesuai digunakan apabila melibatkan **bilangan data yang banyak** dan **telah diisih**. Pada asasnya, proses carian perduaan adalah seperti dalam Rajah 3.1.23.



Rajah 3.1.23 Proses carian perduaan

Terdapat tiga proses perbandingan yang boleh dilaksanakan, iaitu:

- 1 Bandingkan unsur di tengah sama ada **sama dengan** unsur yang dicari.
- 2 Bandingkan unsur di tengah sama ada **lebih kecil daripada** unsur yang dicari.
- 3 Bandingkan unsur di tengah sama ada **lebih besar daripada** unsur dicari.



Pada 31 Oktober 2013, lima orang rakyat Malaysia telah menerima MIMOS *Prestigious Award* (MPA) atas pencapaian mereka yang cemerlang dalam penyelidikan, pembangunan, dan pengkomersialan untuk meningkatkan sains global di Malaysia di samping melakukan inovasi dalam bidang sains komputer. Penerima-penerima anugerah ialah:

Dr. Zahidah Zulkifli (UM): Penyelidikan dalam bidang paleografi dan perpustakaan digital manuskrip Melayu

Dr. Teoh Wei Lin (USM): Penyelidikan reka bentuk optimum berdasarkan anggaran parameter

Dr. Salfarina Abdullah (UPM): Sumbangan dalam pembangunan seni bina perisian

Dr. Afnizanfaizal Abdullah (UTM): Penyelidikan dalam Kecerdasan Buatan dan Bioinformatik Kumpulan (AIBIG)

Dr. Muhammad Zaini Ahmad (UKM): Penyelidikan dalam Metode Numerik bagi Perkomputeran Fungsi Kabur-bernilai dan Aplikasinya kepada Persamaan Perbezaan.

Sumber:
<http://www.mimos.my/paper/honouring-achievements-that-drive-malysias-global-competitiveness/>

CONTOH 7

Contoh 7 menunjukkan pencarian nombor (iaitu 2) dalam suatu senarai nombor menggunakan kaedah carian perduaan (*binary search*). Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.24. Dalam kaedah ini, nombor dalam senarai ialah nombor-nombor yang telah diisih.

```
<html>
  <body>
    <script>
      var no = [1,2,3,4,5];
      var noCarian = 2;
      var indeksAwal = 0, indeksAkhir = no.length - 1,
      indeksTengah = Math.floor((indeksAkhir + indeksAwal) / 2);
      document.write("No Carian:", noCarian);
      while(no[indeksTengah] != noCarian && indeksAwal < indeksAkhir)
      {
        if (no[indeksTengah] > noCarian)
        {
          indeksAkhir = indeksTengah - 1;
          document.write("<br>No tengah yg dirujuk: ",
            no[indeksTengah]," (Pergi ke sebelah kiri) ");
        }
        else
        {
          if (no[indeksTengah] < noCarian)
          {
            indeksAwal = indeksTengah + 1;
            document.write("<br>No tengah yg dirujuk: ",
              no[indeksTengah]," (Pergi ke sebelah kanan) ");
          }
          indeksTengah = Math.floor((indeksAkhir + indeksAwal)/2);
        }
      }
      if (no[indeksTengah] == noCarian)
      {
        document.write('<br>No tengah yg dirujuk:',no[indeksTengah],' (Jumpa) ');
      }
    </script>
  </body>
</html>
```

Nombor yang telah diisih

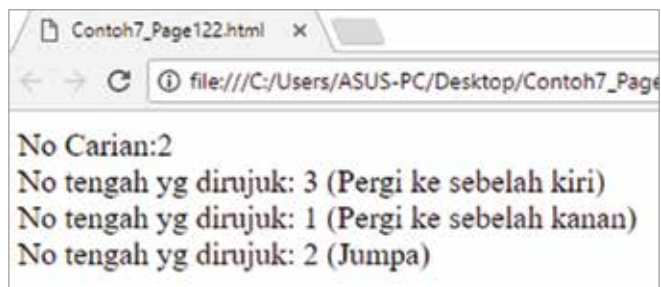
Selagi nombor yang dirujuk TIDAK SAMA dengan nombor yang dicari

Kiraan semula indeksTengah

Jika nombor yang dirujuk SAMA dengan nombor yang dicari

Tahukah Anda

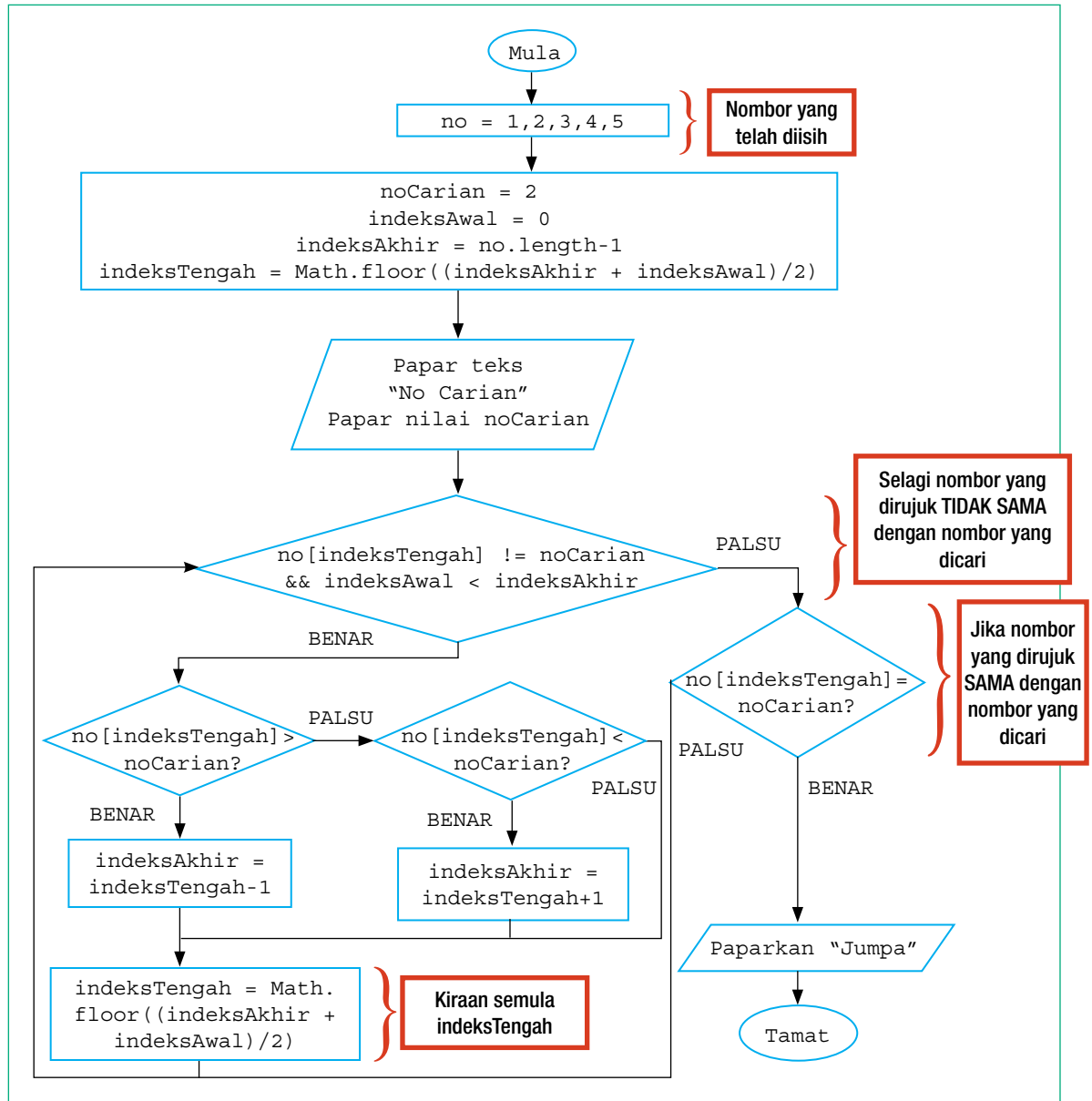
Penggunaan simbol-simbol “”, “”, + dan , dalam *JavaScript* adalah berbeza dengan *Java*.



```
Contoh7_Page122.html x
file:///C:/Users/ASUS-PC/Desktop/Contoh7_Page122.html
No Carian:2
No tengah yg dirujuk: 3 (Pergi ke sebelah kiri)
No tengah yg dirujuk: 1 (Pergi ke sebelah kanan)
No tengah yg dirujuk: 2 (Jumpa)
```

Rajah 3.1.24 Atur cara *JavaScript* dan output (carian perduaan)

Carta alir untuk Contoh 7 ditunjukkan pada Rajah 3.1.25.



Rajah 3.1.25 Carta alir (carian perduaan)

CONTOH 8

Contoh 8 menunjukkan penentuan nombor genap daripada satu senarai nombor dengan menggunakan kaedah carian perduaan (*binary search*). Atur cara JavaScript dan output ditunjukkan dalam Rajah 3.1.26.

Carian nombor genap atau ganjil masih boleh dilaksanakan dengan menggunakan carian perduaan. Walau bagaimanapun, penentuan sama ada nombor dalam senarai adalah nombor genap atau ganjil perlu ➔

dilaksanakan dahulu. Untuk menentukan sama ada nombor ialah nombor genap, nombor perlu dibahagi 2 dan baki mestilah 0, manakala untuk menentukan sama ada nombor ialah nombor ganjil, nombor perlu dibahagi 2 dan bakinya mestilah 1.

Nombor genap	Nombor ganjil
Penentuan nombor genap, nombor perlu dibahagi 2 dan baki mestilah 0 Operator '%' digunakan untuk operasi mendapatkan baki	Penentuan nombor ganjil, nombor perlu dibahagi 2 dan baki mestilah 1 Operator '%' digunakan untuk operasi mendapatkan baki
<code>no[i]%2==0</code>	<code>no[i]%2==1</code>

```

<html>
<body>
<script>
  var no = [11,23,38,48,150], i;
  var noGenap;

  for(i=0; i<5; i++)
  {
    if ((no[i]%2 == 0))
    {
      noGenap = no[i];
      var indeksAwal = 0, indeksAkhir = no.length - 1,
          indeksTengah = Math.floor((indeksAkhir + indeksAwal)/2);
      while(no[indeksTengah] != noGenap && indeksAwal < indeksAkhir)
      {
        if (no[indeksTengah] > noGenap)
        {
          indeksAkhir = indeksTengah - 1;
        }
        else
        {
          if (no[indeksTengah] < noGenap)
          {
            indeksAwal = indeksTengah + 1;
          }
        }
        indeksTengah = Math.floor((indeksAkhir + indeksAwal)/2);
      }

      if (no[indeksTengah] == noGenap)
      {
        document.write("<br>Nombor tengah yang dirujuk: ",
          no[indeksTengah], " (Jumpa nombor genap) ");
      }
    }
  }
</script>
</body>
</html>

```

Senarai nombor yang telah diisi

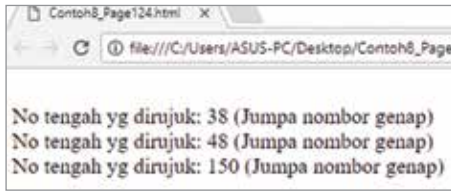
Tentukan nombor genap (nombor yang dicari)

Selagi nombor yang dirujuk tidak sama dengan nombor genap yang dicari

Kiraan semula indeks tengah

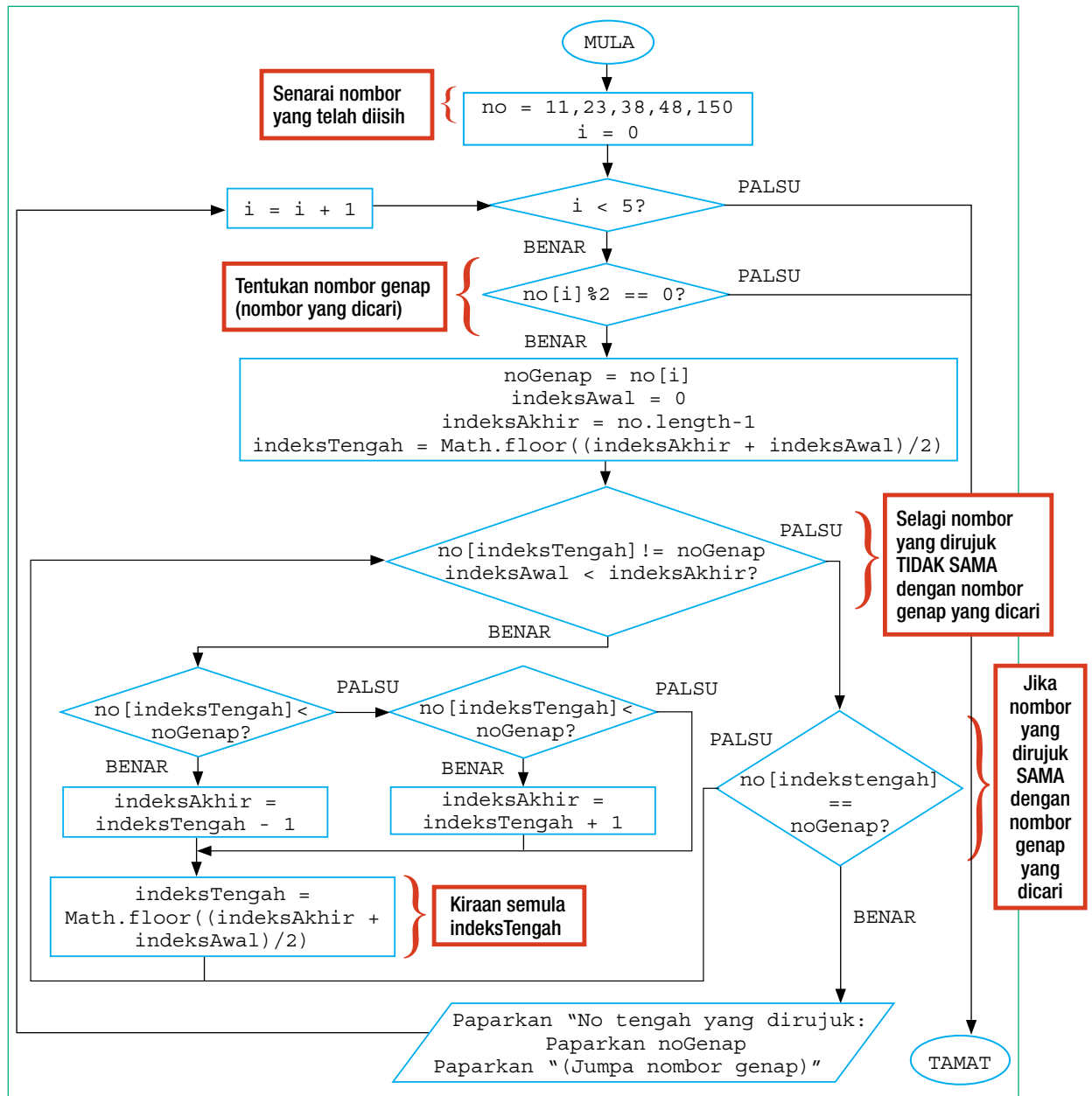
Jika nombor yang dirujuk sama dengan nombor genap yang dicari





Rajah 3.1.26 Atur cara JavaScript dan output bagi carian perduaan (carian nombor genap)

Carta alir untuk Contoh 8 ditunjukkan pada Rajah 3.1.27.

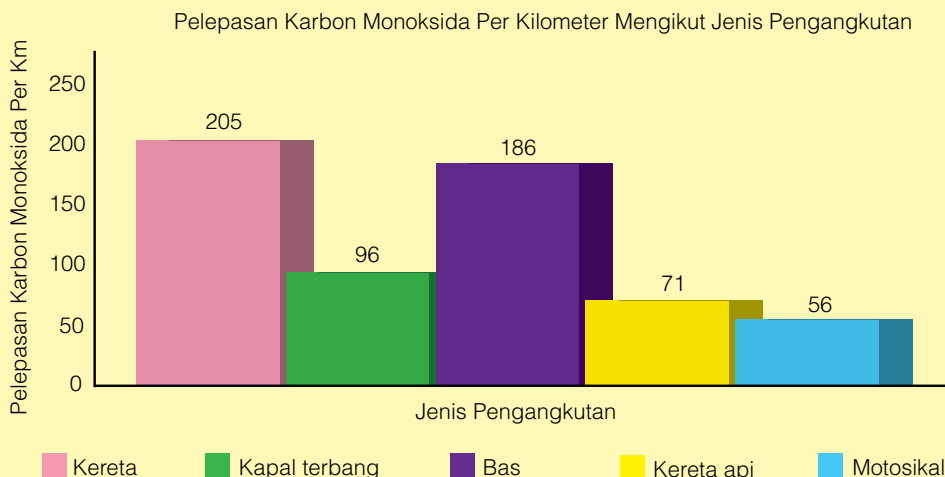


Rajah 3.1.27 Carta alir carian perduaan (carian nombor genap)

Objektif: Menghasilkan atur cara penskripan klien untuk mencari nilai daripada satu senarai nilai menggunakan kaedah carian perduaan (*binary search*).

75% daripada karbon monoksida dilepaskan daripada pengangkutan seperti kereta dan bas. Di kawasan bandar, pelepasan karbon monoksida yang berbahaya menyebabkan pencemaran udara sebanyak 50% hingga 90%.

Diberi maklumat statistik pelepasan karbon monoksida berdasarkan jenis pengangkutan seperti di bawah:



Bahagian 1:

Tulis atur cara penskripan klien yang dapat:

- menerima senarai nilai pelepasan karbon monoksida yang ditunjukkan dalam rajah di atas (205, 96, 186, 71, 56).
- Seterusnya, mencari nilai 186, iaitu pelepasan monoksida oleh sebuah bas menggunakan Carian Perduaan (*binary search*).
- Paparkan mesej "Pelepasan monoksida oleh sebuah bas" jika nilai di (b) dijumpai.

Bahagian 2:

Tulis carta alir yang lengkap bagi atur cara yang telah dibina pada Bahagian 1.

Pengiraan untuk Mendapatkan Nilai-nilai Maksimum, Minimum dan Mean

Mendapatkan Nilai Maksimum

Nilai maksimum merujuk nilai terbesar dalam suatu senarai nilai. Proses mencari nilai maksimum dapat dilaksanakan seperti yang ditunjukkan pada Rajah 3.1.28.



Rajah 3.1.28 Proses mencari nilai maksimum

Terdapat hanya satu proses perbandingan, iaitu bandingkan nilai yang dirujuk dalam senarai sama ada lebih besar daripada nilai maksimum.

CONTOH 9

Contoh 9 menunjukkan proses mencari nombor terbesar (nilai maksimum) di dalam suatu senarai nombor. Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.29.

```

<html>
  <body>
    <script>
      var no = [5,1,4,12,8], noMaksimum = -1,
          i,bilNombor = no.length;
      for(i=0; i<bilNombor; i++)
      {
        if (no[i] > noMaksimum)
        {
          noMaksimum = no[i];
        }
      }
      document.write(noMaksimum);
    </script>
  </body>
</html>

```

Beri nilai awalan kepada noMaksimum.
Dianggap -1 adalah noMaksimum (pilih nombor paling rendah)

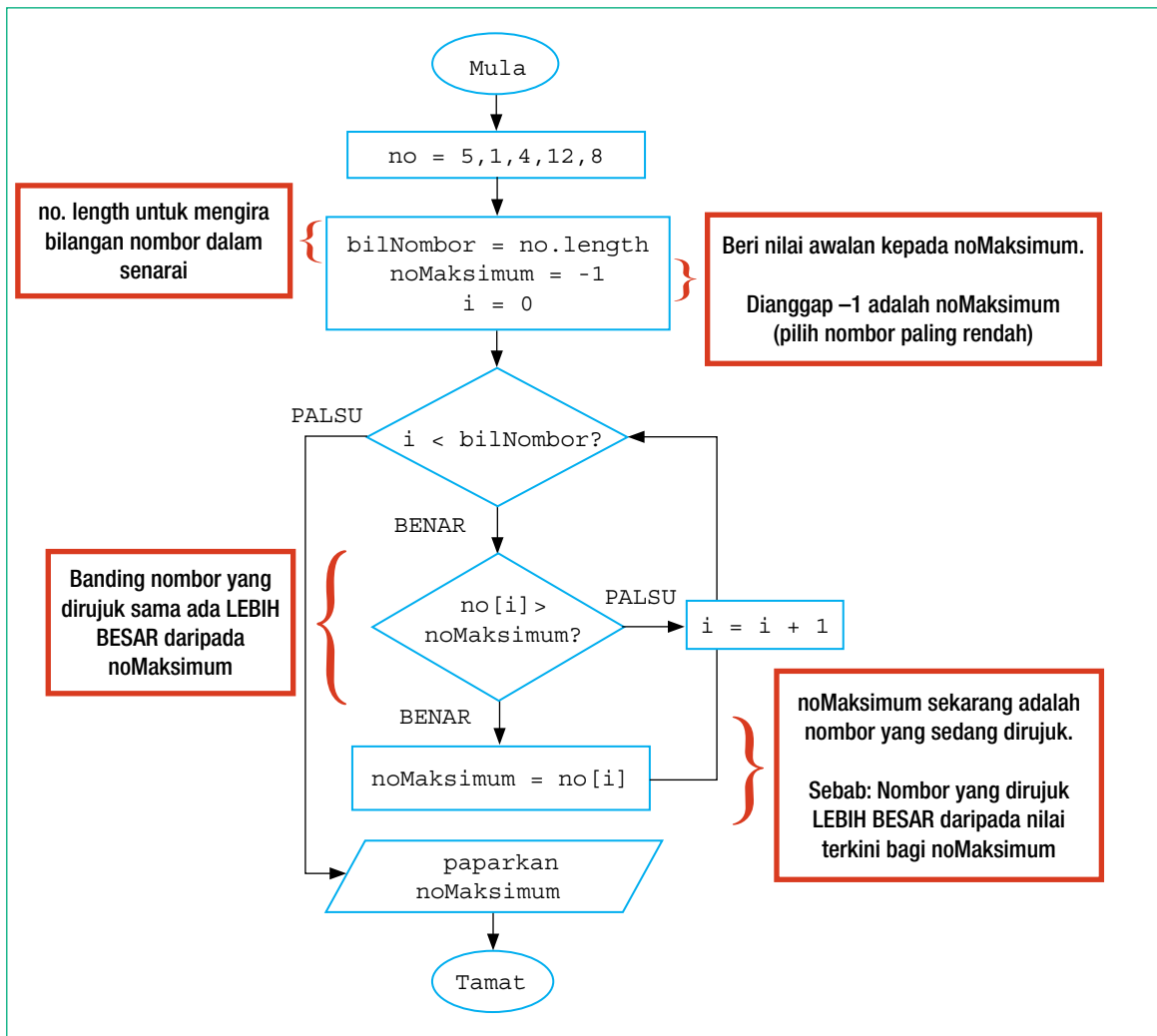
no.length untuk mengira bilangan nombor dalam senarai

Banding nombor yang dirujuk LEBIH BESAR daripada noMaksimum

noMaksimum sekarang adalah nombor yang sedang dirujuk. Sebab: Nombor yang dirujuk LEBIH BESAR daripada nilai terkini bagi noMaksimum

Rajah 3.1.29 Atur cara *JavaScript* dan output (mendapatkan nilai maksimum)

Carta alir untuk Contoh 9 ditunjukkan pada Rajah 3.1.30.



Rajah 3.1.30 Carta alir (mendapatkan nilai maksimum)

Proses mencari nilai maksimum boleh digunakan untuk mencari nilai-nilai lain seperti bangunan tertinggi dan jarak terpanjang.

CONTOH 10

Contoh 10 menunjukkan terdapat satu senarai nilai jarak, iaitu jarak perjalanan dari rumah ke sekolah bagi 5 orang murid. Atur cara akan dapat membantu mencari jarak yang paling panjang diambil oleh murid untuk datang ke sekolah.

Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.31. Proses yang dilaksanakan dalam atur cara untuk mencari jarak terpanjang adalah seperti yang berikut:

- 1 Menetapkan nilai awal bagi jarak terpanjang. Dapatkan nilai paling rendah seperti -1 untuk mewakili nilai awal bagi jarak terpanjang.
- 2 Seterusnya, rujuk jarak pertama dalam senarai.
- 3 Bandingkan jarak pertama dengan nilai awal jarak terpanjang. Jika jarak tersebut lebih besar daripada nilai awal tadi, maka jarak terpanjang telah bertukar kepada jarak pertama. Jika tidak, nilai awal jarak terpanjang akan kekal.
- 4 Ulang langkah yang sama dengan jarak-jarak yang lain dalam senarai.



UJI MINDA

6 7 8 9 4 5 3

Nyatakan nilai maksimum bagi senarai nilai yang ditunjukkan di atas. Kemudian, cuba ikuti proses untuk mendapatkan nilai maksimum bagi suatu senarai seperti yang ditunjukkan dalam Contoh 10.

```

<html>
  <body>
    <script>
      var jarak = [10,8,8,12,5], jarakTerpanjang = -1, i,
      jarak.length untuk mengira bilangan nilai jarak dalam senarai
      bilJarak = jarak.length;
      for(i = 0; i < bilJarak; i++)
      {
        if (jarak[i] > jarakTerpanjang)
        {
          jarakTerpanjang = jarak[i];
        }
      }
      document.write("Jarak terpanjang yang diambil oleh murid ke sekolah (di antara 5 orang murid): " + jarakTerpanjang + " km");
    </script>
  </body>
</html>

```

Senarai jarak yang diambil oleh 5 murid untuk datang ke sekolah

Dianggap -1 adalah jarak terpanjang

Rujuk nilai jarak dalam senarai satu per satu menggunakan indeks bagi array. Bandingkan jarak tersebut dengan jarak terpanjang tadi.

Jika jarak yang dirujuk adalah lebih panjang daripada jarakTerpanjang, maka jadikan nilai jarak yang dirujuk menjadi jarakTerpanjang yang terkini.

Paparkan jarakTerpanjang setelah semua nilai jarak dalam senarai dirujuk.



Rajah 3.1.31 Atur cara *JavaScript* dan output (mendapatkan jarak terpanjang ke sekolah)

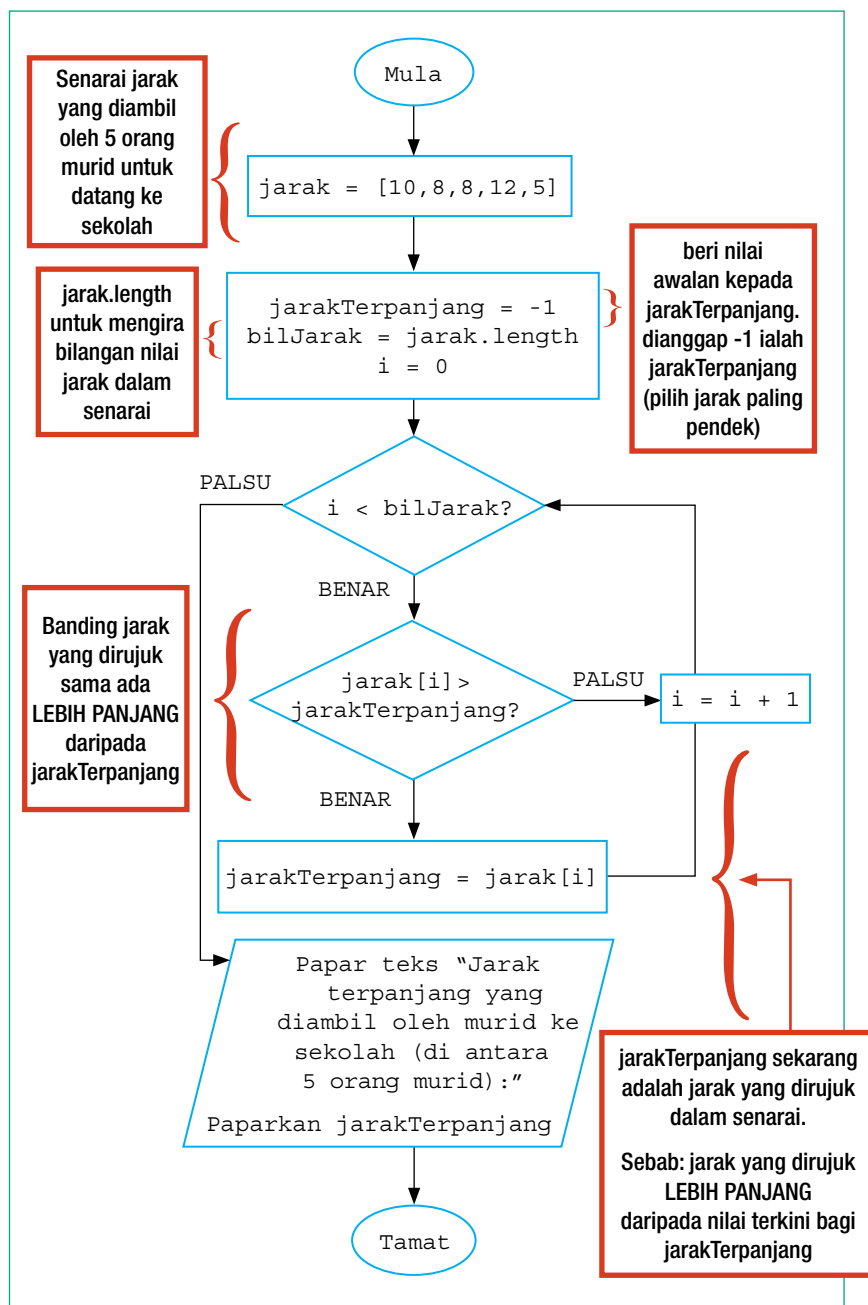
Carta alir untuk Contoh 10 ditunjukkan pada Rajah 3.1.32.



INOVASI SAINS KOMPUTER

Pada 8 Mei 2017, Ekspo *Innovative Research, Invention and Application* (I-RIA) telah diadakan oleh pihak Universiti Utara Malaysia dan kerajaan negeri Kedah untuk mencungkil bakat-bakat inovasi dalam kalangan pelajar. Ekspo ini menjadi platform untuk institusi pendidikan di Malaysia dan Wilayah Indonesia–Malaysia–Thailand mempamerkan produk inovatif merangkumi kategori kejuruteraan komputer, sains komputer dan pengaturcaraan, multimedia, pengurusan data dan storan.

Sumber:
<http://www.uum.edu.my/index.php/en/uum-news/3950-i-ria-2017-draws-record-participation>



Rajah 3.1.32 Carta alir (mendapatkan jarak terpanjang ke sekolah)

Mendapatkan Nilai Minimum

Nilai minimum merujuk nilai terkecil dalam suatu senarai nilai. Proses mencari nilai minimum dapat dilaksanakan seperti ditunjukkan pada Rajah 3.1.33.



Rajah 3.1.33 Proses mencari nilai minimum

Terdapat hanya satu proses perbandingan sahaja, iaitu: banding nilai yang dirujuk dalam senarai lebih kecil daripada nilai minimum.

CONTOH 11

Contoh 11 menunjukkan proses mencari nombor terkecil (nilai minimum) di dalam suatu senarai nombor. Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.34.

```
<html>
  <body>
    <script>
      var no = [5, -1, 4, 12, 8], noMinimum = 999,
          i, bilNombor = no.length;
      for(i = 0; i < bilNombor; i++)
      {
        if (no[i] < noMinimum)
        {
          noMinimum = no[i];
        }
      }
      document.write(noMinimum);
    </script>
  </body>
</html>
```

Beri nilai awalan kepada noMinimum. Dianggap 999 adalah noMinimum (pilih nombor paling tinggi)

no.length untuk mengira bilangan nombor dalam senarai

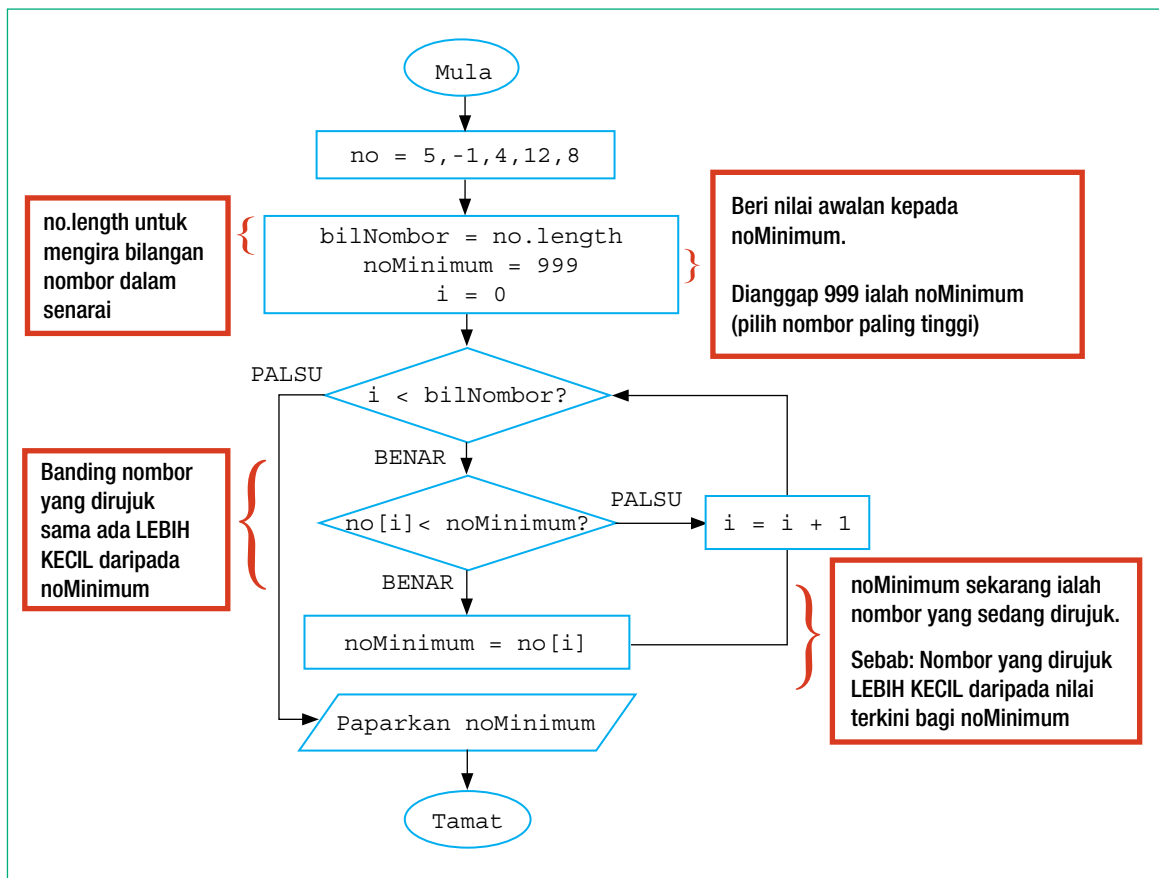
Banding nombor yang dirujuk sama ada LEBIH KECIL daripada noMinimum

noMinimum sekarang ialah nombor yang sedang dirujuk. Sebab: Nombor yang dirujuk LEBIH KECIL daripada nilai terkini bagi noMinimum



Rajah 3.1.34 Atur cara *JavaScript* dan output (mendapatkan nilai minimum)

Carta alir untuk Contoh 11 ditunjukkan pada Rajah 3.1.35.



Rajah 3.1.35 Carta alir (mendapatkan nilai minimum)

Proses mencari nilai minimum boleh digunakan untuk mencari nilai-nilai lain seperti kotak paling ringan dan sungai terpendek.



UJI MINDA

Berikut menunjukkan suatu senarai nilai.

7 2 3 5 6 9 4

Bagaimanakah caranya untuk mendapatkan nilai minimumnya?

CONTOH 12

Contoh 12 menunjukkan berat beg sekolah (dalam kg) bagi enam orang murid yang diambil dan direkodkan seperti dalam senarai. Atur cara akan dapat menentukan berat beg yang paling ringan antara beg tersebut.

Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.36. Proses yang dilaksanakan dalam atur cara untuk mencari berat beg paling ringan adalah seperti yang berikut:

- 1 Menetapkan nilai awal bagi berat beg paling ringan. Dapatkan nilai paling tinggi seperti 999 untuk mewakili nilai awal bagi berat terendah.

- 2 Seterusnya, rujuk berat pertama dalam senarai.
- 3 Bandingkan berat pertama dengan nilai awal berat terendah. Jika nilai berat tersebut lebih kecil daripada nilai awal tadi, maka berat paling ringan telah bertukar kepada berat pertama. Jika tidak, nilai awal berat terendah akan kekal.
- 4 Ulang langkah yang sama dengan berat-berat yang lain dalam senarai.

```

<html>
  <body>
    <script>
      var berat = [3,3,2,5,4,5], beratPalingRingan = 999, i,
          bilBerat = berat.length;

      for(i = 0; i < bilBerat; i++)
      {
        if (berat[i] < beratPalingRingan)
        {
          beratPalingRingan = berat[i];
        }
      }

      document.write("Berat beg paling ringan (di kalangan 6 orang murid): "
        + beratPalingRingan + " kg");
    </script>
  </body>
</html>

```

Senarai berat beg sekolah daripada 6 orang murid

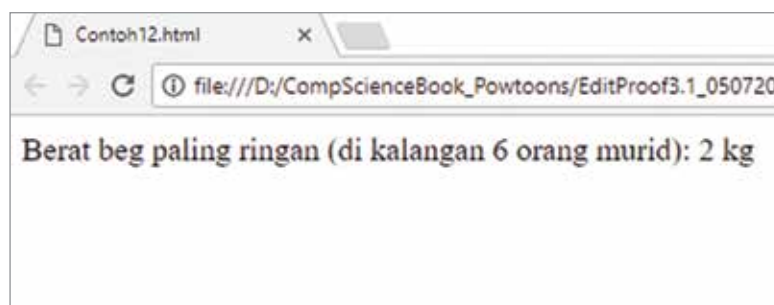
berat.length untuk mengira bilangan nilai berat dalam senarai

Dianggap 999 adalah berat terendah

Rujuk nilai berat beg dalam senarai satu persatu menggunakan indeks bagi tatasusunan (array). Bandingkan berat tersebut dengan berat terendah tadi.

Jika nilai berat yang dirujuk adalah lebih kecil daripada berat terendah, maka jadikan nilai berat yang dirujuk menjadi berat terendah yang terkini.

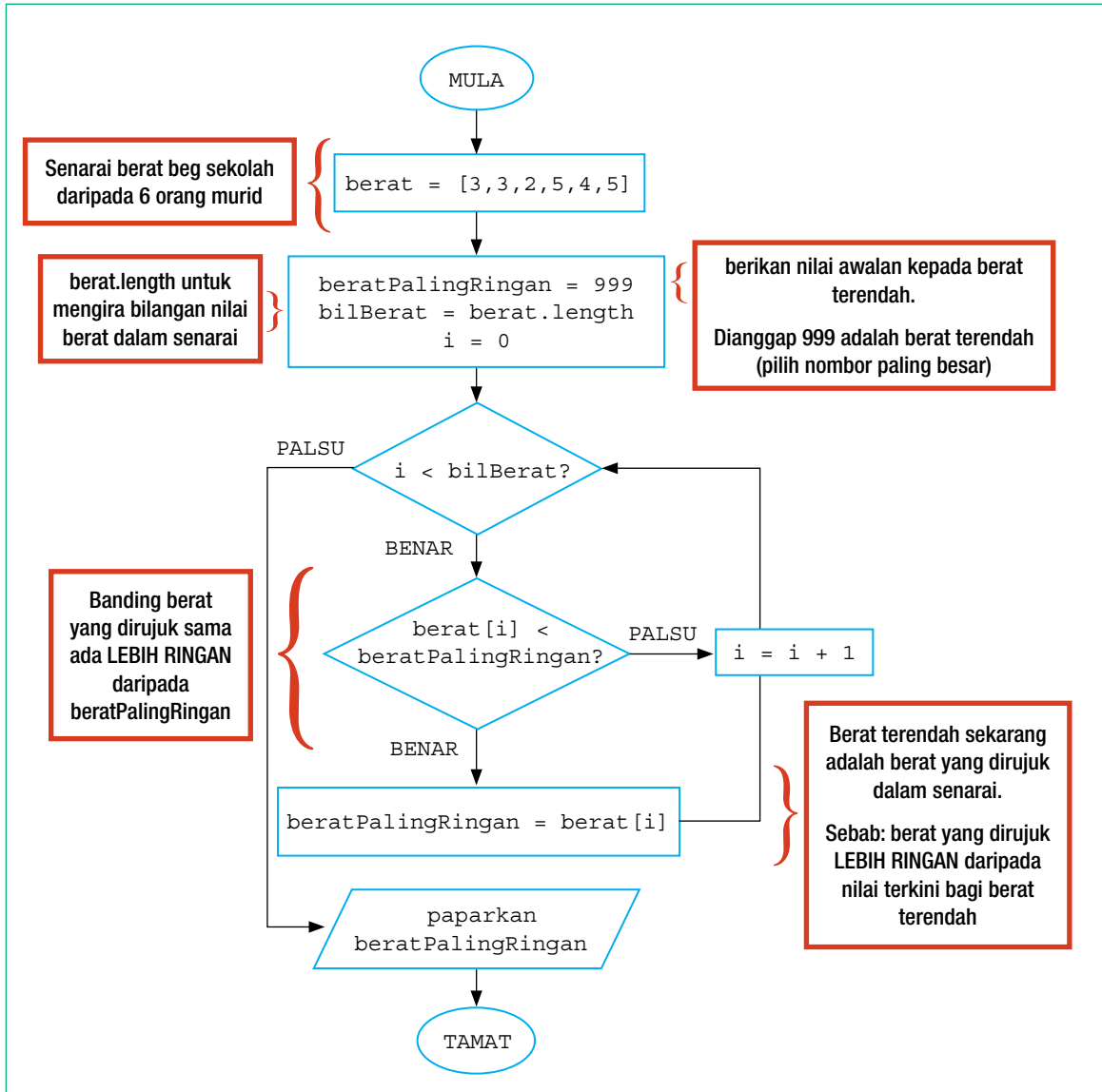
Paparkan berat terendah setelah semua nilai berat dalam senarai dirujuk.



Rajah 3.1.36 Atur cara JavaScript dan output (mendapatkan berat beg paling ringan)



Carta alir untuk Contoh 12 ditunjukkan pada Rajah 3.1.37.



Rajah 3.1.37 Carta alir (mendapatkan berat beg paling ringan)

AKTIVITI 5

Kerja Individu

Carta Alir Bagi Mencari Nilai Maksimum Dan Nilai Minimum

Bina carta alir bagi suatu aplikasi yang mempunyai satu senarai kos membuat kuih bagi enam jenis kuih, iaitu 3.00, 2.50, 1.50, 3.10, 4.00, 2.00 dan mencari serta memaparkan kos kuih yang tertinggi (maksimum) dan yang terendah (minimum).

Mendapatkan Nilai *Mean*

Mean merujuk purata nilai dalam suatu senarai nilai. Untuk mendapatkan purata, yang berikut ialah langkah-langkah yang perlu dilaksanakan:

- 1 Jumlahkan nilai-nilai dalam senarai nilai tersebut
- 2 Dapatkan bilangan nilai yang terdapat dalam senarai
- 3 Nilai *mean* dikira dengan membahagikan nilai jumlah dengan bilangan nilai yang terdapat dalam senarai (Formula: $Mean = \text{Jumlah}/\text{Bilangan}$)

CONTOH 13

Contoh 13 menunjukkan proses mengira dan mendapatkan nilai purata (*mean*) bagi sebilangan nombor. Nombor-nombor dimasukkan oleh pengguna dan seterusnya atur cara mengira jumlah nombor-nombor tersebut. Purata akan dikira dengan menggunakan formula purata = jumlah ÷ bilangan.

Untuk mengira nilai purata, nilai jumlah dan bilangan perlu diperolehi dahulu. Formula bagi jumlah ialah jumlah = jumlah + no manakala formula bagi bilangan ialah pembilang = pembilang + 1.

Contoh atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.38(a) dan 3.1.38(b).

```
<html>
  <body>
    <script>
      var no, pembilang=0, ulang=false, jumlah=0, purata=0;
      do
      {
        no = parseInt (prompt("Masukkan satu nombor:"));
        pembilang = pembilang + 1;
        jumlah = jumlah + no;
        ulang = confirm("Ulang sekali lagi?");
      }while(ulang == true);

      purata = jumlah/pembilang;
      document.write(purata);

    </script>
  </body>
</html>
```

Memberi nilai awal kepada pembilang, jumlah dan purata (iaitu nilai awal ialah 0) Sebab: Tiada nilai lagi untuk dikira

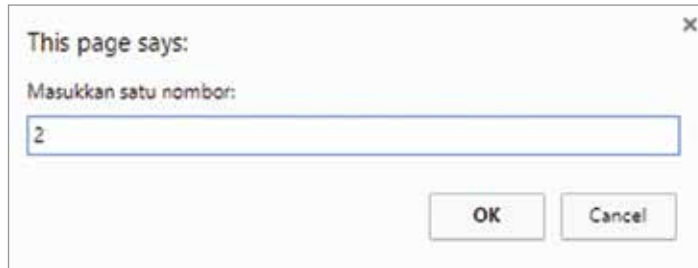
Mengira bilangan (dengan menambah 1 kepada pembilang) dan mengira jumlah (dengan menambah nilai kepada jumlah)

Mengira purata nilai. Formula: jumlah/pembilang

nombor dimasukkan oleh pengguna (*prompt*) dan pastikan bernilai integer (*parseInt*)

Rajah 3.1.38(a) Atur cara *JavaScript* (mendapatkan nilai *mean*)

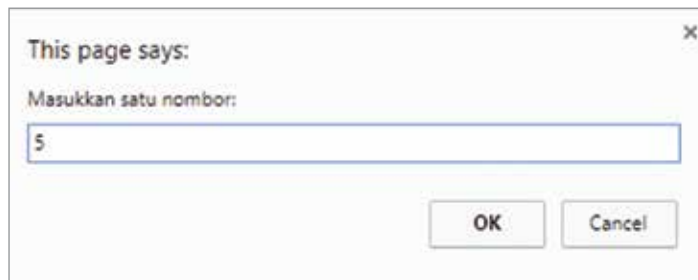
Meminta pengguna memasukkan satu nombor.



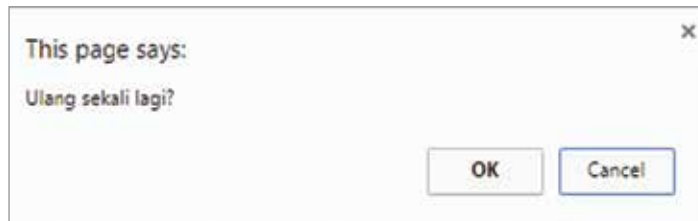
Meminta pengguna klik "OK" jika pengguna ingin memasukkan nombor lagi



Meminta pengguna memasukkan satu nombor.



Meminta pengguna klik "Cancel" jika pengguna ingin tamatkan kemasukkan nombor.

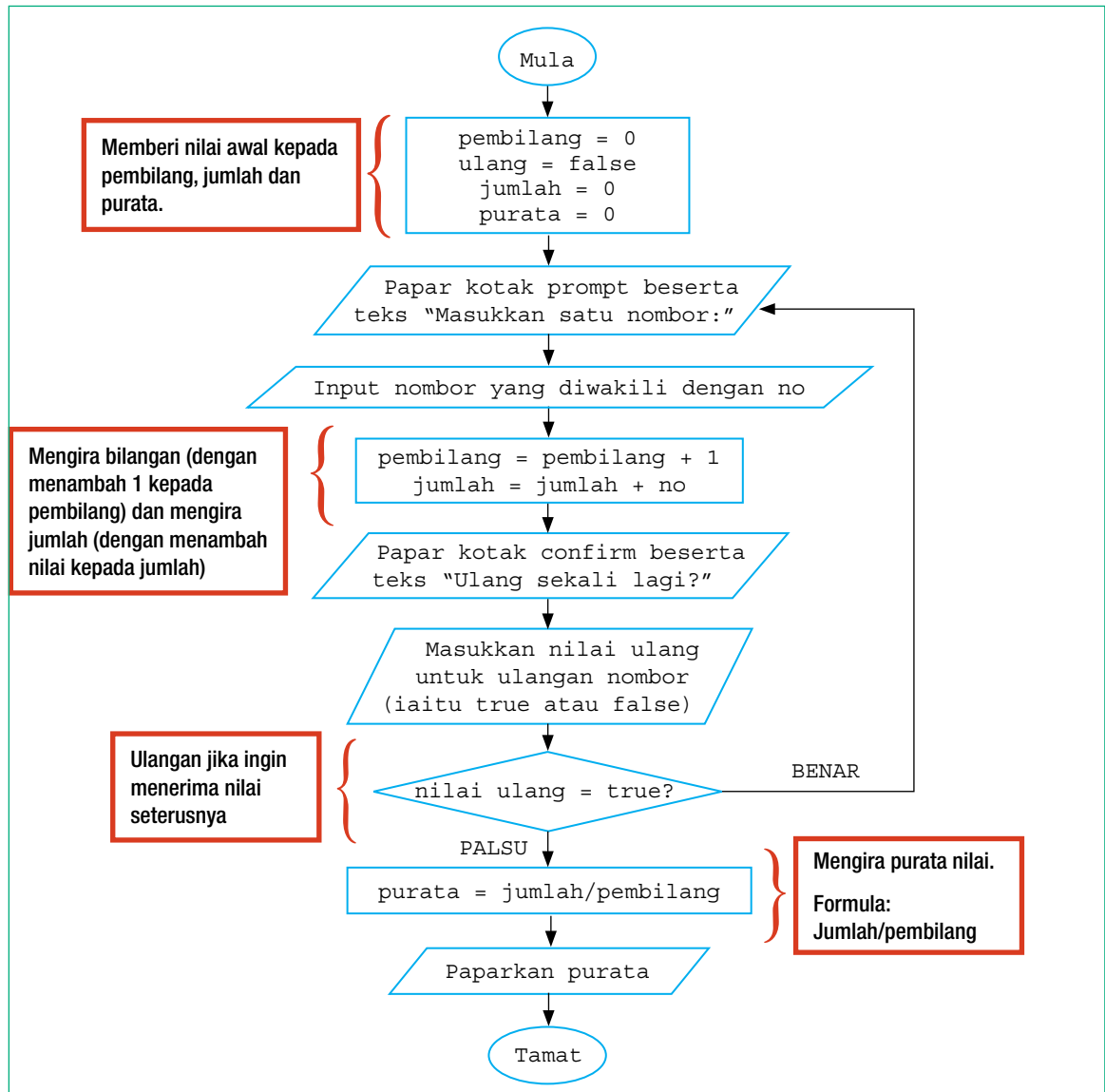


Memaparkan nilai purata.



Rajah 3.1.38(b) Output (mendapatkan nilai *mean*)

Carta alir untuk Contoh 13 ditunjukkan pada Rajah 3.1.39.



Rajah 3.1.39 Carta alir (mendapatkan nilai *mean*)

AKTIVITI

6

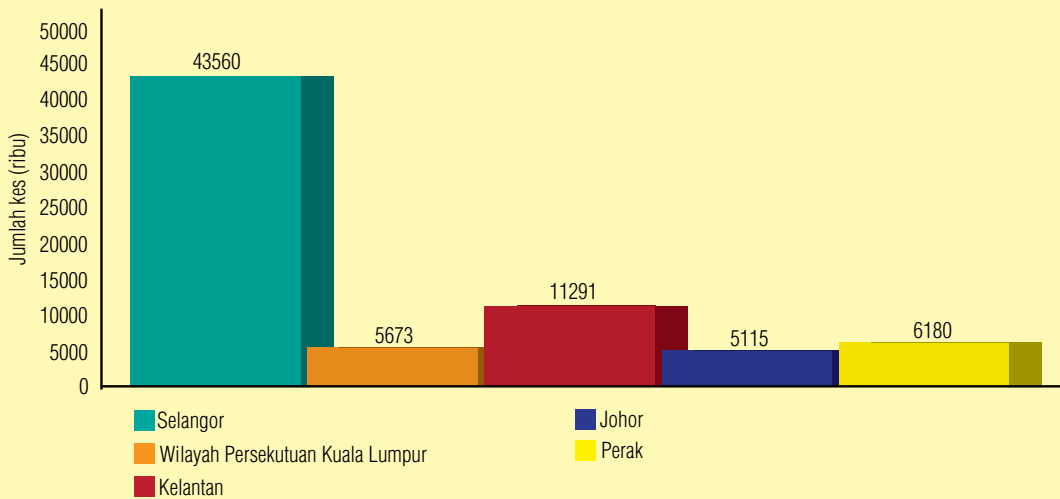
Kerja Individu

Carta Alir Bagi Mencari Nilai *Mean*

Objektif: Menunjukkan langkah-langkah untuk mencari nilai *mean* daripada satu senarai nilai.

Wabak denggi merupakan wabak yang semakin menular di Malaysia. Setiap tahun, jumlah keseluruhan kes yang melibatkan wabak denggi semakin meningkat. Carta palang pada halaman yang seterusnya telah menunjukkan jumlah kes keseluruhan wabak denggi di lima buah negeri di Malaysia pada tahun 2015.

Jumlah Kes Keseluruhan Wabak Denggi di Lima Buah Negeri di Malaysia Pada Tahun 2015



Sumber:

Portal Kementerian Kesihatan Malaysia

<http://www.moh.gov.my/index.php/pages/view/1674>

Berdasarkan graf di atas, lukis carta alir menunjukkan langkah-langkah untuk mendapatkan nilai *mean* atau purata bagi jumlah kes keseluruhan wabak denggi di lima buah negeri di Malaysia pada tahun 2015.



INOVASI SAINS KOMPUTER

Peningkatan jumlah kematian akibat wabak denggi telah mendorong pembangun aplikasi di Malaysia, iaitu Hilman Nordin dari Universiti Malaya untuk membangunkan aplikasi *iContageo*. Aplikasi ini membantu rakyat Malaysia untuk menjangkakan lokasi “hotspot” bagi wabak denggi berdasarkan aduan daripada rakyat. Hilman Nordin merupakan salah seorang yang terlibat dalam membangunkan aplikasi tersebut.

Sumber:

<https://govinsider.asia/inclusive-gov/malaysia-launches-dengue-alerts-app/>

Mendapatkan Nilai *Count*

Count mengira bilangan nilai yang terdapat dalam suatu senarai. Untuk mengira dan mendapatkan nilai *count*, yang berikut ialah langkah-langkah yang perlu dilaksanakan:

- 1 Menggunakan suatu perwakilan sebagai pembilang dan memberikan nilai awalan pembilang sebagai 1.
- 2 Nilai *count* dikira dengan menambah pembilang dengan 1 bagi setiap nilai yang dicapai atau dirujuk (Formula: pembilang = pembilang + 1).
- 3 Memaparkan nilai akhir pembilang, iaitu nilai *count*.

CONTOH 14

Contoh 14 menunjukkan proses mengira bilangan nama yang dimasukkan oleh pengguna.

Atur cara akan mengira bilangan nama tersebut dengan menggunakan formula pembilang = pembilang + 1 sebaik sahaja nama dimasukkan oleh pengguna. Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.40(a) dan 3.1.40(b).


```

<html>
  <body>
    <script>
      var nama, pembilang = 0,
          ulang = false;
      do
      {
        nama = prompt("Masukkan nama anda:");
        pembilang = pembilang + 1;
        ulang = confirm("Ulang sekali lagi?");
      }while(ulang == true);

      document.write(pembilang);

    </script>
  </body>
</html>

```

Memberi nilai awal kepada pembilang (iaitu nilai awal ialah 0)
Sebab: Tiada nilai lagi untuk dikira

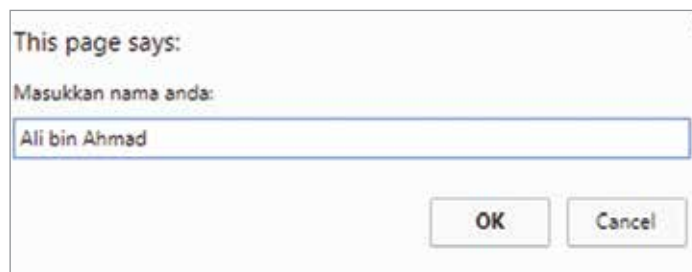
Ulangan jika ingin menerima nilai seterusnya

Menerima sebarang nilai (contoh: nama)

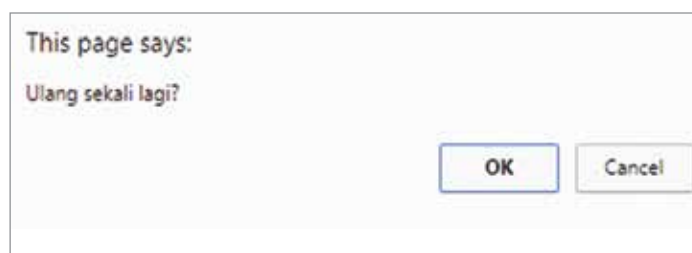
Mengira bilangan (dengan menambah 1 kepada pembilang)

Rajah 3.1.40(a) Atur cara JavaScript (mendapatkan nilai count)

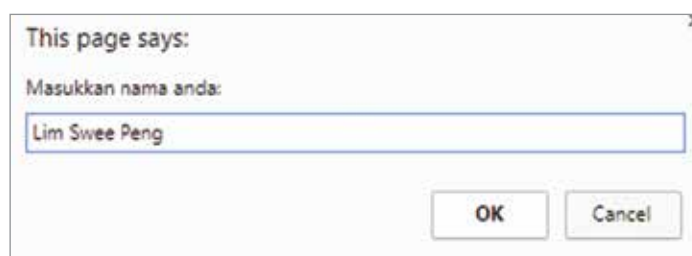
Meminta pengguna memasukkan nama.



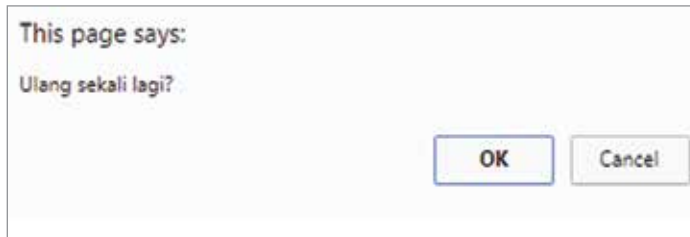
Meminta pengguna klik "OK" jika pengguna ingin memasukkan nama lagi.



Meminta pengguna memasukkan nama.



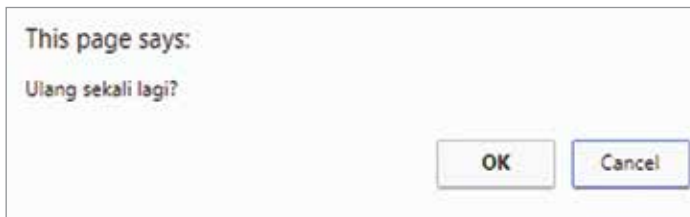
Meminta pengguna klik “OK” jika pengguna ingin memasukkan nama lagi.



Meminta pengguna memasukkan nama.



Meminta pengguna klik “Cancel” jika pengguna ingin tamatkan kemasukan nama.

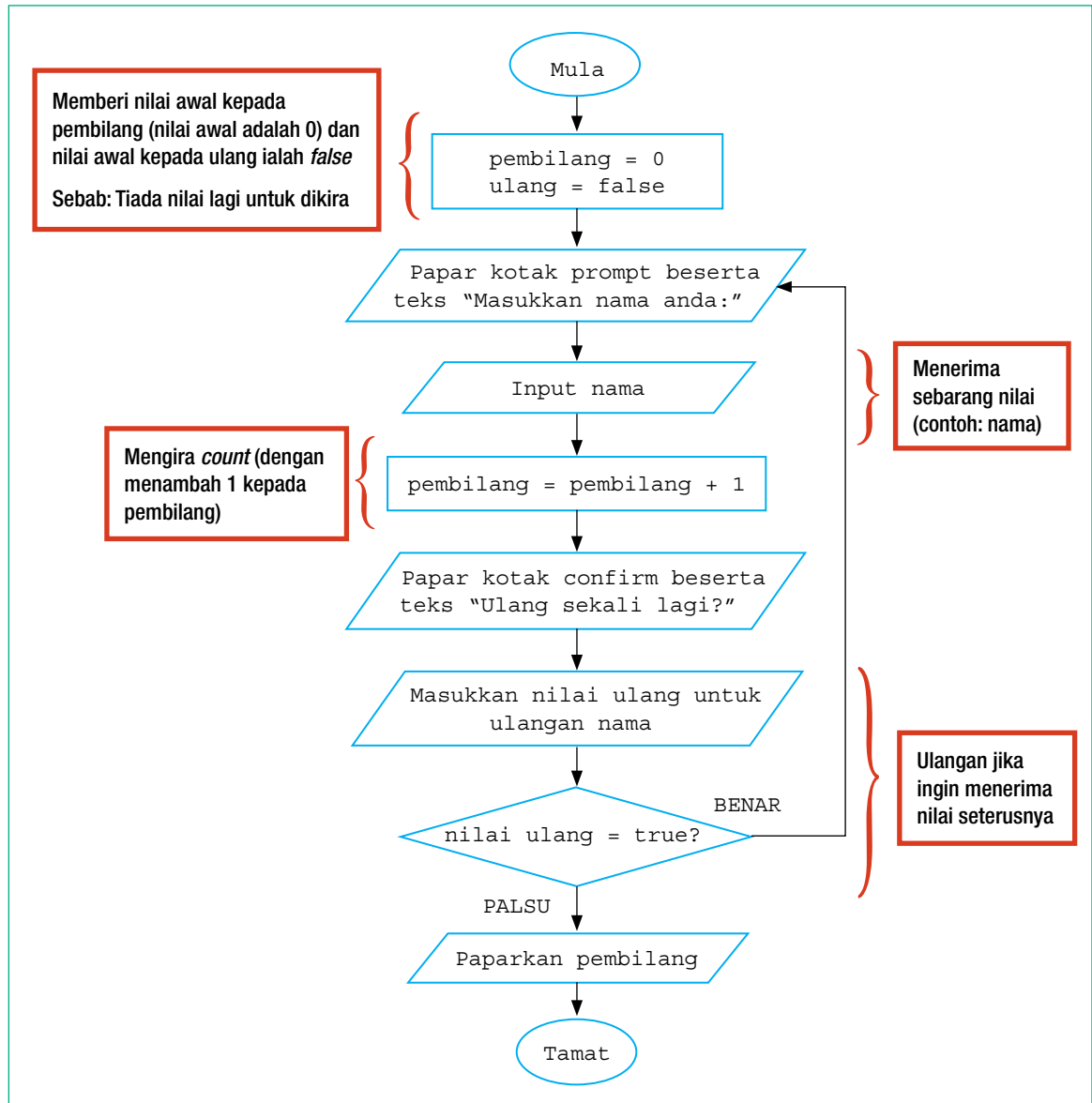


Memaparkan bilangan nama yang telah dimasukkan.



Rajah 3.1.40(b) Output (mendapatkan nilai *count*)

Carta alir untuk Contoh 14 ditunjukkan pada Rajah 3.1.41.



Rajah 3.1.41 Carta alir (mendapatkan nilai *count*)

AKTIVITI

7

Kerja Individu

Carta Alir Bagi Mencari Nilai *Count*

Lakukan aktiviti ini di dalam makmal komputer.

Bahagian 1:

- 1 Buka perisian *Notepad*.

2 Taip atur cara HTML dan *JavaScript* yang diberikan:

```
<html>
  <body>
    <script>
      var nama, pembilang=0, ulang=false;

      do
      {
        nama = prompt("LARIAN AMAL SEKOLAH BESTARI \nMasukkan nama
          peserta:");
        pembilang=pembilang+1;

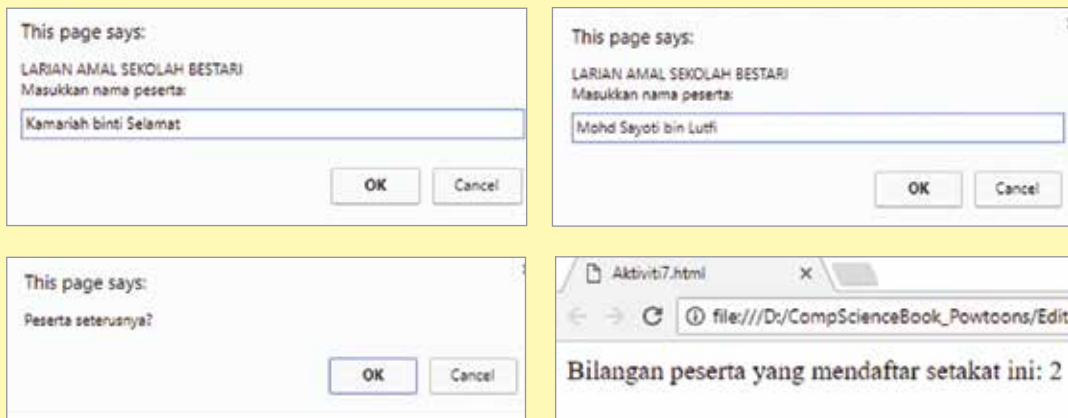
        ulang = confirm("Peserta seterusnya?" ) ;
      }while(ulang==true);

      document.write("Bilangan peserta yang mendaftar setakat ini: " +
        pembilang);
    </script>
  </body>
</html>
```

3 Simpan (save) sebagai Aktiviti7.html

4 Buka fail Aktiviti7.html melalui pelayar web (*web browser*) seperti *Internet Explorer* atau *Google Chrome*.

5 Berikut adalah contoh output yang dihasilkan:



Murid boleh memasukkan beberapa nama peserta dan pastikan bilangan (output) yang dipaparkan adalah betul.

Bahagian 2:

1 Ambil sekeping kertas yang baharu.

2 Lukis carta alir bagi atur cara yang telah dibina di Bahagian 1, iaitu atur cara bagi mengira dan mendapatkan bilangan peserta larian amal Sekolah Bestari berdasarkan nama-nama peserta yang didaftarkan.

Pengumpulan nilai dalam struktur giliran (*queue*)

Giliran (*Queue*) merupakan struktur data linear, iaitu pengumpulan nilai dalam ruang ingatan secara teratur. *Queue* boleh membuang nilai lama, iaitu nilai yang terawal dimasukkan. Kaedah ini dipanggil FIFO (*first-in, first-out*).

Untuk memahami konsep *queue* ini, cuba murid bayangkan sistem angka giliran di kaunter bank. Pelanggan akan dipanggil dan dilayan mengikut turutan nombor tiket yang diambil.

Operasi *queue*

Terdapat dua operasi penting dalam *queue*, iaitu:

- 1 *enqueue* – Memasukkan data ke dalam *queue*
- 2 *dequeue* – Membuang data lama dalam *queue*

Operasi *queue* boleh dilakukan menggunakan fungsi `push()` dan `shift()` dalam Rajah 3.1.42 atau `unshift()` dan `pop()` bagi objek tatasusunan seperti yang ditunjukkan dalam Rajah 3.1.43.

Kelebihan

Keempat-empat fungsi ini sangat mudah digunakan dengan hanya memanggil fungsi-fungsi tersebut.

Kekangan

Fungsi-fungsi ini tidak sesuai digunakan jika:

- Mempunyai *queue* yang besar (mempunyai *queue* yang panjang atau nilai *queue* yang banyak) kerana pelaksanaannya menggunakan tatasusunan (*array*).
- Nilai-nilai perlu digerakkan kedudukannya setiap kali fungsi `shift()` dipanggil dan digunakan.

Penggunaan `push()` dan `shift()` dalam JavaScript

- Nilai awal dimasukkan melalui bahagian belakang ruang ingatan *array*.
- Nilai awal dikeluarkan melalui bahagian depan ruang ingatan *array*.

	depan		belakang
<code>push(5)</code>			5
	depan		belakang
<code>push(-1)</code>		5	-1
	depan		belakang
<code>push(12)</code>	5	-1	12
	depan		belakang
<code>shift()</code>	-1	12	



UJI MINDA

Pada pendapat anda, bagaimanakah fungsi *queue* dapat diaplikasikan dalam sistem nombor giliran di sebuah bank?

Penerangan:

- 1 Arahan `push(5)` akan memasukkan nilai 5 dari arah belakang ke dalam *queue*.
- 2 Arahan `push(-1)` akan memasukkan nilai -1 dari arah belakang ke dalam *queue*, iaitu selepas nilai 5 dimasukkan. Begitu juga dengan arahan `push(12)`.
- 3 Arahan `shift()` akan mengeluarkan nilai terawal yang dimasukkan, iaitu nilai 5 dalam contoh di atas.

CONTOH 15

Contoh 15 menunjukkan pengumpulan nombor ke dalam senarai yang dikenali sebagai giliran (*queue*) menggunakan subatur cara `push()`. Seterusnya, proses mengeluarkan nombor daripada giliran juga dilaksanakan menggunakan subatur cara `shift()`. Nombor pertama yang diletakkan akan dikeluarkan daripada giliran. Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.42.

```
<html>
  <body>
    <script>
      var queueNo = [], noBaru, noKeluar;

      noBaru = 5;
      queueNo.push(noBaru);
      document.write("<br>Nombor Baru Masuk: ", noBaru);

      noBaru = -1;
      queueNo.push(noBaru);
      document.write("<br>Nombor Baru Masuk: ", noBaru);

      noBaru = 12;
      queueNo.push(noBaru);
      document.write("<br>Nombor Baru Masuk: ", noBaru);

      document.write("<br><br>Senarai Nombor Terkini Dalam Giliran
        (Queue):<br>-----");
      document.write("<br>", queueNo);

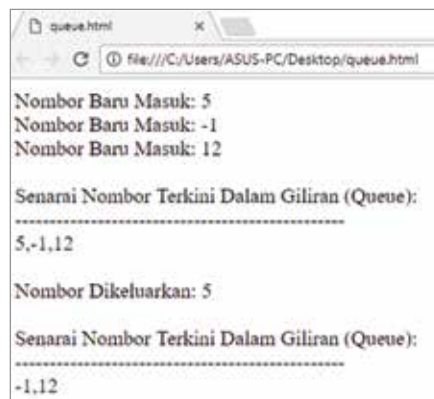
      noKeluar = queueNo.shift();
      document.write("<br><br>Nombor Dikeluarkan: ", noKeluar);
      document.write("<br><br>Senarai Nombor Terkini Dalam Giliran
        (Queue):<br>-----");
      document.write("<br>", queueNo);

    </script>
  </body>
</html>
```

Queue kosong

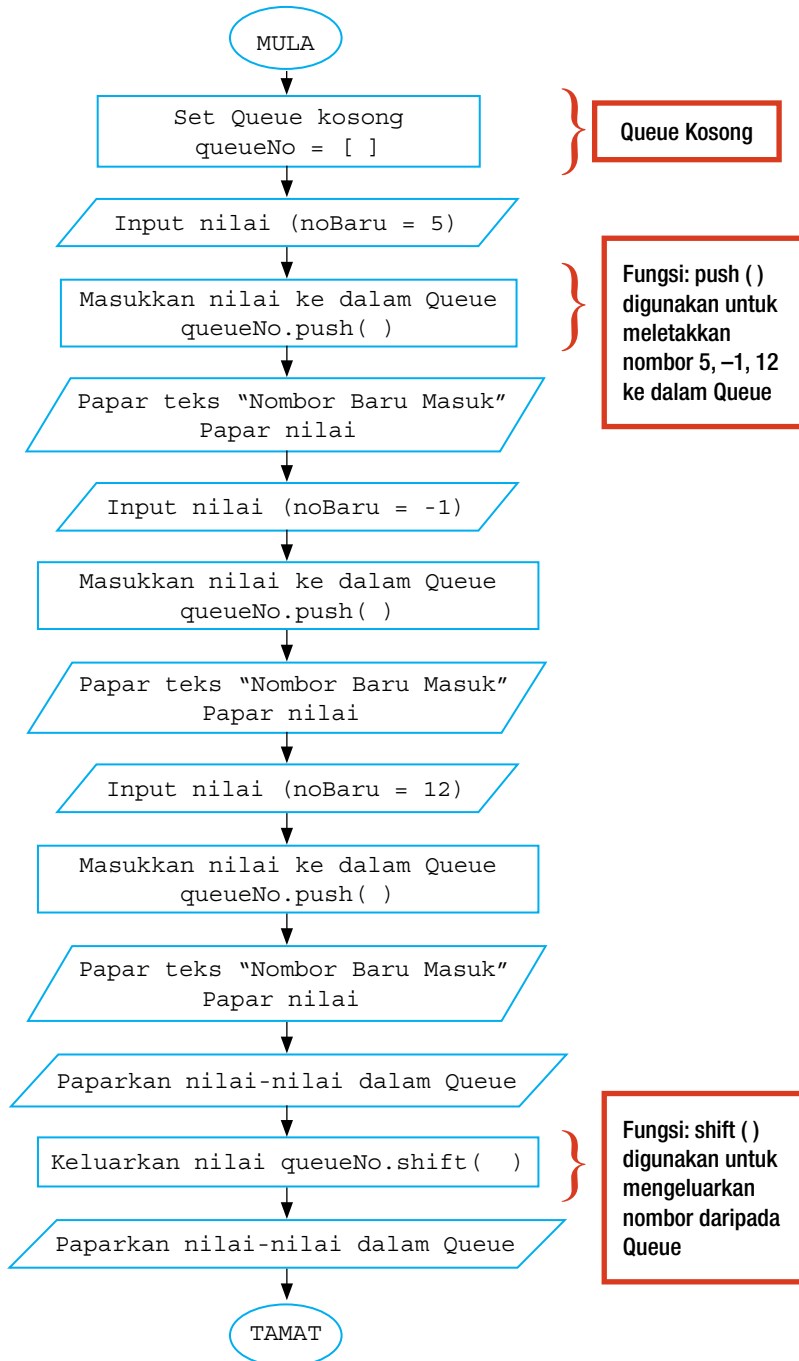
Fungsi push () digunakan untuk meletakkan nombor 5, -1, 12 ke dalam Queue.

Fungsi shift() untuk mengeluarkan nombor daripada Queue



Rajah 3.1.42 Atur cara *JavaScript* dan output (pengumpulan nilai-nilai dalam struktur *queue* – subatur cara `push()` dan `shift()`)

Carta alir untuk Contoh 15 ditunjukkan pada Rajah 3.1.43.



Rajah 3.1.43 Carta alir (pengumpulan nilai-nilai dalam struktur *queue*)

Penggunaan `unshift ()` dan `pop ()` dalam *Javascript*

- Nilai awal dimasukkan melalui bahagian depan ruang ingatan *array*.
- Nilai awal dikeluarkan melalui bahagian belakang ruang ingatan *array*.

	depan		belakang
<code>unshift (5)</code>	5		
	depan		belakang
<code>unshift (-1)</code>	-1	5	
	depan		belakang
<code>unshift (12)</code>	12	-1	5
	depan		belakang
<code>pop ()</code>		12	-1

Penerangan:

- 1 Arah `unshift (5)` akan memasukan nilai 5 dari arah hadapan ke dalam *queue*.
- 2 Arah `unshift (-1)` akan memasukkan nilai -1 dari arah hadapan ke dalam *queue*. Begitu juga dengan arahan `unshift (12)`.
- 3 Arah `pop ()` akan mengeluarkan nilai terawal yang dimasukkan, iaitu nilai 5 walaupun 5 berada di belakang *queue*.

CONTOH 16

Contoh 16 menunjukkan pengumpulan nombor ke dalam senarai yang dikenali sebagai giliran (*queue*) menggunakan subatur cara `unshift ()`. Seterusnya, proses mengeluarkan nombor daripada giliran juga dilaksanakan menggunakan subatur cara `pop ()`. Nombor pertama yang diletakkan akan dikeluarkan daripada giliran. Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.44.

```

<html>
  <body>
    <script>
      var queueNo = [], noBaru, noKeluar;

      noBaru = 5;
      queueNo.unshift (noBaru);
      document.write("<br>Nombor Baru Masuk: ", noBaru);

      noBaru = -1;
      queueNo.unshift (noBaru);
      document.write("<br>Nombor Baru Masuk: ", noBaru);

      noBaru = 12;
      queueNo.unshift (noBaru);
      document.write("<br>Nombor Baru Masuk: ", noBaru);

      document.write("<br><br>Senarai Nombor Terkini Dalam Giliran
        (Queue):<br>-----");
      document.write("<br>", queueNo);

      noKeluar = queueNo.pop();
      document.write("<br><br>Nombor Dikeluarkan: ", noKeluar);
      document.write("<br><br>Senarai Nombor Terkini Dalam Giliran
        (Queue):<br>-----");
      document.write("<br>", queueNo);

    </script>
  </body>
</html>

```

Queue kosong

Fungsi: `unshift ()` digunakan untuk meletakkan nombor 5, -1, 12 ke dalam Queue

Fungsi: `pop ()` untuk mengeluarkan nombor daripada Queue


```

contoh15.html x
file:///C:/Users/AGUS-PC/Desktop/contoh15.html

Nombor Baru Masuk: 5
Nombor Baru Masuk: -1
Nombor Baru Masuk: 12

Senarai Nombor Terkini Dalam Giliran (Queue):
-----
12,-1,5

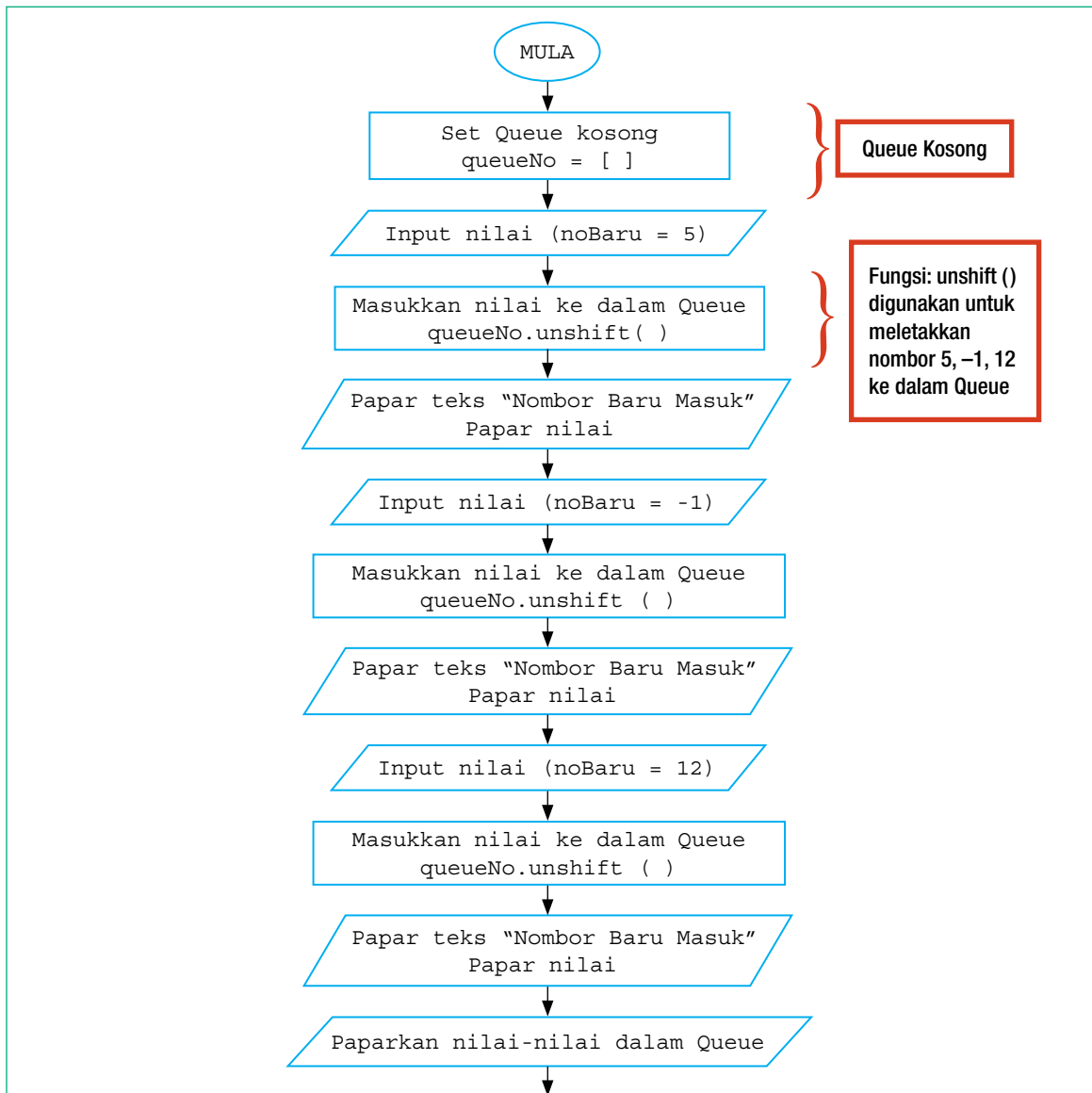
Nombor Dikeluarkan: 5

Senarai Nombor Terkini Dalam Giliran (Queue):
-----
12,-1

```

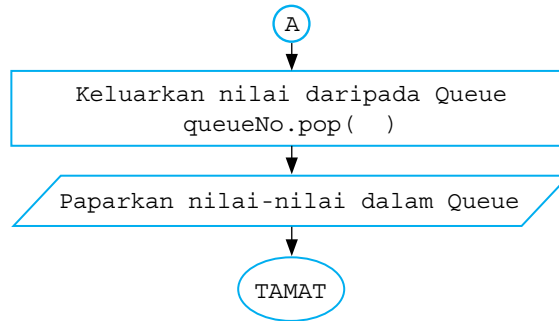
Rajah 3.1.44 Atur cara JavaScript dan output (pengumpulan nilai-nilai dalam struktur *queue* – subatur cara `unshift()` dan `pop()`)

Carta alir untuk Contoh 16 ditunjukkan dalam Rajah 3.1.45.



Queue Kosong

Fungsi: `unshift()` digunakan untuk meletakkan nombor 5, -1, 12 ke dalam Queue



Rajah 3.1.45 Carta alir (pengumpulan nilai-nilai dalam struktur *queue*)

3.1.3 Atur cara yang Mempunyai Pemalar, Pemboleh ubah dan Jenis Data Berlainan bagi Bahasa Penskripan Klien

Seperti yang telah dipelajari sebelum ini, atur cara penskripan klien boleh ditulis menggunakan *JavaScript*. Atur cara *JavaScript* klien boleh ditulis dan dibenam (*embed*) dalam dokumen HTML dan diinterpretasikan oleh pelayar (*browser*).

Dalam topik ini, murid akan menggunakan pemalar, pemboleh ubah dan jenis data yang berlainan dalam atur cara *bubble sort*, *selection sort*, *binary search*, mendapatkan nilai maksimum, minimum, purata dan nilai *count*. Begitu juga atur cara dalam pengumpulan nilai dalam *queue*.

Pemboleh ubah

Pemboleh ubah ialah ruang ingatan yang digunakan untuk menyimpan nilai sementara sewaktu atur cara diproses. Nilainya boleh berubah-ubah mengikut penggunaan pemboleh ubah atau arahan kepada pemboleh ubah. Setiap pemboleh ubah akan diberi nama mengikut nilai yang disimpan. Nama yang diberi kepada pemboleh ubah dikenali sebagai pengecam.

Contoh:

- Nilai gaji pekerja. Nama pemboleh ubah yang boleh diberi ialah gajiPekerja.
- Nama pelajar. Nama pemboleh ubah yang boleh diberi ialah namaPelajar.

Pemalar

Pemalar merupakan ruang ingatan yang digunakan untuk menyimpan nilai sementara sewaktu atur cara diproses. Pemalar sama seperti pemboleh ubah tetapi nilainya tidak boleh berubah-ubah. Nilai tidak boleh berubah kerana nilai yang ditentukan itu merupakan suatu nilai yang telah ditetapkan dalam atur cara.

Contoh nilai pemalar:

- Nilai pi (π), iaitu bernilai 3.142 (ditulis dalam bentuk tiga titik perpuluhan).
- Bilangan hari dalam seminggu, iaitu tujuh hari.

Const ialah kata kunci yang digunakan dalam *JavaScript* untuk merujuk pemalar.

Jenis Data

Jenis data merupakan jenis nilai atau pelbagai set data yang digunakan dalam atur cara. Jadual 3.1.2 menunjukkan jenis data asas dalam *JavaScript*.

Jadual 3.1.2 Jenis data asas dalam *JavaScript*

Jenis data	Contoh data	
Nombor	5	100.50
Rentetan (<i>String</i>)	"Ali bin Abu"	"BMW 318i"
<i>Boolean</i>	true	false

Murid boleh melakukan proses-proses seperti mengisih (kaedah *bubble sort* atau *selection sort*), mencari nilai tertentu (kaedah *binary search*), mencari nilai maksimum (*max*), mencari nilai minimum (*min*), mengira nilai purata (*mean*) dan mengira bilangan nilai (*count*) kepada pelbagai jenis nilai atau data yang telah diterangkan sebelum ini.

CONTOH 17

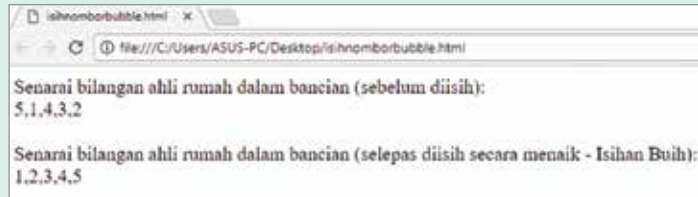
Contoh 17 menunjukkan senarai bilangan ahli rumah dalam bancian penduduk diambil dan direkodkan. Bilangan ahli rumah dimasukkan ke dalam pemboleh ubah `banciPenduduk` dan seterusnya diisih secara menaik menggunakan isihan buih. Atur cara *JavaScript* dan contoh output ditunjukkan pada Jadual 3.1.3.

Jadual 3.1.3 Contoh atur cara *JavaScript* dan output isihan buih (*bubble sort*) (Jenis data nombor)

Jenis data	Contoh atur cara penskripan bagi isihan buih (<i>Bubble Sort</i>)
Nombor	<pre><html> <body> <script> var banciPenduduk=[5,1,4,3,2]; 1 var bilNombor = banciPenduduk.length, i, j; var sementara = banciPenduduk[0]; document.write("Senarai bilangan ahli rumah dalam bancian (sebelum diisih):
"); document.write(banciPenduduk); document.write("

Senarai bilangan ahli rumah dalam bancian (selepas diisih secara menaik - Isihan Buih):
"); for (i=0; i<bilNombor-1; i++) { for (j=0; j<bilNombor-i-1; j++) { if (banciPenduduk[j] > banciPenduduk[j+1]) 2 { sementara = banciPenduduk[j]; banciPenduduk[j] = banciPenduduk[j+1]; banciPenduduk[j+1] = sementara; } } } document.write(banciPenduduk); </script> </body> </html></pre>

Contoh output



Penerangan:

- 1 Menerima satu senarai bilangan ahli rumah yang berjenis nombor.
- 2 Mengisih bilangan ahli rumah tersebut secara menaik menggunakan kaedah isihan buih.

CONTOH 18

Contoh 18 menunjukkan senarai nama pesakit berjenis *String* disimpan dalam pemboleh ubah `namaPesakit` dan seterusnya diisih secara menaik mengikut susunan abjad menggunakan kaedah isihan buih. Atur cara *JavaScript* dan contoh output ditunjukkan pada Jadual 3.1.4.

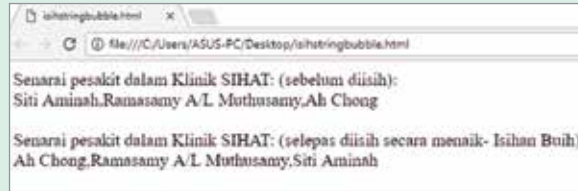
Jadual 3.1.4 Contoh atur cara *JavaScript* dan output isihan buih (*bubble sort*) (Jenis data *String*)

Jenis data	Contoh atur cara penskripan bagi Isihan Buih (<i>Bubble Sort</i>)
<i>String</i>	<pre><html> <body> <script> var namaPesakit = ["Siti Aminah", "Ramasamy A/L Muthusamy", "Ah Chong"]; 1 var bilPesakit = namaPesakit.length, i, j; var sementara = namaPesakit[0]; document.write("Senarai pesakit dalam Klinik SIHAT: (sebelum diisih):
"); document.write(namaPesakit); document.write("

Senarai pesakit dalam Klinik SIHAT: (selepas diisih secara menaik- Isihan Buih):
"); for (i=0; i<bilPesakit-1; i++) { for (j=0; j<bilPesakit-i-1;j++) { if (namaPesakit[j] > namaPesakit[j+1]) { sementara=namaPesakit[j]; namaPesakit[j] = namaPesakit[j+1]; namaPesakit[j+1] = sementara; } } } document.write(namaPesakit); </script> </body> </html></pre> <p>2</p>



Contoh output



Penerangan:

- 1 Menerima satu senarai nama yang berjenis *String*.
- 2 Mengisih nama-nama secara turutan menaik mengikut abjad menggunakan kaedah isihan buih.

CONTOH 19

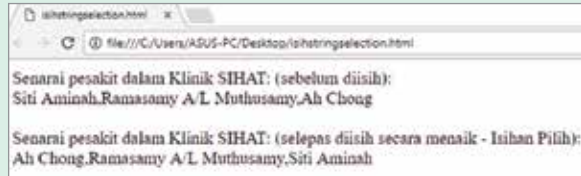
Contoh 19 menunjukkan senarai nama pesakit berjenis *String* disimpan dalam pemboleh ubah `namaPesakit` dan seterusnya diisih secara menaik mengikut susunan abjad menggunakan kaedah isihan pilih. Atur cara *JavaScript* dan contoh output ditunjukkan pada Jadual 3.1.5.

Jadual 3.1.5 Contoh atur cara *JavaScript* dan output isihan pilih (*selection sort*) (Jenis data *String*)

Jenis data	Contoh atur cara penskripan bagi isihan pilih (<i>Selection Sort</i>)
<i>String</i>	<pre><html> <body> <script> var namaPesakit=["Siti Aminah", "Ramasamy A/L Muthusamy", "Ah Chong"]; 1 var i,j,min,sementara; var bilPesakit = namaPesakit.length; document.write("Senarai pesakit dalam Klinik SIHAT: (sebelum diisih):
"); document.write(namaPesakit); document.write("

Senarai pesakit dalam Klinik SIHAT: (selepas diisih secara menaik - Isihan Pilih):
"); for (i = 0; i < bilPesakit - 1; i++) { min = i; for(j = i+1; j < bilPesakit; j++) { if (namaPesakit[j] < namaPesakit[min]) { min = j; } } if (min != i){ sementara = namaPesakit[i]; namaPesakit[i] = namaPesakit[min]; namaPesakit[min] = sementara; } } document.write(namaPesakit); </script> </body> </html></pre>

Contoh output



Penerangan:

- 1 Menerima senarai nama yang berjenis *String*.
- 2 Mengisih nama-nama secara turutan menaik mengikut abjad menggunakan kaedah isihan pilih.

CONTOH 20

Contoh 20 menunjukkan senarai nama pesakit berjenis *String* disimpan dalam pemboleh ubah `namaPesakit`. Seterusnya, proses mencari pesakit bernama "Ramasamy A/L Muthusamy" dilaksanakan menggunakan kaedah carian perduaan. Atur cara *JavaScript* dan contoh output ditunjukkan pada Jadual 3.1.6.

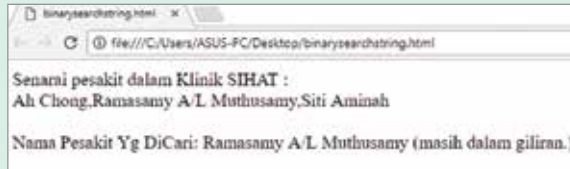
Jadual 3.1.6 Contoh atur cara *JavaScript* dan output carian perduaan (*binary search*) (Jenis data *String*)

Jenis data	Contoh atur cara penskripan carian perduaan (<i>Binary Search</i>)
<i>String</i>	<pre><html> <body> <script> var namaPesakit=["Ah Chong", "Ramasamy A/L Muthusamy", "Siti Aminah"]; 1 var namaPesakitCarian="Ramasamy A/L Muthusamy"; var indeksAwal = 0, indeksAkhir = namaPesakit.length - 1, indeksTengah = Math.floor((indeksAkhir + indeksAwal)/2); document.write("Senarai pesakit dalam Klinik SIHAT :
"); document.write(namaPesakit); document.write("

Nama Pesakit Yg DiCari: ", namaPesakitCarian); while(namaPesakit[indeksTengah] != namaPesakitCarian && indeksAwal < indeksAkhir) { if (namaPesakit[indeksTengah] > namaPesakitCarian) { indeksAkhir = indeksTengah - 1; } else { if (namaPesakit[indeksTengah] < namaPesakit) { indeksAwal = indeksTengah + 1; } } indeksTengah = Math.floor((indeksAkhir + indeksAwal)/2); } if (namaPesakit[indeksTengah]==namaPesakitCarian) { document.write(" (masih dalam giliran.)"); 3 } </script> </body> </html></pre>



Contoh output



Penerangan:

- 1 Mempunyai senarai nama pesakit yang berjenis *String*.
- 2 Mencari pesakit bernama "Ramasamy A/L Muthusamy" menggunakan carian perduaan.
- 3 Mesej "masih dalam giliran." akan dipaparkan jika pesakit masih dalam senarai pesakit.

CONTOH 21

Contoh 21 menunjukkan nombor-nombor iaitu 5.5, -1.3, dan 12.95 berjenis *double* diletakkan ke dalam *Queue* menggunakan fungsi `unshift()`. Seterusnya, fungsi `pop()` digunakan untuk mengeluarkan nombor dalam *Queue* tersebut, iaitu nombor 5.5 (nombor) pertama diletakkan. Atur cara *JavaScript* dan contoh output ditunjukkan pada Jadual 3.1.7.

Jadual 3.1.7 Contoh atur cara *JavaScript* dan output dalam pengumpulan nilai dalam *Queue* (Jenis data nombor)

Jenis data	Contoh atur cara penskripan bagi pengumpulan nilai dalam struktur <i>queue</i>
Nombor	<pre> <html> <body> <script> var queueNo = [], noBaru, noKeluar; noBaru = 5.5; queueNo.unshift(noBaru); document.write("
Nombor Baru Masuk: ", noBaru); noBaru = -1.3; queueNo.unshift(noBaru); document.write("
Nombor Pelajar Masuk: ", noBaru); noBaru = 12.95; queueNo.unshift(noBaru); document.write("
Nombor Pelajar Masuk: ", noBaru); document.write("

Senarai Nombor Pelajar Terkini Dalam Giliran (Queue):
-----"); document.write("
", queueNo); noKeluar = queueNo.pop(); 2 document.write("

Nombor Pelajar Dikeluarkan: ", noKeluar); document.write("

Senarai Nombor Terkini Dalam Giliran (Queue):
-----"); document.write("
", queueNo); </script> </body> </html> </pre>

Contoh output

```

D:\contoh21.html x
file:///C:/Users/ASUS-PC/Desktop/Aturcara_3.1.3

Nombor Baru Masuk: 5.5
Nombor Masuk: -1.3
Nombor Masuk: 12.95

Senarai Nombor Terkini Dalam Giliran (Queue):
-----
12.95,-1.3,5.5

Nombor Dikeluarkan: 5.5

Senarai Nombor Terkini Dalam Giliran (Queue):
-----
12.95,-1.3
    
```

Penerangan:

- 1 Meletakkan senarai nombor, iaitu 5.5, -1.3 dan 12.95 ke dalam *Queue* menggunakan arahan `unshift()`.
- 2 Mengeluarkan nombor terawal yang dimasukkan ke dalam *Queue* menggunakan arahan `pop()`.

CONTOH 22

Contoh 22 menunjukkan nama-nama pesakit seperti "Siti Maimunah", "Ramasamy A/L Muthusamy" dan "Ah Chong" diletakkan ke dalam *Queue* dengan menggunakan fungsi `unshift()`. Seterusnya, fungsi `pop()` digunakan untuk mengeluarkan nama dalam *Queue* tersebut, iaitu "Siti Maimunah" (nama pertama) yang diletakkan. Atur cara *JavaScript* dan contoh output ditunjukkan pada Jadual 3.1.8.

Jadual 3.1.8 Contoh atur cara *JavaScript* dan output bagi pengumpulan nilai dalam *Queue* (Jenis data *String*)

Jenis data	Contoh atur cara penskripan bagi pengumpulan nilai-nilai dalam struktur <i>queue</i>
<i>String</i>	<pre> <html> <body> <script> var queueNama = [], namaBaru, namaKeluar; namaBaru = "Siti Maimunah"; queueNama.unshift(namaBaru); document.write("
Nama Pesakit Baru Masuk: ", namaBaru); namaBaru = "Ramasamy A/L Muthusamy"; queueNama.unshift(namaBaru); document.write("
Nama Pesakit Masuk: ", namaBaru); namaBaru = "Ah Chong"; queueNama.unshift(namaBaru); document.write("
Nama Pesakit Masuk: ", namaBaru); document.write("

Senarai Nama Pesakit Terkini Dalam Giliran (Queue):
-----"); document.write("
", queueNama); namaKeluar = queueNama.pop(); 2 document.write("

Nama Pesakit Dikeluarkan: ", namaKeluar); document.write("

Senarai Nama Pesakit Terkini Dalam Giliran (Queue):
-----"); document.write("
", queueNama); </script> </body> </html> </pre> <p>1</p>

Contoh output

```
nama01.java
C:\Users\Ariana\Documents\nama01.java
Nama Pesakit Baru Masuk: Siti Maimunah
Nama Pesakit Masuk: Ramasamy A/L Muthusamy
Nama Pesakit Masuk: Ah Chong
Senarai Nama Pesakit Terkini Dalam Giliran (Queue):
-----
Ah Chong,Ramasamy A/L Muthusamy,Siti Maimunah
Nama Pesakit Dikeluarkan: Siti Maimunah
Senarai Nama Pesakit Terkini Dalam Giliran (Queue):
-----
Ah Chong,Ramasamy A/L Muthusamy
```

Penerangan:

- 1 Meletakkan senarai berjenis *String* ke dalam *Queue* menggunakan arahan `unshift ()`.
- 2 Mengeluarkan nama pertama dimasukkan ke dalam *Queue* menggunakan arahan `pop ()`.

AKTIVITI

8

Kerja Individu

Carta Alir *Binary Search*

Objektif: Menunjukkan langkah-langkah untuk mencari suatu nilai daripada satu senarai nilai dengan menggunakan kaedah carian perduaan (*binary search*).

Terdapat satu senarai barangan runcit yang menentukan barangan runcit yang dikenakan *Goods and Services Tax* (GST) (bercukai) dan barangan runcit yang tidak dikenakan GST (tidak bercukai) sepertimana yang disiarkan oleh Jabatan Kastam Diraja Malaysia seperti berikut:

Nama barang	Kadar GST Piawai 6%	Kadar GST Sifar 0%
Abalon	✓	
Agar-agar	✓	
Aneka bebola hasil laut/daging/sayuran	✓	
Asam jawa	✓	
Asam pedas <i>paste</i>	✓	
Ayam		✓
Ayam belanda (<i>Turkey</i>)	✓	

(Dikemas kini pada 2 Januari 2015)

Sumber: Jabatan Kastam Diraja Malaysia

Bahagian 1:

Bina atur cara penskripan klien yang dapat melaksanakan arahan-arahan berikut:

- (a) menyimpan senarai nama barang dalam pemboleh ubah bernama *barang* berjenis *String*.
- (b) mencari nama barang, iaitu "Ayam" dengan menggunakan carian perduaan (*binary search*).
- (c) memaparkan mesej "Kadar GST Sifar 0%" jika nama barang pada (b) dijumpai.

Bahagian 2:

Bina carta alir lengkap bagi atur cara yang telah dibina pada Bahagian 1.



UJI MINDA

Dua puluh empat kumpulan telah layak untuk memasuki Pertandingan Bola Jaring Peringkat Zon Mentakab, Pahang. Sekiranya nama kumpulan tersebut disusun mengikut tatasusunan, berapakah item dalam tatasusunan yang boleh dikenal pasti oleh carian perduaan (*binary search*) untuk mengetahui lokasi tertentu bagi sesuatu pasukan dalam tatasusunan?

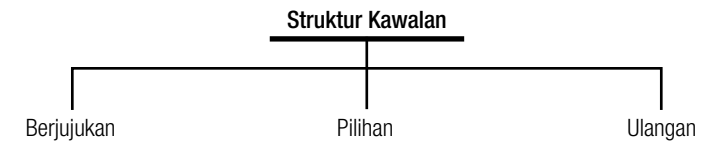
Penggunaan Struktur Kawalan dalam Bahasa Penskripan Klien

Struktur kawalan akan mengawal aliran atur cara dalam satu blok kod atur cara. Pernyataan ini akan menentukan sama ada pernyataan akan dilaksanakan atau tidak, termasuk mengulang pernyataan dalam satu blok kod.

Contoh kawalan:

- Mesti memasukkan kad ATM (*Auto Teller Machine*) dahulu sebelum memasukkan nombor pin untuk pengesahan semasa menggunakan mesin ATM.
- Meneruskan transaksi ATM jika pelanggan bank memasukkan nombor pin yang sah.
- Mengulang masuk nombor pin sebanyak tiga kali jika pin nombor yang dimasukkan adalah tidak sah.

Rajah 3.1.46 menunjukkan struktur kawalan dalam *JavaScript*.



Rajah 3.1.46 Struktur kawalan dalam *JavaScript*

Struktur Kawalan Jujukan

Struktur kawalan jujukan bermaksud arahan atau pernyataan dilaksanakan mengikut tertib sepertimana yang diarahkan dalam suatu kod atur cara tanpa perlu memeriksa sebarang syarat. Jadual 3.1.9 ialah sintaks bagi struktur kawalan jujukan yang ditunjukkan oleh pseudokod dan carta alir:

Jadual 3.1.9 Contoh pseudokod dan carta alir (Struktur kawalan berjjukan)

Pseudokod	Carta Alir
1 Mula	<pre> graph TD M([Mula]) --> I[/Input Nama Anda/] I --> C[/Cetak "Hello" dan Nama Anda/] C --> T([Tamat]) </pre>
2 Input Nama Anda	
3 Cetak "Hello" dan Nama Anda	
4 Tamat	


Penerangan:

- 1 Arahan dilaksanakan satu persatu, iaitu baris demi baris.
- 2 Bermula dengan Input Nama.
- 3 Seterusnya mencetak "Hello" berserta Nama tersebut.

CONTOH 23

Contoh 23 menunjukkan atur cara *JavaScript* yang menggunakan struktur kawalan jujukan yang pernyataannya dilakukan satu persatu untuk menerima satu nama dan memaparkan semula nama tersebut seperti yang ditunjukkan pada Jadual 3.1.10.

Jadual 3.1.10 Contoh atur cara *JavaScript* dan output (Struktur kawalan jujukan)


Atur cara	Contoh Output
<pre><html> <head> <title>Struktur Kawalan Jujukan</title> </head> <body> <script> var namaAnda; namaAnda="Siti Aminah"; 1 document.write("Hello " + namaAnda); 2 </script> </body> </html></pre>	

Penerangan:

- 1 Atur cara menerima nama, iaitu Siti Aminah.
- 2 Seterusnya mencetak nama tersebut semula.

Jadual 3.1.11 pula menunjukkan contoh atur cara *JavaScript* dan output untuk menerima 2 nilai berjenis integer, iaitu tahun semasa dan tahun lahir. Seterusnya, kiraan umur dibuat berdasarkan tahun-tahun tersebut dan umur dipaparkan pada akhir atur cara.

Jadual 3.1.11 Contoh atur cara *JavaScript* dan output (Struktur kawalan jujukan)

Atur cara	Contoh Output
<pre><html> <head> <title>Struktur Kawalan Jujukan</title> </head> <body> <script> var tahunSemasa, tahunLahir, umur; tahunSemasa = 2017; 1 tahunLahir = 1973; 2 umur = tahunSemasa - tahunLahir; 3 document.write("Umur Anda: " + umur); 4 </script> </body> </html></pre>	

Penerangan:

- 1 Atur cara menerima tahun semasa, iaitu 2017.
- 2 Seterusnya atur cara menerima tahun lahir, iaitu 1973.
- 3 Atur cara mengira umur dengan melaksanakan operasi tolak.
- 4 Akhirnya, atur cara memaparkan nilai umur tersebut.

Tahukah Anda

JavaScript ialah bahasa pengaturcaraan yang dibangunkan oleh Brendan Eich. *JavaScript* boleh memproses nombor yang mengubah suai kandungan sesuatu atur cara. Ketika dalam pembangunannya, *JavaScript* dikenali sebagai *Mocha* dan kemudahan *Livescript*, seterusnya *Livescript*. Skrip sintaksnya hampir menyerupai *Java*, maka diberikan namanya sebagai *JavaScript*.

Sumber:

https://www.w3.org/community/webbed/wiki/A_Short_History_of_JavaScript

Antara pernyataan yang terdapat dalam struktur kawalan jujukan ialah input data, proses seperti pengiraan (operasi aritmetik), dan paparan atau cetakan output. Input data boleh diterima daripada pengguna (secara interaktif) atau ditetapkan dalam atur cara.

Contoh input data (secara interaktif)

- `namaAnda = prompt("Masukkan nama anda: ");`

Contoh input data (ditetapkan dalam atur cara)

- `tahunSemasa=2017;`

Contoh operasi aritmetik

- `umur=tahunSemasa-tahunLahir;`
- `diskaun=(50/100)*100;`

Jadual 3.1.12 menunjukkan operator aritmetik yang boleh digunakan dalam *JavaScript*.

Jadual 3.1.12 Operator aritmetik yang boleh digunakan dalam *JavaScript*

Operasi	Simbol operator aritmetik
Campur	+
Tolak	-
Darab	*
Bahagi	/
Modulus (baki kepada hasil bahagi)	%

Contoh paparan:

- `window.alert("Hello " + namaAnda);`
- `document.write("Umur Anda: " + umur);`

Struktur Kawalan Pilihan

Struktur kawalan pilihan memerlukan pernyataan bersyarat yang akan melaksanakan satu pernyataan atau satu blok kod yang ditutupi dengan "{" dan "}" dalam *JavaScript*. Pernyataan bersyarat dilaksanakan dengan menggunakan pernyataan yang dipanggil ungkapan logik yang terdiri daripada operator hubungan.

Jadual 3.1.13 menunjukkan operator hubungan yang boleh terdapat dalam ungkapan logik. Ungkapan logik akan menghasilkan nilai *Boolean*, iaitu *true* atau *false* untuk menentukan sama ada satu pernyataan atau satu blok kod akan dilaksanakan atau tidak. Dua pernyataan *JavaScript* yang digunakan untuk struktur kawalan pilihan.

- `if...`
- `if...else...`

Jadual 3.1.13 Operator hubungan yang terdapat dalam ungkapan logik

Syarat	Simbol Operator Hubungan
Sama dengan	==
Tidak sama dengan	!=
Lebih besar daripada	>
Lebih kecil daripada	<
Lebih besar atau sama dengan	>=
Lebih kecil atau sama dengan	<=

Pernyataan *if*

Jika syarat dalam *if* adalah `true`, pernyataan akan dilaksanakan. Jika sebaliknya, pernyataan tidak akan dilaksanakan dan dilangkau. Jadual 3.1.14 ialah sintaks struktur kawalan pilihan yang ditunjukkan oleh pseudokod dan carta alir.

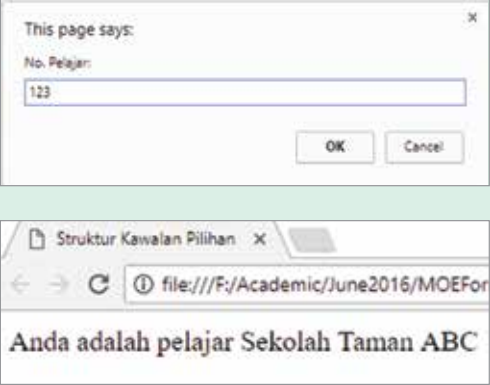
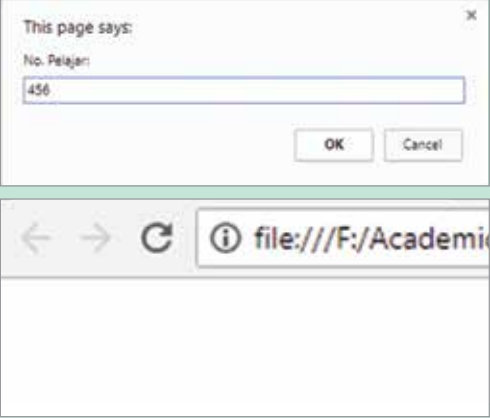
Jadual 3.1.14 Pseudokod dan carta alir (*if*)

Pseudokod	Carta Alir
<ol style="list-style-type: none"> 1 Mula 2 Pernyataan 1 3 Jika Syarat BENAR 3.1 Pernyataan 2 4 Tamat 	<pre> graph TD Mula([Mula]) --> P1[Pernyataan 1] P1 --> D{Syarat BENAR?} D -- BENAR --> P2[Pernyataan 2] D -- PALSU --> T([Tamat]) P2 --> T </pre>

CONTOH 24

Contoh 24 menunjukkan penggunaan `if` dalam atur cara *JavaScript* iaitu nombor pelajar yang diterima diuji sekiranya nombor pelajar ialah bernilai 123. Mesej “Anda adalah pelajar Sekolah Taman ABC” akan dipaparkan jika nombor pelajar ialah 123, dan tiada sebarang mesej dipaparkan jika sebaliknya (nombor pelajar selain 123). Atur cara *JavaScript* dan output ditunjukkan pada Jadual 3.1.15.

Jadual 3.1.15 Contoh atur cara *JavaScript* dan output (*if*)

Atur cara	Contoh Output
<pre><html> <head> <title>Struktur Kawalan Pilihan </title> </head> <body> <script> var noPelajar; 1 noPelajar = prompt("No. Pelajar: "); 2 if (noPelajar == 123) 3 3 document.write("Anda adalah pelajar Sekolah Taman ABC"); </script> </body> </html></pre>	<p>(memenuhi syarat <code>if</code>)</p>  <p>(tidak memenuhi syarat <code>if</code>)</p> 

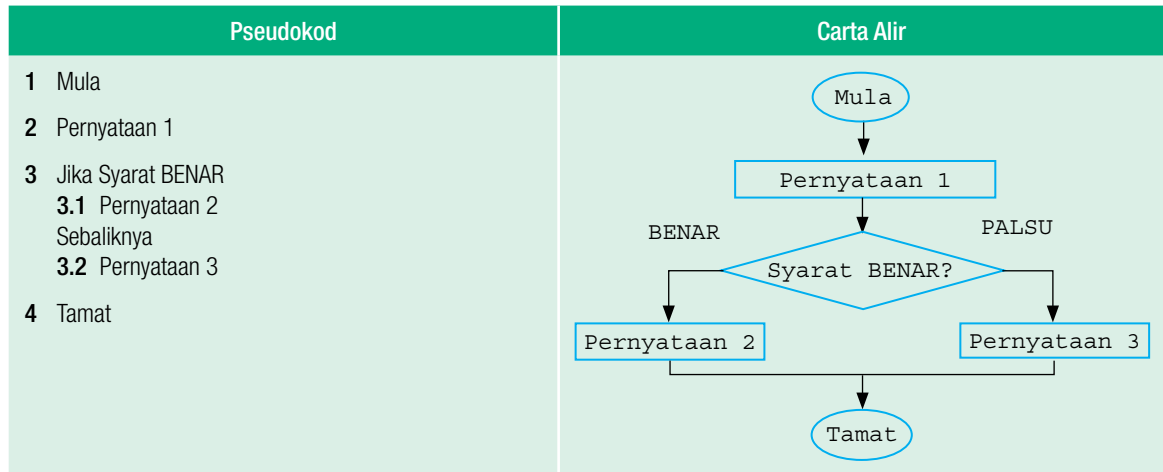
Penerangan:

- 1 Atur cara menerima nombor pelajar.
- 2 Seterusnya, menentukan jika nombor pelajar bersamaan dengan 123.
- 3 Jika nombor pelajar bersamaan dengan 123, “Anda adalah pelajar Sekolah Taman ABC” akan dipaparkan (memenuhi syarat `if`). Jika nombor pelajar tidak bersamaan dengan 123, tiada arahan dilakukan (tidak memenuhi syarat `if`).

Pernyataan *if...else*

Banyak syarat dalam pernyataan *if* boleh didapati dan boleh digabungkan bersama-sama dengan *else* dan *else...if*. Jika syarat tidak ditepati (jika syarat adalah *false*), bahagian *else* atau *else...if* akan dirujuk untuk mengenal pasti blok kod yang dilaksanakan. Jadual 3.1.16 ialah sintaks struktur kawalan *if...else* yang ditunjukkan oleh pseudokod dan carta alir.

Jadual 3.1.16 Pseudokod dan carta alir (*if...else*)



CONTOH 25

Contoh 25 menunjukkan penggunaan *if...else* dalam atur cara *JavaScript*, iaitu nombor pelajar yang diterima akan diuji sama ada nombor pelajar bernilai 123 atau tidak. Mesej "Anda adalah pelajar Sekolah Taman ABC" dipaparkan jika nombor pelajar ialah 123 dan mesej "Anda BUKAN pelajar Sekolah Taman ABC" dipaparkan jika nombor pelajar bukan 123. Atur cara *JavaScript* dan output ditunjukkan pada Jadual 3.1.17.

Jadual 3.1.17 Contoh atur cara *JavaScript* dan output (*if...else*)

Atur cara	Contoh Output
<pre><html> <head> <title>Struktur Kawalan Pilihan </title> </head> <body> <script> var noPelajar; noPelajar = prompt("No. Pelajar: "); 1 if (noPelajar == 123) 2 3 document.write("Anda adalah pelajar Sekolah Taman ABC"); else 4 document.write("Anda BUKAN pelajar Sekolah Taman ABC"); </script> </body> </html></pre>	<p>(memenuhi syarat <i>if</i>)</p>

Atur cara	Contoh Output
	<p>(tidak memenuhi syarat <code>if</code>)</p> 

Penerangan:

- 1 Atur cara menerima nombor pelajar.
- 2 Menentukan sama ada nombor pelajar adalah 123.
- 3 Jika nombor pelajar adalah 123, “Anda adalah pelajar Sekolah Taman ABC” dipaparkan (memenuhi syarat `if`).
- 4 Sebaliknya, “Anda bukan pelajar Sekolah Taman ABC” akan dipaparkan (tidak memenuhi syarat `if`).

Dalam suatu pernyataan `if...else`, dua atau lebih syarat *Boolean* boleh diuji serentak dengan menggabungkan syarat *Boolean* melalui operator logikal. Operator-operator logikal yang lazim digunakan ialah **AND**, **OR** dan **NOT**. Jadual 3.1.18 menunjukkan operator logikal yang boleh terdapat dalam ungkapan logik.

Jadual 3.1.18 Operator logikal dalam ungkapan logik

Syarat	Simbol Operator Logikal
AND	&&
OR	
NOT	!

Operator Logikal AND

Operator logik AND digunakan apabila terdapat dua atau lebih syarat *Boolean* perlu digabungkan dan semua syarat perlu benar sebelum melaksanakan pernyataan seterusnya.

CONTOH 26

Contoh 26 menunjukkan markah peperiksaan mestilah antara 0 hingga 100. Oleh sebab itu, untuk menentukan sama ada markah yang dimasukkan adalah antara 0 hingga 100, operator logikal “&&” perlu digunakan. Contoh: `if (markahPeperiksaan >= 0 && markahPeperiksaan <=100)`. Atur cara *JavaScript* dan output ditunjukkan pada Jadual 3.1.19.



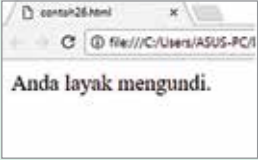


Jadual 3.1.19 Contoh atur cara *JavaScript* dan output (AND)


Atur cara

```
<html>
  <body>
    <script>
      var markahPeperiksaan = 67, gred;
      if (markahPeperiksaan >= 0 && markahPeperiksaan <= 39)
      {
        gred = "Gred G";
      }
      else
      {
        if (markahPeperiksaan >= 40 && markahPeperiksaan <= 44)
        {
          gred = "Gred E";
        }
        else
        {
          if (markahPeperiksaan >= 45 && markahPeperiksaan <= 49)
          {
            gred = "Gred D";
          }
          else
          {
            if (markahPeperiksaan >= 50 && markahPeperiksaan <= 59)
            {
              gred = "Gred C";
            }
            else
            {
              if (markahPeperiksaan >= 60 && markahPeperiksaan <= 64)
              {
                gred = "Gred C+";
              }
              else
              {
                if (markahPeperiksaan >= 65 && markahPeperiksaan <= 69)
                {
                  gred = "Gred B";
                }
                else
                {
                  if (markahPeperiksaan >= 70 && markahPeperiksaan <= 74)
                  {
                    gred = "Gred B+";
                  }
                  else
                  {
                    if (markahPeperiksaan >= 75 && markahPeperiksaan <= 79)
                    {
                      gred = "Gred A-";
                    }
                    else
                    {
                      if (markahPeperiksaan >= 80 && markahPeperiksaan <= 89)
                      {
                        gred = "Gred A";
                      }
                    }
                  }
                }
              }
            }
          }
        }
      }
    }
  }

```


Jadual 3.1.20 Contoh atur cara JavaScript dan output (OR)

Atur cara	Contoh Output
<pre>html> <body> <script> 1 var umur = 25, statusDaftar = true; 2 if (umur >= 21 statusDaftar == true) { document.write("Anda layak mengundi."); 3 } 4 else { document.write("Anda tidak layak mengundi."); } </script> </body> </html></pre>	<p>(memenuhi kedua-dua syarat if) Umur: 25 StatusDaftar: true</p> 
<pre><html> <body> <script> var umur = 25, statusDaftar = false; if (umur >= 21 statusDaftar == true) { document.write("Anda layak mengundi."); } else { document.write("Anda tidak layak mengundi."); } </script> </body> </html></pre>	<p>(memenuhi salah satu syarat if) Umur: 25 StatusDaftar: false</p> 
<pre><html> <body> <script> var umur = 19, statusDaftar = true; if (umur >= 21 statusDaftar == true) { document.write("Anda layak mengundi."); } else { document.write("Anda tidak layak mengundi."); } </script> </body> </html></pre>	<p>(memenuhi salah satu syarat if) Umur: 19 StatusDaftar: true</p> 

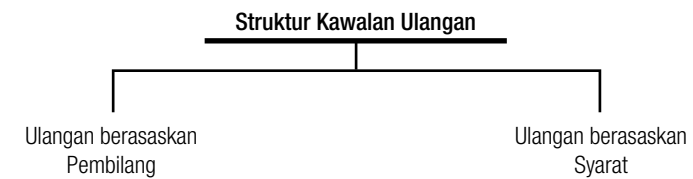
Atur cara	Contoh Output
<pre> <html> <body> <script> var umur = 19, statusDaftar = false; if (umur >= 21 statusDaftar == true) { document.write("Anda layak mengundi."); } else { document.write("Anda tidak layak mengundi."); } </script> </body> </html> </pre>	<p>(tidak memenuhi kedua-dua syarat if) Umur: 19 StatusDaftar: false</p> 

Penerangan:

- 1 Menetapkan nilai "umur = 25" dan "statusDaftar = true".
- 2 Menyemak nilai umur dan statusDaftar menggunakan arahan if. Menyemak sama ada memenuhi salah satu syarat iaitu umur >= 25 atau statusDaftar = true.
- 3 Jika salah satu syarat dipenuhi, mesej "Anda layak mengundi" akan dipaparkan.
- 4 Sebaliknya, jika kedua-dua syarat tidak dipenuhi, mesej "Anda tidak layak mengundi" akan dipaparkan.

Struktur Kawalan Ulangan

Struktur kawalan ulangan memerlukan pernyataan bersyarat yang akan melaksanakan satu pernyataan atau satu blok kod berulang kali. Struktur kawalan ulangan melaksanakan satu pernyataan yang diberi selagi syaratnya adalah *true*. Rajah 3.1.47 menunjukkan dua jenis struktur kawalan ulangan.

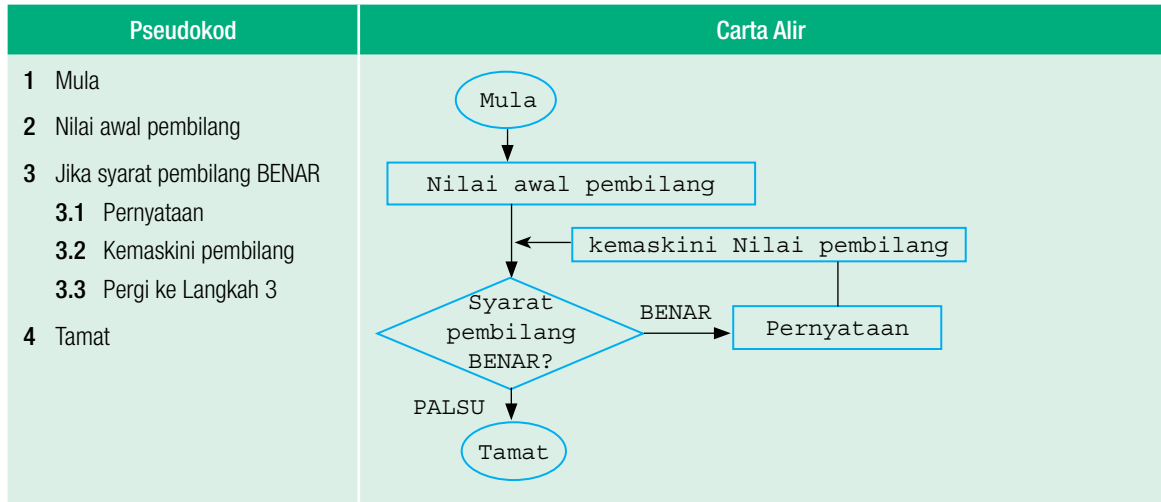


Rajah 3.1.47 Jenis struktur kawalan ulangan

Ulangan berasaskan pembilang

Ulangan berasaskan pembilang menggunakan pernyataan *for...* dalam *JavaScript*. Ulangan berasaskan pembilang akan menggunakan satu pemboleh ubah untuk membuat ulangan kepada pernyataan dan syaratnya bergantung kepada pembilang. Jadual 3.1.21 adalah sintaks bagi struktur ulangan berasaskan pembilang yang ditunjukkan oleh pseudokod dan carta alir.

Jadual 3.1.21 Pseudokod dan carta alir (ulangan berasaskan pembilang)



Untuk melaksanakan struktur kawalan berasaskan pembilang, pernyataan `for` akan digunakan. `for` mempunyai tiga komponen, iaitu:

- Nilai awal – Nilai permulaan untuk membilang
- Nilai tamat – Nilai untuk menamatkan proses membilang yang menentukan ulangan tamat atau diteruskan
- Nilai kemas kini – Proses untuk mengemas kini nilai pembilang

CONTOH 28

Contoh 28 menunjukkan blok kod akan berterusan mencetak “Salam Sejahtera!” selagi pembilang kurang atau sama dengan 3 (atau sebanyak 3 kali). Atur cara *JavaScript* dan output ditunjukkan pada Jadual 3.1.22.

Jadual 3.1.22 Contoh atur cara *JavaScript* dan output (ulangan berasaskan pembilang)

Atur cara	Contoh Output
<pre> <html> <body> <script> var pembilang; 1 for(pembilang=1; pembilang<=3; pembilang++) { document.write("Salam Sejahtera!
"); 5 } </script> </body> </html> </pre>	
<div style="border: 2px solid red; padding: 5px; margin-top: 10px;"> <p>2 Nilai awal pembilang. Contoh: pembilang di tetapkan sebagai 1</p> </div>	<div style="border: 2px solid red; padding: 5px; margin-top: 10px;"> <p>3 Nilai tamat pembilang. Ini ialah syarat bagi ulangan. Contoh: Ulangan akan diteruskan apabila pembilang kurang atau sama dengan 3. Ulangan akan berhenti apabila pembilang melebihi 3.</p> </div>
	<div style="border: 2px solid red; padding: 5px; margin-top: 10px;"> <p>4 Nilai kemas kini pembilang. Contoh: pembilang ditambah 1 (pembilang++)</p> </div>

Penerangan:

- 1 `for` digunakan dalam *JavaScript* untuk melaksanakan ulangan.
- 2 Terdapat nilai pembilang yang diberi nilai awal 1.
- 3 Terdapat nilai tamat pembilang, iaitu 3 untuk mengira ulangan sebanyak 3 kali (bermula 1 hingga 3).
- 4 pembilang `++` adalah untuk kemas kini nilai pembilang dengan menambah satu kepada nilai pembilang (sehingga syarat pembilang `<=3` tidak dipenuhi).
- 5 Mesej "Salam Sejahtera!" akan dipaparkan sebanyak 3 kali.


Kawalan ulangan ini juga mengenali kata kunci `break` dan `continue`. Kata kunci `break` akan menyebabkan penamatan atau pemberhentian suatu ulangan. Penamatan atau pemberhentian berlaku di mana-mana sahaja di dalam blok kod tersebut.

Kata kunci `continue` pula akan terus melaksanakan dan menghabiskan ulangan blok kod atau pernyataan dan memeriksa syarat sama ada syarat tersebut `true`. Jika `true`, ulangan akan disambung dan diteruskan sekali lagi.

CONTOH 29

Contoh 29 menunjukkan penggunaan `break`, iaitu atur cara akan berhenti apabila pembilang bernilai 2 dan menamatkan ulangan walaupun nilai tamat pembilang ialah 3. Atur cara *JavaScript* dan output ditunjukkan dalam Jadual 3.1.23.

Jadual 3.1.23 Contoh atur cara *JavaScript* dan output (`break`)

Atur cara	Contoh Output
<pre><html> <body> <script> var pembilang; for(pembilang=1; pembilang<=3; pembilang++) { document.write("Salam Sejahtera!
"); 1 if (pembilang==2) break; 2 } </script> </body> </html></pre>	


Penerangan:

- 1 Atur cara akan berterusan mencetak "Salam Sejahtera!" selagi pembilang kurang atau sama dengan 3.
- 2 Jika pembilang bersamaan dengan 2, blok kod ulangan akan dihentikan dan ditamatkan oleh pernyataan `break`.

CONTOH 30

Contoh 30 menunjukkan penggunaan `continue`, iaitu atur cara akan diteruskan apabila pembilang bernilai 2 dan meneruskan ulangan. Atur cara *JavaScript* dan output ditunjukkan dalam Jadual 3.1.24.

Jadual 3.1.24 Contoh atur cara *JavaScript* dan output (continue)

Atur cara	Contoh Output
<pre> <html> <body> <script> var pembilang; for(pembilang=1; pembilang<=3; pembilang++) { document.write("Salam Sejahtera!
"); 1 if (pembilang==2) 2 continue; } </script> </body> </html> </pre>	

Penerangan:

- 1 Atur cara akan berterusan mencetak "Salam Sejahtera!" selagi pembilang kurang atau sama dengan 3.
- 2 Jika pembilang bersamaan dengan 2, blok kod ulangan akan tetap diteruskan oleh pernyataan `continue`.

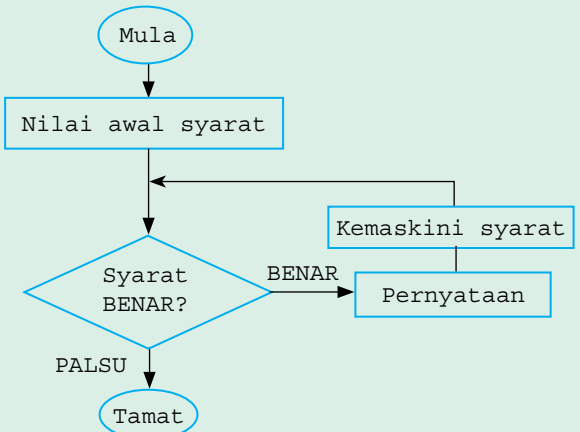
Ulangan berasaskan syarat

Ulangan berasaskan syarat dengan menggunakan dua jenis pernyataan dalam *JavaScript*, iaitu:

- `while...`
- `do...while`

Pernyataan `while` melaksanakan suatu pernyataan sekali lagi selagi syarat adalah `true` manakala pernyataan `do...while` akan melaksanakan pernyataan atau blok kod sekurang-kurangnya sekali dan akan meneruskan pernyataan selagi syarat adalah `true`. Jadual 3.1.25 dan Jadual 3.1.26 menunjukkan struktur ulangan berasaskan syarat.

Jadual 3.1.25 Struktur ulangan berasaskan syarat (`while...`)

Jenis Ulangan	Pseudokod	Carta Alir
<code>while...</code>	<ol style="list-style-type: none"> 1 Mula 2 Nilai awal syarat 3 Jika Syarat BENAR <ol style="list-style-type: none"> 3.1 Pernyataan 3.2 Kemas kini syarat 3.3 Pergi ke Langkah 3 4 Tamat 	

Jadual 3.1.26 Struktur ulangan berkawalan syarat (do... while)

Jenis Ulangan	Pseudokod	Carta Alir
do...while...	<ol style="list-style-type: none"> 1 Mula 2 Nilai awal syarat 3 Pernyataan 4 Jika syarat BENAR <ol style="list-style-type: none"> 4.1 Kemaskini syarat 4.2 Pergi ke Langkah 3 5 Tamat 	<pre> graph TD Mula([Mula]) --> Nilai[Nilai awal syarat] Nilai --> Pernyataan[Pernyataan] Pernyataan --> Syarat{Syarat BENAR?} Syarat -- BENAR --> Kemaskini[Kemaskini syarat] Kemaskini --> Pernyataan Syarat -- PALSU --> Tamat([Tamat]) </pre>

CONTOH 31

Contoh 31 menunjukkan "Salam Sejahtera!" akan dipaparkan dahulu sekurang-kurangnya sekali dan syarat akan disemak selepas paparan. Jika syarat dipenuhi paparan akan dilaksanakan semula. Atur cara JavaScript dan output ditunjukkan pada Jadual 3.1.27.

Jadual 3.1.27 Contoh atur cara JavaScript dan output (do...while)

Atur cara	Contoh Output
<pre> <html> <body> <script> var ulang=1; 1 do{ document.write("Salam Sejahtera!
"); 2 ulang++; 3 }while(ulang<=3); 4 </script> </body> </html> </pre>	<p>(syarat dipenuhi) nilai awal bagi pembolehubah ulang ialah 1</p>
<pre> <html> <body> <script> var ulang=4; do{ document.write("Salam Sejahtera!
"); ulang++; }while(ulang<=3); </script> </body> </html> </pre>	<p>(syarat tidak dipenuhi) nilai awal bagi pembolehubah ulang ialah 4</p>



Penerangan:

- 1 Menetapkan syarat `ulang = 1` sebagai nilai awal.
- 2 Mesej “Salam Sejahtera!” akan dicetak.
- 3 Kemas kini syarat (`ulang++`).
- 4 Semak syarat sama ada `ulang <= 3`. Jika syarat dipenuhi, “Salam Sejahtera!” akan dicetak sekali lagi dan langkah 3 hingga 4 akan diulang sehingga syarat tidak lagi dipenuhi.

Tahukah Anda

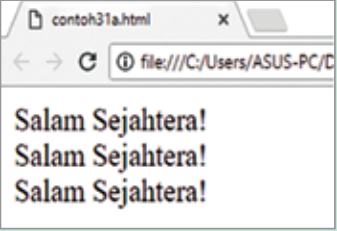
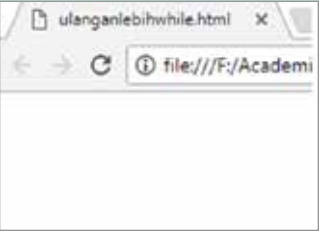


Syarat tamat bagi ulangan boleh juga bergantung kepada nilai input pengguna dan nilai kiraan semasa.

CONTOH 32

Contoh 32 menunjukkan syarat akan disemak dahulu sebelum “Salam Sejahtera!” dipaparkan. Jika syarat dipenuhi, “Salam Sejahtera!” akan dipaparkan dan syarat akan disemak semula. Atur cara *JavaScript* dan output ditunjukkan pada Jadual 3.1.28.

Jadual 3.1.28 Contoh atur cara *JavaScript* dan output (`while`)

Atur cara	
<pre><html> <body> <script> var ulang=1; 1 while (ulang<=3) 2 { document.write("Salam Sejahtera!
"); 3 ulang++; 4 } </script> </body> </html></pre>	
Contoh Output	
(syarat dipenuhi)	(syarat tidak dipenuhi – nilai awal ulang = 4)
	

Penerangan:

- 1 Menetapkan syarat `ulang=1` sebagai nilai awal.
- 2 Syarat `ulang<=3` disemak dahulu.
- 3 Jika syarat dipenuhi, cetak “Salam Sejahtera!”. Jika syarat tidak dipenuhi, iaitu menetapkan nilai `ulang = 4` sebagai nilai awal, “Salam Sejahtera!” tidak akan dipaparkan.
- 4 Kemas kini nilai syarat (`ulang++`). Semak semula syarat sebelum “Salam Sejahtera!” dicetak semula.

Tahukah Anda



`break` dan `continue` juga boleh terdapat dalam ulangan ini. Pernyataan `continue` akan melaksanakan pernyataan penambahan sebelum syarat diperiksa.

Objektif: Menggunakan struktur kawalan (jujukan, pilihan, ulangan) dalam atur cara.

Tulis dan hasilkan satu dokumen HTML yang mempunyai *JavaScript* bagi satu ulangan yang dapat menerima dan membuat ulangan daripada 1 sehingga 10. Setiap ulangan akan memeriksa sama ada nombor yang dirujuk adalah nombor genap atau ganjil dan paparkan satu mesej seperti yang ditunjukkan dalam contoh output di bawah.

Contoh Output:

- 1 ialah nombor ganjil
- 2 ialah nombor genap
- 3 ialah nombor ganjil
- 4 ialah nombor genap
- 5 ialah nombor ganjil
- 6 ialah nombor genap
- 7 ialah nombor ganjil
- 8 ialah nombor genap
- 9 ialah nombor ganjil
- 10 ialah nombor genap

3.1.5 Kelebihan *Standard Library*

Pengenalan *Standard Library*

Standard library dalam pengaturcaraan merupakan koleksi kaedah atau fungsi yang **disediakan** dan **diguna pakai** sewaktu implementasi dalam kod atur cara. *Standard library* perlu ditakrifkan dalam spesifikasi bahasa pengaturcaraan.

Standard library juga merupakan sebahagian daripada arahan dalam atur cara, termasuklah definisi bagi algoritma yang biasa digunakan, struktur data, dan mekanisme bagi input dan output.

Antara *standard library* yang terdapat dalam *JavaScript*:

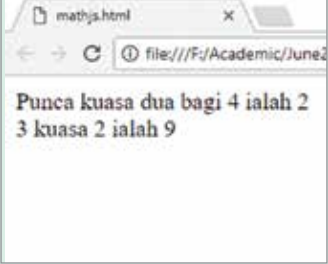
- `math.js` bagi fungsi-fungsi matematik seperti `math.sqrt()` dan `math.pow()`
- `date.js` bagi fungsi-fungsi tarikh seperti `date.now()` dan `date.format()`

Antara kelebihan *Standard Library*:

- 1 Digunakan tanpa mengetahui cara pelaksanaannya. Contoh pada Jadual 3.1.29 panggil `math.pow()` daripada *standard library* `math.js` tanpa perlu mengetahui bagaimana operasi kiraan kuasa (*power*) dilaksanakan.

- 2 Boleh digunakan berulang-ulang kali apabila diperlukan dalam atur cara dengan hanya memanggil fungsinya.
- 3 Dapat mengurangkan masa pembangunan atur cara kerana tidak perlu menulis pernyataan berulang-ulang kali.

Jadual 3.1.29 Contoh atur cara *JavaScript* (`math.sqrt ()` dan `math.pow ()`)

Atur cara	Contoh Output
<pre><html> <head> <script src="math.js"> </script> </head> <body> <script> document.write("Punca kuasa dua bagi 4 ialah ", math.sqrt(4)); document.write("
3 kuasa 2 ialah ", math.pow(3, 2)); </script> </body> </html></pre>	

3.1.6 Penggunaan *Standard Library* dalam Bahasa Penskripan Klien

math.js

`math.js` ialah *standard library* yang kerap diguna pakai dalam *JavaScript* yang terdiri daripada fungsi *built-in* dan pemalar dengan pelbagai jenis data seperti nombor. *Standard library* dalam *JavaScript* berada dalam pakej “.js” dan perlu disimpan dahulu. Seterusnya `math.js` akan dimuatkan dan seterusnya boleh digunakan dalam atur cara.

`math.js` boleh didapati daripada pelbagai pengurus pakej seperti `npm` dan `bower` tetapi boleh juga dimuat turun daripada laman web: <http://mathjs.org/download.html>. Rajah 3.1.48 menunjukkan tangkap layar (*screenshot*) laman web <http://mathjs.org/download.html> yang mengandungi fail `math.js` untuk dimuat turun dan disimpan.



Bower vs NPM
 Sumber: ng-learn.org/2013/11/Bower-vs-npm/



Rajah 3.1.48 math.js

math.js boleh juga digunakan dalam node.js dalam pelayan (*server*) dan dalam pelayar (*browser*) untuk atur cara klien. *Library* perlu dimuatkan dan disetkan sebelum boleh digunakan.

Atribut src pada <script> (Atur cara)

math.js boleh dimasukkan sebagai atribut pada tag <script> dalam atur cara *JavaScript*. Cara-caranya ialah:

- 1 *Download* fail math.js. Rajah 3.1.48 menunjukkan math.js ialah fail *JavaScript* yang boleh dimuat turun daripada satu laman web.
- 2 Gunakan fungsi-fungsi tertakrif dalam math.js. Jadual 3.1.30 menunjukkan fungsi-fungsi dalam math.js.

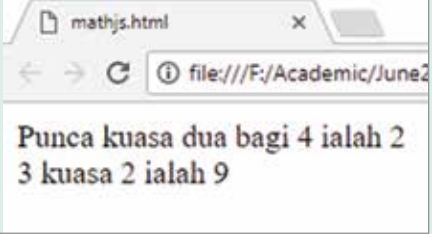
Jadual 3.1.30 Fungsi-fungsi dalam math.js

Fungsi	Penerangan
math.add(x, y)	Menambah dua nombor
math.divide(x, y)	Membahagi dua nombor
math.subtract(x, y)	Menolak dua nombor
math.pow(x, y)	Mengira kuasa kepada nombor
math.sqrt(x)	Mengira punca kuasa bagi nombor
math.cube(x)	Mendarab tiga nombor yang sama (x * x * x)
math.sort(x)	Mengisih unsur-unsur dalam satu matriks

CONTOH 33

Contoh 33 menunjukkan penggunaan `math.sqrt()` dan `math.pow()` daripada fail `math.js` yang dimuat turun. Atur cara *JavaScript* dan output ditunjukkan pada Jadual 3.1.31.

Jadual 3.1.31 Contoh atur cara *JavaScript* (`math.js` daripada sumber muat turun)

Atur cara	Contoh Output
<pre><html> <head> <script src="math.js"> 1 </head> <body> <script> document.write("Punca kuasa dua bagi 4 ialah ", math.sqrt(4)); 2 document.write("
3 kuasa 2 ialah ", math.pow(3, 2)); 3 </script> </body> </html></pre>	

Penerangan:

- 1 Pada `<head>` tetapkan sumber *standard library* yang dirujuk, iaitu `math.js`.
- 2 Sekarang, fungsi-fungsi dalam `math.js` boleh digunakan. Contoh: `math.sqrt(4)`. `math.sqrt(4)` akan mengira punca kuasa dua bagi 4 dan memaparkan hasilnya.
- 3 `math.pow(3, 2)` pula akan melaksanakan operasi 3 kuasa 2 dan memaparkan hasilnya.

Pelayar (Browser)

Pelayar (*browser*) ialah satu program yang mempunyai antara muka bergrafik yang memaparkan fail-fail HTML dan digunakan untuk navigasi WWW (*World Wide Web*).

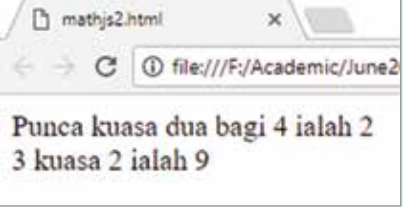
`math.js` juga boleh dimuatkan sebagai fail *JavaScript* biasa dalam pelayar (*browser*). Cara-caranya ialah:

- 1 Dapatkan sumber pautan bagi kandungan `math.js`. Contoh 34 menunjukkan <http://cdnjs.cloudflare.com/ajax/libs/mathjs/3.7.0/math.min.js> ialah pautannya.
- 2 Gunakan fungsi-fungsi tertakrif di dalam `math.js`. Sila rujuk math.js.org/docs/reference/functions.html untuk fungsi lain-lain dalam `math.js`.

CONTOH 34

Contoh 34 menunjukkan penggunaan `math.sqrt()` dan `math.pow()` daripada sumber pautan <http://cdnjs.cloudflare.com/ajax/libs/mathjs/3.7.0/math.min.js>. Atur cara *JavaScript* dan output ditunjukkan dalam Jadual 3.1.32.

Jadual 3.1.32 Contoh atur cara *JavaScript* dan output (`math.js` daripada sumber pautan)

Atur cara	Contoh Output
<pre><html> <head> <script src="http://cdnjs.cloudflare.com/ ajax/libs/mathjs/3.7.0/math.min.js"> 1 </script> </head> <body> <script> document.write("Punca kuasa dua bagi 4 ialah ", math.sqrt(4)); 2 document.write("
3 kuasa 2 ialah ", math.pow(3, 2)); 3 </script> </body> </html></pre>	

Penerangan:

- 1 Pada `<head>` tetapkan sumber yang dirujuk, iaitu <http://cdnjs.cloudflare.com/ajax/libs/mathjs/3.7.0/math.min.js>.
- 2 Sekarang, fungsi-fungsi daripada pautan di atas boleh digunakan. Contoh: `math.sqrt(4)`. Tugas fungsi `math.sqrt(4)` ialah mengira punca kuasa dua bagi 4 dan `document.write()` memaparkan hasilnya.
- 3 `math.pow(3, 2)` pula akan melaksanakan operasi 3 kuasa dua dan memaparkan hasilnya.

date.js

Antara *standard library* yang lain adalah `date.js`.

- 1 *Download* fail `date.js` di mana `date.js` ialah fail *JavaScript* yang mengandungi kandungan fungsi-fungsi untuk operasi matematik.
- 2 Gunakan fungsi-fungsi tertakrif dalam `date.js`. Jadual 3.1.33 menunjukkan fungsi-fungsinya.

Jadual 3.1.33 Fungsi-fungsi dalam `date.js`

Fungsi	Penerangan
<code>Date.today()</code>	Menghasilkan tarikh hari ini
<code>Date.parse('today')</code>	Menukar bentuk objek kepada bentuk objek <i>Date</i>
<code>Date.today().add().days()</code>	Menambah hari kepada hari ini

Sila rujuk <http://www.datejs.com/2007/11/27/getting-started-with-datejs/> untuk lain-lain fungsi dalam `date.js`.

CONTOH 35

Contoh 35 menunjukkan penggunaan `Date.today()` daripada fail `date.js` yang dimuat turun. Atur cara *JavaScript* dan output ditunjukkan pada Jadual 3.1.34.

Jadual 3.1.34 Contoh atur cara *JavaScript* dan output (`date.js` daripada sumber muat turun)

Atur cara	Contoh Output
<pre><html> <head> <script src="date.js"> 1 </script> </head> <body> <script> var hariIni= Date.today(); 2 document.write(hariIni); 3 </script> </body> </html></pre>	

Penerangan:

- 1 Pada `<head>` tetapkan sumber *standard library* yang dirujuk, iaitu "date.js".
- 2 Sekarang, fungsi-fungsi dalam `date.js` boleh digunakan. Contoh: `Date.today()`.
- 3 `Date.today()` akan mengenal pasti butiran tarikh dan masa terkini dan memaparkan hasilnya (mengikut format yang telah ditetapkan).

AKTIVITI 10

Kerja Individu

Menggunakan *Standard Library*

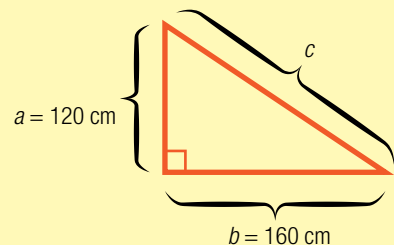
Objektif: Penggunaan *Standard Library* dalam Bahasa Penskripan Klien

Tulis dan hasilkan satu dokumen HTML yang mempunyai *JavaScript* yang dapat mengira dan memaparkan nilai sisi hipotenus bagi maklumat segi tiga seperti yang ditunjukkan pada rajah yang ditunjukkan di bawah.

Diberi, formula teorem *Pythagoras*, bagi mencari sisi hipotenus adalah seperti yang berikut:

$$c^2 = a^2 + b^2$$

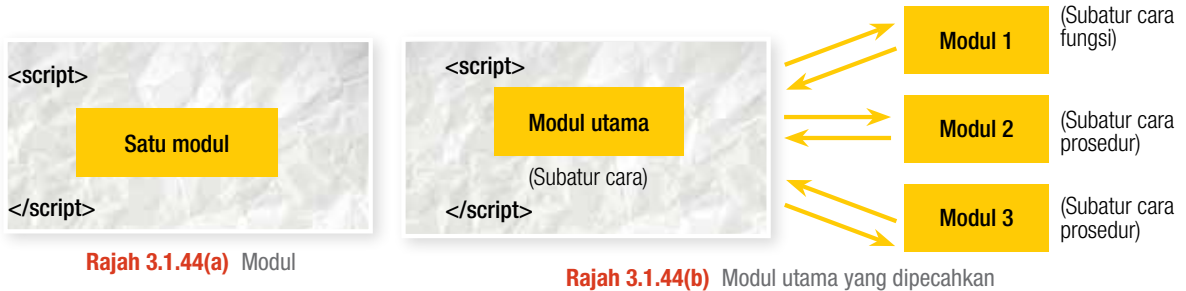
Gunakan *standard library* `math.js` untuk membantu pengiraan sisi hipotenus di sebelah.



3.1.7 Penggunaan *Procedure* dan *Function* untuk Meringkaskan Atur Cara dalam Bahasa Penskripan Klien

Satu atur cara *JavaScript* mempunyai banyak barisan arahan yang dihimpunkan dalam satu blok kod yang dikenali sebagai modul seperti yang ditunjukkan pada Rajah 3.1.44(a).

Barisan arahan tersebut boleh dipecahkan mengikut tugas masing-masing dan dihimpunkan dalam modul yang berasingan yang dipanggil subatur cara seperti yang ditunjukkan dalam Rajah 3.1.44(b). Subatur cara `main()` akan memanggil modul untuk dilaksanakan.



Terdapat dua jenis subatur cara, iaitu:

- Prosedur (*procedure*)
- Fungsi (*function*)

Perbezaan antara Prosedur (*procedure*) dengan Fungsi (*function*) dalam Atur Cara Bahasa Penskripan Klien

Dalam konteks atur cara, prosedur (*procedure*) tidak memulangkan nilai manakala fungsi (*function*) akan memulangkan nilai. Perbandingan dirumuskan seperti dalam Jadual 3.1.35.

Jadual 3.1.35 Perbandingan prosedur dan fungsi

Aspek perbandingan	Prosedur	Fungsi
Memulangkan nilai	Tidak memulangkan nilai	Memulangkan nilai
Contoh definisi	<pre>function toCelcius (fahrenheit) { var celcius; celcius = (5/9)* fahrenheit - 32; document.write ("Suhu adalah" + celcius + " darjah Celcius); }</pre>	<pre>function toCelcius (fahrenheit) { var celcius; celcius = (5/9)* fahrenheit - 32); return celcius; }</pre>
Contoh panggilan	<code>toCelcius (fahrenheit);</code>	<code>celcius = toCelcius (fahrenheit);</code>

Cara Memanggil/Menggunakan *Procedure* dan *Function* untuk Meringkaskan Atur Cara Bahasa Penskripan Klien

Subatur cara akan menggunakan subatur cara lain bertujuan supaya kod pernyataan-pernyataan dalam subatur cara lain akan dilaksanakan. Umumnya, hubungan dua subatur cara ialah pemanggil dan dipanggil. Subatur cara (pemanggil) memanggil **nama** subatur cara (dipanggil) dalam badan subatur cara (pemanggil) seperti yang ditunjukkan dalam Rajah 3.1.45.



Rajah 3.1.45 Subatur cara pemanggil dan dipanggil

CONTOH 36

Contoh 36 menunjukkan cara memanggil subatur cara JumlahNo() berulang-ulang kali dengan nilai yang berbeza-beza diberikan kepada subatur cara. Subatur cara JumlahNo() mengira dan menjumlahkan nombor dalam senarai subatur cara yang sama dan dapat digunakan juga untuk menjumlahkan nombor bagi jenis data yang berbeza. Atur cara dan output ditunjukkan pada Jadual 3.1.36.

Jadual 3.1.36 Contoh atur cara JavaScript dan output (pemanggil subatur cara)

Atur cara	Contoh output
<pre> <html> <body> <script> var SenaraiPertama = [2,4,12,77]; var SenaraiKedua = [333,222,111]; var SenaraiKetiga = [5.1,5.2,5.6,5.0]; var jum1, jum2,jum3; jum1 = JumlahNo (SenaraiPertama); jum2 = JumlahNo (SenaraiKedua); jum3 = JumlahNo (SenaraiKetiga); document.write(jum1 + "
"); document.write(jum2 + "
"); document.write(jum3 + "
"); function JumlahNo (nombor) { var i, bilNo, jumlah= 0; bilNo = nombor.length; for (i=0; i<bilNo; i++) { jumlah = jumlah + nombor[i]; } return jumlah; } </script> </body> </html> </pre>	<pre> 95 666 20.9 </pre>
<p>1</p> <p>6</p> <p>4</p>	<p>Senarai nombor</p> <p>Memanggil Function JumlahNo()</p> <p>Mencetak jumlah</p> <p>Fungsi yang dipanggil iaitu Function JumlahNo()</p> <p>5</p>

Penerangan:

- 1 Terdapat tiga senarai nombor yang berlainan saiz dan jenis data, iaitu `SenaraiPertama`, `SenaraiKedua`, dan `SenaraiKetiga`.
- 2 Atur cara ingin menjumlahkan setiap senarai nombor.
- 3 Atur cara memanggil fungsi `JumlahNo()` dengan menghantar senarai nombor yang berbeza.
- 4 Fungsi `JumlahNo()` dibina untuk menerima senarai nombor dan menjumlahkan nombor-nombor dalam senarai.
- 5 Hasil jumlah dipulangkan kepada subatur cara yang memanggil.
- 6 Hasil jumlah diterima daripada subatur cara memanggil dan disimpan ke dalam pemboleh ubah berbeza, iaitu `jum1`, `jum2` dan `jum3`. Seterusnya setiap jumlah dicetak dengan menggunakan pernyataan `document.write()`.

Parameter

Setiap prosedur (*procedure*) dan fungsi (*function*) mempunyai kurungan. Dalam kurungan terdapat nama parameter yang dipisahkan dengan simbol “,” (*comma*).

function nama_function (parameter1, parameter2, ...)

Parameter ataupun argumen membolehkan prosedur dan fungsi menghantar nilai (pemanggil) dan diterima oleh prosedur dan fungsi yang dipanggil. Ungkapan parameter rasmi (*formal parameter*) digunakan untuk merujuk parameter bagi subatur cara yang dipanggil. Ungkapan parameter sebenar (*actual parameter*) pula digunakan untuk merujuk pemboleh ubah di dalam subatur cara pemanggil. Jadual 3.1.37 menunjukkan contoh atur cara *JavaScript* dan output untuk menghantar nilai parameter kepada subatur cara yang dipanggil.

CONTOH 37

Contoh 37 menunjukkan subatur cara `toCelsius()` dipanggil dan menghantar nilai Fahrenheit. Subatur cara `toCelsius()` menerima nilai Fahrenheit tersebut dan melaksanakan penukaran nilai Fahrenheit kepada Celsius dengan menggunakan formula $(5/9) \times (\text{fahrenheit} - 32)$. Atur cara *JavaScript* dan output ditunjukkan dalam Jadual 3.1.37.

Jadual 3.1.37 Contoh atur cara *JavaScript* dan output (parameter)

Atur cara	
<pre><html> <body> <script> var fahrenheit = 77; document.write("Suhu adalah " + toCelsius (fahrenheit) + " darjah Celsius.");</pre>	

Atur cara

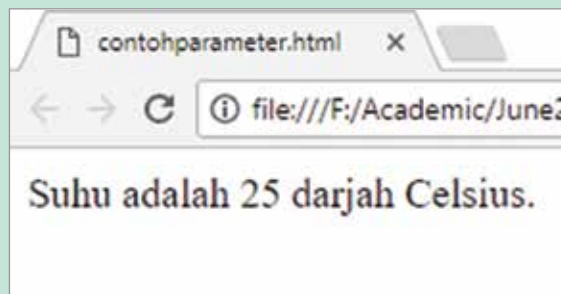
```
function toCelsius (fahrenheit) {  
    return (5/9) * (fahrenheit - 32);  
}  
</script>  
</body>  
</html>
```

Fungsi dipanggil

Parameter rasmi

Menerima nilai fahrenheit 3

Contoh output



Penerangan:

- 1 Menerima nilai suhu fahrenheit, iaitu, 77.
- 2 Mencetak suhu dalam nilai celcius dengan memanggil fungsi `toCelcius (fahrenheit)` kemudian menghantar nilai `fahrenheit=77` kepada fungsi yang dipanggil.
- 3 Fungsi yang dipanggil akan menerima nilai fahrenheit dan menjalankan operasi aritmetik $(5/9) * (fahrenheit - 32)$ untuk menukarnya kepada celcius.

Memulangkan nilai

Fungsi (*function*) akan mengembalikan kawalan kepada pemanggil. Sesetengah fungsi (*function*) bukan sahaja mengembalikan kawalan tetapi juga mengembalikan nilai. Dalam contoh atur cara pada Jadual 3.1.38, atur cara akan mengembalikan nilai kepada fungsi (*function*) pemanggil. Jadual 3.1.38 menunjukkan contoh atur cara *JavaScript* dan output dalam memulangkan nilai daripada subatur cara yang dipanggil kepada yang memanggil.

CONTOH 38

Contoh 38 menunjukkan hasil nilai kiraan $(5/9) * (fahrenheit - 32)$ akan dipulangkan kepada `toCelcius ()` iaitu fungsi yang memanggil dan seterusnya memaparkan nilai tersebut. Atur cara *JavaScript* dan output ditunjukkan dalam Jadual 3.1.38.

Jadual 3.1.38 Contoh atur cara JavaScript dan output (pulang nilai)

Atur cara

```
<html>
<body>

<script>
  var fahrenheit=77;
  document.write("Suhu adalah " + toCelsius(fahrenheit) + " darjah
    Celsius.")

  function toCelsius(fahrenheit)
  {
    return (5/9) * (fahrenheit-32);
  }


</script>

</body>
</html>
```

Diagram illustrating the execution flow of the JavaScript code:

- 1** (highlighted in pink): `var fahrenheit=77;` - Assigns the value 77 to the variable `fahrenheit`.
- 2** (highlighted in pink): `document.write("Suhu adalah " + toCelsius(fahrenheit) + " darjah Celsius.")` - Calls the `toCelsius` function, passing the value of `fahrenheit` as an argument.
- 3** (highlighted in pink): `function toCelsius(fahrenheit)` - The start of the function definition.
- 4** (highlighted in pink): `return (5/9) * (fahrenheit-32);` - The function returns the result of the arithmetic operation.

Contoh output



The screenshot shows a browser window with the URL `file:///F:/Academic/June` and the output text: "Suhu adalah 25 darjah Celsius."

Penerangan:

- 1 Menerima nilai suhu fahrenheit, iaitu 77.
- 2 Mencetak suhu dalam nilai celsius dengan memanggil fungsi `toCelsius(fahrenheit)` kemudian menghantar nilai `fahrenheit=77` kepada fungsi yang dipanggil.
- 3 Fungsi yang dipanggil akan menerima nilai fahrenheit dan menjalankan operasi aritmetik $(5/9) * (fahrenheit-32)$ untuk menukarnya kepada nilai celsius.
- 4 Hasil aritmetik akan dipulangkan kepada fungsi memanggil untuk dicetak.

AKTIVITI 11

Kerja Individu

Penggunaan *Procedure* dan *Function* dalam Bahasa Penskripan Klien

Objektif: Meringkaskan atur cara dengan menggunakan *procedure* dan *function*

Tulis dan hasilkan satu dokumen HTML yang mempunyai JavaScript yang dapat mengira dan memaparkan hasil darab "5 * 5". Gunakan fungsi (*function*) `KiraDarab()` untuk mengira hasil darab "5 * 5" tersebut.

Penggunaan Tatasusunan (*Array*) dalam Bahasa Penskripan Klient

Penggunaan tatasusunan dalam bahasa penskripan klient adalah sama dengan penggunaan tatasusunan *Java* yang telah dipelajari semasa di Tingkatan 4. Penggunaan tatasusunan adalah untuk menyimpan satu senarai nilai dalam satu pemboleh ubah.

- Pengisytiharan tatasusunan (*array*) dalam bahasa penskripan klient

Sintaks:

```
var nama_tatasusunan = [unsur1, unsur2, ...];
```

Contoh pengisytiharan:

```
var no = [ ];
```

- Memberikan nilai awalan tatasusunan (*array*) dalam bahasa penskripan klient
Nilai awal atau dikenali sebagai nilai pertama boleh diletakkan di dalam tatasusunan sebelum nilai-nilai tersebut diubah pada arahan atur cara yang seterusnya.

Contoh:

(a) Memberikan nilai awalan semasa pengisytiharan:

```
var no = [5, -1, 4, 12, 8];
```

(b) Memberikan nilai awalan selepas pengisytiharan:

- Nilai diberikan kepada pemboleh ubah `no` dengan merujuk indeks yang bermula dengan 0.
- Indeks ialah nombor yang mewakili setiap lokasi dalam tatasusunan.
- Indeks 0 merujuk nilai pertama dalam tatasusunan, indeks 1 merujuk nilai kedua dalam tatasusunan dan seterusnya sehingga indeks `n`, nilai `n` dalam tatasusunan.

```
var no    =    [];
no[0]    =    5;
no[1]    =    -1;
no[2]    =    4;
no[3]    =    12;
no[4]    =    8;
```

- Input nilai tatasusunan (*array*) dalam atur cara bahasa penskripan klient

Input ialah nilai yang diterima daripada pengguna atur cara. Sekiranya pengguna atur cara memasukkan nilai menggunakan kotak *prompt*, yang berikut ialah cara input nilai tatasusunan:

```
var input;
var no = [ ];
input = prompt("Sila masukkan satu nombor:");
no[0]=input;
input = prompt("Sila masukkan satu nombor:");
no[1]=input;
input = prompt("Sila masukkan satu nombor:");
no[2]=input;
```

- Pemrosesan nilai tatasusunan (*array*) dalam atur cara bahasa penskripan klien

Pemrosesan bermaksud melaksanakan operasi-operasi seperti operasi aritmetik dan operasi logik. Antara operasi aritmetik yang boleh dilaksanakan adalah seperti mendapatkan jumlah nilai-nilai, manakala operasi logik adalah seperti mencari sesuatu nilai dalam tatasusunan.

CONTOH 39

Contoh 39 menunjukkan nombor-nombor dalam tatasusunan dijumlahkan dengan dua cara yang berbeza. Jadual 3.1.39(a) merujuk setiap nombor dengan lokasi indeks masing-masing dijumlahkan nilainya manakala Jadual 3.1.39(b) pula merujuk setiap nombor dengan indeks yang diubah oleh pembilang dalam gelung `for` dijumlahkan nilainya.

- Mendapatkan jumlah nilai dalam tatasusunan
 - (a) Mendapatkan nombor dalam senarai satu per satu dan menjumlahkannya

Jadual 3.1.39(a) Atur cara *JavaScript* dan output (Kira Jumlah)

Atur cara	Contoh Output
<pre><html> <body> <script> var no = [5, -1, 4, 12, 8]; var jumlah = 0; jumlah = no[0] + no[1] + no[2] + no[3] + no[4]; document.write (jumlah); </script> </body> </html></pre>	28

- (b) Menggunakan ulangan untuk mendapatkan nombor dalam senarai dan menjumlahkannya

Jadual 3.1.39(b) Contoh atur cara *JavaScript* dan output (Kira Jumlah) menggunakan `for`

Atur cara	Contoh Output
<pre><html> <body> <script> var no = [5, -1, 4, 12, 8]; var jumlah = 0; var i; for (i=0; i<5; i++) { jumlah = jumlah + no[i]; } document.write (jumlah); </script> </body> </html></pre>	28

CONTOH 40

Contoh 40 menunjukkan proses mencari satu nombor dalam tatasusunan. Gelung `for` digunakan untuk merujuk satu per satu nombor di dalam tatasusunan dengan menggunakan indeks dan seterusnya menyemak sama ada nombor tersebut ialah 12 dengan menggunakan pernyataan `for`. Atur cara *JavaScript* dan output ditunjukkan dalam Jadual 3.1.40.

- Atur cara akan mencari nilai daripada suatu senarai nombor dalam tatasusunan.

Jadual 3.1.40 Contoh atur cara *JavaScript* dan output (Cari Nilai)

Atur cara	Contoh Output
<pre><html> <body> <script> var no = [5, -1, 4, 12, 8]; var i; for(i=0; i<5; i++) { if (no[i] == 12) document.write (no[i] + " dijumpai"); } </script> </body> </html></pre>	12 dijumpai

CONTOH 41

Contoh 41 menunjukkan nombor-nombor dalam tatasusunan diisih secara menaik dengan menggunakan kaedah isihan buih. Atur cara *JavaScript* dan output ditunjukkan dalam Jadual 3.1.41.

- Atur cara akan mengisihkan suatu senarai nombor dalam tatasusunan.

Jadual 3.1.41 Contoh atur cara *JavaScript* dan output (isihan buih)

Atur cara	Contoh Output
<pre><html> <body> <script> var no = [5, -1, 4, 12, 8]; var i, j, sementara; for(i=0; i<5; i++) { document.write (" " + no[i]); } for(i=0; i<5-1; i++) { for(j=0; j<5-i-1; j++) {</pre>	5 -1 4 12 8 Senarai nombor (telah diisih secara menaik): -1 4 5 8 12

Atur cara	Contoh Output
<pre> if (no[j] > no[j+1]) { sementara = no[j]; no[j] = no[j+1]; no[j+1] = sementara; } } document.write ("

Senarai nombor (telah diisih secara menaik):

"); for(i=0; i<5; i++) { document.write (" " + no[i]); } </script> </body> </html> </pre>	

- Memaparkan nilai tatasusunan (*array*) dalam atur cara bahasa penskripan klien

Memaparkan bermaksud aktiviti menyerahkan atau menunjukkan hasil nilai-nilai dalam tatasusunan kepada pengguna atur cara.

- Paparan juga boleh dilaksanakan dengan memanggil nama pemboleh ubah tatasusunan sahaja (tanpa merujuk indeks).

Contoh:

```
document.write (" " + no);
```

- Paparan boleh dilaksanakan dengan memanggil satu per satu nilai dalam tatasusunan apabila merujuk indeks (satu per satu indeks).

```
document.write (" " + no[0] + " " + no[1] + " " + no[2]);
for(i=0; i<5; i++)
    document.write (" " + no[i]);
```

- Membina atur cara lengkap tatasusunan (*array*) dalam bahasa penskripan klien

CONTOH 42

Contoh 42 menunjukkan atur cara yang dapat menerima senarai nilai suhu (dalam *Celsius*) bagi beberapa buah negara di dunia dan mengisih serta memaparkan senarai negara mengikut turutan nilai suhu daripada yang terendah kepada yang tertinggi (susunan menaik). Diberi nilai-nilai suhu adalah seperti dalam Jadual 3.1.42.



Jadual 3.1.42 Senarai nilai suhu (dalam Celcius)

Negara	Suhu
Algeria	51.3°C (124.3 °F)
Ghana	43.3°C (109.9 °F)
Nigeria	46.4°C (115.5 °F)
Afrika Selatan	50.0°C (122.0 °F)
Tunisia	55.0°C (131.0 °F)

Atur cara *JavaScript* dan output ditunjukkan dalam Rajah 3.1.46 dan Rajah 3.1.47.

```
<html>
  <body>
    <p>Senarai negara dan suhu (belum diisih): </p>
    <script>
      1 var negara = ["Algeria","Ghana","Nigeria","Afrika Selatan","Tunisia"];
      2 var suhu = [51.3, 43.3, 46.4, 50.0, 55.0];
        var i, j, sementara, sementara2;
        for(i=0; i<5; i++)
      3 {
          document.write (" "+ negara[i] + " " + suhu[i]);
          document.write ("<br>");
        }
        for(i=0; i<5-1; i++)
      4 {
          for(j=0; j<5-i-1; j++)
            {
              if (suhu[j] > suhu[j+1])
                {
                  sementara = suhu[j];
                  suhu[j] = suhu[j+1];
                  suhu[j+1] = sementara;
                }
              sementara2 = negara[j];
              negara[j] = negara[j+1];
              negara[j+1] = sementara2;
            }
        }

        document.write ("<br><br>Senarai negara beserta suhu (telah
          diisih):<br><br>");

      6 for(i=0; i<5; i++)
        {
          document.write (" "+ negara[i]+" "+ suhu[i]);
          document.write ("<br>");
        }

    </script>
  </body>
</html>
```

Rajah 3.1.46 Atur cara *JavaScript* (isih tatasusunan)

```
codeOutputContoh43.htm X
file:///D:/CompScienceBook_Powtoon...

Senarai negara dan suhu (belum diisih):
Algeria 51.3
Ghana 43.3
Nigeria 46.4
Afrika Selatan 50
Tunisia 55

Senarai negara beserta suhu (telah diisih):
Ghana 43.3
Nigeria 46.4
Afrika Selatan 50
Algeria 51.3
Tunisia 55
```

Rajah 3.1.47 Output (isih tatasusunan)

Penerangan:

- 1 Terdapat satu tatasusunan, iaitu senarai negara.
- 2 Satu lagi tatasusunan, iaitu suhu bagi setiap negara tersenarai pada (1).
- 3 Paparkan negara dan suhu.
- 4 Laksanakan isihan buih.
- 5 Setiap kali suhu disusun atau ditukar kedudukan, nama negara juga akan ditukar kedudukannya mengikut suhunya.
- 6 Paparkan negara dan suhu yang telah diisih.

AKTIVITI 12

Kerja Individu

Tatasusunan dalam atur cara Bahasa Penskripan Klien

Objektif: Menggunakan tatasusunan bagi menyimpan dan memproses nilai-nilai.

Lakukan aktiviti ini di dalam makmal komputer.

Langkah-langkah:

- 1 Buka perisian *Notepad*.
- 2 Bina atur cara HTML dan *JavaScript* yang dapat:
 - (a) menyimpan senarai ketinggian bagi tujuh orang pelajar dalam satu tatasusunan.
 - (b) mengisih dan memaparkan ketinggian pelajar-pelajar tersebut.



- 4 Simpan (save) sebagai Aktiviti12.html.
- 5 Buka fail Aktiviti12.html melalui pelayar web (*web browser*) seperti *Internet Explorer* atau *Google Chrome*. Contoh output adalah seperti yang berikut:



3.1.9 Atur Cara Interaktif yang Mengandungi Pengisytiharan Data, Struktur Kawalan, *Procedure* dan *Function* serta Tatasusunan

Dalam bahagian ini, kita akan mempelajari cara untuk membangunkan laman web yang dapat berinteraksi dengan pengguna atau juga dikenali sebagai laman web interaktif. Murid boleh membina laman web mudah dengan menggunakan HTML/CSS dan menambah bahasa *JavaScript* untuk berinteraksi.

Kini, *JavaScript* digunakan dalam pelbagai persekitaran seperti berinteraksi dengan pelayan, menambah animasi pada laman web, memanipulasi imej dan menyimpan serta memproses data pengguna. Sebagai permulaan, kita boleh menggunakan butang atau tag HTML `<button>` untuk pengguna berinteraksi dengan laman web.

- **Tag HTML `<button>`**

Tag HTML `<button>` boleh digunakan untuk laman web interaktif, iaitu butang akan terhasil dalam dokumen HTML (laman web).

CONTOH 43

Contoh 43 pada Jadual 3.1.43 dan Jadual 3.1.44 menunjukkan contoh atur cara *JavaScript* dan output yang menggunakan tag HTML `<button>` untuk menghasilkan butang untuk diklik dan digunakan “onclick=” untuk melaksanakan arahan selepas butang diklik.


Jadual 3.1.43 Contoh atur cara *JavaScript* dan output `<button>`

Atur cara	Contoh Output
<pre><html> <body> <button onclick = "window.alert('Hai!')"> Klik Sini </button> </body> </html></pre>	

Penerangan:

- 1 Butang dengan tag HTML `<button>` dihasilkan.

Jadual 3.1.44 Contoh atur cara *JavaScript* dan output `<onclick=>`

Atur cara	Contoh Output
<pre><html> <body> <button onclick = "window.alert('Hai!')"> Klik Sini </button> </body> </html></pre> <p>Gunakan button “onclick”</p>	

Penerangan:

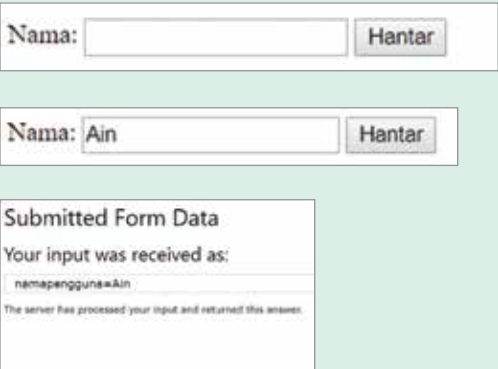
- 2 Apabila pengguna mengklik butang “klik sini”, kotak pop-up alert dengan mesej “Hai!” dipaparkan.

CONTOH 44

Contoh 44 menggunakan medan teks daripada `<form>` yang disediakan oleh HTML. Seterusnya, nilai dalam medan teks akan dihantar kepada halaman “action_page.php” selepas butang “Hantar” diklik. Jadual 3.1.45 menunjukkan contoh atur cara HTML dan output yang menggunakan tag HTML `<form>`.



Jadual 3.1.45 Contoh atur cara HTML dan output `<form>`

Atur cara	Contoh Output
<pre><html> <body> 1 <form action="\action_page.php"> Nama: 3 <input type="text" name="namapengguna" value=""> 2 <input type="submit" value="Hantar"> </form> </body> </html></pre>	

Penerangan:

- 1 Tag HTML `<form>` digunakan untuk menghasilkan borang (`form`).
- 2 Borang mempunyai butang "Hantar" berjenis `submit`.
- 3 Teks daripada medan "Nama:" berjenis `text` dihantar kepada halaman "action_page.php".

Jika butang "Hantar" diklik, data pada borang akan dihantar kepada satu halaman yang dipanggil "action_page.php". Pelayan akan memproses input yang dimasukkan pada borang dan memulangkan jawapan seperti yang dikehendaki atau diarahkan seterusnya. Murid akan mempelajarinya pada topik yang seterusnya, iaitu dalam topik Bahasa Penskripan Pelayan.

- Menggunakan tag HTML "`<button>`" dan memanggil *procedure/function* untuk melaksanakan arahan dalam atur cara interaktif.

Untuk topik ini, kita akan menggunakan butang tag HTML "`<button>`" untuk membina laman web interaktif. Sebagai contoh, setelah butang diklik, suatu *procedure/function* akan dipanggil untuk melaksanakan sesuatu tugas/arahan.

Seperti yang telah dipelajari sebelum ini, iaitu dalam topik 3.1.7, *procedure/function* ialah suatu kaedah dalam pembangunan atur cara yang melibatkan pembangunan asas kepada blok-blok dalam atur cara. Blok ialah kawasan/seksyen sesuatu arahan, iaitu kod-kod atur cara ditempatkan untuk melaksanakan sesuatu tugas tertentu. Tujuannya untuk mengurangkan pengulangan ketika menulis arahan/kod yang sama. Misalnya, apabila arahan/kod tersebut diperlukan semula, blok/seksyen tersebut akan dipanggil semula untuk melaksanakan arahan/kod tersebut.

Tahukah Anda

- Tag HTML `<form>` Unsur `<button>` juga boleh digunakan dalam borang HTML (*HTML form*). Unsur HTML `<form>` akan menghasilkan satu borang yang dapat mengumpul input/data daripada pengguna dan terdapat beberapa unsur input seperti medan teks, *checkbox*, *radio button* dan *submit button*.



Akses QR code di atas untuk melihat animasi pengaturan cara berbasaskan web.

CONTOH 45

Contoh 45 menunjukkan paparan senarai nombor yang belum diisih dan telah diisih yang boleh diletakkan pada blok-blok yang berbeza dengan menggunakan *procedure/function*, iaitu *procedure/function* **SebelumIsih()** dan **SelepasIsih()**. Dua `<button>` digunakan untuk pengguna berinteraksi dengan laman web bagi memilih nombor sebelum diisih atau selepas diisih.

Atur cara *JavaScript* dan output bagi Isihan Buih (*Bubble Sort*) menggunakan butang secara interaktif dan *procedure/function* untuk blok arahan penskripan ditunjukkan pada Rajah 3.1.48 dan Rajah 3.1.49.

```
<html>
  <body>

    <p>Senarai nombor sebelum diisih: </p>

    <button onclick="sebelumIsih()" > Sebelum Isih </button>

    <script>
      function sebelumIsih()
      {
        var no = [5, 1, 4, 3, 2];
        document.write(no);
      }
    </script>

    <p>Senarai nombor selepas diisih (Isihan Buih): </p>

    <button onclick="selepasIsih()" > Selepas Isih </button>

    <script>
      function selepasIsih()
      {
        var no = [5, 1, 4, 3, 2];
        var panjang = no.length, i, j;
        var sementara;
        for (i=0; i<panjang; i++)
        {
          for (j=0; j<panjang-i-1; j++)
          {
            if (no[j] > no[j+1])
            {
              sementara = no[j];
              no[j] = no[j+1];
              no[j+1] = sementara;
            }
          }
        }
        document.write(no);
      }
    </script>
  </body>
</html>
```

Rajah 3.1.48 Atur cara *JavaScript* (butang interaktif `<button>`)



Contoh output:

Senarai nombor sebelum diisih:

Senarai nombor selepas diisih (Isihan Buih):

(Selepas klik "Sebelum Isih")

5 1 4 3 2

(Selepas klik "Selepas Isih")

1 2 3 4 5

Rajah 3.1.49 Output (butang interaktif <button>)

Penerangan:

- 1 Terdapat 2 butang, iaitu "Sebelum Isih" dan "Selepas Isih".
- 2 Selepas butang "Sebelum Isih" diklik, atur cara memanggil function `SebelumIsih()`.
- 3 Selepas butang "SelepasIsih" diklik, atur cara memanggil function `SelepasIsih()`.
- 4 Senarai nombor belum diisih dan hasil senarai nombor telah diisih dipaparkan pada dokumen atau halaman baharu kerana penggunaan `document.write(no)`.

Suatu atur cara boleh terdapat lebih daripada satu *procedure/function*. Setiap *procedure/function* mempunyai arahannya sendiri dan boleh memanggil *procedure/function* yang lain.

CONTOH 46

Contoh 46 menunjukkan atur cara bagi isihan buih (*bubble sort*) menggunakan butang secara interaktif dan beberapa *function* untuk mengumpulkan semua arahan penskripan:

Untuk mengisi senarai nombor tersebut, arahan/kod bagi proses mengisi boleh diletakkan pada blok kod yang lain. Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.50 dan Rajah 3.1.51.

```
<html>
  <body>
    <p>Senarai nombor sebelum diisih: </p>
    <button onclick="sebelumIsih()"> Sebelum Isih </button>

    <script>
      function sebelumIsih()
      {
```

Memanggil *function* `sebelumIsih()`

```

    var no = [5, 1, 4, 3, 2];
    document.write(no); 3
  }
</script>

<p>Senarai nombor selepas diisih (Isihan Buih): </p>
<button onclick="selepasIsih()"> Selepas Isih </button>

<script>
  function selepasIsih()
  {
    var no = [5, 1, 4, 3, 2];
    var panjang = no.length, i, j;
    var sementara;
    isihanBuih(no, panjang, sementara);
  }
</script>

<script>
  function isihanBuih(no, panjang, sementara)
  {
    var i, j;

    for (i=0; i<panjang; i++)
    {
      for (j=0; j<panjang-i-1; j++)
      {
        if (no[j] > no[j+1])
        {
          tukarKedudukan(no, sementara, j, j+1);
        }
      }
    }
    document.write(no); 3
  }
</script>

<script>
  function tukarKedudukan(no, sementara, i, j)
  {
    sementara = no[i];
    no[i] = no[j];
    no[j] = sementara;
  }
</script>
</body>
</html>

```

Rajah 3.1.50 Atur cara *JavaScript* (subatur cara) memanggil subatur cara lain



Penerangan:

- 1 function `selepasIsih()` memanggil subatur cara lain iaitu `isihanBuih()`.
- 2 function `isihanBuih()` pula memanggil subatur cara `tukarKedudukan()` untuk melaksanakan arahan tukar kedudukan nombor.
- 3 Senarai nombor belum diisih dan senarai nombor telah diisih dipaparkan.

Senarai nombor sebelum diisih:

Sebelum Isih

Senarai nombor selepas diisih (Isihan Buih):

Selepas Isih

(Selepas klik "Sebelum Isih")

5 1 4 3 2

(Selepas klik "Selepas Isih")

1 2 3 4 5

Rajah 3.1.51 Output (subatur cara) memanggil subatur cara lain

- **Tag HTML `<a>...` atau *hyperlink***

Hyperlink atau tag HTML `<a>...` digunakan supaya pengguna dapat berinteraksi dengan laman web. Cara untuk melaksanakan arahan atur cara penskripan untuk ke tapak web atau laman web melalui teks atau imej dengan mengklik pada teks atau imej untuk ke tapak web atau laman web yang ditentukan seperti yang berikut.

CONTOH 47

Contoh 47 menunjukkan penggunaan *hyperlink* secara interaktif (melalui teks). Atur cara *JavaScript* dan output ditunjukkan pada Rajah 3.1.52.

```
<html>
  <body>
    <a href = "https://www.mohe.gov.my/en/" > Laman Web KPT Malaysia </a>
  </body>
</html>
```

1

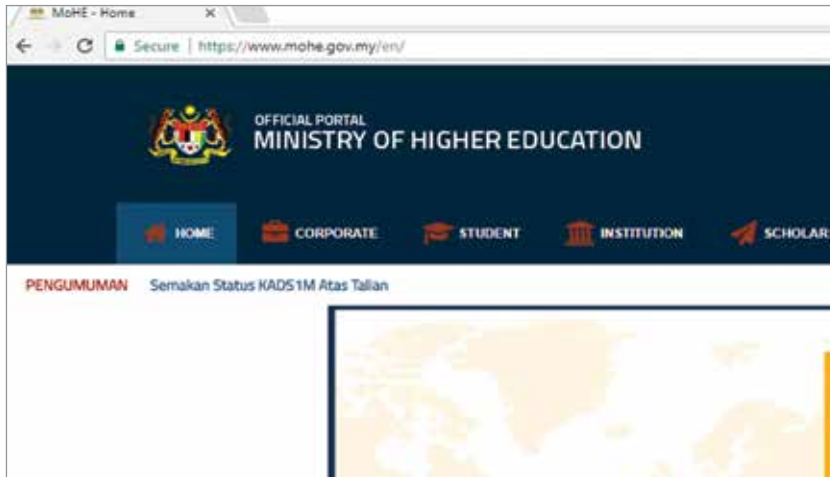
teks

2



Contoh output:

[Laman Web KPT Malaysia](https://www.mohe.gov.my/en/)



Rajah 3.1.52 Atur cara *JavaScript* dan output (interaktif secara hyperlink teks)

Penerangan:

- 1 URL `https://www.mohe.gov.my/en/` dalam tag `<a>` ialah tapak web yang ditetapkan.
- 2 Selepas pengguna yang mengklik teks "Laman Web KPT Malaysia", tapak web akan dibuka.

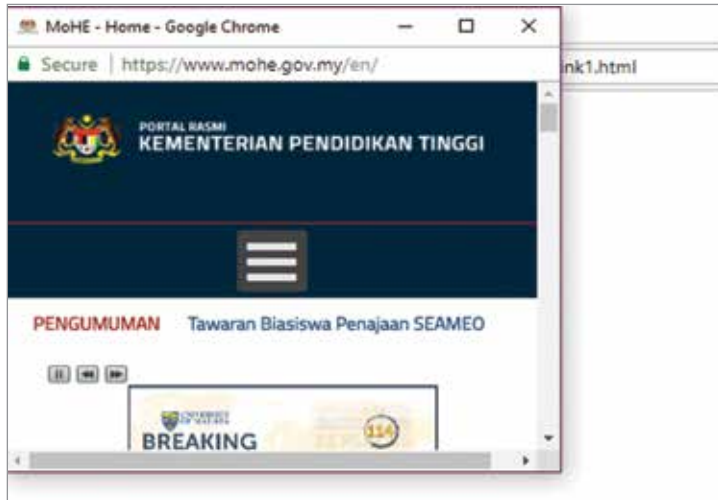
CONTOH 48

Contoh 48 menunjukkan penggunaan *hyperlink* secara interaktif (melalui teks) dan *procedure/function* untuk blok arahan penskripan, iaitu selepas pengguna mengklik pautan (link) subatur cara lamanweb() dilaksanakan iaitu satu dokumen HTML baharu (tetingkap baharu) dibuka dan dipaparkan. Atur cara *JavaScript* dan output ditunjukkan dalam Rajah 3.1.53.

```
<html>
  <body>
    <a href = "javascript:lamanWeb()"> Laman Web KPT Malaysia </a>
    <script>
      function lamanWeb()
      {
        window.open("https://www.mohe.gov.my/en/", "jav",
          "width=400,height=300,resizable=yes");
      }
    </script>
  </body>
</html>
```



Contoh output:



Rajah 3.1.53 Atur cara JavaScript dan output (interaktif secara hyperlink – panggil subatur cara)

Penerangan:

- 1 URL `https://www.mohe.gov.my/en/` dalam tag `<a>` ialah tapak web yang ditetapkan.
- 2 Selepas pengguna mengklik teks “Laman Web KPPT Malaysia”, tapak web akan dibuka pada satu dokumen HTML yang baharu.

Tahukah Anda



Fungsi `window.open()` akan membuka satu tettingkap pelayar (*browser*) dan memuatkan sumber URL dalam bentuk tettingkap atau tab. Sekiranya sumber tidak dijumpai pada tettingkap yang telah terbuka, satu tettingkap baharu akan dibuka dan sumber URL tersebut akan dimuatkan. `jav` menetapkan jenis bagi kandungan sumber URL yang dibuka.

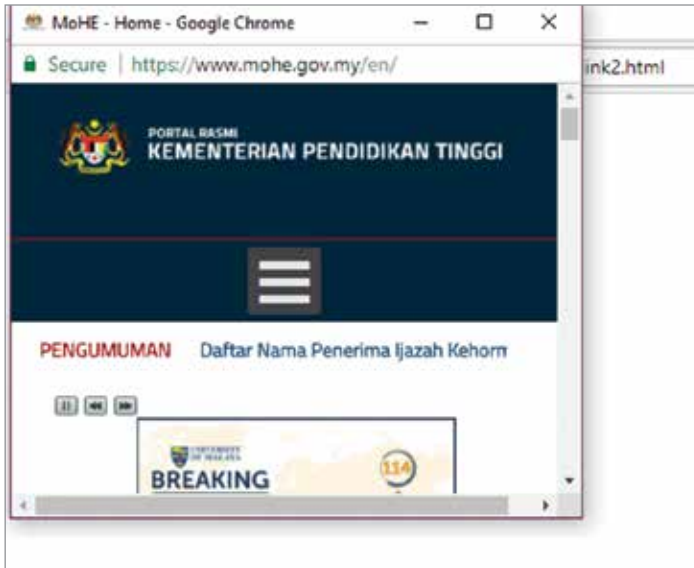
Sumber:
<https://developer.mozilla.org/en-US/docs/Web/API/Window/open>

CONTOH 49

Contoh 49 menunjukkan penggunaan *hyperlink* secara interaktif (melalui imej) dan *procedure/function* untuk blok arahan penskripan, iaitu selepas pengguna mengklik logo (iaitu imej), dokumen HTML baharu dibuka dan dipaparkan. Atur cara JavaScript dan output ditunjukkan pada Rajah 3.1.54.

```
<html>
  <body>
    <a href = "javascript:lamanweb()" > 
    </a>
    <script>
      function lamanweb()
      {
        window.open("https://www.mohe.gov.my/en/", "jav",
          "width=400,height=300,resizable=yes");
      }
    </script>
  </body>
</html>
```

Contoh output:



Rajah 3.1.54 Atur cara JavaScript dan output (interaktif secara hyperlink – imej)

Penerangan:

- 1 URL `https://www.mohe.gov.my/en/` dalam function `lamanweb()` ialah tapak web yang ditetapkan.
- 2 Selepas pengguna mengklik imej, iaitu daripada sumber imej `src="logo_kpt_2012.png"`, function `lamanweb()` dipanggil dan tapak web akan dibuka pada satu dokumen HTML yang baharu.

AKTIVITI

13

Kerja Individu

Atur Cara Interaktif dalam Bahasa Penskripan Klien

Objektif: Menggunakan pautan teks dan imej sebagai atur cara interaktif.

Lakukan aktiviti ini di dalam makmal komputer.

Langkah-langkah:

- 1 Pilih 4 Kerajaan Negeri dan dapatkan pautan ke 4 portal web tersebut. Senarai Kerajaan Negeri adalah seperti yang berikut:
 - (a) Kerajaan Negeri Selangor
 - (b) Kerajaan Negeri Johor



- (c) Kerajaan Negeri Melaka
- (d) Kerajaan Negeri Perak
- (e) Kerajaan Negeri Pahang
- (f) Kerajaan Negeri Perlis
- (g) Kerajaan Negeri Sarawak
- (h) Kerajaan Negeri Terengganu
- (i) Kerajaan Negeri Kedah

2 Pilih logo negeri yang bersesuaian bagi 4 portal tersebut.



Contoh: Kerajaan Negeri Sarawak

- 3 Buka perisian *Notepad*.
- 4 Bina atur cara HTML dan *JavaScript* yang dapat:
 - (a) Memaparkan 4 logo negeri yang telah dipilih.
 - (b) Membuka portal (tapak web) apabila pengguna menekan setiap logo atau teks “Kerajaan Negeri” melalui pautan yang telah diperolehi pada Langkah 1.
- 5 Simpan (*save*) sebagai *Aktiviti13.html*
- 6 Buka fail *Aktiviti13.html* melalui pelayar web (*web browser*) seperti *Internet Explorer* atau *Google Chrome*. Contoh output adalah seperti berikut:



LATIHAN PENGUKUHAN 3.1

- 1 Tulis tag HTML untuk meletakkan imej (`car.png`) di sudut kiri atas dokumen HTML.
- 2 Bina carta alir untuk mendapatkan suhu tertinggi (maksimum) bagi senarai suhu yang diterima daripada penggunaan sebuah sensor suhu.
- 3 Terangkan aplikasi penggunaan struktur ulangan `for`, `while` dan `do...while` secara ringkas dengan memberikan contoh-contoh yang bersesuaian.
- 4 Berikan ungkapan aritmetik *JavaScript* yang menggunakan fungsi matematik yang terdapat dalam `math.js` untuk menulis formula matematik di bawah:

$$E = MC^2$$

- 5 Satu *function* `KiraPurata()` dihasilkan dalam atur cara *JavaScript* seperti di bawah:

```
function KiraPurata(senaraiHarga)
{
  var i, bilangan=senaraiHarga.length, jumlah=0, purata=0;
  for(i=0; i<bilangan; i++)
  {
    jumlah=jumlah+senaraiHarga[i];
  }
  purata = jumlah/bilangan;
  return purata;
}
```

Tulis pernyataan untuk memanggil *function* tersebut.

- 6 Diberi subatur cara *JavaScript* bernama `MAJLISMAKAN()` yang mempunyai tiga parameter, iaitu nama organisasi, pilihan menu sama ada daging atau ayam (1: Daging, 2: Ayam), dan bilangan tetamu. Subatur cara ini akan menyemak pilihan menu yang dipilih dengan menggunakan pernyataan `if` dan seterusnya mengira kos bagi majlis makan malam tersebut berdasarkan bilangan tetamu yang hadir seperti yang berikut:

```
function MAJLISMAKAN(namaOrganisasi, menu, bilTetamu)
{
  if (menu==1)
    kos=15.00;
  else if (menu==2)
    kos=20.00;
  jumlahKos=kos*bilTetamu;

  document.write("<br>Nama Organisasi: " + namaOrganisasi + "<br>Kos
    bagi " + bilTetamu + " bilangan tetamu (RM): " + jumlahKos);
}
```

Tulis pernyataan *JavaScript* untuk memanggil subatur cara `MAJLISMAKAN()` bagi organisasi-organisasi seperti yang berikut:



- (a) Nama organisasi: UNiA, Menu: Daging, Bilangan tetamu: 100
- (b) Nama organisasi: Akuan Sdn Bhd, Menu: Ayam, Bilangan tetamu: 220
- (b) Nama organisasi: Pertama & Co., Menu: Daging, Bilangan tetamu: 110

7 Diberi keratan *JavaScript* yang mempunyai nilai ketumpatan bagi enam jenis bahan. Apakah nilai yang dicetak?

```
var ketumpatan = [0.25, 2.35, 0.77, 20.0, 5.5, 6.0];
document.write(ketumpatan[2]);
```

8 Tulis pengisytiharan tatasusunan yang dapat memegang lima nilai pH bagi lima asid hidroklorik daripada hasil kajian kepekatan asid hidroklorik. Contoh hasil kajian kepekatan asid hidroklorik adalah seperti yang berikut:



Asid Hidroklorik	Kepekatan	pH
A	0.001	3
B	0.1	1
C	0.01	2
D	0.1	1
E	1	0

9 Lengkapkan keratan atur cara *JavaScript* yang berikut dengan mengisi tempat kosong dengan pernyataan `onclick()` yang bersesuaian:

```
<html>
<body>
  <p>Klik butang untuk memaparkan hasil kiraan menggunakan formula
    matematik  $y = mx + c$ .</p>

  <button <input type="text" /> >Kiraan Matematik</button>

  <p id="demo" ></p>
  <script>
    function FungsiMath ()
    {
      var y,m=2,x=10,c=3;
      y=m*x+c;
      document.write("<br>" +y);
    }
  </script>
</body>
</html>
```

10 Tulis pernyataan `onclick()` yang betul untuk menghasilkan satu butang dan paparan seperti di bawah:



3.2

STANDARD PEMBELAJARAN

Pada akhir pembelajaran murid dapat

- 3.2.1 Menjelaskan fungsi laman web sebagai penghubung antara pengguna dengan sistem dalam rangkaian
- 3.2.2 Menghasilkan dan menyimpan data dalam fail teks (.txt)
- 3.2.3 Menggunakan bahasa penskripan pelayan untuk membina laman web yang unik bagi setiap pengguna
- 3.2.4 Menggunakan data yang diimport daripada fail teks untuk menyelesaikan masalah
- 3.2.5 Menggunakan data yang diimport daripada fail pangkalan data untuk menyelesaikan masalah
- 3.2.6 Melakukan pengesahsahihan (*validation*) pada data input daripada pengguna
- 3.2.7 Membina (*create*), mengemas kini (*update*), membuat sandaran (*backup*) dan memulihkan (*restore*) pangkalan data
- 3.2.8 Melaksanakan beberapa bentuk pertanyaan (*query*) pangkalan data
- 3.2.9 Membina daftar (*sign up*) dan log masuk (*login*) pada laman web
- 3.2.10 Menghasilkan atur cara yang dapat mengesahkan pengguna dan mengemas kini data dalam pangkalan data

Bahasa Penskripan Pelayan

3.2.1 Pembangunan Laman Web dengan Menggunakan Bahasa Penskripan Pelayan

Keperluan Laman Web Antara Pengguna dengan Sistem Dalam Rangkaian

Bayangkan anda merupakan seorang pengawas perpustakaan dan merupakan setiausaha bagi badan tersebut. Anda telah diarahkan untuk menulis laporan lengkap mengenai keberkesanan sistem pinjaman buku di perpustakaan sekolah. Apakah perkara pertama yang akan anda lakukan sebelum memulakan penulisan laporan anda?

Sewajarnya kita perlu mencari bahan untuk merangka isi dalam laporan tersebut atau sekurang-kurangnya meneliti format yang sesuai untuk menulis. Situasi ini mempunyai persamaan dengan pembinaan laman web. Sebelum laman web dibina, seseorang pereka web harus merancang terlebih dahulu bagaimanakah sesuatu laman web tersebut dipersembahkan. Hal ini dikatakan demikian kerana setiap laman web yang dibangunkan itu akan menjadi perantara yang akan menentukan cara pengguna akan berinteraksi dengan sistem yang digunakan.

Dalam era teknologi di hujung jari ini, pasti anda sudah mempunyai pengalaman yang berbeza-beza apabila melayari pelbagai jenis laman web dalam Internet. Sebagai contoh, sekiranya anda melayari laman web enjin carian, anda perlu memasukkan kata kunci yang bersesuaian untuk mendapatkan maklumat yang anda kehendaki.

Hal ini disebabkan laman web tersebut membantu pengguna untuk mencapai maklumat yang dikehendaki. Akan tetapi, sekiranya anda melayari laman web mana-mana syarikat penerbangan, anda perlu memasukkan destinasi yang ingin anda lawati serta tarikh untuk sistem memaparkan harga tiket yang anda inginkan.

Setiap laman web akan menghendaki pengguna memasukkan input yang berbeza bagi membenarkan sistem memaparkan maklumat mengikut keperluan pengguna tersebut.

Cuba anda bayangkan tanpa laman web, bagaimanakah anda akan berinteraksi dengan sistem perbankan, enjin carian ataupun media sosial? Mampukah anda berinteraksi dengan sistem?

Kesimpulannya, kewujudan laman web ini adalah untuk menjadi perantara dalam menentukan bagaimanakah pengguna boleh berinteraksi dengan sesuatu sistem dalam Internet. Oleh sebab itu, sebelum sesuatu laman web dibina, kita perlu terlebih dahulu mengenal pasti, apakah tujuan laman web itu dibina. Perkara ini dikatakan demikian kerana paparan dalam laman web akan menentukan cara seseorang pengguna itu dapat memanipulasikan maklumat dalam sesebuah sistem dan menentukan sejauh manakah capaian pengguna terhadap sistem tersebut.

Konsep Pelayan

Apabila kita ke sebuah restoran, kita akan memanggil pelayan restoran untuk membuat pesanan makanan yang kita pilih berdasarkan menu yang diberikan. Seterusnya pelayan akan memastikan pesanan klien akan diproses oleh tukang masak dan dihantar semula kepada klien mengikut kehendak klien. Dalam analogi ini, pelanggan restoran ialah satu pihak yang meminta servis daripada pelayan restoran dan pelayan restoran ialah pihak lain yang akan memenuhi kehendak klien.

Begitu jugalah konsep pelanggan/pelayan model yang digunakan bagi menyokong komunikasi menerusi Internet. Model ini berasaskan pengagihan fungsi antara klien dengan pelayan.

Klien ialah apa-apa mesin atau proses yang memohon servis yang spesifik daripada pelayan manakala pelayan ialah mesin atau proses yang memberikan servis kepada klien. Proses klien dan pelayan ini boleh berlaku dalam satu sistem komputer yang sama atau dalam sistem komputer yang berbeza yang dihubungkan melalui rangkaian. Dalam persekitaran Internet dan rangkaian, pelayar web akan bertindak sebagai klien web manakala pelayan web pula adalah sebagai pelayan.

Bahasa Penskripan Pelayan

Bahasa penskripan pelayan ialah program yang dilaksanakan di sebelah pelayan atau pelayan web yang melatari pembangunan web.

Ibarat sebuah rumah yang perlu disokong oleh pelbagai fungsi untuk melengkapkan fungsi rumah itu sendiri seperti pendawaian elektrik, sistem paip dan sistem saliran, begitu juga fungsi bahasa penskripan pelayan dalam sesebuah laman web. Salah satu fungsi bahasa penskripan pelayan ialah mengakses data daripada pangkalan data. Sekiranya pengguna memohon suatu laman web dipaparkan dengan menggunakan pelayar, permintaan tersebut akan dihantar kepada pelayan web. Kemudian, pelayan web akan mencari laman web yang dikehendaki dan memindahkan laman web tersebut kepada pelayar untuk diterjemahkan dan dipaparkan kepada pengguna.

Tahukah Anda

Tim Berners-Lee telah membangunkan *World Wide Web* (WWW) kepada majikannya, CERN pada tahun 1989. Pada tahun 1990, beliau telah mencipta pelayar web yang pertama untuk WWW.

Sumber:
<https://www.w3.org/People/Berners-Lee>



Bahasa Penskripan Pelayan
<https://goo.gl/0xKutg>



Web Pages and
Web Apps
goo.gl/QfnxT2



APLIKASI HARIAN

Perl telah merangkumi banyak bidang IT. *Perl* merupakan tulang belakang analisis bioinformatik dalam bidang sains yang berkembang pesat. *Perl* memberi sumbangan besar dalam bidang biologi, terutamanya dalam Projek Genom Manusia.

Sumber:
http://www.perlmonks.org/?node_id=823275

Antara sebab utama penskripan pelayan diperlukan ialah bahasa penskripan pelayan tidak bergantung pada pelayar. Jika dibandingkan dengan penskripan klien yang bergantung pada pelayar, isu akan timbul apabila penskripan klien yang digunakan tidak disokong oleh pelayar. Keadaan ini berbeza dengan penskripan pelayan yang kesemua pemprosesan dilakukan di sebelah pelayan, maka isu pemprosesan disekat oleh pelayar tidak akan timbul.

Namun begitu, kadangkala bahasa penskripan pelayan tidak begitu selamat memandangkan penskripan pelayan boleh digunakan untuk menggodam laman web. Oleh sebab itu, laman web yang bagus perlu menggunakan kedua-dua jenis bahasa penskripan klien dan pelayan bagi menyokong proses pengesahan di sebelah klien dan juga disebelah pelayan untuk tujuan keselamatan laman web tersebut.

Memandangkan penskripan pelayan berupaya untuk memberikan respons mengikut kehendak pengguna, bahasa penskripan pelayan digunakan bagi aplikasi seperti yang berikut:

- Papan pesanan
- Buku pelawat blog

Terdapat banyak pilihan bahasa penskripan pelayan yang boleh digunakan oleh laman web bagi menyokong interaksi dengan pelayan web untuk memanipulasikan data. Antara bahasa penskripan pelayan adalah seperti yang berikut:

- ASP
- *JavaScript*
- PHP
- Python
- *Perl*

AKTIVITI

1

Kerja Individu

Pemerhatian terhadap Fungsi Laman Web dalam Pelbagai Bidang

Objektif: Mengenal pasti fungsi laman web dalam pelbagai bidang

Sebagai salah seorang penulis bagi buletin sekolah, anda ditugaskan untuk mengkaji sektor-sektor yang menggunakan laman web untuk menyediakan perkhidmatan atau produk kepada pengguna bagi ruangan *Teknologi dan Anda*.

- 1 Senaraikan beberapa sektor berkaitan yang akan anda kaji.
- 2 Terangkan bagaimana laman web digunakan bagi sektor-sektor yang telah anda kaji dalam soalan 1 secara jelas.



3.2.2 Penghasilan dan Penyimpanan Data dalam Fail Teks (.txt)

Sesuatu data yang dimasukkan oleh pengguna daripada laman web boleh disimpan ke dalam pelayan dalam bentuk fail teks. Sebagai contoh, apabila pengguna (*user*) memasukkan nama serta kata laluan, maklumat tersebut boleh disimpan dalam fail teks dalam komputer pelayan. Untuk melakukan tugas sedemikian, kita akan menggunakan fungsi "fopen" sebagai arahan yang akan dilaksanakan dalam pelayan.

Skrip *Hypertext Preprocessor* atau PHP dilaksanakan oleh komputer pelayan. PHP menyediakan keupayaan pengaturcaraan untuk memproses sesuatu tugas dan akan dilaksanakan oleh pelayan. Rajah 3.2.1 ialah contoh ringkas penskripan PHP dan hasilnya akan dikembalikan kepada pelayar. Beberapa operasi asas terhadap fail teks adalah seperti yang ditunjukkan dalam Rajah 3.2.2.

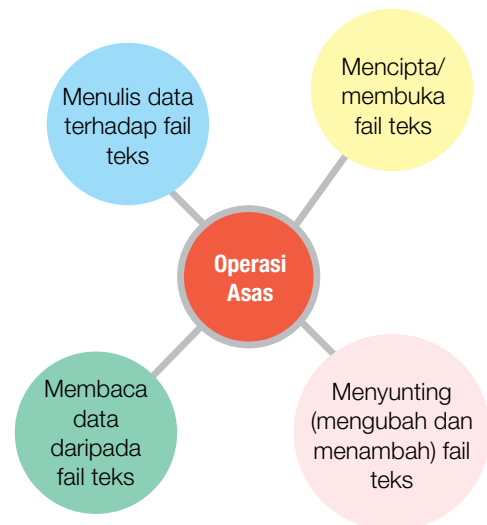
Tahukah
Anda



Hypertext Preprocessor (PHP) menyokong jenis fail untuk tujuan operasi input dan output ke atas fail seperti fail teks. Dengan menggunakan PHP, kita boleh membaca/menulis data daripada /ke fail teks.

```
<html>
<body>
  <h1> Halaman PHP saya yang pertama </h1>
  <?php
    echo "Selamat Datang";
  ?>
</body>
</html>
```

Rajah 3.2.1 Contoh ringkas penskripan PHP



Rajah 3.2.2 Operasi asas terhadap fail teks

Mencipta/membuka fail teks

Sebelum sebarang operasi terhadap fail teks dilaksanakan, fail teks tersebut mestilah dibuka terlebih dahulu. Berikut merupakan sintaks untuk mencipta/membuka fail teks, iaitu menggunakan fungsi "fopen" yang akan dimasukkan dalam laman web.

```
$f=fopen("nama fail teks", mod capaian);
```

CONTOH 1

Contoh 1 menerangkan penggunaan fungsi `fopen`.

```
$f=fopen ("LogMasuk.txt", "w");
```

"`$f`" di sini merujuk pemboleh ubah bagi mewakili fail teks yang akan digunakan dalam pelayan. "`LogMasuk`" ialah nama fail teks yang kita cadangkan untuk menyimpan teks manakala "`w`" bermaksud bahawa fail tersebut dalam mod *writing* (bertulis). Selain mod "`w`", mod "`a`" (*appending*) juga boleh digunakan sebagai cara untuk menambah kandungan teks ke dalam fail teks tersebut.

Mod capaian sebarang fail akan menentukan operasi yang boleh dilaksanakan oleh fail teks tersebut. Jadual 3.2.1 menunjukkan ringkasan beberapa mod capaian fail teks.

Tahukah Anda

Apakah fail PHP?

- ✓ Fail PHP boleh mengandungi teks, *JavaScript* dan kod PHP.
- ✓ Kod PHP dilakukan terhadap pelayan, dan keputusannya kembali kepada pelayar seperti HTML.
- ✓ Fail PHP mempunyai sambungan ".php".

Sumber:
goo.gl/FumQuW

Jadual 3.2.1 Mod capaian dan penerangan

Mod capaian	Penerangan
<code>r</code>	Fail dibuka hanya untuk dibaca sahaja
<code>w</code>	Fail dibuka hanya untuk ditulis sahaja (bermula dari awal fail)
<code>a</code>	Fail dibuka untuk ditulis sahaja (bermula di akhir fail)
<code>r +</code>	Fail dibuka untuk dibaca/ditulis. Penunjuk fail bermula daripada permulaan fail tersebut.
<code>w +</code>	Fail dibuka untuk dibaca/ditulis. Memadamkan kandungan fail atau mencipta fail baharu sekiranya fail belum wujud. Penunjuk fail bermula daripada permulaan fail tersebut.
<code>a +</code>	Fail dibuka untuk dibaca/ditulis. Kandungan fail tersebut terpelihara. Mencipta fail baharu sekiranya fail belum wujud. Penunjuk fail bermula daripada penghujung fail tersebut.

Contoh-contoh dalam pelajaran ini akan menggunakan fail teks kosong yang dipanggil "LogMasuk.txt". Namun, anda juga boleh membuat fail teks anda sendiri jika anda mahu.

Mari kita cuba untuk membuka fail teks dengan menggunakan mod "`w`" (*writing*) dan seterusnya menyimpan (menulis) sesuatu teks ke dalam fail teks tersebut.

Antara langkah yang perlu ada fail PHP ialah:

- 1 Buka fail menggunakan `fopen`.
- 2 Tulis isi kandungan fail yang telah dibuka dengan teks menggunakan `fwrite`.
- 3 Tutup fail yang telah dibuka menggunakan `fclose`.

CONTOH 2

Contoh 2 menerangkan penggunaan fungsi `fopen` serta fungsi `fwrite` dalam atur cara bagi laman web `SimpanTeks.php`.

```
<?php
    // untuk membuka fail teks dengan menggunakan mod w
    $f = fopen("LogMasuk.txt", "w"); 1
    fwrite($f, "Selamat Datang"); 2
    // menutup fail setelah digunakan
    fclose($f); 3
?>
```

Rajah 3.2.3 Atur cara bagi laman web `SimpanTeks.php`

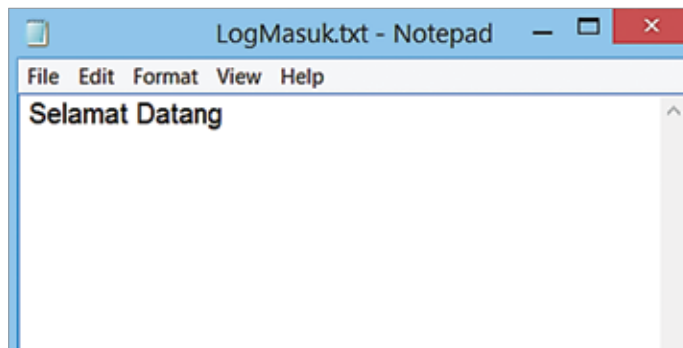
Penerangan:

Pernyataan `$f = fopen("LogMasuk.txt", "w");` pada 1 adalah untuk membuka fail teks yang bernama "LogMasuk.txt". Hasil pelaksanaan pernyataan ini ialah fail tersebut akan dicipta dalam *folder* yang berada dalam komputer pelayan, dan dirujuk dengan pemboleh ubah penunjuk "\$f". Pada ketika ini, isi kandungan fail tersebut masih kosong.

Pernyataan `fwrite($f, "Selamat Datang");` pada 2 seterusnya akan menulis/mencatatkan perkataan "Selamat Datang" dalam fail tersebut.

Sekiranya tiada lagi teks yang hendak ditulis ke dalam fail teks tersebut, maka pada akhir pernyataan mestilah mempunyai pernyataan yang akan menutup fail tersebut. Seperti dalam Rajah 3.2.3 pada 3, kita menggunakan pernyataan `fclose($f);`

Apabila kita menyemak isi kandungan fail teks "LogMasuk.txt" dengan membukanya melalui perisian *Notepad*, maka kandungannya adalah seperti yang ditunjukkan dalam contoh output dalam Rajah 3.2.4.



Rajah 3.2.4 Paparan output fail `LogMasuk.txt`

Tahukah Anda



Penyataan *if* diperlukan supaya sekiranya pengguna menekan butang "Masuk", atur cara akan melaksanakan pernyataan seperti membuka fail, menulis teks sebagai kandungan fail teks dan seterusnya menutup fail teks tersebut. Teks yang ditulis dalam fail teks tersebut diwakili dengan pemboleh ubah seperti '\$nama', '\$tarikh' dan '\$log'.

Objektif: Menghasilkan satu fail teks dan menyimpan data ke dalam fail tersebut.

Katakan anda ingin merekodkan nama pengguna yang telah mengunjungi laman web sekolah anda. Langkah-langkah yang perlu dilaksanakan ialah:

- 1 Tulis atur cara HTML mudah bagi menghasilkan satu dokumen HTML (laman web) seperti yang ditunjukkan pada Rajah 3.2.5.
- 2 Pastikan terdapat satu medan input untuk pengguna memasukkan nama mereka semasa melayari laman web. Pengguna juga perlu diminta untuk memasukkan nama mereka.
- 3 Pastikan juga terdapat satu butang "Masuk" yang akan mengambil nama dalam medan input dan menyimpan nama tersebut ke dalam fail teks dalam komputer pelayan. Nama pengguna perlu disimpan dalam pelayan untuk tujuan semakan pada masa akan datang.
- 4 Setelah butang ditekan, nama pengguna hendaklah disimpan dalam satu fail teks.

Nota: Berdasarkan nama-nama yang tersimpan dalam fail teks tersebut, kita juga dapat mengetahui bilangan pengguna yang telah mengunjungi laman web tersebut.

- 5 Semak fail teks tersebut dengan membukanya semula dan memastikan nama-nama pengunjung laman web sekolah anda telah berada dalam fail tersebut.



Rajah 3.2.5 Contoh laman web yang akan dihasilkan

Dalam laman web yang ditunjukkan pada Rajah 3.2.5, kita akan menggunakan objek borang (*form*). Objek borang merupakan objek yang penting dalam laman web kerana kita memerlukan elemen seperti kotak teks (*textbox*) untuk memasukkan data nama murid serta elemen butang (*button*) untuk tindakan yang akan dilakukan setelah maklumat dimasukkan. Dalam hal ini, setelah butang "Masuk" diklik, pengguna akan dihalakan ke dalam objek borang (*form*) yang sama. Perkara ini boleh kita kenal pasti pada pernyataan `<form method='POST'>` yang atribut "action" tidak dinyatakan dalam pernyataan tersebut. Rajah 3.2.6 menunjukkan atur cara HTML menggunakan objek borang.

```

<html>
  <head><title>Write to a text file</title></head>
  <body>
    <h1>Selamat Datang</h1>
    <h1>Ke Sekolah Menengah Bistari Jaya</h1>
    <form method='POST'> 1
      <p>Masukkan Nama Anda
      <input type = 'text' name='namapengguna'></p>
      <input type = 'submit' name='Submit' value='Masuk'>
    </form>
    <?php
      if (isset($_POST["Submit"])) 2
      {
        $f = fopen("LogMasuk.txt", "a"); 3
        $nama = $_POST['namapengguna']; 4
        $tarikh = date('d/m/Y h:i:s a', time()); 5
        $log = $nama.":". $tarikh.PHP_EOL; 6
        fwrite($f, $log); 7
        fclose($f); 8
      }
    ?>
  </body>
</html>

```

Rajah 3.2.6 Contoh atur cara yang menggunakan objek borang

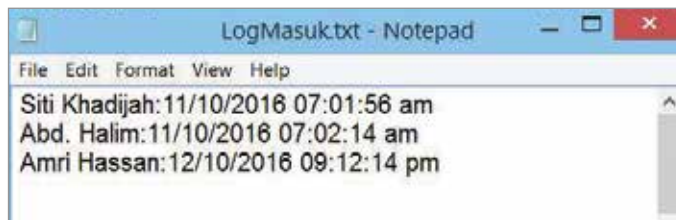
Penerangan:

- 1 Gunakan objek borang yang mempunyai elemen kotak teks dan butang tindakan.
- 2 Menyemak sama ada butang "Masuk" telah ditekan.
- 3 Membuka fail teks dengan mod capaian "a" (*append*) yang bertujuan menambah isi kandungan fail teks tersebut.
- 4 Mengumpukkan nilai daripada kotak teks ke dalam pemboleh ubah '\$nama'.
- 5 Mengumpukkan nilai daripada penggunaan fungsi 'date' ke dalam pemboleh ubah '\$tarikh'.
- 6 Mengumpukkan nilai cantuman daripada pemboleh ubah '\$nama' dan pemboleh ubah '\$tarikh' teks ke dalam pemboleh ubah '\$log'.
- 7 Mencatatkan nilai pemboleh ubah '\$log' ke dalam fail teks.
- 8 Menutup fail teks yang telah dibuka.

Hasil pelaksanaan laman web ini ialah satu fail teks yang bernama "LogMasuk.txt" akan terhasil dan apabila kita membuka fail tersebut dengan menggunakan perisian *Notepad*, kita akan mendapati isi kandungan fail tersebut tercatat **nama pengguna** berserta **tarikh** dan **masa** ketika laman web ini dilawati. Contoh fail teks LogMasuk.txt ditunjukkan dalam Rajah 3.2.7.

Fungsi date digunakan bagi mengubah format nilai tarikh yang diperolehi dari panggilan fungsi time() dimana:

- d : memberikan nilai hari bagi bulan tertentu dari 01 hingga 31
 - m : memberikan nilai nombor bulan tertentu dari 01 hingga 12
 - Y : memberikan nilai tahun tertentu sebanyak 4 digit
 - h : memberikan nilai masa dalam jam dari 01 hingga 12
 - i : memberikan nilai masa dalam minit dari 1 hingga 59
 - a : memberikan nilai masa dalam huruf kecil sama ada 'am' atau 'pm'
- Format lain yang boleh digunakan seperti date('D/M/Y H:i:S A', time())



Rajah 3.2.7 Contoh fail teks "LogMasuk.txt"

AKTIVITI

3

Kerja Kumpulan

Menghasilkan Fail Teks dan Menyimpan Data

Objektif: Menghasilkan fail teks dan menyimpan data ke dalam fail teks

Anda dan rakan anda merupakan pengawas sekolah. Pada setiap hari Isnin dan Rabu anda ditugaskan untuk menulis nama murid yang datang lewat ke sekolah. Nama di bawah merupakan murid yang datang lewat pada hari Isnin dan Rabu.

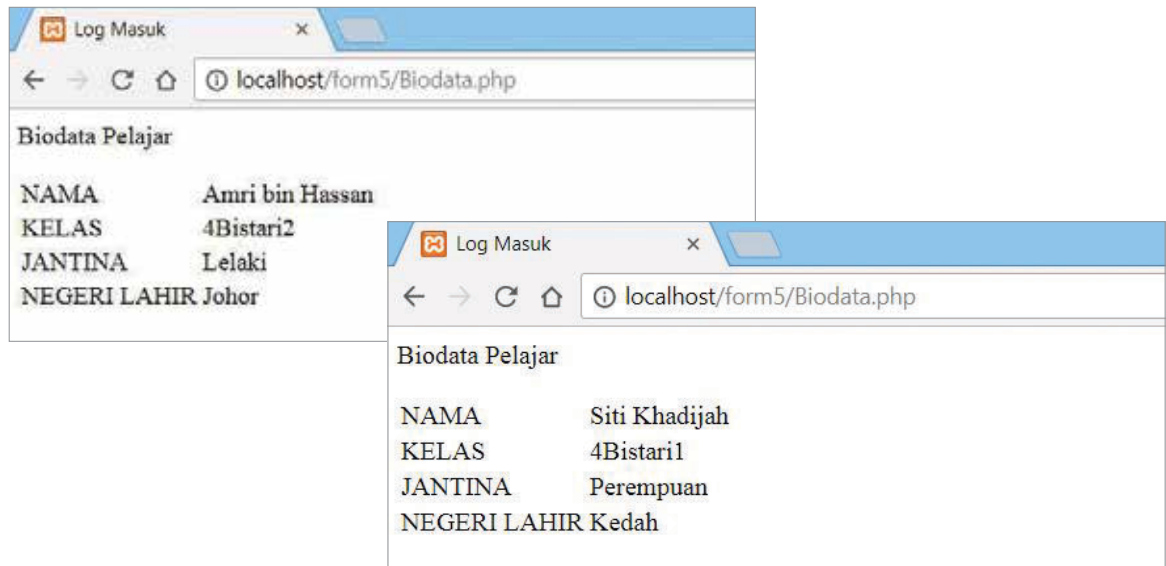
Isnin	Rabu
Raja	Kumutha
Nadia	Jason
Fadzliana	Hartini
Embong	Rukumani

Berdasarkan maklumat di atas, bagaimanakah cara untuk anda merekodkan nama murid yang datang lewat ke sekolah dengan menghasilkan fail teks? Melalui data yang disimpan, bolehkah anda mengenal pasti nama murid yang paling kerap datang lewat ke sekolah pada setiap bulan?



3.2.3 Penggunaan Bahasa Penskripan Pelayan untuk Membina Laman Web yang Unik bagi Setiap Pengguna

Sesuatu laman web kelihatan unik bagi setiap pengguna apabila laman web memaparkan maklumat tentang pengguna tersebut sahaja seperti biodata murid, maklumat terperinci buku, transaksi rekod kerja harian dan lain-lain. Rajah 3.2.8 menunjukkan laman web unik bagi setiap pengguna secara ringkas.



Rajah 3.2.8 Laman web unik bagi setiap pengguna

PROJEK

1

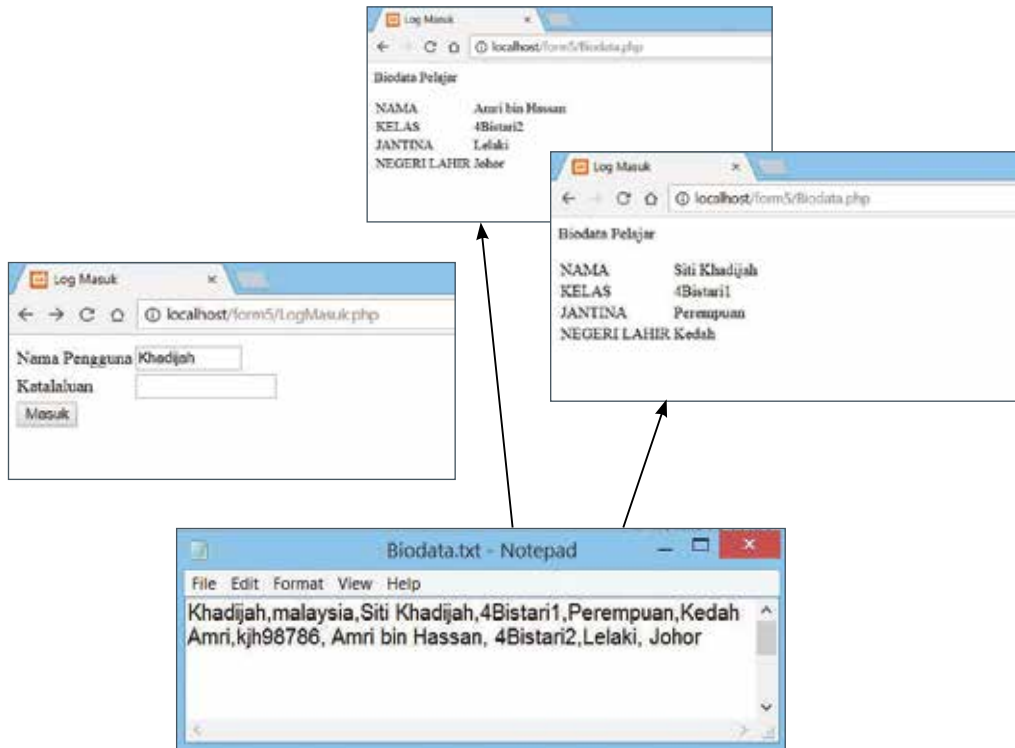
Kerja Individu

Membangunkan Laman Web

Membangunkan laman web berdasarkan senario atau bidang kerja dengan menggunakan bahasa penskripan pelayan.

Situasi:

Katakan kita ingin membina laman web yang akan memaparkan maklumat murid yang log masuk ke dalam sistem maklumat murid. Pertama sekali, murid dikehendaki memasukkan nama pengguna dan kata laluan dalam laman web log masuk. Sekiranya murid berjaya masuk dalam laman web log masuk, seterusnya maklumat biodata murid akan dipaparkan berdasarkan data yang telah disimpan terlebih dahulu di dalam fail teks. Sementara waktu, kita beranggapan bahawa fail teks bertindak seperti fail pangkalan data yang akan menyimpan maklumat seperti nama pengguna, kata laluan, nama murid, kelas, jantina serta negeri lahir. Ringkasan bagi projek ini seperti yang ditunjukkan pada Rajah 3.2.9.



Rajah 3.2.9 Ringkasan laman web yang ingin dibangunkan

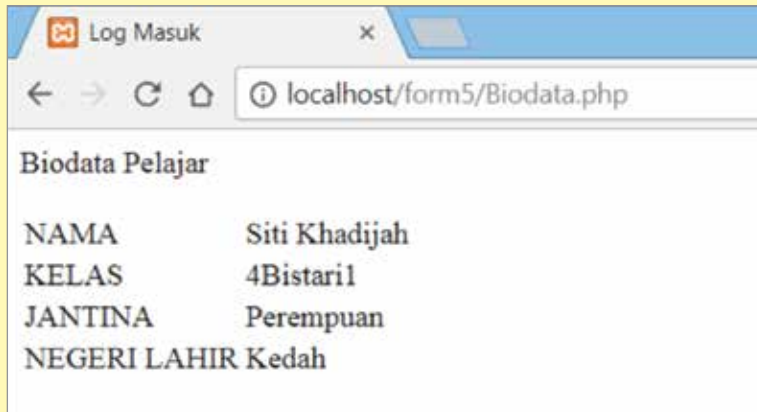
Fail teks bernama Biodata.txt hendaklah disediakan terlebih dahulu dengan menggunakan perisian *Notepad*. Fail teks ini mengandungi maklumat nama pengguna, katalaluan, nama murid, kelas, jantina dan negeri kelahiran. Setiap baris dalam fail tersebut mempunyai 6 item data dan dipisahkan dengan simbol koma. Atur cara bagi laman web LogMasuk.php akan dibangunkan bagi membenarkan pengguna memasukkan data nama pengguna dan katalaluan.

Atur cara bagi laman web Biodata.php akan dibangunkan bagi mencapai data yang telah disimpan dalam fail teks Biodata.txt. Sekiranya nama pengguna dan katalaluan yang dimasukkan berpadanan dengan data yang ada dalam fail teks tersebut maka maklumat lain seperti nama murid, kelas, jantina serta negeri kelahiran akan dipaparkan.

Rajah 3.2.10 menunjukkan atur cara bagi laman web Biodata.php dan Rajah 3.2.11 pula adalah paparan output bagi laman web tersebut.

```
<html>
  <head>
    <title>Log Masuk</title>
    <?php
      $nama = $_POST['namapengguna'];
      $katalaluan = $_POST['katalaluan'];
      $jumpa = False;
    ?>
  </head>
  <body>
    <p>Biodata Murid</p>
```





Rajah 3.2.11 Contoh paparan output bagi laman web Biodata.php

Penerangan:

Gunakan objek borang yang mempunyai elemen kotak teks dan butang tindakan.

Simbol	Penerangan
1	Pelaksanaan pernyataan ulangan membaca kandungan fail teks sehingga penghujung fail teks.
2	<pre>\$medan = explode(',', fgets(\$f));</pre> Pernyataan bertujuan untuk memisahkan teks dengan simbol koma yang dibaca daripada fail teks. Contoh kandungan fail teks ada simbol koma. <i>Khadijah, Malaysia, Siti Khadijah, 4Bistari1, Perempuan, Kedah</i> Bilangan pemboleh ubah adalah sebanyak bilangan simbol (koma + 1).
3	Umpukkan nilai yang telah dipisahkan ke dalam pemboleh ubah yang berkenaan seperti <code>\$user</code> , <code>\$pass</code> , <code>\$namapenuh</code> , <code>\$kelas</code> , <code>\$jantina</code> dan <code>\$negeri</code> .
4	Pernyataan bertujuan untuk membandingkan (<i>string compare</i>) nilai pemboleh ubah <code>\$nama</code> dengan pemboleh ubah <code>\$user</code> .
5	Pernyataan bertujuan untuk membandingkan (<i>string compare</i>) nilai pemboleh ubah <code>\$katalaluan</code> dengan pemboleh ubah <code>\$pass</code> . Sekiranya teks bagi kedua-dua pemboleh ubah adalah sama maka hasilnya ialah 0.

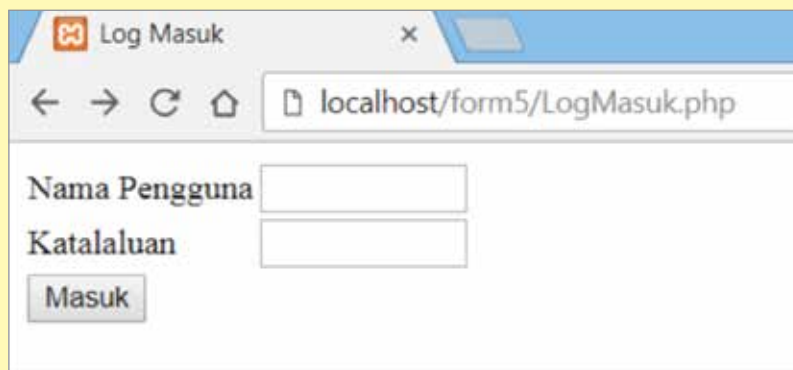
Rajah 3.2.12 menunjukkan atur cara skrip sebelah klien bagi laman web LogMasuk.php dan Rajah 3.2.13 pula ialah paparan output bagi laman web tersebut.

```

<html>
  <head>
    <title>Log Masuk</title>
  </head>
  <body>
    <form action = "Biodata.php" method = "POST"> 1
      <table>
        <tr>
          <td>Nama Pengguna</td>
          <td><input name = "namapengguna" type = "text" size = "10"> 2
        </td>
        </tr>
        <tr>
          <td>Katalaluan</td>
          <td><input name = "katalaluan" type = "Password" size = "15"> 3
        </td>
        </tr>
        <tr>
          <td><input Type = "submit" Name = "submit" Value = "Masuk"> 4
        </td>
        </tr>
      </table>
    </form>
  </body>
</html>

```

Rajah 3.2.12 Atur cara skrip sebelah klien bagi laman web LogMasuk.php



Rajah 3.2.13 Contoh paparan output bagi laman web LogMasuk.php

Kandungan fail teks boleh juga terdiri daripada nombor, teks dan kombinasi nombor dan teks. Contohnya: Ali,24,32,48, Matematik,3.20.



Penerangan:

Simbol	Penerangan
1	Gunakan objek borang (<i>form</i>).
2	Objek borang yang mempunyai elemen kotak teks yang bernama 'namapengguna'.
3	Objek borang yang mempunyai elemen kotak teks yang bernama 'katalaluan'. Nota: Kotak teks ini akan memaparkan simbol '*' apabila pengguna memasukkan kata laluan.
4	Menyediakan butang tindakan yang bernama 'submit'.

AKTIVITI

4

Kerja Individu

Membina Laman Web Unik

Ah Seng memiliki sebuah pasar raya di bandar Jengka. Untuk mendapatkan lebih banyak pelanggan, Ah Seng ingin membina laman web untuk pasar rayanya agar lebih dikenali oleh orang ramai. Laman web yang dicadangkan berupaya untuk merekodkan maklumat pengguna yang mengunjungi laman web tersebut dan setiap pengguna mempunyai kata laluan. Kemudian, setelah pengguna berjaya memasuki laman web tersebut, pengguna dapat melihat barang-barang yang ada di pasar rayanya.

Berdasarkan situasi ini, bagaimanakah anda, sebagai murid Sains Komputer, ingin membina laman web seperti yang dikehendaki oleh Ah Seng?



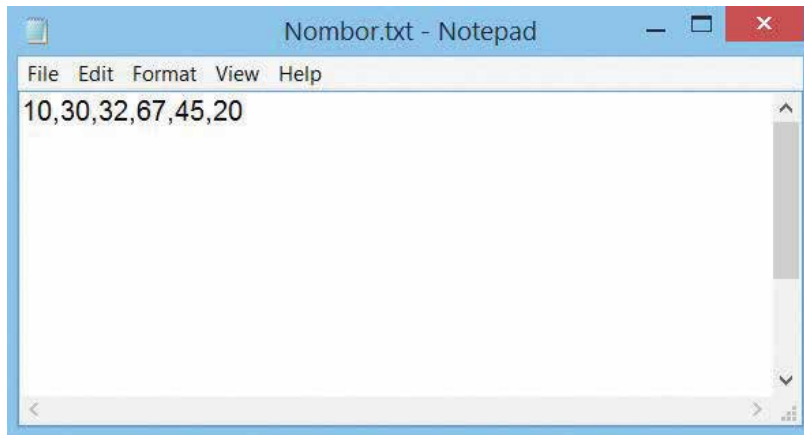
3.2.4

Penggunaan Data yang Diimport daripada Fail Teks untuk Menyelesaikan Masalah

Apabila kita mempunyai banyak data yang perlu diproses/dimanipulasikan, adalah lebih baik sekiranya data-data tersebut disimpan dalam fail (contohnya fail teks). Data-data dalam fail kemudiannya dapat diproses tanpa memerlukan satu persatu data yang diinput oleh pengguna.

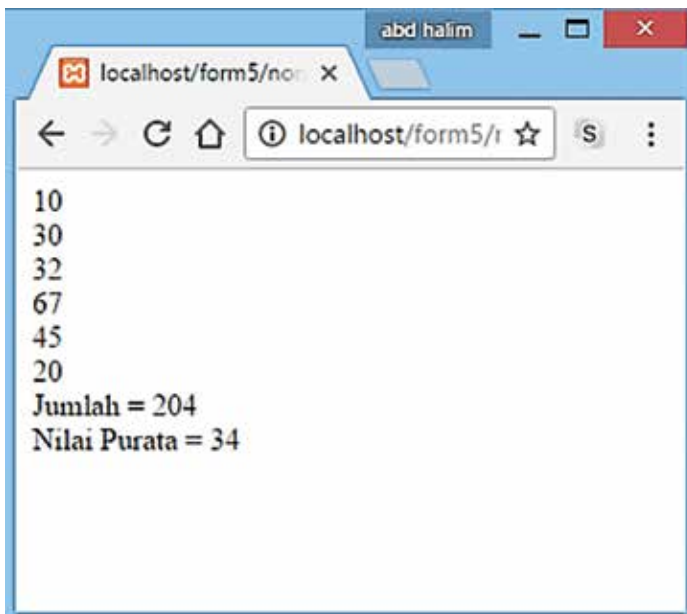
Katakan kita mempunyai senarai nombor integer seperti 10, 30, 32, 6, 45 dan 20. Kita boleh menyimpan nombor-nombor ini ke dalam satu fail teks untuk dijadikan sebagai input dalam laman web bagi mencari jumlah, nilai purata dan sebagainya.

Untuk menyimpan nombor-nombor ini, kita akan menggunakan perisian *Notepad* dan disimpan sebagai 'Nombor.txt' seperti yang ditunjukkan dalam Rajah 3.2.14.



Rajah 3.2.14 Kandungan fail teks Nombor.txt

Seterusnya, kita akan membina laman web yang boleh menginput data ini dan memprosesnya bagi mencari jumlah dan nilai purata seperti paparan output yang ditunjukkan dalam Rajah 3.2.15. Rajah 3.2.16 menunjukkan atur cara bagi laman web *Purata.php*.



Rajah 3.2.15 Contoh paparan output laman web pengiraan nilai purata

Tahukah Anda

Fungsi "explode()" pada PHP adalah untuk memisahkan teks menjadi beberapa bahagian teks.

Contohnya:

Teks seperti:

1, Johor, Johor Bahru

boleh dipisahkan menjadi

- 1
 - Johor
 - Johor Bahru
- dan datanya boleh disimpan dalam tatasusunan yang bersaiz tiga elemen iaitu elemen pertama menyimpan nilai '1', elemen kedua menyimpan nilai 'Johor' dan elemen ketiga menyimpan nilai 'Johor Bahru'.

```

<?php
    $f = fopen("Nombor.txt","r"); 1
    while (!feof($f)) 2
    {
        $arrNombor = explode(',',$f); 3
    }
    $bilangan = count($arrNombor); 4
    $jumlah = 0;
    for ($x = 0;$x < $bilangan; $x++) 5
    {
        print $arrNombor[$x]."<br>"; 6
        $jumlah = $jumlah + $arrNombor[$x];
    }
    $purata = $jumlah / $bilangan;
    print "Jumlah = ".$jumlah."<br>";
    print "Nilai Purata = ".$purata."<br>";
    // tutup fail yang telah dibuka
    fclose($f); 7
?> //penamat untuk php

```

Rajah 3.2.16 Atur cara sebelah pelayan bagi laman web Purata.php

Penerangan:

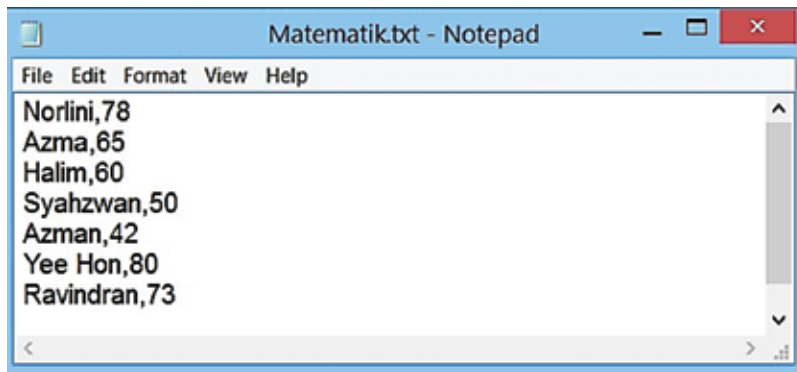
- 1 Membuka fail teks yang bernama 'Nombor.txt' yang mengandungi nombor-nombor.
- 2 Struktur pengulangan untuk proses membaca teks daripada fail teks sehingga penghujung fail.
- 3 Pada setiap baris teks yang dibaca daripada fail teks, data nombor-nombor yang dipisahkan dengan simbol koma akan disimpan dalam tatasusunan yang bernama '\$arrNombor'.
- 4 Penggunaan fungsi 'count' untuk menentukan bilangan elemen bagi tatasusunan '\$arrNombor'.
- 5 Struktur pengulangan sebanyak '\$bilangan' kali ulangan.
- 6 Melaksanakan penjumlahan nombor. Hasil penjumlahan akan disimpan dalam pemboleh ubah '\$jumlah'.
- 7 Menutup fail teks yang telah dibuka.

CONTOH 3

Contoh 3 menerangkan pembinaan laman web yang akan menyenaraikan nama murid, markah serta gred yang diperoleh bagi markah tersebut. Katakan kita telah mempunyai data yang disimpan dalam fail teks yang bernama 'Matematik.txt' seperti yang ditunjukkan dalam Rajah 3.2.17. Melalui data-data ini, kita boleh memprosesnya dengan membangunkan laman web yang boleh mencapai dan seterusnya membaca data-data yang terdapat dalam fail teks tersebut. Andaikan masalah yang ingin kita selesaikan ialah membina laman web yang akan menyenaraikan nama murid, markah serta gred yang diperoleh bagi markah tersebut.



Kataiah penetapan gred markah ditentukan dengan mematuhi syarat seperti yang ditunjukkan dalam Rajah 3.2.18.

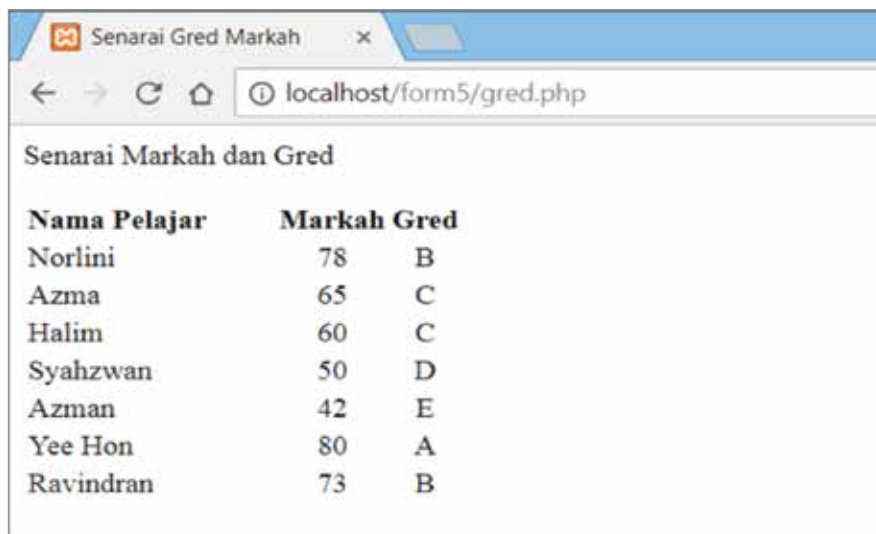


Rajah 3.2.17 Contoh kandungan fail teks "Matematik.txt"

Markah	Gred
80 – 100	A
70 – 79	B
60 – 69	C
50 – 59	D
0 – 49	E

Rajah 3.2.18 Syarat penetapan gred markah

Hasil pelaksanaan yang dikehendaki adalah seperti yang ditunjukkan dalam Rajah 3.2.19.



Rajah 3.2.19 Paparan output bagi laman web gred.php

Rajah 3.2.20 menunjukkan senarai atur cara sebelah pelayan laman web gred.php bagi memproses dan menghasilkan gred markah.

```
<html>
  <head>
    <title>Senarai Gred Markah</title>
  </head>
  <body>
    <p>Senarai Markah dan Gred</p>
    <?php
      //membuka fail untuk membaca kandungan fail
      $f = fopen("Matematik.txt","r");
      $valid = false; // menilaiawalkan dengan nilai false
      print "<table>";
      print "<th align = 'left' width = '130'>Nama Murid</th>";
      print "<th align = 'center'>Markah</th>";
      print "<th align = 'center'>Gred</th>";
      while (!feof($f) )
      {
        $medan = explode (',' ,fgets($f));
        $nama = $medan[0];
        $markah = $medan[1];
        //tentukan gred markah bermula disini
        if ($markah >= 80)
        {
          $gred = "A";
        } elseif ($markah >= 70)
        {
          $gred = "B";
        } elseif ($markah >= 60 )
        {
          $gred = "C";
        } elseif ($markah >= 50)
        {
          $gred = "D";
        } else
        {
          $gred = "E";
        }
        print "<tr>";
        print "<td>".$nama."</td>";
      }
    </?php>
  </body>
</html>
```



```

        print "<td align = 'center'>".$markah."</td>";
        print "<td align = 'center'>".$gred."</td>";
        print "</tr>";
    } // penamat untuk while
    print "</table>";
    fclose($f); // menutup fail yang telah dibuka
?> //penamat untuk php
</body>
</html>

```

Rajah 3.2.20 Atur cara sebelah pelayan bagi laman web gred.php

Penerangan:

- 1 Membuka fail teks yang bernama 'Matematik.txt' yang mengandungi nama murid dan markah.
- 2 Struktur pengulangan untuk proses membaca teks daripada fail teks sehingga penghujung fail.
- 3 Pada setiap baris teks yang dibaca daripada fail teks, data nama dan markah yang dipisahkan dengan simbol koma akan disimpan dalam tatasusunan yang bernama '\$medan'.
- 4 Proses menetapkan nilai gred.
- 5 Menutup fail teks yang telah dibuka.

3.2.5 Menggunakan Data yang Diimport dari Fail Pangkalan Data untuk Menyelesaikan Masalah

Bahagian ini menerangkan bahasa penskripan pelayan PHP yang berinteraksi dengan pangkalan data *MySQL*. Langkah-langkah yang perlu diikuti untuk mencapai data daripada pangkalan data ialah:

- 1 Membuat sambungan ke pangkalan data
- 2 Memilih nama pangkalan data
- 3 Membuat pertanyaan terhadap pangkalan data untuk memperoleh data yang dikehendaki
- 4 Menamatkan sambungan ke pangkalan data

Keempat-empat langkah ini merupakan langkah yang berturutan (*sequential*) yang bermaksud kita tidak boleh memperoleh data yang dikehendaki tanpa melakukan proses penyambungan ke atas pangkalan data terlebih dahulu.



Tutorial *MySQL*
goo.gl/L8veAS

**Tahukah
Anda**



Medan-medan tertentu dalam jadual bagi pangkalan data seperti kata laluan, nombor kad kredit menyimpan data yang telah dienkrapsikan (*encrypted*) ini adalah untuk mengelakkan capaian data secara tidak sah.

Membuat Sambungan ke Pangkalan Data MySQL

Untuk membuat sambungan kepada pangkalan data pelayan, kita akan menggunakan fungsi kendalian "mysqli_connect()". Sintaks fungsi kendalian "mysqli_connect()" adalah seperti yang berikut.

```
$con = mysqli_connect("hoskomputer", "namapengguna", "katalaluan");
```

Contoh:

```
$con = mysqli_connect("localhost", "halim", "1234");
```

Memilih Nama Pangkalan Data MySQL

Seterusnya, apabila sambungan berjaya dibuat, kita akan menggunakan fungsi "mysqli_select_db()" bagi memilih nama pangkalan data. Sintaks fungsi kendalian "mysqli_select_db()" adalah seperti yang berikut.

```
mysqli_select_db("namasambungan", "namapangkalan data");
```

Contoh:

```
mysqli_select_db($con, "dbPelajar");
```

Membuat Pertanyaan terhadap Pangkalan Data untuk Memperoleh Data yang Dikehendaki

Kita akan menggunakan fungsi "mysqli_query" bagi mendapatkan data daripada jadual dalam pangkalan data tersebut. Sintaks fungsi kendalian "mysqli_query" adalah seperti yang berikut.

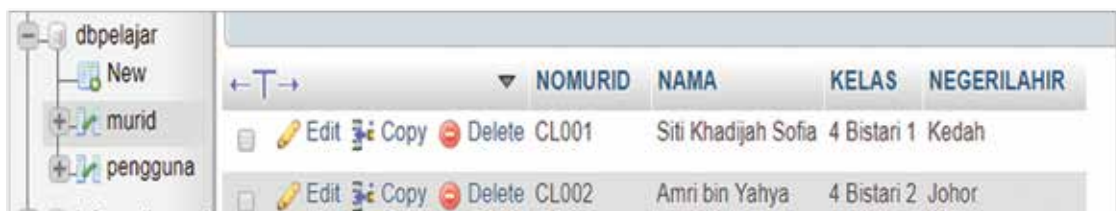
```
$hasil = mysqli_query("namasambungan", "penyataan SQL untuk mendapatkan data");
```

Contoh:

```
$hasil = mysqli_query($con, "SELECT * from MURID");
```

CONTOH 4

Contoh 4 menunjukkan pangkalan data dbPelajar yang mempunyai jadual (*table*) Murid seperti yang berikut:



	NOMURID	NAMA	KELAS	NEGERILAHIR
	CL001	Siti Khadijah Sofia	4 Bistari 1	Kedah
	CL002	Amri bin Yahya	4 Bistari 2	Johor

Atur cara yang ditunjukkan dalam Rajah 3.2.21 adalah untuk membuat pertanyaan terhadap pangkalan data bagi memperoleh data yang dikehendaki. ➔

```

$con = mysqli_connect("localhost","root",""); 1
if (!$con)
{
    die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
}
mysqli_select_db($con,"dbPelajar"); 2

$hasil = mysqli_query($con,"SELECT * from MURID"); 3

mysqli_close(); 4

```

Rajah 3.2.21 Atur cara sebelah pelayan untuk membuat pertanyaan ke atas pangkalan data

CONTOH 5

Contoh 5 menunjukkan pangkalan data dbPelajar yang mempunyai jadual (*table*) Murid. Kita ingin membuat laman web yang akan menyenaraikan semua rekod murid.

Untuk tujuan menyenaraikan rekod murid, kita akan menggunakan fungsi

`mysqli_fetch_array`

Contoh:

```
$row = mysqli_fetch_array($hasil);
```

Di sini hasil penggunaan fungsi `mysqli_fetch_array` merupakan satu set rekod yang disimpan dalam pemboleh ubah tatasusunan yang bernama `row`.

Oleh sebab rekod tersebut disimpan dalam pemboleh ubah `row`, kita akan menggunakan mekanisme ulangan seperti `while` bagi tujuan menyenaraikan rekod murid tersebut.

Atur cara dalam Rajah 3.2.22 ialah contoh penggunaan dalam laman web `Senarai.php` dan akan menghasilkan output seperti yang ditunjukkan dalam Rajah 3.2.23.

```

<html>
<head>
    <title>Senarai Maklumat Murid</title>
</head>
<body>
<p>Senarai Maklumat Murid</p>
<?php

```

```

$con = mysqli_connect("localhost","root","");
if (!$con)
{
    die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
}
mysqli_select_db($con, "dbPelajar");
print "<table border='1'>";
print "<tr>";
print "<th>No Murid</th>";
print "<th>Nama</th>";
print "<th>Kelas</th>";
print "<th>Negeri Kelahiran</th>";
print "</tr>";

$hasil = mysqli_query($con,"SELECT * FROM MURID");
while($row = mysqli_fetch_array($hasil))
{
    $nomurid = $row['NOMURID'];
    $nama = $row['NAMA'];
    $kelas = $row['KELAS'];
    $negeri = $row['NEGERILAHIR'];
    print "<tr>";
    print "<td>.$nopelajar."</td>";
    print "<td>.$nama."</td>";
    print "<td>.$kelas."</td>";
    print "<td>.$negeri."</td>";
    print "</tr>";
}
print "</table>";
mysqli_close($con);
?>
</body>
</html>

```

Mengambil nilai dari tatasusunan *row* dan mengumpukkannya ke dalam pemboleh ubah yang berkenaan.

Rajah 3.2.22 Atur cara sebelah pelayan bagi laman web Senarai.php



No Murid	Nama	Kelas	Negeri Kelahiran
CL001	Siti Khadijah Sofia	4 Bistari 1	Kedah
CL002	Amri bin Yahya	4 Bistari 2	Johor

Rajah 3.2.23 Contoh paparan output bagi laman web Senarai.php

3.2.6 Melakukan Pengesahsahihan (*Validation*) pada Data Input daripada Pengguna

Suatu laman web yang terdapat elemen kemasukan data dalam bentuk borang (*form*) biasanya akan mengandungi medan input seperti kotak teks (*textbox*), butang radio (*radio button*), senarai pilih (*listbox*) serta butang hantar (*submit button*). Bukan semua medan input ini merupakan medan input yang wajib bagi pengguna memasukkan data untuk diproses.

Berdasarkan Rajah 3.2.24, terdapat beberapa medan input yang mesti diisi dengan data dan ada juga beberapa medan input yang dirasakan tidak semestinya perlu diisi dengan data sebelum maklumat murid baharu tersebut disimpan dalam proses pendaftaran murid baharu. Medan input yang bertanda "*" merupakan yang mesti diisi oleh pengguna. Sekiranya medan input ini tidak diisi, maka akan terdapat ralat semasa proses penyimpanan data ke dalam pangkalan data dan proses pendaftaran melalui laman web tersebut akan gagal.

Tahukah
Anda



Salah satu kaedah penggodam untuk mendapatkan data adalah melalui *form*. Pengesahsahihan data yang betul adalah penting untuk melindungi data dalam *form* anda.

Rajah 3.2.24 Laman web dalam bentuk borang yang mengandungi medan input

Tahukah Anda

Pengesahsahihan pada data input dalam komputer pelayan kadangkala tidak mencukupi untuk menjamin pengesahan *form* yang berjaya dan selamat. Pembangun laman web mempertimbangkan cara melakukan pengesahsahihan data input pada komputer pelanggan dengan menggunakan bahasa penskripan pelanggan seperti *JavaScript*. Hal ini bermakna laman web tersebut menjadi lebih responsif. Dengan kaedah ini, *form* tidak akan dihantar sekiranya pengesahsahihan gagal dan pengguna akan mendapat maklum balas dengan segera.

Dalam bab ini, kita akan membina laman web yang akan melakukan pengesahsahihan (*validation*) pada input daripada pengguna untuk mengelakkan terjadinya ralat semasa proses penyimpanan data ke dalam pangkalan data, seperti yang ditunjukkan dalam Rajah 3.2.25. Rajah 3.2.26 menunjukkan paparan output laman web *DaftarPelajar.php*.

Ada beberapa cara pengesahsahihan pada input boleh dilakukan. Antara cara yang lazim dilakukan adalah seperti yang berikut:

- 1 Pastikan kandungan medan input yang wajib diisi dengan data tidak kosong.
- 2 Semak kandungan medan input sama ada mematuhi format yang ditetapkan seperti medan input untuk e-mel atau nombor telefon.
- 3 Semak kandungan medan input untuk angka sama ada mematuhi kriteria tertentu seperti nilai markah antara nilai 0 hingga 100.
- 4 Semak sama ada *Form* telah dihantar kepada komputer pelayan.


```

<html>
<head>
  <title>Daftar Maklumat Murid Baru</title>
  <style>
    error {color: #FF0000;}
  </style>
</head>
<body>
  <?php
    $errName = $errNoMurid = $errEmail = "";
    if ($_SERVER["REQUEST_METHOD"] == "POST") 1
    {
      if (empty($_POST["Nama"])) ←
        $errName = "Sila Masukkan Nama";
      else
        $errName = "";
      if (empty($_POST["NoMurid"])) ← 2
        $errNoMurid = "Sila Masukkan Nombor Murid";
      else
        $errNoMurid = "";
      if (empty($_POST["email"]))
        $errEmail = "Sila Masukkan email"; ←
      else
      {
        $email = $_POST["email"];
        // periksa samaada alamat email SAH
        if (!filter_var($email,FILTER_VALIDATE_EMAIL)) 3
          $errEmail = "Alamat email dimasukkan tidak mengikut
            format";
        else
          $errEmail = "";
      } //penamat untuk else
    } // penamat untuk if
  ?> //penamat untuk php
  <h1>Daftar Maklumat Murid Baru</h1>
  <form method = "POST" action = "<?php echo
    htmlspecialchars($_SERVER["PHP_SELF"]);?>">
    <table>
      <tr>
        <td>No Murid *</td>
        <td><input name = "NoMurid" type = "text" size = "5">
          <span class = "error"><?php echo $errNoMurid;?></span>
        </td>
      </tr>
    </table>
  </tr>
</tr>

```



```

<td>Nama Murid *</td>
<td><input name = "Nama" type = "text" size = "30">
    <span class = "error"><?php echo $errName;?></span>
</td>
</tr>
    <tr>
        <td>Alamat</td>
        <td><textarea name = "Alamat" rows = "4" cols =
            "50"></textarea></td>
    </tr>
    <tr>
        <td>No Telefon</td>
        <td><input name = "Telefon" type = "text" size = "10"></td>
    </tr>
    <tr>
        <td>E-mail *</td>
        <td><input name = "email" type = "text" size = "30">
            <span class = "error"><?php echo $errEmail;?></span>
        </td>
    </tr>

    <tr>
        <td><input type = "submit" Name = "submit" Value =
            "Daftar"></td>
    </tr>
</table>
</form>
</body>
</html>

```

Rajah 3.2.25 Atur cara bagi laman web DaftarPelajar.php

Rajah 3.2.26 Laman web dalam bentuk borang yang mengandungi medan input

Penerangan:

Pernyataan `if ($_SERVER["REQUEST_METHOD"] == "POST")` pada **1** bertujuan untuk memeriksa jenis kiriman borang data.

Pernyataan `if (empty($_POST["Nama"]))` pada **2** bertujuan untuk memeriksa kandungan medan 'Nama'. Sekiranya kandungan medan "nama" adalah kosong, maka pemboleh ubah `errName` akan menyimpan perkataan "Sila Masukkan Nama".

Pernyataan `if (empty($_POST["NoMurid"]))` pada **2** bertujuan untuk memeriksa kandungan medan 'NoMurid'. Sekiranya kandungan medan "NoMurid" adalah kosong, maka pemboleh ubah `errNoMurid` akan menyimpan perkataan "Sila Masukkan Nombor Murid".

Pernyataan `if (empty($_POST["email"]))` pada **2** bertujuan untuk memeriksa kandungan medan 'email'. Sekiranya kandungan medan "email" adalah kosong, maka pemboleh ubah `errEmail` akan menyimpan perkataan "Sila Masukkan email", jika tidak pernyataan `if (!filter_var($email, FILTER_VALIDATE_EMAIL))` pada **3** bertujuan untuk memeriksa sama ada *email* yang dimasukkan mengikut format yang betul. Sekiranya *email* yang dimasukkan pada medan *email* tidak mengikut format yang betul maka pemboleh ubah `errEmail` akan menyimpan perkataan "Alamat *email* dimasukkan tidak mengikut format".

Kandungan nilai pemboleh ubah `errName`, `errNoMurid`, `errEmail` akan dipaparkan dalam bentuk tulisan berwarna merah seperti yang dinyatakan dalam pernyataan

```
<style>
    .error {color: #FF0000;}
</style>
```

Fungsi `htmlspecialchars()` pada baris 36 dalam atur cara `DaftarPelajar.php` akan menukarkan abjad - abjad khas yang telah ditetapkan bagi entiti HTML seperti :

```
& kepada &amp;      " kepada &quot;    ' kepada &# 039;
< kepada &lt;      > kepada &gt;
```

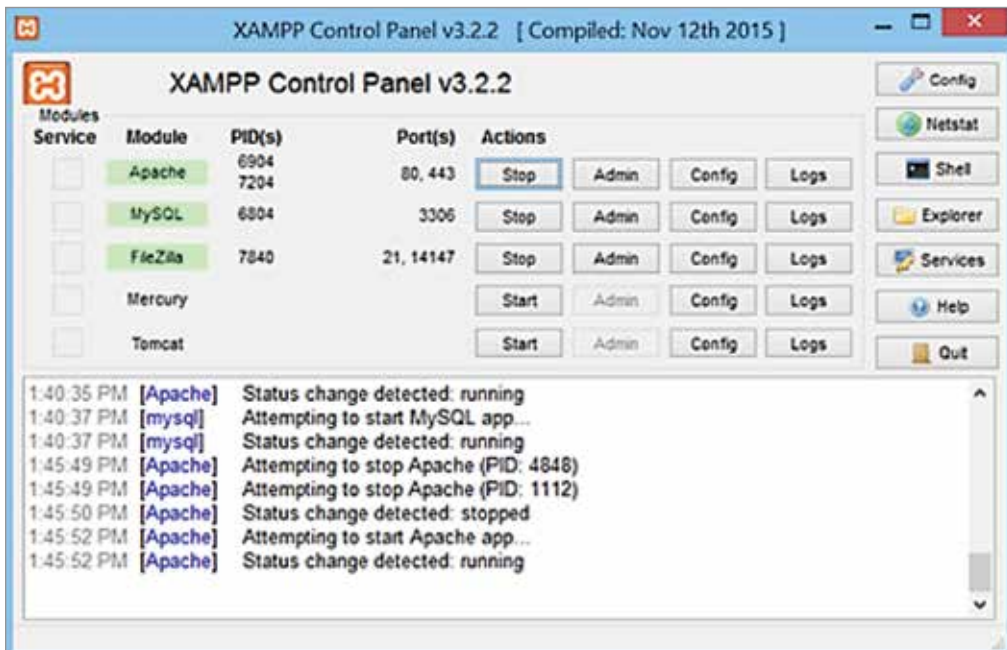
3.2.7 Membina, Mengemas kini, Membuat Sandaran (*Backup*) dan Memulihkan Pangkalan Data

Dewasa ini, banyak aplikasi menggunakan pangkalan data bagi menyimpan data. Dalam bahagian ini kita akan mempelajari bagaimana untuk:

- membina pangkalan data
- mengemas kini pangkalan data
- membuat sandaran (*backup*) pangkalan data
- memulihkan (*restore*) pangkalan data

Membina pangkalan data

Sebelum kita membina pangkalan data, **Web Server Apache** dan aplikasi *database MySQL* hendaklah dilancarkan terlebih dahulu. Kedua-dua aplikasi ini boleh dilancarkan daripada tettingkap **XAMPP Control Panel** seperti yang ditunjukkan pada Rajah 3.2.27(a).



Rajah 3.2.27(a) XAMPP Control Panel

Pada tettingkap XAMPP Control Panel tersebut, klik butang 'Start' pada bahagian Apache dan MySQL. Apabila bahagian Apache dan MySQL menunjukkan warna hijau, langkah seterusnya ialah kita membuka pelayar (*browser*) dan menaip `http://localhost/dashboard` pada bar alamat di pelayar tersebut. Sekiranya tiada sebarang masalah, pelayar web kita akan memaparkan tettingkap seperti Rajah 3.2.27(b).

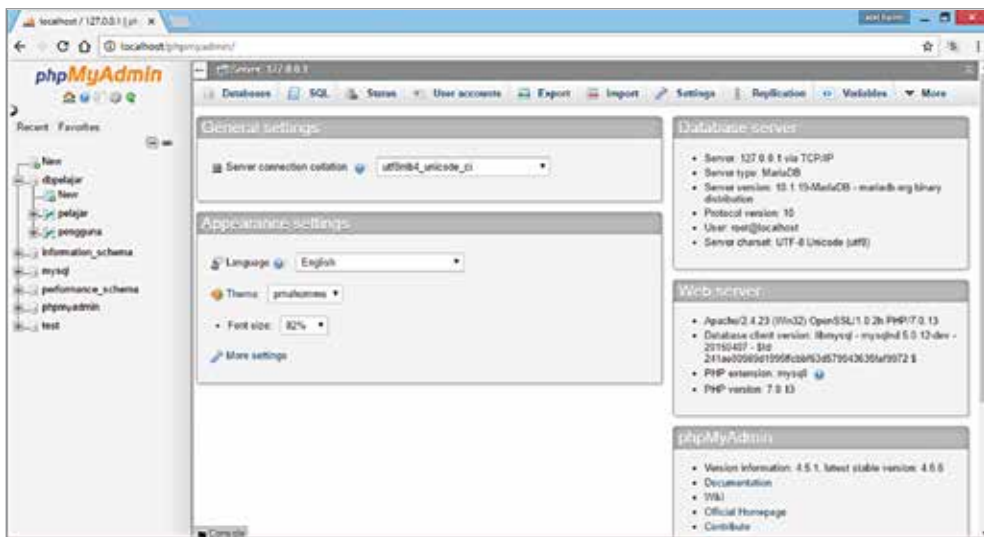


Rajah 3.2.27(b) Tettingkap XAMPP Control Panel

Untuk membina pangkalan data *MySQL*, kita akan menggunakan aplikasi yang ada dalam XAMPP tersebut, iaitu "phpMyAdmin". "phpMyAdmin" ialah aplikasi GUI (*Graphical User Interface*) untuk *MySQL* yang berasaskan web yang akan kita gunakan untuk :

- membina pangkalan data
- mengemas kini pangkalan data
- membuat sandaran (*backup*)
- memulihkan (*restore*) pangkalan data

Setelah kita klik pada bahagian "phpMyAdmin", aplikasi ini akan kita gunakan untuk membina pangkalan data seperti yang ditunjukkan dalam Rajah 3.2.27(c).



Rajah 3.2.27(c) Membina pangkalan data

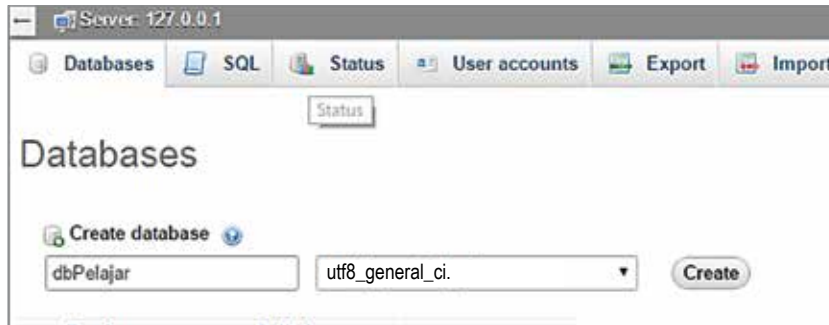
Langkah-langkah:

- 1 Untuk membuat pangkalan data baharu, klik menu *Databases* pada bahagian kiri atas laman web, seperti yang ditunjukkan dalam Rajah 3.2.28.



Rajah 3.2.28 Membuat pangkalan data baharu

- 2 Pada bahagian 'Databases' ini, kita akan memasukkan nama pangkalan data (katakan "dbPelajar") yang ingin dibina di dalam ruangan "Create database". Rajah 3.2.29 menunjukkan tempat untuk memasukkan nama bagi pangkalan data.



Rajah 3.2.29 Masukkan nama pangkalan data

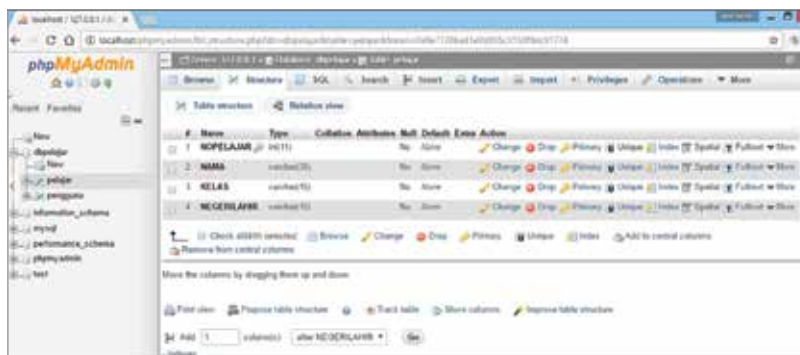
- 3 Setelah memberikan nama pangkalan data, pilih ruangan "Collation". *Collation* mengandungi piawaian (*standard*) penyusunan huruf, angka dalam jadual sesuatu pangkalan data. Pilih *utf8_general_ci*.
- 4 Klik butang 'Create' untuk membina pangkalan data baharu.

Mengemas kini Pangkalan Data

Dari semasa ke semasa kadangkala adanya keperluan untuk kita mengubah atau menambah medan dalam jadual (*table*) kita bagi sesuatu Pangkalan Data. Kita boleh melakukan perkara ini dengan menggunakan "phpMyAdmin". Rajah 3.2.30 menunjukkan paparan skrin untuk melakukan kemas kini medan dalam jadual.

Yang berikut ialah langkah-langkah untuk menambah atau mengubah medan dalam jadual:

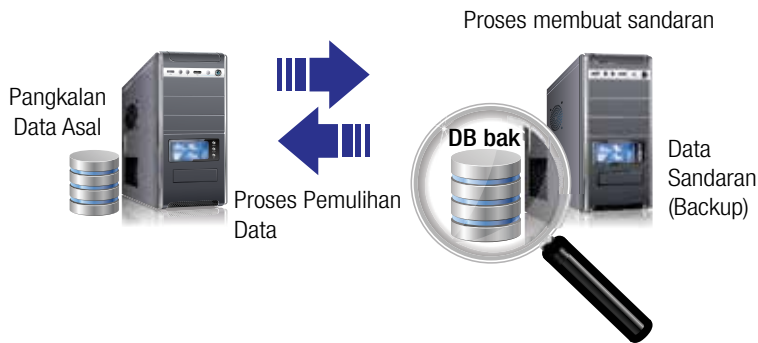
- 1 *Login* terlebih dahulu dalam halaman "phpMyAdmin" anda.
- 2 Klik pangkalan data yang ingin kita kemas kini, katakan pangkalan data tersebut ialah "dbPelajar".
- 3 Klik menu 'Structure'.
- 4 Untuk mengubah nama medan, kita klik pada bahagian 'Change' manakala untuk menghapuskan nama medan, kita klik pada bahagian 'Drop'.



Rajah 3.2.30 Menambah atau mengubah medan dalam jadual

Membuat sandaran (*backup*) pangkalan data

Dewasa ini, banyak aplikasi menggunakan pangkalan data bagi menyimpan data. Disebabkan terlalu banyak kebergantungan aplikasi yang menggunakan pangkalan data, maka membuat sandaran (*backup*) data merupakan aspek yang penting bagi menjamin keselamatan data tetap dipelihara sekiranya berlaku perkara-perkara yang boleh menyebabkan kerosakan atau kehilangan data. Rajah 3.2.31 menunjukkan proses membuat sandaran dan pemulihan data.



Rajah 3.2.31 Proses membuat sandaran dan pemulihan data

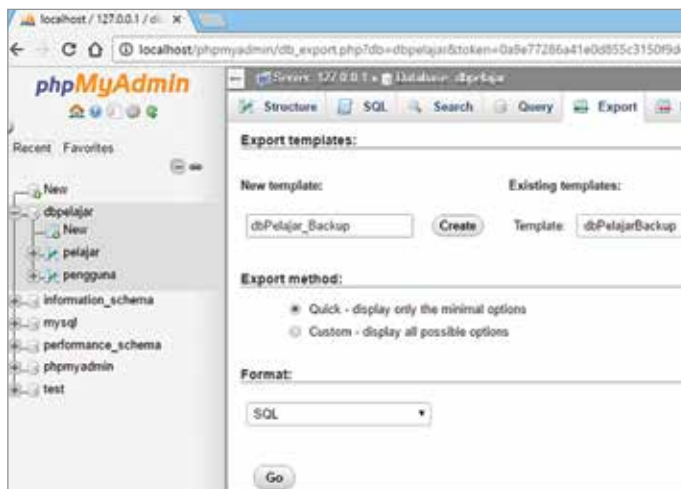
Tahukah
Anda



Memiliki sandaran data bagi apa-apa jua aplikasi juga tidak akan dianggap sempurna sekiranya tiada kemudahan pemulihan (*restore*) ke atas pangkalan data. Kedua-dua kemudahan ini, iaitu sandaran (*backup*) dan pemulihan (*restore*) merupakan salah satu pelengkap bagi menjamin keselamatan data.

Cara Membuat Data Sandaran (Backup) dengan menggunakan "phpMyAdmin"

- 1 *Login* terlebih dahulu dalam halaman "phpMyAdmin" anda.
- 2 Klik pangkalan data yang ingin kita buat sandaran. Katakan pangkalan data tersebut ialah "dbPelajar".
- 3 Klik menu "Export".
- 4 Taip nama fail sandaran yang ingin dicipta pada ruangan 'New Template'. Katakan nama fail tersebut ialah 'dbPelajarBackup'. Rajah 3.2.32 menunjukkan skrin paparan bagi membuat fail sandaran.



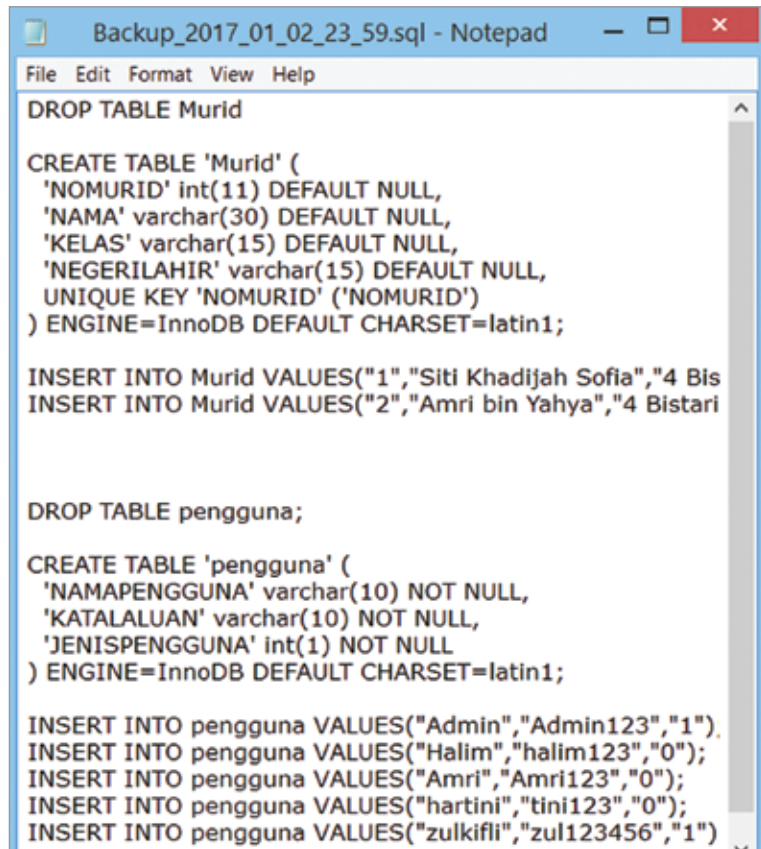
Rajah 3.2.32 Membuat fail sandaran

- 5 Klik butang 'Go'.

Tahukah Anda

Selalunya pentadbir sistem akan membuat fail sandaran pangkalan data dengan menyimpan nama fail sandaran tersebut dengan nama 'Backup' diikuti dengan tarikh dan masa proses tersebut dilakukan. Contoh nama fail sandaran seperti 'Backup-dbPelajar_2016-12-31_08_45_11.sql' yang bermaksud pangkalan data telah dibuat sandaran pada 31 Disember 2016 pada masa 8.45 pagi. Kaedah memberikan nama fail sandaran sedemikian akan memudahkan kita memilih fail sandaran yang terkini semasa proses pemulihan data (*Restore*) nanti.

Pangkalan data telah berjaya dibuat sandaran apabila fail sandaran yang bernama 'dbPelajarBackup.sql' dicipta. Contoh isi kandungan fail tersebut adalah seperti yang ditunjukkan dalam Rajah 3.2.33.



```
Backup_2017_01_02_23_59.sql - Notepad
File Edit Format View Help
DROP TABLE Murid

CREATE TABLE 'Murid' (
  'NOMURID' int(11) DEFAULT NULL,
  'NAMA' varchar(30) DEFAULT NULL,
  'KELAS' varchar(15) DEFAULT NULL,
  'NEGERILAHIR' varchar(15) DEFAULT NULL,
  UNIQUE KEY 'NOMURID' ('NOMURID')
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

INSERT INTO Murid VALUES("1","Siti Khadijah Sofia","4 Bis
INSERT INTO Murid VALUES("2","Amri bin Yahya","4 Bistari

DROP TABLE pengguna;

CREATE TABLE 'pengguna' (
  'NAMAPENGGUNA' varchar(10) NOT NULL,
  'KATALALUAN' varchar(10) NOT NULL,
  'JENISPENGGUNA' int(1) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

INSERT INTO pengguna VALUES("Admin","Admin123","1");
INSERT INTO pengguna VALUES("Halim","halim123","0");
INSERT INTO pengguna VALUES("Amri","Amri123","0");
INSERT INTO pengguna VALUES("hartini","tini123","0");
INSERT INTO pengguna VALUES("zulkifli","zul123456","1")
```

Rajah 3.2.33 Contoh isi kandungan fail sandaran

Tahukah Anda

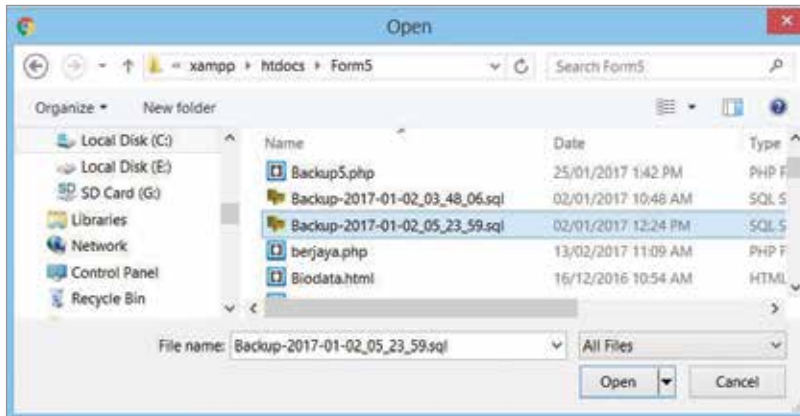
Dalam MySQL kita telahpun menggunakan arahan 'Export' sebagai cara untuk membuat sandaran. Kali ini untuk memulihkan pangkalan data, kita akan gunakan arahan "Import". Ini bermakna sebelum kita hendak memulihkan pangkalan data, kita mestilah terlebih dahulu mempunyai fail sandaran.

Memulihkan (*Restore*) Pangkalan Data

Memulihkan (*Restore*) pangkalan data merupakan aktiviti mengembalikan pangkalan data kepada keadaan asal semula semasa kita membuat fail sandaran (*backup*) yang terkini.

Langkah-Langkah untuk memulihkan pangkalan data.

- 1 *Login* terlebih dahulu dalam halaman "phpMyAdmin" anda.
- 2 Klik nama pangkalan data untuk memulihkan pangkalan data.
- 3 Klik menu "Import".
- 4 Pada ruangan "file to import", klik butang "Choose File" dan pilih nama fail sandaran yang pernah kita buat sebelumnya. Rajah 3.2.34 menunjukkan skrin paparan untuk memilih nama fail sandaran yang pernah dibuat.



Rajah 3.2.34 Memilih fail sandaran yang pernah dibuat

- 5 Cari fail sandaran yang telah dicipta sebelumnya seperti yang ditunjukkan pada Rajah 3.2.35.
- 6 Klik 'GO'.



Rajah 3.2.35 Paparan output setelah berjaya memulihkan (*restore*) pangkalan data dbpelajar

3.2.8 Melaksanakan Beberapa Bentuk Pertanyaan (*Query*) Pangkalan Data

Sesuatu maklumat yang telah disimpan dalam pangkalan data boleh diperoleh semula dengan menggunakan mekanisme Pertanyaan (*Query*). Struktur pertanyaan yang digunakan kepada pangkalan data akan menggunakan format yang telah ditetapkan dan kebanyakan sistem pengurusan pangkalan data menggunakan *Structured Query Language* (SQL).

Penggunaan pertanyaan ini akan mencari serta menyusun data daripada satu atau lebih jadual (*table*). Terdapat beberapa fungsi kendalian dalam pertanyaan ini bagi memudahkan pencarian data berdasarkan kriteria carian yang boleh digunakan.

Terdapat **dua** jenis pertanyaan iaitu:

- (i) Pertanyaan Tindakan – bentuk pertanyaan yang akan menjalankan tugas bersama-sama data yang sedia ada dari pangkalan data. Antara bentuk tindakan adalah seperti mencipta jadual baharu, menambah, mengemas kini atau menghapuskan data dalam jadual.
- (ii) Pertanyaan Memilih – bentuk pertanyaan yang akan mengambil data yang sedia ada dari pangkalan data untuk digunakan. Hasil dari pertanyaan boleh dipaparkan pada skrin, dicetak atau disalin/disimpan.

CONTOH 6

Pertanyaan Tindakan bagi membina jadual bernama 'MURID' dalam pangkalan data adalah seperti yang ditunjukkan dalam Rajah 3.2.36.

```
CREATE TABLE 'MURID' (  
  'NOMURID' int(11) DEFAULT NULL,  
  'NAMA' varchar(30) DEFAULT NULL,  
  'KELAS' varchar(15) DEFAULT NULL,  
  'NEGERILAHIR' varchar(15) DEFAULT NULL,  
  UNIQUE KEY 'NOMURID' ('NOMURID')  
);
```

Rajah 3.2.36 Arahan dalam bentuk Pertanyaan Tindakan

CONTOH 7

Pertanyaan Tindakan bagi membina jadual bernama 'PENGGUNA' dalam pangkalan data adalah seperti yang ditunjukkan dalam Rajah 3.2.37.

```
CREATE TABLE 'PENGGUNA' (  
  'NAMAPENGGUNA' varchar(10) NOT NULL,  
  'KATALALUAN' varchar(10) NOT NULL,  
  'JENISPENGGUNA' int(1) NOT NULL  
)
```

Rajah 3.2.37 Arahan dalam bentuk Pertanyaan Tindakan bagi membina jadual 'PENGGUNA'

CONTOH 8

Pertanyaan Tindakan bagi menambah data dalam jadual 'MURID' adalah seperti yang ditunjukkan dalam Rajah 3.2.38.

```
INSERT INTO MURID VALUES("1","Siti Khadijah Sofia","4 Bistari 1","Kedah");  
INSERT INTO MURID VALUES("2","Amri bin Yahya","4 Bistari 2","Johor");
```

Rajah 3.2.38 Arahan dalam bentuk Pertanyaan Tindakan bagi menambah data dalam Jadual 'MURID'

CONTOH 9

Beberapa Pertanyaan Memilih bagi mendapatkan set data daripada jadual 'MURID' ditunjukkan dalam Rajah 3.2.39.

```
SELECT * FROM MURID;
```

```
SELECT * FROM MURID WHERE NEGERILAHIR = 'KEDAH';
```

```
SELECT NAMA, KELAS FROM MURID;
```

Rajah 3.2.39 Arahan dalam bentuk Pertanyaan Memilih bagi mendapatkan set data daripada jadual 'MURID'

CONTOH 10

Contoh 10 menunjukkan atur cara dalam laman web Negeri.php dengan menggunakan Pertanyaan Memilih bagi menyenaraikan semua maklumat murid yang berasal dari negeri tertentu. Input bagi laman web tersebut mempunyai kotak pilihan yang membolehkan pengguna boleh memilih nama negeri. Rajah 3.2.40 menunjukkan atur cara dalam laman web Negeri.php dan paparan outputnya seperti yang ditunjukkan dalam Rajah 3.2.41.

Penyelesaian:

Negeri.php

```
<html>
  <body>

    <h3>Senarai Nama Murid Berdasarkan Negeri Kelahiran</h3>
    <form action = "PelajarNegeri.php" method = "POST">
    Pilih Negeri :
    <select id = "PilihanNegeri" name = "Negeri">
      <option value = "Johor">Johor</option>
      <option value = "Kedah">Kedah</option>
      <option value = "Kelantan">Kelantan</option>
      <option value = "Labuan">Labuan</option>
      <option value = "Melaka">Melaka</option>
      <option value = "Negeri Sembilan">Negeri Sembilan</option>
      <option value = "Pahang">Pahang</option>
      <option value = "Perak">Perak</option>
      <option value = "Perlis">Perlis</option>
      <option value = "Pulau Pinang">Pulau Pinang</option>
      <option value = "Sabah">Sabah</option>
      <option value = "Sarawak">Sarawak</option>
      <option value = "Selangor">Selangor</option>
      <option value = "Terengganu">Terengganu</option>
      <option value = "Kuala Lumpur">Kuala Lumpur</option>
    </select>
    <input type = "submit" value = "Proses" name = "submit">
  </form>
</body>
</html>
```

Rajah 3.2.40 Atur cara dalam laman web Negeri.php



Output:



Rajah 3.2.41 Paparan output bagi laman web Negeri.php

Rajah 3.2.42 menunjukkan atur cara dalam laman web PelajarNegeri.php bagi tujuan menyenaraikan semua maklumat murid yang berasal dari negeri tertentu yang telah dipilih dari laman web Negeri.php, manakala paparan outputnya seperti yang ditunjukkan dalam Rajah 3.2.43.

PelajarNegeri.php

```
<html>
  <body>
    <?php
      $namaNegeri = $_POST["Negeri"];
      $con = mysqli_connect("localhost","root","");
      if (!$con)
        die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
      mysqli_select_db($con,"dbPelajar");
      print "<h3>Senarai Nama Murid Berdasarkan Negeri Kelahiran</h3>";
      print "<h3>Negeri :".$namaNegeri."</h3>";
      print "<table border = '1'>";
      print "<tr>";
      print "<th>No Murid</th>";
      print "<th>Nama</th>";
      print "<th>Kelas</th>";
      print "</tr>";
      $sql = "SELECT * FROM MURID WHERE NEGERILAHIR = '".$namaNegeri.'";
      echo $sql;
      $result = mysqli_query($con,$sql);
      while ($row = mysqli_fetch_array($result))
      {
        $nomurid = $row['NOMURID'];
        $nama = $row['NAMA'];
        $kelas = $row['KELAS'];
        print "<tr>";
```



```

print "<td>".$nomurid."</td>";
print "<td>".$nama."</td>";
print "<td>".$kelas."</td>";
print "</tr>";
}
print "</table>";
?>
mysqli_close($con);
</body>
<html>

```

Rajah 3.2.42 Atur cara dalam laman web PelajarNegeri.php



Rajah 3.2.43 Paparan output bagi laman web PelajarNegeri.php

3.2.9 Membina Daftar (*Sign Up*) dan Log Masuk (*Login*) dalam Suatu Laman Web

Kebanyakan laman web pada masa ini menghendaki pengguna mendaftar sebagai pengguna/ahli yang sah sebelum dibenarkan untuk mencapai data daripadanya. Setelah pengesahan dilakukan, barulah pengguna dapat melayari laman web tersebut.

Contoh-contoh laman web yang memerlukan pendaftaran (*sign up*) adalah seperti laman web keahlian kelab/persatuan sekolah, laman web syarikat untuk pekerja-pekerja, laman web media sosial dan sebagainya.

Laman web tersebut hanya membenarkan perkongsian data atau maklumat seperti gambar, video, artikel dan lain-lain kepada ahli-ahli berdaftar sahaja. Dalam bahagian ini, kita akan membina laman web daftar (*sign up*) dan seterusnya laman web masuk (*login*) untuk pengguna berdaftar.

CONTOH 11

Contoh 11 menerangkan pembinaan laman web pendaftaran ahli untuk Kelab Catur sekolah. Rajah 3.2.44 menunjukkan atur cara dalam laman web DaftarAhli.php manakala paparan outputnya seperti yang ditunjukkan dalam Rajah 3.2.45. Proses pendaftaran ahli akan dilaksanakan oleh atur cara dalam laman web ProsesDaftar.php seperti yang ditunjukkan dalam Rajah 3.2.46.

Kita akan membina laman web daftar (*sign up*) untuk Kelab Catur Sekolah.

Penyelesaian:

Yang berikut ialah kandungan fail 'DaftarAhli.php'.

```
<html>
  <head>
    <title>Daftar Ahli Baru</title>
  </head>
  <body>
    <form action = "ProsesDaftar.php" method = "POST">
      <table border = "0">
        <tr>
          <td style = "background-color:#00FF00;" align =
            "center">SelamatDatang</td>
          <td style = "background-color: #00FF00;" align = "center">
            Daftar Ahli Baru</td>
        </tr>
        <tr>
          <td><img src = "LogoKelab.png"></td>
          <td width = "60%">
            <table>
              </tr>
              <td>Nama Pengguna</td>
              <td><input name = "namapengguna" size = "10"
                type = "text"></td>
            </tr>
            <tr>
              <td>Katalaluan</td>
              <td><input name = "katalaluan" size = "15"
                type = "password"></td>
            </tr>
            <tr>
              <td>Jenis Keahlian</td>
              </td>
              <select name = "jenis">
                <option value = "Ahli Biasa">Ahli Biasa</option>
                <option value = "Pengerusi">Pengerusi</option>
                <option value = "Pentadbir">Pentadbir</option>
              </select>
            </td>
          </tr>
        </table>
      </table>
    </form>
  </body>
</html>
```



```

        <tr>
            <td><input name = "submit" value = "Daftar"
                type = "submit"></td>
        </tr>
    </table>
</form>
</body>
</html>

```

Rajah 3.2.44 Atur cara dalam laman web DaftarAhli.php

Output:



Rajah 3.2.45 Paparan output bagi laman web Daftar.php

Yang berikut ialah kandungan fail 'ProsesDaftar.php'.

```

<?php
    $con = mysqli_connect("localhost","root","");
    if (!$con)
    {
        die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
    }
    mysqli_select_db($con,"dbPelajar");
    $namapengguna = $_POST['namapengguna'];
    $katalaluan = $_POST['katalaluan'];
    $jenis = $_POST['Ahli'];
    $sql = "INSERT INTO PENGGUNA (NAMAPENGGUNA, KATALALUAN, JENISPENGGUNA)
        VALUES ('$namapengguna', '$katalaluan', '$Ahli)";
    print $sql;
    $result = mysqli_query($con,$sql);
    header('location:Admin.php');
    mysqli_close($con);
?>

```

Rajah 3.2.46 Atur cara dalam laman web ProsesDaftar.php

CONTOH 12

Contoh 12 menerangkan pembinaan laman web log masuk (*sign in*) untuk ahli kelab yang telah berdaftar. Rajah 3.2.47 menunjukkan atur cara dalam laman web LogMasuk.php manakala paparan outputnya seperti yang ditunjukkan di Rajah 3.2.48.

Proses pengesahan menyemak sama ada pengguna telah mendaftar sebagai ahli kelab atau tidak akan dilakukan oleh atur cara dalam laman web ProsesMasuk.php seperti yang ditunjukkan di Rajah 3.2.49. Sekiranya pengguna telah berdaftar sebagai ahli kelab maka laman web Masuk.php akan dipaparkan seperti yang ditunjukkan di Rajah 3.2.49. Rajah 3.2.50 menunjukkan paparan output bagi laman web Masuk.php.

```
<html>
  <head>
    <title>Log Masuk</title>
  </head>
  <body>
    <table border = "0">
      <tr>
        <td style = "background-color: #00FF00;" align = "center">Selamat
          Datang</td>
        <td style = "background-color: #00FF00;" align = "center">Log Masuk</td>
      </tr>
      <tr>
        <td><img src = "LogoKelab.png"></td>
        <td width = "60%">
          <form action = "ProsesMasuk.php" method = "POST">
            <table>
              <tr>
                <td>Nama Pengguna</td>
                <td><input name = "namapengguna" size = "10" type = "text"></td>
              </tr>
              <tr>
                <td>Katalaluan</td>
                <td><input name = "katalaluan" size = "15" type = "password">
                  </td>
              </tr>
              <tr>
                <td><input name = "submit" value = "Masuk" type = "submit"></td>
              </tr>
            </table>
          </form>
        </td>
      </tr>
    </table>
  </body>
</html>
```

Rajah 3.2.47 Atur cara dalam laman web LogMasuk.php

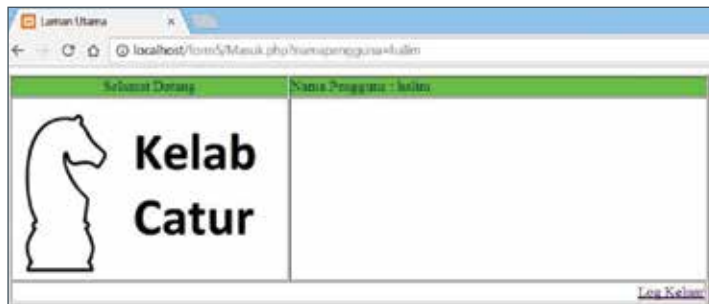


Rajah 3.2.48 Paparan output bagi laman web LogMasuk.php

```
<?php
$con = mysqli_connect("localhost","root","");
if (!$con)
{
    die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
}
mysqli_select_db($con,"dbPelajar");
$namapengguna = $_POST['namapengguna'];
$katalaluan = $_POST['katalaluan'];

$rekod = mysqli_query($con, "SELECT * FROM PENGGUNA where NAMAPENGGUNA =
'$namapengguna' and KATALALUAN = '$katalaluan'");
$hasil = mysqli_num_rows($rekod);
if ($hasil>0)
{
    header("location:Masuk.php?namapengguna = ".$namapengguna);
}
else
{
    header("location:LogMasuk.php"); // kembali ke laman web LogMasuk.php
}
?>
```

Rajah 3.2.49 Atur cara dalam laman web ProsesMasuk.php



Rajah 3.2.50 Paparan output bagi laman web Masuk.php

```

<html>
  <?php
    $nama = $_GET['namapengguna'];
  ?>
  <head>
    <title>Laman Utama</title>
  </head>
  <body>
    <table border = "1">
      <tr>
        <td style = "background-color: #00FF00;" align = "center">Selamat
          Datang</td>
        <td colspan = "2" style = "background-color: #00FF00;" align =
"left">Nama
          Pengguna :<?php print $nama?>
        </td>
      </tr>
      <tr>
        <td width = "40%"><img src = "LogoKelab.png"></td>
        <td colspan = "2" valign = "top"></td>
      </tr>
      <tr>
        <td colspan = "2" align = "right"><a href = "KelabCatur.php">
          Log Keluar</a></td>
      </tr>
    </table>
  </body>
</html>

```

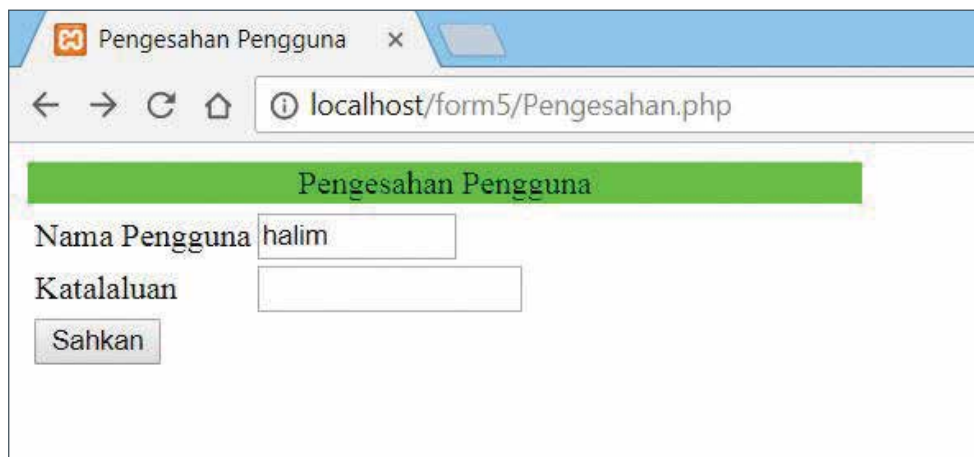
Rajah 3.2.51 Atur cara dalam laman web Masuk.php

3.2.10 Pengesahan Pengguna dan Pengemaskinian Data dalam Pangkalan Data

Pengesahan pengguna (*user verification*) merupakan salah satu ciri keselamatan data dalam pangkalan data yang boleh dicapai oleh pengguna dalam laman web. Kemudahan capaian ini seterusnya akan membataskan pengguna untuk melakukan sesuatu perubahan ke atas pangkalan data. Sebagai contoh peringkat pengguna yang berbeza memberikan hak capaian yang berbeza serta aktiviti yang boleh dilakukan ke atas pangkalan data. Laman web untuk pekerja di sesebuah syarikat hanya membenarkan pekerja yang boleh mencapai pangkalan data tersebut. Untuk membuat laman web yang dapat mengesahkan pengguna tertentu, kita memerlukan satu jadual dalam pangkalan data yang menyimpan nama pengguna, serta kata laluan pengguna tersebut.

Bermula dari laman Pengesahan.php seperti yang ditunjukkan dalam Rajah 3.2.52, pengguna dikehendaki memasukkan nama pengguna dan katalaluan. Proses pengesahan pengguna akan dilaksanakan oleh atur cara laman web Sahkan.php seperti yang ditunjukkan dalam Rajah 3.2.53.

Sekiranya nama pengguna dan katalaluan yang dimasukkan berpadanan dengan rekod yang wujud di dalam jadual bagi pangkalan data maka laman web seperti dalam Rajah 3.2.54(a) akan paparkan. Sekiranya tidak, laman web seperti dalam Rajah 3.2.54(b) akan dipaparkan.



Rajah 3.2.52 Paparan laman web Pengesahan.php

```
<?php
    session_start();
    $_SESSION['PenggunaSah'] = 0;
    $con = mysqli_connect("localhost","root","");
    if (!$con)
    {
        die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
    }
    mysqli_select_db($con,"dbPelajar");
```

```

$namapengguna = $_POST['namapengguna'];
$katalaluan = $_POST['katalaluan'];
$rekod = mysqli_query($con, "SELECT * FROM PENGGUNA where NAMAPENGGUNA =
    '$namapengguna' and KATALALUAN = '$katalaluan'");
$hasil = mysqli_num_rows($rekod);
if($hasil>0)
{
    $_SESSION['PeggunaSah'] = 1;
    header("location:Pengesahan.php");
}
else
{
    $_SESSION['PeggunaSah'] = 0;
    header("location:Pengesahan.php");
}
mysqli_close($con);
?>

```

Rajah 3.2.53 Atur cara dalam laman web Pengesahan.php



Rajah 3.2.54(a) Paparan output bagi laman web Sah.php

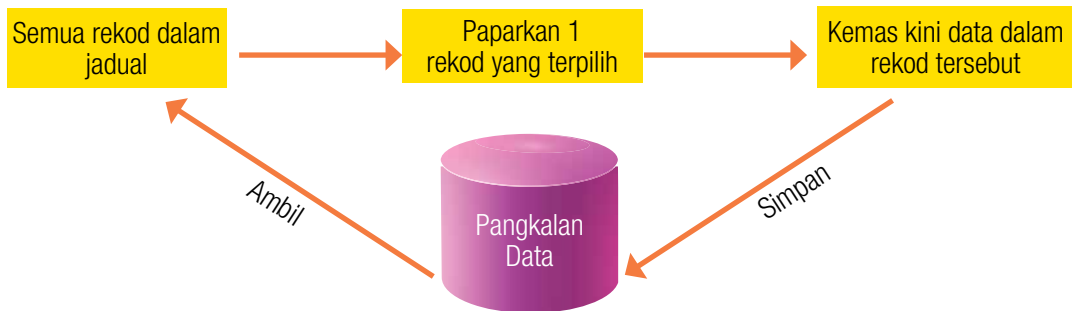


Rajah 3.2.54(b) Paparan output bagi laman web Sah.php



Pengemaskinian Data dalam Pangkalan Data

Untuk mengemas kini data dalam pangkalan data, kita memerlukan pencarian rekod yang hendak dikemaskinikan terlebih dahulu dalam jadual (*table*) bagi pangkalan data. Rajah 3.2.55 menunjukkan proses pengemaskinian data dalam pangkalan data. Bermula dari menyenaraikan terlebih dahulu semua rekod yang ada di dalam jadual, pengguna seterusnya akan memilih rekod yang hendak dikemaskinikan. Setelah pengguna memilih rekod yang hendak dikemas kini, laman web akan memaparkan satu rekod yang telah dipilih dan proses pengemaskinian data dalam pangkalan data akan dilakukan oleh laman web yang seterusnya.



Rajah 3.2.55 Proses pengemaskinian data

CONTOH 13

Contoh 13 menunjukkan atur cara dalam laman web `Senarai.php` yang akan menyenaraikan semua rekod daripada Jadual bagi pangkalan data. Rajah 3.2.56 menunjukkan atur cara dalam laman web `Senarai.php`, manakala paparan outputnya seperti yang ditunjukkan dalam Rajah 3.2.57.

```
<html>
<head>
  <title>Senarai Maklumat Murid</title>
</head>
<body>
  <p>Senarai Maklumat Murid</p>
  <?php
    $con = mysqli_connect("localhost","root","");
    if (!$con)
    {
      die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
    }
    mysqli_select_db($con,"dbPelajar");
```

```

print "<table border='1'>";
print "<tr>";
print "<th>No Murid</th>";
print "<th>Nama</th>";
print "<th>Kelas</th>";
print "<th>Negeri Kelahiran</th>";
print "</tr>";
$hasil = mysqli_query($con, "SELECT * FROM MURID");
while($row = mysqli_fetch_array($hasil))
{
    $lnk = "<a href = 'Kemaskini.php?nomurid=$nomurid'>Kemaskini</a>";
    print "<tr>";
    print "<td>".$nomurid."</td>";
    print "<td>".$nama."</td>";
    print "<td>".$kelas."</td>";
    print "<td>".$negeri."</td>";
    print "<td>".$lnk."</td>";
    print "</tr>";
}
print "</table>";
mysqli_close($con);
?>
</body>
</html>

```

Rajah 3.2.56 Atur cara dalam laman web Senarai.php

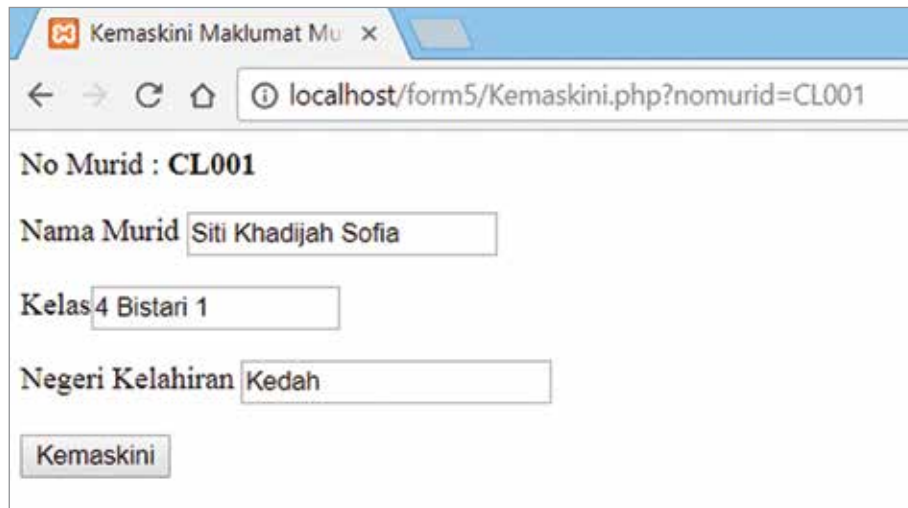


The screenshot shows a web browser window titled "Senarai Maklumat Murid" with the URL "localhost/form5/Senarai.php". The page content includes the heading "Senarai Maklumat Murid Untuk Dikemas kini" and a table with the following data:

No Murid	Nama	Kelas	Negeri Kelahiran	Tindakan
CL001	Siti Khadijah Sofia	4 Bistari 1	Kedah	Kemaskini
CL002	Amri bin Yahya	4 Bistari 2	Johor	Kemaskini

Rajah 3.2.57 Atur cara dalam laman web Senarai.php

Seterusnya, kita akan membina laman web yang bertujuan untuk memaparkan rekod yang telah dipilih untuk tujuan melakukan pengemaskinian data seperti dalam Rajah 3.2.58.



Rajah 3.2.58 Paparan output laman web Kemaskini.php

Rajah 3.2.59 ialah kandungan laman web untuk memaparkan rekod yang telah dipilih.

Proses pengemaskinian rekod akan dilakukan oleh atur cara dalam laman web ProsesKemaskini.php seperti yang ditunjukkan dalam Rajah 3.2.60.

```

<html>
  <head>
    <title>Kemaskini Maklumat Murid</title>
  </head>
  <body>
    <?php
      $noM = $_GET ['nomurid'];
      $con = mysqli_connect ("localhost","root","");
      if (!$con)
      {
        die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
      }
      mysqli_select_db($con,"dbPelajar");
      $sql = "SELECT * FROM MURID WHERE NOMURID ='".$noM."'";
      // contoh $sql = "SELECT * FROM MURID WHERE NOMURID = 'CL001'";
      $result = mysqli_query($con,$sql);
      $row = mysqli_fetch_array($result);
      $nama = htmlspecialchars($row['NAMA'], ENT_QUOTES);
      $kelas = $row['KELAS'];
      $negeri = $row['NEGERILAHIR'];
    ?>

```

```

<form action = "ProsesKemaskini.php" method = "POST">
  <p>No Murid : <b><?php print $noP;?></b></p>
  <p><input type = "hidden" name = "nomurid" value = '<?php print
    $noP;?>'></p>
  <p>Nama Murid
  <input name = "nama" type = "text" value = '<?php print $nama;?>'>
  </p>
  <p>Kelas<input name = "kelas" type = "text" size = "15" value =
    '<?php print $kelas; ?>'> </p>
  <p>Negeri Kelahiran <input name = "negeri" type = "text" size
    = "20" value = '<?php print $negeri;?>'>
  </p>
  <p><input type = "submit" value = "Kemaskini"></p>
</form>
</body>
</html>

```

Rajah 3.2.59 Atur cara bagi laman web Kemaskini.php

```

<?php
  $con = mysqli_connect("localhost","root","");
  if (!$con)
  {
    die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
  }
  mysqli_select_db($con,"dbPelajar");
  $nomurid = $_POST['nomurid'];
  $nama = $_POST['nama'];
  $kelas = $_POST['kelas'];
  $negeri = $_POST['negeri'];
  $sql = "update murid set NAMA = '$nama', KELAS = '$kelas',
    NEGERILAHIR='$negeri' where NOMURID = '$nomurid'";
  $result = mysqli_query($con,$sql);
  header('location:Senarai.php');
?>

```

Rajah 3.2.60 Atur cara bagi laman web ProsesKemaskini.php

LATIHAN PENGUKUHAN 3.2

- 1 Apakah yang dimaksudkan dengan bahasa penskripan pelayan?
- 2 Senaraikan **dua** aplikasi yang menggunakan bahasa penskripan pelayan.
- 3 Mengapakah bahasa penskripan pelayan diperlukan dalam pembangunan web?
- 4 Berikan **dua** operasi asas terhadap fail teks yang biasa digunakan laman web PHP.
- 5 Jelaskan maksud bagi setiap tag di bawah:

```
<form action = "Ucapan.php" method = "POST"> a  
  Nama : <input type = "text" name = "txtNama"> b  
  Umur : <input type = "text" name = "txtUmur"> c  
  <input type = "Submit" value = "Hantar"> d  
</form>
```

- 6 Dalam PHP, kita boleh mencapai data daripada borang (*form*) dengan menggunakan kendalian `$_GET` atau `$_POST`. Apakah perbezaan antara GET dengan POST?
- 7 Dalam satu laman web, terdapat satu borang (*form*) untuk pengguna memasukkan data. Pada pendapat anda, bolehkah terdapat lebih daripada satu borang dalam satu laman web?
- 8 Apakah kegunaan fungsi *explode* dalam PHP?
- 9 Yang berikut ialah contoh arahan MySQL. Apakah nilai dalam Jadual 'TEST' apabila arahan tersebut dilaksanakan?

```
INSERT INTO TEST (a,b,c) VALUES (1,2,3);
```

A	B	C

- 10 Apakah output yang akan dipaparkan pada pelayar (*browser*) sekiranya pengguna memasukkan nilai 17 dan klik butang 'Hantar' berdasarkan kandungan fail PHP yang berikut:

Nama fail : **borang.php**

```
<html>  
<body>  
  <form action = "berjaya.php" method = "post">  
    Age: <input type = "text" name = "umur" />  
    <input type = "submit" value = "Hantar" />  
  </form >  
</body>  
</html>
```

Nama fail: **berjaya.php**

```
<html>  
<body>  
  Saya berumur <?php print $_POST['umur']; ?> tahun.  
</body>  
</html>
```

3.3

STANDARD PEMBELAJARAN

Pada akhir pembelajaran murid dapat

3.3.1 Mengkaji dan merumus prinsip reka bentuk laman web dari sudut kesesuaian pengguna dan tujuan laman web

3.3.2 Mereka bentuk kerangka aplikasi yang akan dibangunkan

3.3.3 Membina laman web mudah dengan menggunakan *Hypertext Markup Language (HTML)* yang mengandungi *frame*, *header*, *paragraph* dan *image*

3.3.4 Menggunakan *Cascading Style Sheets (CSS)* untuk menggayakan *text*, *font*, *background*, *tables*, *borders* dan *position*

3.3.5 Menggunakan bahasa penskripan klien untuk membina laman web pengguna yang interaktif mengandungi fitur berikut:

- (i) Pengesahan data
- (ii) Butang pilihan
- (iii) Kotak *popup*

3.3.6 Menggunakan bahasa penskripan pelayan untuk membina laman web yang boleh mencapai dan mengemas kini data dalam pangkalan data

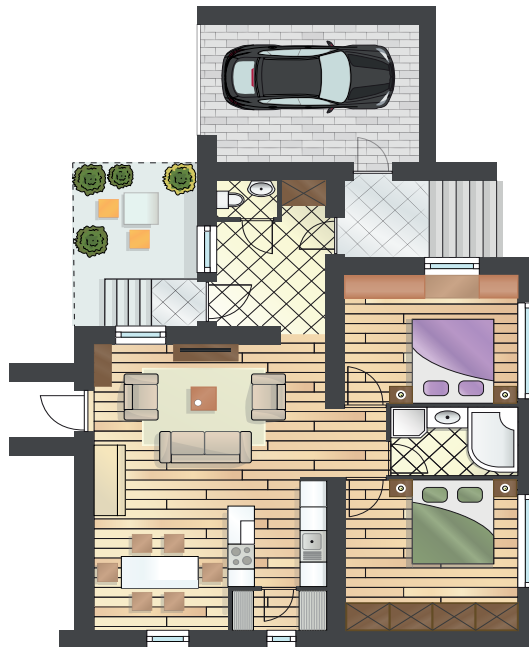
3.3.7 Menghasilkan sebuah laman web interaktif untuk kegunaan pengguna dan pentadbir sistem bagi menyelesaikan masalah



Laman Web Interaktif

3.3.1 Prinsip Reka Bentuk Laman Web

Prinsip merujuk asas yang menjadi dasar kepada perkembangan atau pembinaan sesuatu benda. Reka bentuk pula merujuk susunan atau struktur dalam proses pembinaan sesuatu perkara. Oleh sebab itu, mereka bentuk laman web boleh diibaratkan seperti proses merancang, membina dan mengubah suai serta menata hias kediaman rumah. Setiap ruang yang dibina dalam kediaman memainkan fungsi yang berlainan. Lantaran itu, struktur pembinaan dan corak tatahias bagi setiap ruang juga akan berbeza mengikut kesesuaian fungsinya. Konsep, saiz dan kedudukan setiap ruang kediaman seperti ruang tamu, dapur dan juga bilik tidur akan ditentukan terlebih dahulu semasa fasa perancangan mengikut jenis kediaman yang dibina seperti banglo, kondominium atau rumah teres seperti yang ditunjukkan dalam Rajah 3.3.1.



Rajah 3.3.1 Contoh susun atur kediaman mengikut jenis ruang

Dalam proses mereka bentuk laman web, terdapat beberapa prinsip atau elemen reka bentuk laman web yang perlu diberi perhatian agar laman web yang dibina tidak kelihatan pelik atau memberikan pengalaman yang tidak menyenangkan kepada pengguna. Hal ini dikatakan demikian kerana elemen-elemen tersebut mampu memberikan pengalaman yang berbeza dan mempengaruhi kepuasan pengguna untuk melayari laman web yang dibina.

Antara prinsip asas yang harus diambil kira semasa mereka bentuk laman web adalah seperti yang ditunjukkan dalam Rajah 3.3.2.



10 Top Principles of Effective Web Design
<https://goo.gl/kd1Rzh>



Rajah 3.3.2 Prinsip-prinsip reka bentuk laman web

Jenis Persekitaran Web yang Dibina

Sebelum sesuatu laman web dibina, kita harus memahami persekitaran laman web tersebut. Persekitaran laman web ini merangkumi beberapa faktor teknikal seperti yang berikut:

- Kesesuaian penggunaan pelbagai jenis pelayar
- Kelajuan capaian Internet
- Keperluan perisian sokongan untuk paparan
- Keperluan sistem komputer dan peranti mudah alih yang digunakan oleh pengguna
- Keperluan saiz paparan monitor yang digunakan

Terdapat beberapa jenis pelayar yang boleh digunakan bagi memaparkan laman web, contohnya *Internet Explorer*, *Google Chrome* dan *Firefox* seperti yang ditunjukkan dalam Rajah 3.3.3. Setiap pelayar biasanya mempunyai beberapa versi yang boleh digunakan untuk memaparkan laman web.

Oleh sebab itu, pereka web sewajarnya mempertimbangkan jenis dan versi pelayar yang boleh digunakan untuk memaparkan laman web yang dibina dan seharusnya tidak tertumpu pada satu jenis atau satu versi sahaja.

Tahukah Anda



Google Chrome merupakan pelayar yang mendahului carta populariti pelayar sejak lima tahun lalu.



Statistik Populariti Pelayar
goo.gl/BQULLf



UJI MINDA

Nyatakan pelayar-pelayar lain yang anda tahu selain yang telah dipelajari dalam topik ini.



Internet Explorer



Google Chrome



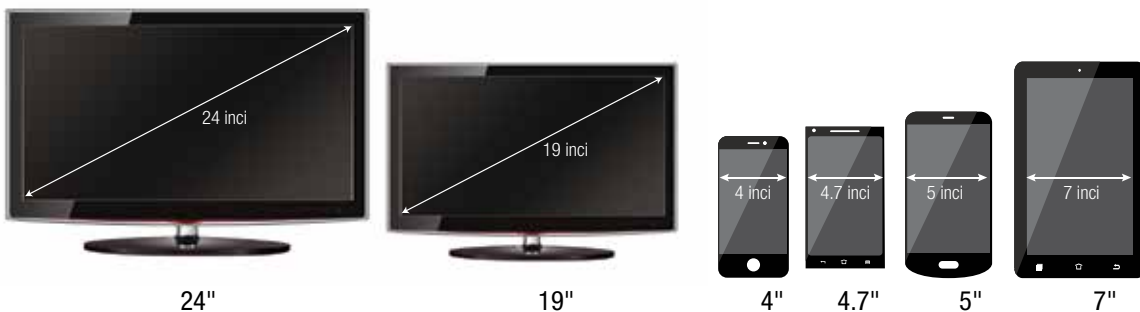
Firefox

Rajah 3.3.3 Jenis pelayar untuk memaparkan laman web

Selain kepelbagaian jenis pelayar, kepelbagaian kaedah capaian Internet yang digunakan bagi setiap pengguna seperti kelajuan capaian juga harus diambil kira. Sebagai contoh, sekiranya laman web dibina untuk tujuan hiburan seperti permainan dalam talian, permainan ini mungkin memerlukan capaian yang lebih laju berbanding dengan laman web yang dibina untuk sumber maklumat seperti berita.

Kadangkala sesetengah laman web memerlukan perisian sokongan khas untuk menyokong paparan seperti animasi, audio atau video. Pereka web perlu peka tentang pemilihan jenis perisian sokongan, misalnya memilih perisian yang diketahui umum dan digunakan secara meluas. Contohnya, bagi animasi, perisian sokongan yang biasa digunakan adalah seperti *Flash*.

Kepelbagaian saiz skrin seperti yang ditunjukkan dalam Rajah 3.3.4 juga tidak boleh dipandang remeh. Hal ini dikatakan demikian kerana saiz paparan skrin akan memberikan impak secara terus pada kualiti paparan web berdasarkan saiz resolusi yang digunakan. Bagi skrin berjenis LCD (*liquid crystal display*), saiz resolusi diukur berdasarkan saiz skrin paparan. Contohnya, skrin monitor bersaiz 19 inci biasanya mempunyai saiz resolusi 1440×900 manakala skrin bersaiz 22 inci mempunyai saiz resolusi 1680×1050 . Semakin tinggi bilangan resolusi, bermakna semakin tinggi bilangan piksel dan semakin jelas dan cantik kualiti paparan. Oleh itu, pereka web harus memastikan laman web yang dipaparkan perlu fleksibel selari dengan kepelbagaian saiz skrin yang tersedia untuk digunakan oleh pengguna.



Rajah 3.3.4 Contoh beberapa saiz paparan skrin dalam inci

Penekanan Maklumat dalam Web

Prinsip penekanan merujuk maklumat atau kandungan yang ingin diberikan perhatian dalam laman web tersebut. Dalam setiap laman web yang dibina, pereka web perlu menentukan maklumat atau kandungan penting yang harus diberi penekanan dan teknik atau kaedah yang sesuai yang perlu digunakan untuk menarik perhatian pengguna laman web.

Terdapat beberapa cara yang boleh digunakan untuk menarik perhatian pengguna. Antara caranya adalah seperti yang berikut:

- Penggunaan kesan khas seperti *drop shadow*, tekstur atau cahaya dan bentuk.
- Penggunaan ruang putih untuk menjadikan elemen-elemen kelihatan lebih terserlah (seperti yang ditunjukkan dalam Rajah 3.3.5)
- Penggunaan fungsi *bold*, *italic*, warna yang berbeza dan *border*.



Website Design
goo.gl/FPejLC



Rajah 3.3.5 Contoh penggunaan ruang putih bagi menonjolkan elemen imej serta pilihan bar navigasi yang digunakan



MALYSIAKU!

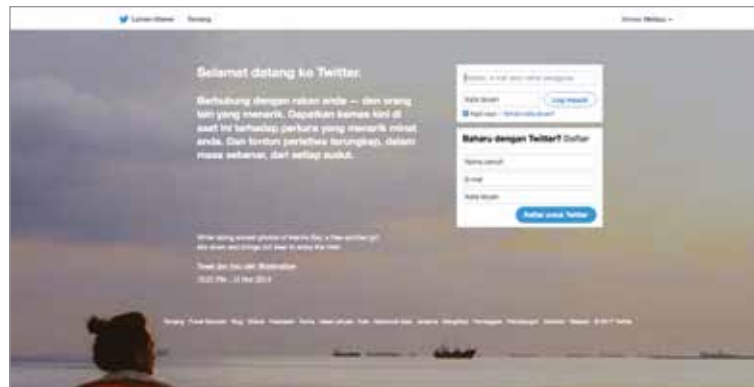
Sistem Analisis Peperiksaan Sekolah (SAPS) ialah sistem yang disediakan oleh Kementerian Pendidikan Malaysia (KPM). Tujuan sistem ini dibangunkan adalah untuk menganalisis data peperiksaan dalaman setiap sekolah di seluruh Malaysia. Sistem ini ialah salah satu revolusi dalam pendidikan di Malaysia yang boleh dimanfaatkan oleh ibu bapa untuk memantau perkembangan prestasi anak-anak mereka.

Perbezaan Elemen dalam Web

Konsep perbezaan digunakan untuk menghasilkan tarikan visual terhadap suatu elemen yang ingin ditekankan dalam laman web. Semakin besar perbezaan suatu elemen yang ingin ditonjolkan berbanding dengan elemen lain di persekitarannya, maka elemen tersebut akan kelihatan lebih berbeza. Penggunaan konsep perbezaan ini juga mampu menghasilkan sistem navigasi yang lebih mudah dan dapat memandu pelawat untuk mencapai maklumat yang dikehendaki atau mengarah mereka ke tempat yang ingin ditujui.

Antara kaedah perbezaan yang boleh digunakan adalah yang seperti berikut:

- Penggunaan label dan pautan yang berbeza.
- Penggunaan *border*, warna yang berbeza dan juga kesan khas.
- Penggunaan ruang putih, teks terbalik, teks berhuruf condong (*italic*) dan saiz teks yang berbeza.
- Penggunaan prinsip perbezaan warna dalam memberi kelainan pada butang *login* (seperti yang ditunjukkan pada Rajah 3.3.6).



Rajah 3.3.6 Contoh laman web *Twitter* yang menggunakan prinsip perbezaan bagi memberikan penegasan kepada butang *login*

Typography

Prinsip *typography* merujuk penyusunan teks yang dipaparkan dalam laman web. Penyusunan teks yang baik memudahkan pemahaman dalam proses pembacaan dan juga meningkatkan elemen mesra pengguna dalam laman web. Penggunaan grafik dan teks yang saling melengkapi mampu membentuk imej dalam minda pembaca. Selain itu, *typography* juga mampu menyampaikan mesej atau idea pereka web yang mungkin bersifat profesional kepada pembacanya dengan lebih baik.

Terdapat beberapa corak penyusunan teks bagi laman web. Corak penyusunan ini boleh digunakan mengikut gaya pembacaan pengguna. Rajah 3.3.7 menunjukkan salah satu corak penyusunan dan paparan teks berasaskan skrin. Corak penyusunan teks lain yang mengikut gaya pembacaan pengguna adalah seperti yang berikut:

- Corak paparan berasaskan kertas
- Corak paparan secara *Z-shaped*
- Corak paparan secara *F-shaped*



Rajah 3.3.7 Contoh corak penyusunan teks berasaskan skrin

Selain itu, semasa membuat pemilihan tentang jenis fon yang digunakan dalam laman web untuk dipaparkan menerusi pelayar, adalah lebih baik kita memilih atau menggunakan kumpulan fon yang dikategorikan sebagai senarai selamat digunakan atau disokong oleh kebanyakan platform. Pemilihan fon yang kurang biasa digunakan boleh menyebabkan sesetengah sistem pengoperasian tidak menyokong jenis fon tersebut dan mempunyai masalah untuk memaparkan kandungan web. Rajah 3.3.8 menunjukkan contoh fon atau teks yang disokong oleh kebanyakan pelayar.



Rajah 3.3.8 Contoh jenis rupa aksara atau fon yang dikategorikan sebagai senarai selamat

Tahukah Anda

Teknologi yang semakin berkembang telah membangunkan teknologi untuk membantu pengguna yang cacat penglihatan melayari Internet. Program *text-to-speech* telah dibangunkan yang menukarkan teks kepada audio yang membolehkan mereka mendapat maklumat secara mendengar. Terdapat juga kemudahan *Braille Keyboards* yang membolehkan mereka menaip.

Sumber:
http://www.huffingtonpost.com/entry/blind-people-internet-accessibility_us_57051a56e4b0537661883b1c



Pada masa ini, pengguna tidak perlu membuang masa untuk ke kompleks beli-belah bagi membeli keperluan mereka. Terdapat banyak laman web untuk perniagaan seperti *Lelong*, *GO SHOP* dan sebagainya untuk mereka membeli sebarang keperluan.



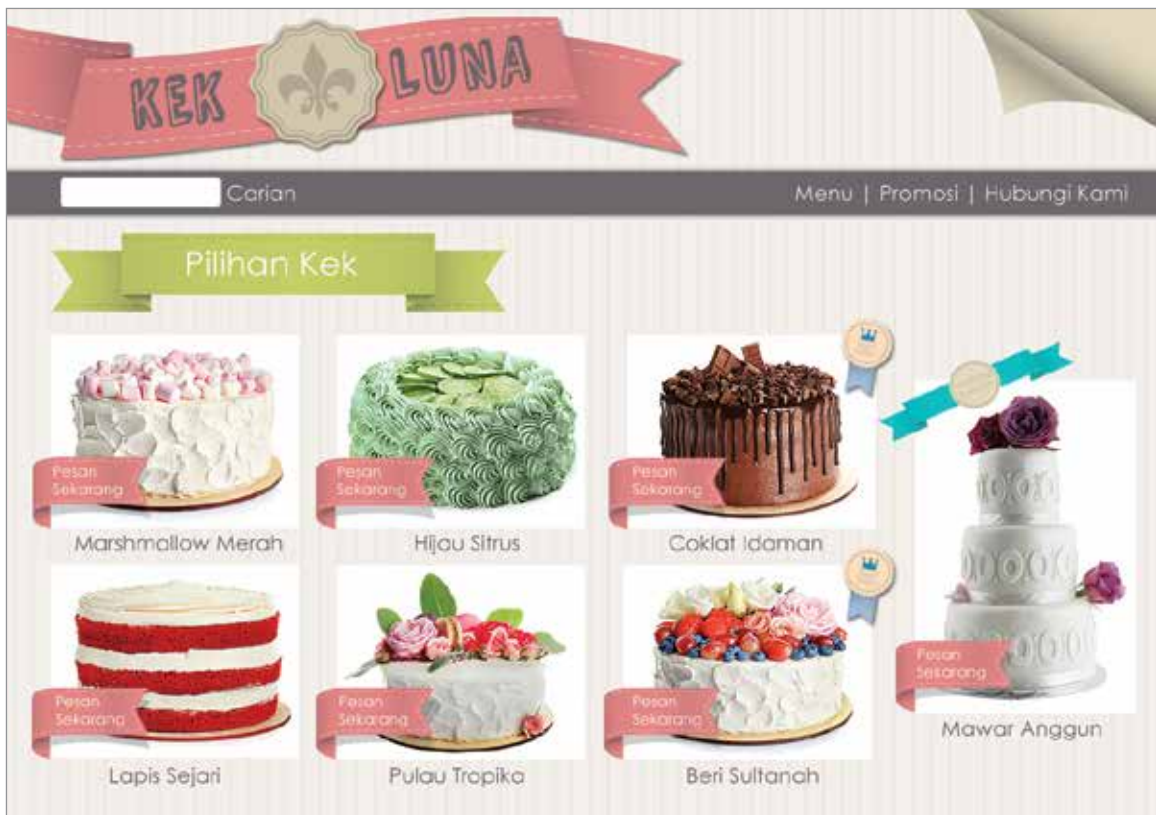
Kesan Warna Dalam
Pembangunan Web
goo.gl/i55yFD

Antara panduan yang berkaitan dengan *typography* adalah seperti yang berikut:

- Seragamkan jenis teks atau fon yang digunakan bagi setiap halaman yang dibina.
- Gunakan saiz teks yang berbeza bagi membezakan tajuk dengan kandungan atau huraian.

Warna dan Grafik

Penggunaan warna dan grafik dalam laman web mampu menonjolkan imej dan identiti sesebuah laman web. Penggunaan warna yang bijak mampu menarik perhatian pelawat serta meningkatkan kebolehbacaan mereka. Penggunaan warna latar belakang dan teks yang senada mampu memberikan pengalaman yang kurang menyenangkan dalam kalangan pengguna. Hal ini dikatakan demikian kerana keadaan ini menyukarkan proses pembacaan mereka. Penggunaan warna yang agak berwarna-warni juga mampu membuatkan pembaca berasa serabut dan kehilangan fokus. Rajah 3.3.9 menunjukkan kombinasi warna latar belakang yang sesuai.



Rajah 3.3.9 Contoh penggunaan warna yang sesuai yang mampu memberikan imej kepada laman web dibina

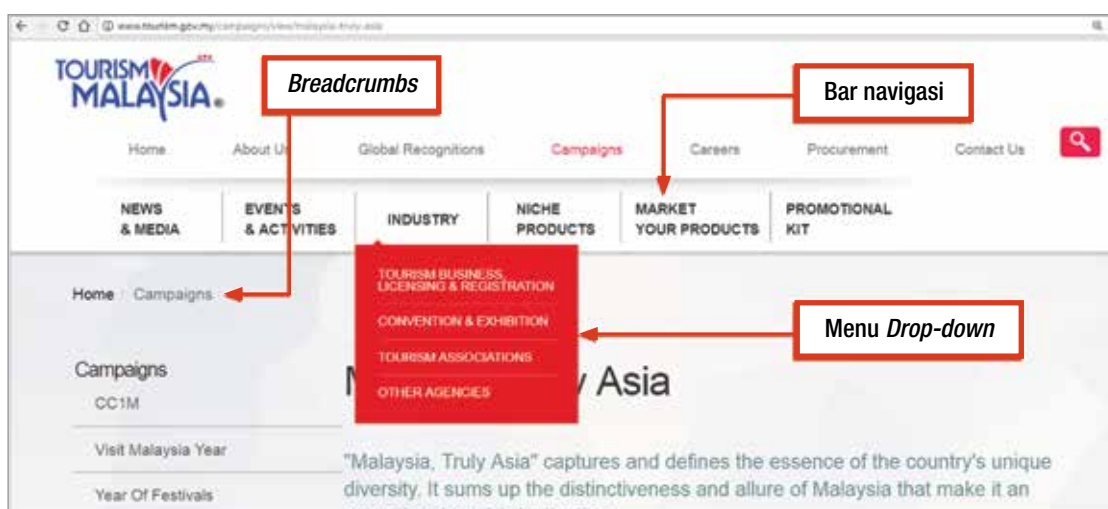
Navigasi

Navigasi merujuk struktur laman web yang akan memudahkan seseorang pengguna melayarinya. Konsep navigasi web ini memberikan pengalaman kepada pelawat laman web untuk mengemudi laman web tersebut mengikut kecekapan masing-masing. Navigasi web yang bagus seharusnya memberikan petunjuk kepada pelawat laman web tentang kedudukan atau lokasi semasa pengguna.

Pereka web biasanya akan menggunakan pelbagai teknik navigasi dalam laman web mereka mengikut kesesuaian. Antaranya adalah seperti yang berikut:

- Bar navigasi
- *Breadcrumbs*
- *Site map*
- Menu *Drop-down*
- Pautan menggunakan grafik

Rajah 3.3.10 menunjukkan beberapa teknik navigasi yang digunakan dalam laman web Tourism Malaysia.



Rajah 3.3.10 Contoh laman web Tourism Malaysia yang menggunakan teknik navigasi *breadcrumbs*, menu *drop-down* dan *navigation bar*.

Keseimbangan Visual

Keseimbangan visual pula merujuk gabungan elemen visual seperti garisan, imej, teks, bentuk, dan warna dalam laman web. Bagi menentukan corak gabungan semua elemen visual tersebut, pereka web juga perlu peka tentang pengguna sasaran laman web tersebut. Sebagai contoh, sekiranya laman web tersebut dibina untuk golongan kanak-kanak, elemen teks yang banyak perlu dielakkan kerana kanak-kanak tidak cenderung untuk membaca. Sebaliknya penggunaan unsur grafik dan imej mesti dilebihkan kerana elemen tersebut mampu menarik minat mereka untuk terus melayari laman web seperti yang ditunjukkan dalam Rajah 3.3.11.

Tahukah Anda

Laman web *ebizmba.com* telah melaporkan 15 laman web kanak-kanak paling popular pada Januari, 2017.

Anda boleh rujuk sumber di bawah untuk melihat laman-laman web tersebut:

Sumber:
goo.gl/DBjBNS



Rajah 3.3.11 Contoh laman web kanak-kanak yang menggunakan unsur grafik dan imej mengikut kesesuaian mereka

Reka bentuk Berasaskan Pengguna

Suatu laman web yang dibina itu akan memenuhi kehendak pengguna apabila laman web tersebut mampu menepati cita rasa pengguna. Untuk itu, pereka web harus mengenali pengguna sasaran mereka dan perkara diinginkan oleh pengguna apabila mereka melayari laman web yang dibina. Sebagai contoh, adakah mereka melayari laman web untuk membaca artikel atau mencari maklumat? Apakah yang diharapkan oleh mereka untuk mencapai tujuan tersebut?

Pertimbangkan laman web enjin carian *Google* yang popular dalam kalangan pengguna. Apakah yang menyebabkan laman web tersebut popular berbanding dengan enjin carian yang lain? Hal ini dikatakan demikian kerana laman web tersebut berjaya memenuhi keinginan pengguna mereka yang melayari laman web tersebut untuk mencari maklumat. *Google* menghasilkan satu laman web yang mudah difahami, hasil carian yang cukup pantas dan maklumat yang sangat padat. Antara muka yang dihasilkan juga sangat ringkas, mudah difahami tanpa iklan dengan pautan yang sangat sedikit dan tujuannya dihasilkan hanya berfokus pada satu objektif, iaitu membenarkan pengguna memasukkan kata kunci carian. Rajah 3.3.12 menunjukkan laman web *Google* yang mempamerkan paparan yang ringkas dan mudah difahami serta sesuai dengan kehendak pengguna yang ingin mencari maklumat.



Rajah 3.3.12 Contoh laman web *www.google.com* yang mempunyai paparan ringkas untuk tujuan carian maklumat.



UJI MINDA

Pengguna seperti Orang Kelainan Upaya (OKU) memerlukan peralatan atau perisian sokongan untuk membantu mereka melayari laman web. Apakah peralatan atau perisian yang sesuai bagi pengguna yang mengalami kecacatan pendengaran?

AKTIVITI

1

Kerja Individu

Prinsip Reka Bentuk Laman Web

Objektif: Mengenal pasti prinsip-prinsip asas reka bentuk laman web yang diimplementasikan dalam laman web media sosial.

Anda dikehendaki melayari laman web media sosial berikut:

- www.facebook.com
- www.instagram.com

Berdasarkan pemerhatian anda, senaraikan prinsip asas reka bentuk laman web yang digunakan dalam setiap laman web tersebut.



3.3.2 Mereka bentuk Kerangka Aplikasi

Penstrukturan Kandungan

Kunci kejayaan penghasilan suatu laman web bergantung pada cara laman web tersebut berfungsi. Walaupun suatu laman web tersebut mengandungi elemen animasi dan grafik yang hebat, tetapi sekiranya laman web tersebut gagal memandu pengguna dalam melaksanakan apa-apa yang dikehendaki oleh mereka seperti proses pembelian produk dilakukan atau cara pengguna memanfaatkan sumber dan aplikasinya, maka kewujudan laman web tersebut bagaikan sia-sia. Untuk membina laman web yang baik, perhatian yang lebih perlu diberikan untuk menentukan cara maklumat dalam laman web tersebut dapat dicapai dan pendekatan yang sesuai bagi pengguna untuk mendapatkan maklumat.

Semasa mereka bentuk kerangka aplikasi yang ingin dibangunkan, terdapat beberapa proses yang perlu dilakukan seperti yang ditunjukkan dalam Rajah 3.3.13.



Rajah 3.3.13 Proses mereka bentuk kerangka aplikasi

Penstrukturan Kandungan atau Maklumat

Apakah kandungan yang perlu dimasukkan dalam laman web anda? Bagaimanakah kandungan maklumat dalam laman web dikelaskan? Persoalan-persoalan tersebut adalah antara perkara penting yang memerlukan seni pengelasan dan pelabelan maklumat dalam laman web.

Dalam proses pembinaan laman web, setiap reka bentuk yang bakal dihasilkan perlu berasaskan pengguna atau pengalaman pengguna semasa melayari laman web. Oleh sebab itu, penglibatan pengguna semasa proses pengumpulan maklumat adalah sangat penting untuk memastikan laman web yang dibina mampu memenuhi kehendak mereka. Antara kaedah yang boleh digunakan bagi mendapatkan input daripada pengguna adalah seperti mengadakan sesi temu bual dengan pengguna, mengedarkan borang kaji selidik kepada sasaran pengguna atau membuat tinjauan cara laman web digunakan oleh pengguna. Antara langkah dalam reka bentuk maklumat adalah seperti yang berikut:

- **Membentuk inventori maklumat**

Langkah pertama yang perlu diambil ialah pembina laman web menyenaraikan semua maklumat yang bakal dimasukkan dalam laman web yang dibina. Senarai atau inventori tersebut bukan sahaja terhad kepada jenis maklumat yang akan dimasukkan, malah perkara yang boleh dilakukan oleh pengguna juga perlu disenaraikan. Hal ini dikatakan demikian kerana senarai atau inventori akan memberikan idea kepada fungsi yang perlu ada dalam laman web tersebut. Sebagai contoh, bagi laman web elektronik dagang, fungsi seperti *zoom in* produk pilihan, *shopping cart*, prosedur penghantaran produk atau respons kepada pengguna adalah antara fungsi penting yang perlu ada bagi laman web tersebut. Rajah 3.3.14 menunjukkan fungsi *Zoom In* yang biasa digunakan dalam laman web elektronik dagang.



Rajah 3.3.14 Contoh fungsi *Zoom In* item

- **Pengelasan maklumat**

Langkah seterusnya ialah pembina laman web mengelaskan senarai maklumat atau inventori maklumat kepada beberapa kategori. Untuk tujuan tersebut, pereka bentuk maklumat boleh melihat persamaan maklumat yang ada bagi mewujudkan kemungkinan-kemungkinan kategori yang perlu ada. Maklumat atau informasi ialah suatu yang subjektif dan corak pengelasan maklumat boleh berbeza-beza bergantung pada perspektif pereka web. Antara kaedah yang boleh digunakan dalam pengelasan maklumat adalah seperti yang ditunjukkan dalam Jadual 3.3.1.

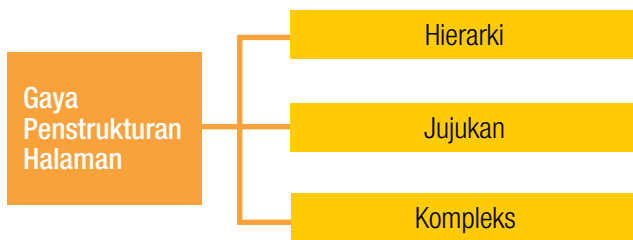
Jadual 3.3.1 Pengelasan maklumat

Kaedah	Contoh
Mengikut turutan abjad	Pengelasan maklumat jualan berasaskan nama pengguna
Kronologi	Pengelasan hasil jualan mengikut tarikh
Kategori atau jenis	Pengelasan jualan mengikut kategori produk
Hierarki	Jenis baju muslimah yang dikelaskan kepada subkategori seperti blaus dan kurung moden
<i>Spatial</i>	Pengelasan maklumat mengikut geografi, contohnya pengelasan jualan mengikut negeri dan tempat
Mengikut urutan magnitud	Pengelasan maklumat jualan bermula daripada jumlah yang kecil kepada jumlah jualan yang besar.

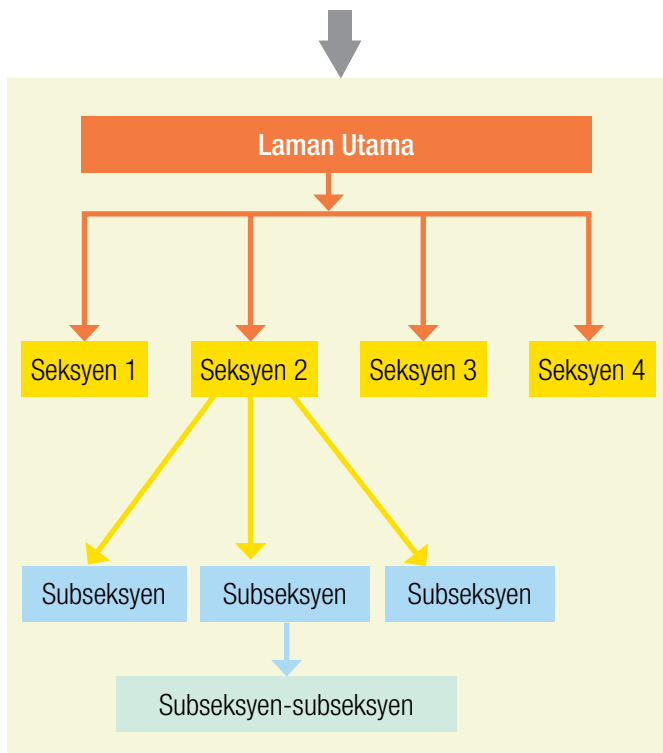
Semasa proses pengelasan maklumat, pereka perlu menetapkan bilangan kategori maklumat yang perlu ada bersesuaian dengan senarai maklumat yang tersedia dan bagaimana setiap kategori tersebut akan divisualkan dalam laman web yang bakal dibina (seperti yang ditunjukkan dalam Rajah 3.3.15). Kaedah terbaik yang dapat dilaksanakan adalah dengan menggunakan gambar rajah atau carta alir bagi menunjukkan hubungan kait bagi setiap kategori maklumat yang ada. Secara tidak langsung, ini juga mampu memberikan idea atau gambaran kasar tentang corak pautan laman web yang bakal dibina.



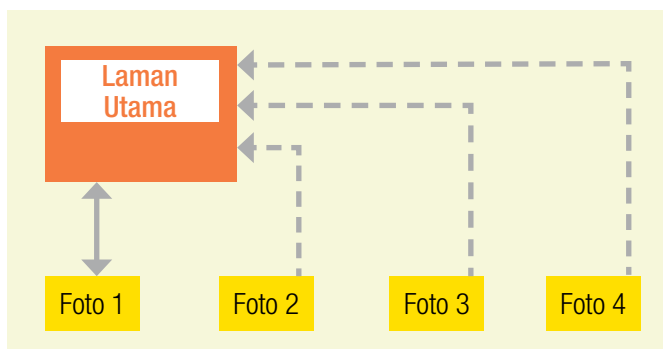
Rajah 3.3.15 Contoh pengelasan maklumat mengikut kategori atau jenis



Rajah 3.3.16 Beberapa gaya penstrukturan halaman



Rajah 3.3.17 Contoh penstrukturan secara hierarki



Rajah 3.3.18 Penstrukturan secara jujukan

Penstrukturan Halaman

Penstrukturan halaman pula menekankan cara setiap halaman dalam laman web tersebut dihubungkan. Kebiasaannya, pereka bentuk maklumat yang profesional akan menggunakan gambar rajah tapak (*site diagram*) sebagai alat untuk menyampaikan struktur laman web ini kepada klien dan sebagai panduan sepanjang proses pembangunan laman web. Gambar rajah tapak ialah *diagram* yang menggunakan simbol kotak bagi mewakili setiap laman web dengan garis atau anak panah bagi mewakili hubungan antara laman web.

Terdapat pelbagai gaya penstrukturan yang dapat digunakan oleh pereka web seperti yang ditunjukkan dalam Rajah 3.3.16. Antaranya adalah seperti yang berikut:

- **Penstrukturan secara hierarki**

Kebanyakan laman web dikelaskan secara hierarki bermula daripada laman pertama yang memaparkan beberapa pilihan kategori maklumat yang ada dalam laman web tersebut untuk dilayari oleh pengguna. Kebiasaannya, kategori yang diwujudkan perlu menepati kesamaan jenis maklumat yang ada bagi setiap kategori tersebut. Contoh penstrukturan secara hierarki ditunjukkan pada Rajah 3.3.17.

- **Penstrukturan secara jujukan**

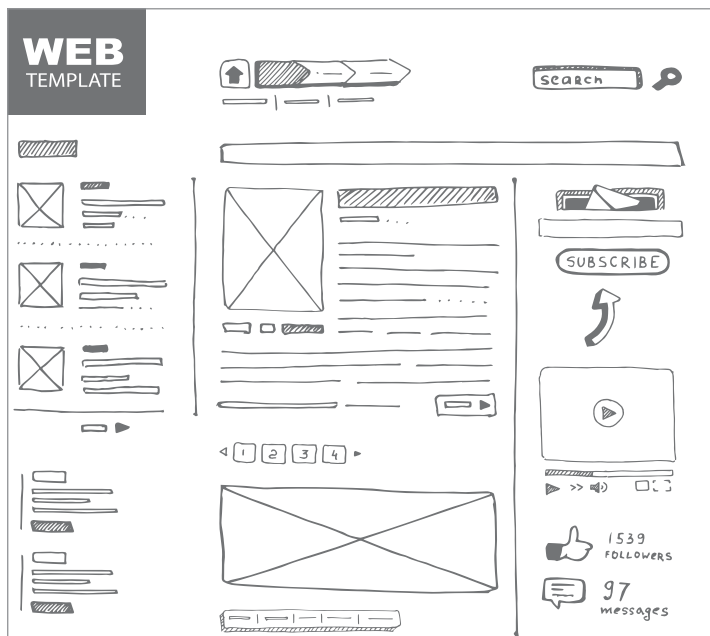
Dalam penstrukturan jujukan, pengguna dipandu dari halaman ke halaman yang lain secara jujukan. Penstrukturan ini lebih sesuai bagi laman web yang berciri penceritaan atau apa-apa maklumat yang bersifat kronologi. Contoh penstrukturan secara jujukan ditunjukkan pada Rajah 3.3.18.

- **Penstrukturan kompleks**

Terdapat laman web komersial yang menawarkan maklumat yang padat dan fungsi yang agak kompleks.

Reka Bentuk Antara Muka

Setelah halaman dalam laman web distrukturkan, fasa seterusnya adalah untuk menentukan bagaimana pengguna laman web dapat mencapai maklumat tersebut. Reka bentuk antara muka menentukan bagaimana struktur laman web secara logikalnya digambarkan termasuk teknik navigasi yang memandu kepada pergerakan pengguna dalam laman web. Pereka bentuk antara muka biasanya akan menggunakan *diagram* atau carta alir bagi menggambarkan reka bentuknya. Rajah biasanya akan menunjukkan bagaimana setiap halaman web berfungsi. Salah satu rajah yang digunakan dalam proses reka bentuk antara muka ialah *wireframe* atau rangka wayar (seperti yang ditunjukkan dalam Rajah 3.3.19). Rangka wayar atau juga dikenali sebagai skema halaman atau pelan halaman ialah panduan visual yang mewakili rangka kerja keseluruhan laman web. Tujuan rangka wayar dibuat ialah untuk menyusun elemen-elemen yang akan dimasukkan dalam laman web bagi menepati konsep laman web tersebut.



Rajah 3.3.19 Contoh wireframe



Contoh Rangka Wayar
goo.gl/z4EUαA

Reka bentuk Navigasi

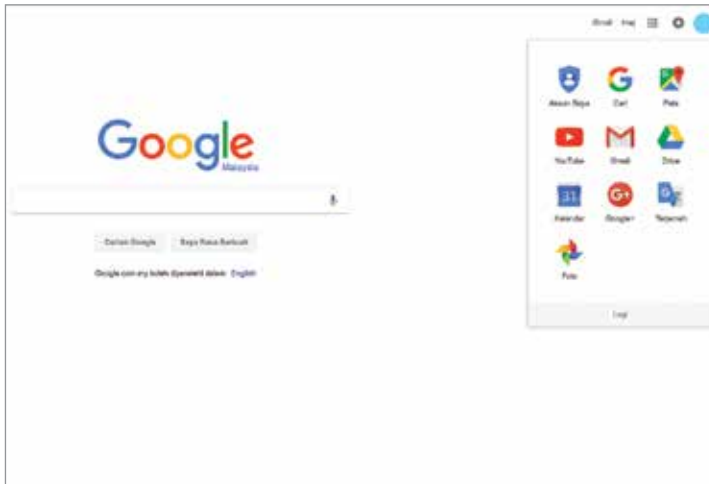
Sekiranya anda mengunjungi kompleks beli-belah, biasanya anda akan menggunakan sistem papan tanda untuk mencari lokasi kedai yang ingin dikunjungi. Begitulah juga konsep yang perlu diterapkan dalam laman web. Bagi membantu pelawat mengetahui kedudukan mereka atau untuk mencapai maklumat yang diinginkan, penggunaan logo, label, pautan atau pintasan lain yang membentuk sistem navigasi digunakan.

Navigasi web merujuk proses objek-objek dalam laman web yang terdiri daripada teks, rajah, audio dan video dihubungkan antara satu sama lain dengan menggunakan pautan hiperteks atau hipermedia. Konsep navigasi web ini memberikan pengalaman kepada pelawat laman web untuk mengemudi laman web tersebut mengikut kecekapan masing-masing. Navigasi web yang bagus seharusnya memberikan petunjuk kepada pelawat laman web tentang kedudukan atau lokasi semasa pengguna.

Fungsi utama konsep navigasi dalam laman web adalah untuk membantu pengguna atau pelawat mengetahui kedudukan semasa mereka. Selain itu, konsep navigasi juga berfungsi untuk memandu pelawat tentang pilihan yang disediakan untuk mereka menerokai lama web tersebut. Pada kebiasaannya, setiap laman web akan menggunakan lebih daripada satu jenis teknik navigasi.

Teknik Navigasi

Terdapat pelbagai teknik navigasi yang boleh dipilih oleh pereka web. Antaranya seperti:



Rajah 3.3.20 Contoh teknik navigasi *toolbar* yang pernah digunakan oleh *Google.com*



Rajah 3.3.21 Contoh bar navigasi menggunakan pautan ikon secara mendatar



Rajah 3.3.22 Contoh teknik navigasi secara *breadcrumbs*

- **Toolbar**

Rajah 3.3.20 menunjukkan teknik navigasi toolbar. Kebanyakan laman web menggunakan *toolbar* secara menegak atau mendatar bagi menyusun pilihan yang ditawarkan kepada pengguna.

- **Bar navigasi**

Rajah 3.3.21 menunjukkan teknik bar navigasi digunakan dengan mengumpulkan sekumpulan pautan teks pada satu laman web. Pautan boleh menggunakan pelbagai elemen multimedia seperti ikon, grafik atau teks. Penggunaan grafik sebagai pautan biasanya bertujuan untuk menjadikan laman web tersebut kelihatan lebih menarik manakala penggunaan pautan teks dapat mempercepat proses muat turun. Teknik ini juga boleh diimplementasikan dengan menyenaraikan elemen teks, ikon atau grafik secara mendatar atau menegak.

- **Breadcrumbs**

Rajah 3.3.22 menunjukkan teknik *breadcrumbs*, iaitu teknik kawalan grafik yang digunakan sebagai bantuan navigasi dalam antara muka pengguna. Teknik ini membolehkan pengguna menjejak lokasi atau kedudukan semasa mereka dalam program, dokumen atau laman web.

- **Peta tapak (sitemap)**

Rajah 3.3.23 merupakan contoh teknik navigasi secara peta tapak atau sitemap. Teknik ini menyenaraikan beberapa pautan laman web yang berkaitan untuk dilayari oleh pengguna.



Rajah 3.3.23 Contoh teknik navigasi secara *peta tapak*

- **Pautan menggunakan grafik**

Teknik navigasi menggunakan grafik seperti yang ditunjukkan dalam Rajah 3.3.24 biasanya banyak digunakan bagi sasaran pengguna seperti kanak-kanak agar lebih mudah bagi mereka untuk melayari laman web.



Rajah 3.3.24 Contoh navigasi menggunakan grafik bagi kanak-kanak dari *smashingmagazine.com*

- **Menu Drop-down**

Rajah 3.3.25 menunjukkan teknik navigasi melalui menu *drop-down*. Teknik ini merupakan elemen kawalan grafik yang menyenaraikan sekumpulan pautan dalam kelompok yang sama. Teknik ini memudahkan pengguna untuk membuat pilihan dengan memaparkan kumpulan pilihan pautan dengan lebih tersusun.



Rajah 3.3.25 Contoh teknik navigasi melalui menu *drop-down*

AKTIVITI
2

2 Kerja kumpulan
Penstrukturan Laman Web dan Teknik Navigasi

Objektif: Membuat tinjauan terhadap penstrukturan laman web dan teknik navigasi yang telah diimplementasikan

Bentuk satu kumpulan kecil. Bagi setiap kumpulan, anda dikehendaki melayari TIGA (3) jenis laman web yang berbeza. Bandingkan ketiga-tiga laman web tersebut berdasarkan faktor yang berikut:

- Jenis penstrukturan halaman yang digunakan
- Teknik navigasi yang digunakan

Berdasarkan tinjauan yang dibuat, apakah kesimpulan yang dapat anda buat?



APLIKASI HARIAN

Menggunakan media sosial untuk memasarkan barang perniagaan merupakan suatu tarikan pada masa ini kepada peniaga-peniaga yang ingin memasarkan barangan atau perkhidmatan mereka.



INOVASI SAINS KOMPUTER

Layari laman web <http://www.telegraph.co.uk/technology/6125914/How-20-popular-websites-looked-when-they-launched.html> untuk melihat perkembangan beberapa paparan laman web yang terkenal pada awal kemunculannya. Bandingkan dengan paparan laman web terkini yang semakin moden dan canggih.

3.3.3

Membina Laman Web Mudah dengan Menggunakan HTML

Apabila kali pertama konsep laman web diperkenalkan pada tahun 1990, laman web hanyalah koleksi dokumen yang dihubungkan dengan menggunakan fungsi pautan teks yang dipaparkan secara maya. Paparan pada laman web juga lebih bersifat statik. Laman web <http://www.telegraph.co.uk/technology/6125914/How-20-popular-websites-looked-when-they-launched.html> ada menyediakan paparan 20 laman web pada awal pelancarannya. Rajah 3.3.26 menunjukkan paparan laman web *Google* pada awal kemunculannya.



Rajah 3.3.26 Paparan *Google* pada awal kemunculannya pada tahun 1996

Pada waktu dahulu, adalah mustahil untuk kita menggunakan laman web bagi tujuan membeli produk, membuat tempahan hotel atau tiket kapal terbang. Punca di sebalik kekurangan ini ialah ketiadaan ruang untuk pengguna menghantar maklumat kepada pelayan untuk diproses pada masa tersebut. Selari dengan arus perubahan zaman, teknologi baharu diperkenalkan bagi memberikan ruang kepada pengguna menghantar maklumat kepada pelayan untuk diproses melalui konsep interaktif laman web. Dengan adanya konsep tersebut, proses membuat tempahan hotel serta pembelian barangan secara dalam talian bukan lagi suatu yang mustahil.

Definisi interaktif sebenarnya sangat luas. Suatu laman web yang mengandungi permainan catur dikelaskan sebagai interaktif kerana mampu memberikan respons kepada pergerakan pengguna yang bermain catur secara maya.

Dewasa ini, penggunaan laman web interaktif melalui rangkaian Internet telah berkembang meliputi pelbagai lapisan masyarakat dengan menghubungkan pelbagai jenis alatan seperti komputer, telefon bimbit, *Media Player* (MP3) dan stesen televisyen untuk tujuan komunikasi dan perkongsian maklumat.

Bagi membenarkan Internet dicapai oleh pengguna Internet, antara muka yang mudah diperlukan. Antara muka inilah yang menjadi sebahagian daripada *World Wide Web* (*WWW*). Setiap dokumen dalam *WWW* dikenali sebagai laman web. Tahukah anda bagaimana laman web dibina?

Apakah HTML?

Setiap laman web ialah fail teks yang ditulis menggunakan *Hypertext Markup Language* atau HTML. *Markup Language* ialah bahasa yang menghuraikan kandungan dan struktur dokumen. Anda boleh mengandaikan HTML sebagai rangka yang memandu kepada struktur setiap laman web. Rajah 3.3.27 menunjukkan struktur HTML.

```
<html>
  <head>
    <title> Tajuk Laman</title>
  </head>
  <body>
    Kandungan bagi laman web ditulis di bahagian ini.
  </body>
</html>
```

Rajah 3.3.27 Struktur HTML

Elemen dan Atribut dalam HTML

Bagi menulis dokumen HTML, satu program yang dikenali sebagai HTML *editor* digunakan. Salah satu HTML *editor* yang digunakan bagi menulis elemen atau tag HTML ialah *text editor*. HTML *text editor* yang biasa digunakan adalah seperti *Notepad*.

Dokumen HTML adalah mengenai konsep bagaimana dokumen dipaparkan. Untuk menerangkan struktur bagi setiap laman web, kod HTML atau elemen HTML digunakan. Elemen biasanya terdiri daripada dua tag yang berpasangan iaitu, *start tag* dan *close tag*. Kandungan yang

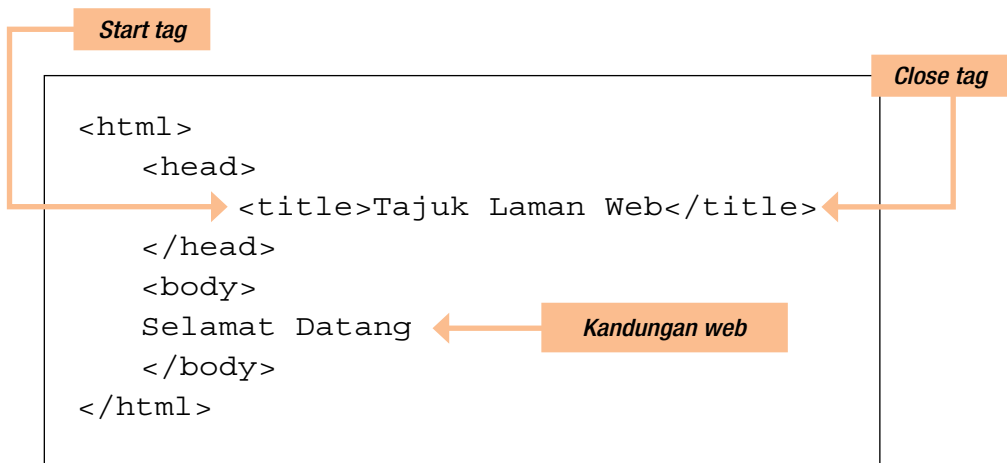


Sekumpulan guru di Wilayah Persekutuan Kuala Lumpur menerima pemberian tablet yang dilengkapi dengan aplikasi *e-tutor* dan *e-paper* bagi menyokong proses pengajaran dan pembelajaran seiring dengan perkembangan ICT dan ledakan maklumat tanpa sempadan. Pemberian gajet ini selaras dengan transformasi dalam Pelan Pembangunan Pendidikan Malaysia 2013 – 2025, iaitu untuk memanfaatkan ICT.



Guru di KL terima gajet goo.gl/wfoKDa

akan dipaparkan pada web akan berada di antara *start tag* dengan *close tag* seperti yang ditunjukkan dalam Rajah 3.3.28 dan Rajah 3.3.29 menunjukkan contoh outputnya.



Rajah 3.3.28 Contoh tag HTML berpasangan



Rajah 3.3.29 Contoh output tag HTML berpasangan

Terdapat juga elemen HTML yang hanya mempunyai *start tag* sahaja. Elemen ini dikenali sebagai *empty element*. Antara contoh *empty element* adalah seperti yang berikut:

-

-

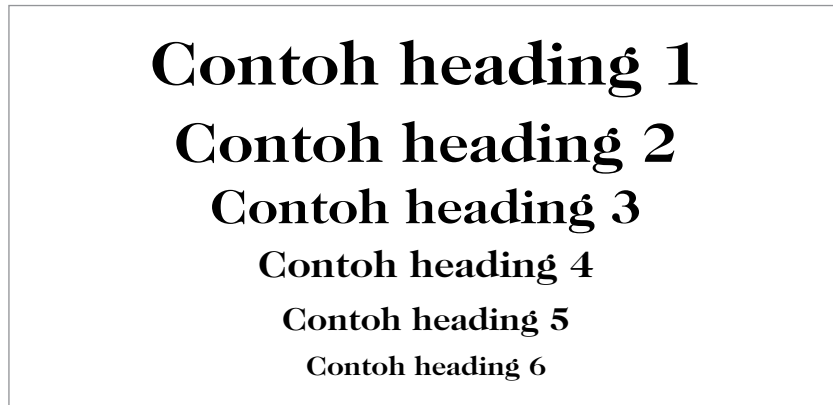
Contoh Penggunaan Elemen *Heading* dalam HTML

Elemen *heading* ialah elemen yang digunakan bagi mentakrifkan “tajuk” dalam laman web. Terdapat enam peringkat *heading* yang ditakrifkan dalam HTML, iaitu bermula daripada “<h1>” digunakan bagi tajuk besar atau penting hingga “<h6>” untuk tajuk kecil atau kurang penting. Rajah 3.3.30 menunjukkan elemen *heading* ini dan Rajah 3.3.31 menunjukkan contoh outputnya.

```
<h1>Contoh heading 1</h1>
<h2>Contoh heading 2</h2>
<h3>Contoh heading 3</h3>
<h4>Contoh heading 4</h4>
<h5>Contoh heading 5</h5>
<h6>Contoh heading 6</h6>
```

Rajah 3.3.30 Enam peringkat *heading*

Output:



Rajah 3.3.31 Contoh output bagi enam *heading*



Pereka tapak web ialah individu yang bertanggungjawab dalam menentukan susun atur, latar belakang dan reka bentuk sesebuah tapak web. Selain memiliki pengetahuan tentang web, pereka tapak web juga perlu kreatif untuk menata hias tapak web.

Contoh Penggunaan Elemen Header dalam HTML

Rajah 3.3.32 menunjukkan elemen `<header>` digunakan untuk memaparkan pengenalan kandungan bagi suatu laman web atau pautan navigasi. Elemen `<header>` biasanya mengandungi:

- Satu atau lebih elemen `<heading>` (`<h1>` – `<h6>`)
- Logo atau ikon
- Maklumat pengarang

```
<html>
  <head></head>
  <body>
    <header>
      <h1>Nilai-nilai Murni</h1>
      <hr>
      <h4>Definisi Nilai-nilai Murni</h4>
      <h4>Contoh Nilai-nilai Murni</h4>
    </header>
    <p>Nilai-nilai murni sewajarnya diterapkan dalam diri setiap murid sejak awal.</p>
  </body>
</html>
```

Rajah 3.3.32 Contoh penggunaan elemen `<header>` dalam HTML

Rajah 3.3.33 menunjukkan contoh output bagi penggunaan elemen <header> yang digunakan dalam HTML.

Output:



Rajah 3.3.33 Contoh output yang mempunyai elemen <header>

Contoh Penggunaan Elemen *Paragraph* dalam HTML

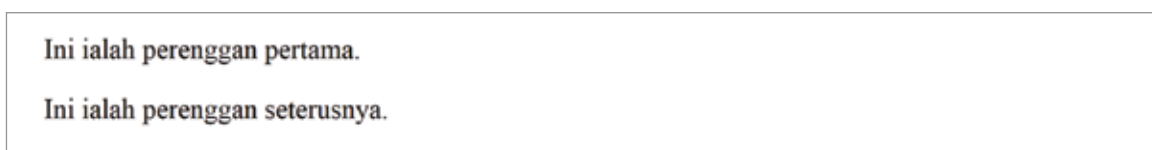
Bagi menyusun teks dalam bentuk perenggan, elemen "<p>" digunakan dalam HTML seperti yang ditunjukkan dalam Rajah 3.3.34.



Rajah 3.3.34 Elemen <p> yang digunakan dalam HTML

Output:

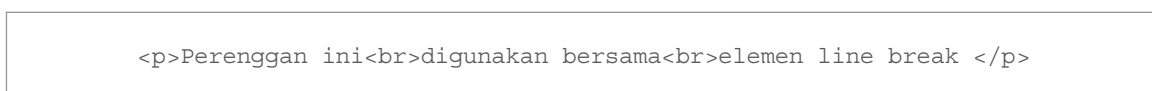
Rajah 3.3.35 menunjukkan hasil output bagi penggunaan elemen <p> dalam HTML.



Rajah 3.3.35 Contoh output yang mempunyai elemen <p>

Contoh Penggunaan Elemen *Line Break* dalam HTML

Bagi memaparkan teks di baris baharu, elemen
 digunakan dalam HTML seperti yang ditunjukkan dalam Rajah 3.3.36.



Rajah 3.3.36 Elemen
 yang digunakan dalam HTML

Output:

Rajah 3.3.37 menunjukkan hasil output bagi penggunaan elemen `
` untuk memulakan baris baharu dalam HTML.

Perenggan ini
digunakan bersama
elemen line break

Rajah 3.3.37 Contoh output yang mempunyai elemen `
`

Contoh Elemen *Image* dalam HTML

Bagi memaparkan grafik dalam laman web, elemen `` digunakan. Elemen `` ialah *empty element*, iaitu elemen ini tidak memerlukan *close tag*. Atribut `src` digunakan bersama `` tag untuk menentukan URL atau alamat imej tersebut diambil. Selain itu, atribut seperti `width` dan `height` juga boleh digunakan bersama bagi elemen ini untuk menetapkan saiz paparan grafik yang dipaparkan seperti yang dikehendaki. Rajah 3.3.38 menunjukkan contoh penggunaan elemen `` dalam HTML. Rajah 3.3.39 menunjukkan hasil output bagi penggunaan elemen `` dalam HTML.

```

```

Rajah 3.3.38 Elemen `` dalam HTML



Rajah 3.3.39 Output bagi elemen `` di Rajah 3.3.38

Contoh Elemen *Frame* dalam HTML

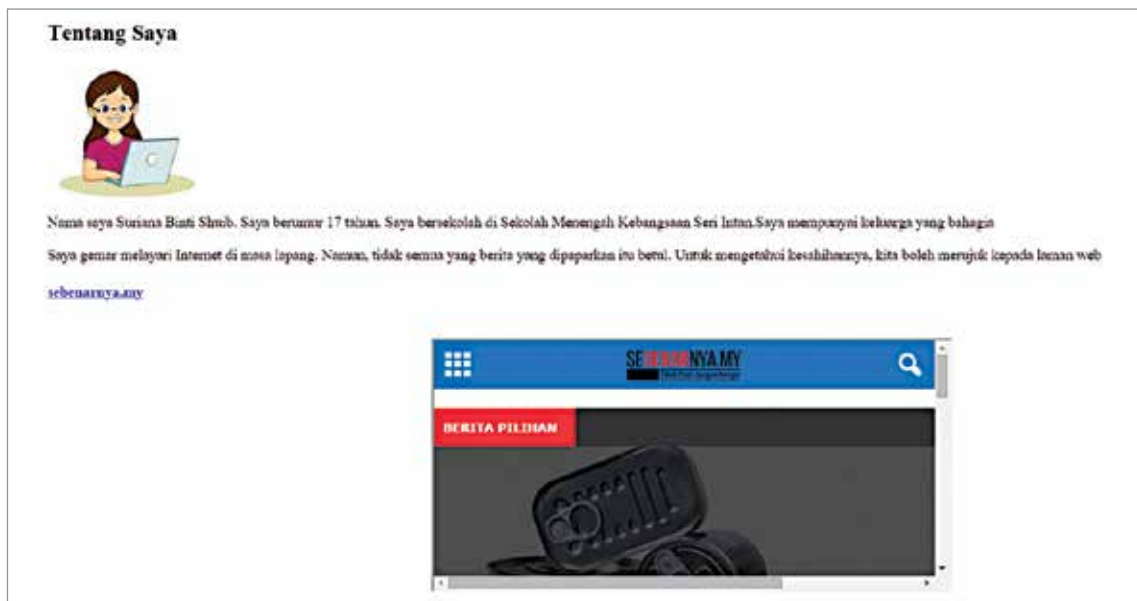
Elemen HTML *iframe* digunakan untuk memaparkan dokumen HTML lain dalam dokumen HTML semasa yang dipaparkan. Atribut *src*, *height* dan *width* digunakan bagi menentukan saiz paparan pada *iframe*.

Rajah 3.3.40 menunjukkan contoh penggunaan elemen `<iframe>` yang digunakan bersama elemen-elemen HTML `<p>`, `
` dan ``. Rajah 3.3.41 menunjukkan hasil output bagi penggunaan `<iframe>` apabila laman web *sebenarnya.my* dipaparkan dalam bingkai atau *frame* dalam laman web 'Tentang Saya'.

```
<html>
<head><title>Contoh Atur cara</title></head>
<body>
  <h2>Tentang Saya</h2>
  
  <p>Nama saya Suriana Binti Shuib. Saya berumur 17 tahun. Saya bersekolah di
    Sekolah Menengah Kebangsaan Seri Intan. Saya mempunyai keluarga yang
    bahagia
  </p>
  <p>Saya gemar melayari Internet di masa lapang. Namun, tidak semua yang
    berita yang dipaparkan itu betul. Untuk mengetahui kesahihannya, kita boleh
    merujuk kepada laman web
    <a href="http://sebenarnya.my"><h4>sebenarnya.my</h4></a>
  </p>
  <br>
  <center>
    <iframe height="40%" src="http://sebenarnya.my"></iframe>
  </center>
</body>
</html>
```

elemen
`<iframe>`

Rajah 3.3.40 Elemen `<iframe>` dalam HTML



Rajah 3.3.41 Contoh output bagi elemen `<iframe>`

Objektif: Meletakkan *frame* dalam laman web

Lakukan aktiviti ini di dalam makmal komputer.

Bahagian 1

- 1 Buka perisian NotePad.
- 2 Taip atur cara HTML berikut:

```
<html>
  <head><title>Pertama</title></head>
  <body>
    <h1>Frame pertama</h1>
  </body>
</html>
```

- 3 Simpan (*save*) sebagai pertama.html

Bahagian 2

- 1 Buka perisian NotePad.
- 2 Taip atur cara HTML berikut:

```
<html>
  <head><title>Kedua</title></head>
  <body>
    <h1>Frame kedua</h1>
  </body>
</html>
```

- 3 Simpan (*save*) sebagai kedua.html

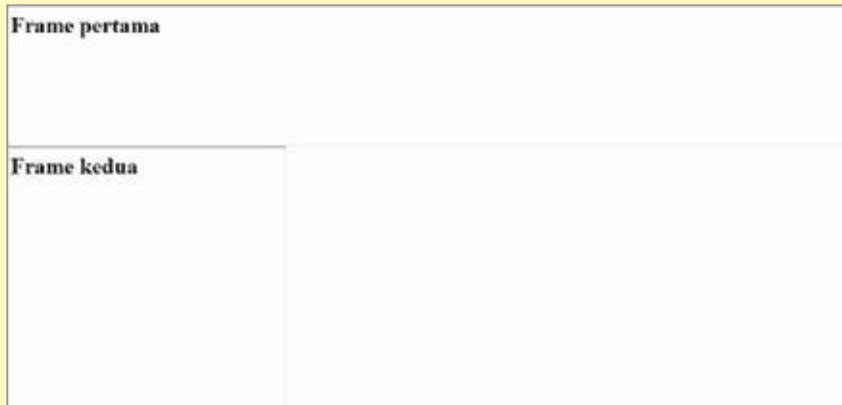
Bahagian 3

- 1 Buka fail NotePad yang baru.
- 2 Taip atur cara HTML berikut:

```
<html>
  <head><title>Gabung</title></head>
  <body>
    <iframe height="30%" width="100%"src="pertama.html">
    </iframe>
    <iframe height="70%" width="30%" src="kedua.html">
    </iframe>
  </body>
</html>
```

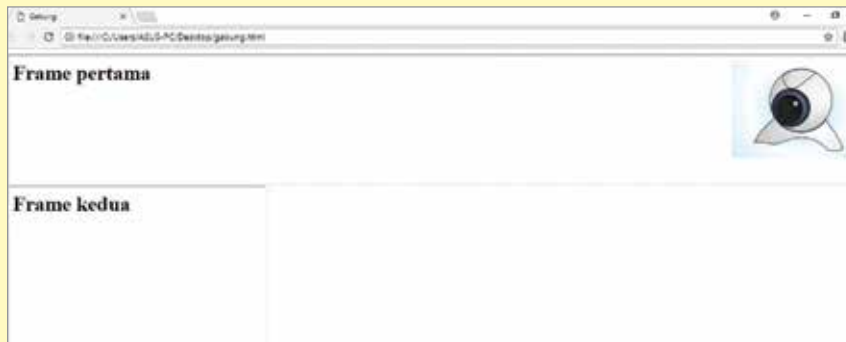
- 3 Simpan (*save*) sebagai gabung.html
- 4 Buka fail gabung.html melalui pelayar web (*web browser*) seperti Internet Explorer atau Google Chrome.

5 Hasil paparan pada pelayan web adalah seperti yang berikut:



6 Dapatkan satu imej daripada sumber Internet (contoh: webcam.jpg).

7 Letakkan imej tersebut dalam frame pertama seperti yang berikut:



3.3.4 Menggunakan *Cascading Style Sheets (CSS)* untuk Menggayakan *Text, Font, Background, Table, Borders* dan *Position*

Apa itu CSS?


Cascading Style Sheet (CSS) ialah salah satu bahasa pengaturcaraan yang merupakan tambahan (*extension*) kepada HTML. CSS membenarkan perubahan kepada gaya sesuatu laman web dan menerangkan persembahan bagi sesuatu dokumen HTML.



What is CSS?
goo.gl/rBLX3S

CONTOH 1

Contoh 1 menunjukkan perbezaan dalam pemaparan teks "Laman Web Sekolah", iaitu gaya warna teks ingin ditukarkan menjadi warna biru.

Contoh tanpa CSS dalam HTML	Contoh CSS dalam HTML
<pre><html> <head> <title>Tanpa Penggunaan CSS</title> </head> <body> <h1>Laman Web Sekolah</h1> </body> </html></pre>	<pre><html> <head> <title>Penggunaan CSS</title> <style> h1 { color: blue; 2 } </style> </head> <body> <h1>Laman Web Sekolah</h1> 1 </body> </html></pre>
<p>Output:</p> 	<p>Output:</p> 

Rajah 3.3.42 Contoh HTML dan Output (tanpa CSS dan dengan CSS).

Penerangan:

- 1 Teks "Laman Web Sekolah" dipaparkan sebagai "heading".
- 2 Teks dipaparkan mengikut gaya yang ditetapkan pada `<style>`, iaitu `color: blue`.

Cara Penulisan CSS dalam Atur Cara HTML

Terdapat tiga cara untuk menulis CSS dalam HTML, iaitu:

- Helaian gaya luar (*External Style Sheet*)
- Helaian gaya dalaman (*Internal Style Sheet*)
- Gaya dalam barisan (*Inline Style*)

Tahukah Anda

Kelebihan *External Style Sheets*

- 1 Anda boleh mengawal beberapa dokumen sekaligus.
- 2 Anda boleh membina kelas untuk stail yang boleh digunakan pada elemen HTML yang berbeza.
- 3 Anda boleh mengumpulkan stail dengan mudah untuk dijadikan lebih efisien.



Tahukah Anda

Kelebihan *Internal Style Sheets*

- 1 *Internal Style Sheets* hanya memberikan kesan pada laman yang sedang dipaparkan.
- 2 *Internal Style Sheets* boleh menggunakan kelas dan ID.
- 3 *Internal Style Sheets* tidak memerlukan anda memuat naik fail yang banyak.



Helaian gaya luar (*External Style Sheet*)

Helaian gaya luar bermaksud tag gaya akan ditulis dan disimpan di dalam satu fail dan fail itu mesti mempunyai *extension* “CSS”. Seterusnya, fail tersebut akan dipanggil daripada tag HTML dengan mengandungi sintaks seperti berikut:

```
<link rel="stylesheet" type="text/css" href="style1.css">
```

CONTOH 2

Contoh 2 menunjukkan cara penulisan CSS menggunakan *External Style Sheet*.

Terdapat dua fail yang perlu disediakan:

- 1 Satu fail bernama “style1.css” perlu disediakan yang mengandungi kandungan yang berikut:

Fail “style1.css”

```
h1 {  
    color: blue;  
}
```

- 2 Satu fail bernama “utama.html” perlu disediakan untuk menggunakan gaya yang telah ditulis dalam “style1.css”:

Fail “utama.html”

```
<html>  
  <head>  
    <title>Penggunaan CSS</title>  
    <link rel="stylesheet" type="text/css"  
      href="style1.css">  
  </head>  
  
  <body>  
    <h1>Laman Web Sekolah</h1>  
  </body>  
</html>
```

Output:



Rajah 3.3.43 Contoh HTML, CSS dan Output (*External Style Sheet*)

Helaian gaya dalaman (*Internal Style Sheet*)

Helaian gaya dalaman bermaksud tag gaya boleh ditulis sama ada di dalam kawasan "<head>" atau "<body>". Anda boleh menulis tag gaya dalam kedua-dua kawasan tersebut. "<style>" boleh diletakkan di mana-mana bahagian sama ada sebelum tag HTML atau selepas tag HTML.

CONTOH 3

Contoh 3 menunjukkan CSS menggunakan *Internal Style Sheet* yang ditulis dalam kawasan "<head>". Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.44.

```
<html>
  <head>
    <title>Penggunaan CSS</title>
    <style>
      h1
      {
        color: blue;
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1>
  </body>
</html>
```

Output:



Rajah 3.3.44 Contoh HTML, CSS dan Output (*Internal Style Sheet* – <head>)

CONTOH 4

Contoh 4 menunjukkan CSS menggunakan *Internal Style Sheet* yang ditulis dalam kawasan "<body>". Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.45.

```
<html>
  <head>
    <title>Penggunaan CSS</title>
  </head>
  <body>
    </style>
    h1
    {
      color: blue;
    }
    </style>
    <h1>Laman Web Sekolah </h1>
  </body>
</html>
```

Output:



Rajah 3.3.45 Contoh HTML, CSS dan Output (*Internal Style Sheet – <body>*)

Gaya dalam barisan (*Inline Style*)

Gaya dalam barisan bermaksud tag gaya akan ditulis pada bahagian/elemen tag. Contohnya, terdapat “<h1>”, iaitu kepala (*heading*) bagi sesuatu tajuk dalam suatu laman web dan warna “<h1>” ingin ditukar kepada warna biru. Oleh sebab itu, tag gaya perlu ditulis dalam tag “<h1>” seperti yang ditunjukkan dalam Contoh 5.

CONTOH 5

Contoh 5 menunjukkan CSS bagi gaya dalam barisan dalam HTML (*Inline Style*). Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.46.



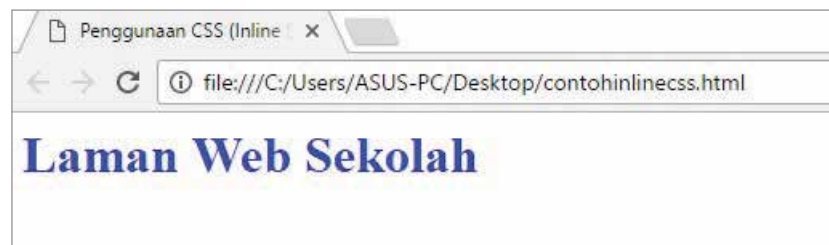
UJI MINDA

Berdasarkan tiga gaya untuk meletakkan CSS, senaraikan perbezaan yang dapat anda kenal pasti.

```
<html>
  <head>
    <title>Penggunaan CSS (Inline Style)</title>
  </head>

  <body>
    <h1 style = "color: blue;" >Laman Web Sekolah</h1>
  </body>
</html>
```

Output:



Rajah 3.3.46 Contoh HTML, CSS dan Output (*Inline Style*)

Tahukah Anda?

Warna dalam CSS boleh dinyatakan dengan kaedah yang berikut:

- Warna *Hexadecimal*
- Warna RGB
- Warna RGBA
- Warna HSL
- Warna HSLA
- Warna yang ditetapkan

Sekiranya terdapat ketiga-tiga gaya sekali di dalam suatu HTML, gaya dalam barisan akan dilaksanakan dahulu berbanding dengan helaian gaya luar dan helaian gaya dalaman.

Membina Laman Web HTML yang Mengandungi CSS untuk Memberikan Gaya kepada *Text*

Terdapat beberapa gaya bagi teks dalam CSS bagi suatu laman web.

Antara gaya tersebut ialah:

- **Warna teks (*colour*)**

Sintaks yang berikut akan menukar warna teks bagi *heading* <h1> kepada warna biru.

```
h1
{
  color: blue;
}
```

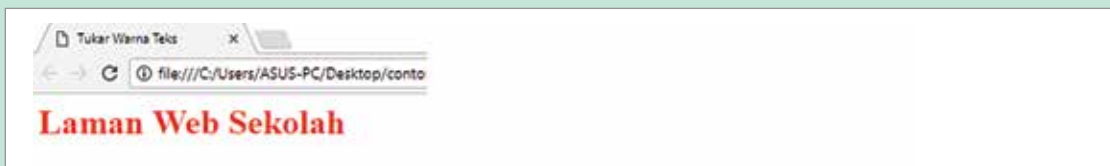
CONTOH 6

Contoh 6 menunjukkan pertukaran warna teks bagi <h1> kepada warna merah. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.47.

```
<html>
  <head>
    <title>Tukar Warna Teks</title>
    <style>
      h1
      {
        color: red; 2
      }
    </style>
  </head>

  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.47 Contoh HTML, CSS dan Output (menukar warna teks)

Penerangan:

- 1 Terdapat *heading* <h1> pada laman web.
- 2 Penetapan CSS `color: red` menjadikan warna teks <h1> menjadi warna merah.

- **Jajaran teks (*Text alignment*)**

Sintaks yang berikut akan menukar jajaran teks bagi <h1> ke bahagian tengah dokumen HTML.

```
h1
{
    text-align: center;
}
```

CONTOH 7

Contoh 7 menunjukkan jajaran teks bagi <h1> ditukar ke bahagian tengah dokumen HTML. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.48.

```
<html>
  <head>
    <title>Jajaran Teks</title>
    <style>
      h1
      {
        text-align: center; 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.48 Contoh HTML, CSS dan Output (jajaran teks)

Penerangan:

- 1 Terdapat *heading* <h1> pada laman web.
- 2 Penetapan CSS `text-align: center` menjadikan jajaran teks berada di bahagian tengah dokumen HTML.

- Bayang teks (*Text shadow*)

Sintaks yang berikut akan menambah bayang-bayang atau *shadow* pada teks kepala (*header text*) sebanyak ketebalan dua piksel, jarak bayang dengan dua piksel dan bayang-bayang akan berwarna hitam.

```
h1
{
    text-shadow: 2px 2px #000000;
}
```




#000000 merujuk kod warna berbentuk *Hexadecimal* dalam HTML.

CONTOH 8

Contoh 8 menunjukkan teks bagi <h1> mempunyai bayang-bayang berwarna hitam dengan ketebalan bayang-bayang sebanyak 2px (piksel) dan jarak antara bayang-bayang dengan teks asal sebanyak 2px (piksel). Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.49.

```
<html>
  <head>
    <title>Bayang Teks</title>
    <style>
      h1
      {
        text-shadow: 2px 2px #000000; 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.49 Contoh HTML, CSS dan Output (*shadow*)

Penerangan:

- 1 Terdapat heading <h1> pada laman web.
- 2 Penetapan CSS `text-shadow: 2px 2px #000000;` menjadikan bayang-bayang pada teks dengan tebal bayang-bayang sebanyak 2px dengan kod warna bayang-bayang #000000, iaitu hitam.

Tahukah Anda

Terdapat dua jenis nama keluarga fon, iaitu

- Nama keluarga “courier”, “arial” dan sebagainya
- Nama generik “serif”, “sans-serif”, “cursive”, “fantasy” dan sebagainya

Membina Laman Web HTML yang Mengandungi CSS untuk Memberikan Gaya kepada *Fon (Font)*

Terdapat beberapa gaya fon dalam CSS bagi suatu laman web. Antara gaya-gaya tersebut ialah:

- Keluarga fon (*Font family*)

```
h1
{
    font-family: "Arial Black";
}
```

Murid boleh merujuk keluarga fon yang telah “disimpan” ke dalam komputer atau merujuk senarai fon yang terdapat dalam *Microsoft Word*.

CONTOH 9

Contoh 9 menunjukkan penukaran gaya fon bagi teks <h1> iaitu menukar keluarga fon (*font family*) kepada "Arial Black". Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.50.

```
<html>
  <head>
    <title>Keluarga Font</title>
    <style>
      h1
      {
        font-family: "Arial Black"; 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.50 Contoh HTML, CSS dan Output (*font family*)

Penerangan:

- 1 Terdapat <h1> pada laman web.
- 2 Penetapan CSS font-family: "Arial Black" menjadikan jenis teks "Arial Black".

- **Saiz fon (*Font size*)**

Saiz fon boleh ditukar mengikut saiz yang ditetapkan. Sintaks yang berikut menunjukkan saiz fon ditukar kepada 70.

```
h1
{
  font-size: 70;
}
```



CSS Font-Size Property
goo.gl/83Krjq

CONTOH 10

Contoh 10 menunjukkan penukaran saiz fon menjadi lebih besar, iaitu sebanyak 70 (atau dikenali sebagai 70 piksel). Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.51.

```
<html>
  <head>
    <title>Saiz Font</title>
    <style>
      h1
      {
        font-size: 70; 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.51 Contoh HTML, CSS dan Output (saiz fon)

Penerangan:

- 1 Terdapat heading <h1> pada laman web.
- 2 Penetapan CSS `font-size: 70` menjadikan saiz teks kepada saiz 70.



Cuba anda tukar kod di sebelah untuk menukar stail fon *italic* kepada **bold** dan *normal*. Selamat mencuba!

- **Stail fon (*Font style*)**

Stail fon boleh ditukar mengikut keperluan di dalam dokumen HTML. Sintaks yang berikut menunjukkan stail fon ditukar kepada bentuk *italic*. Antara stail lain adalah seperti *bold* dan *normal*.

```
h1
{
    font-style: italic;
}
```

CONTOH 11

Contoh 11 menunjukkan penukaran stail fon bagi teks `<h1>` iaitu kepada bentuk italic. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.52.

```
<html>
  <head>
    <title>Stail Font</title>
    <style>
      h1
      {
        font-style: italic; 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.52 Contoh HTML, CSS dan Output (*font style*)

Penerangan:

- 1 Terdapat heading `<h1>` dalam laman web.
- 2 Penetapan CSS `font-style: italic` menjadikan bentuk teks ditetapkan dalam laman web.

Membina Laman Web HTML yang Mengandungi CSS untuk Memberikan Gaya kepada *Background*

Terdapat beberapa gaya yang boleh ditambah pada latar belakang (*background*) dokumen HTML bagi suatu laman web yang menggunakan CSS. Antaranya ialah:

- **Menukar warna latar belakang**

Sintaks yang berikut akan menukarkan latar belakang dokumen HTML menjadi warna hijau muda. Latar belakang dokumen merujuk kepada latar belakang `<body>`.

```
body
{
    background-color: lightgreen;
}
```

CONTOH 12

Contoh 12 menunjukkan penukaran warna latar belakang bagi dokumen HTML (laman web) kepada warna `lightgreen`. Rujuk "htmlcolorcodes.com" untuk warna-warna yang lain. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.53.

```
<html>
  <head>
    <title>Warna Latar Belakang</title>
    <style>
      body
      {
        background-color: lightgreen; 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.53 Contoh HTML, CSS dan Output (warna latar belakang)

Penerangan:

- 1 Terdapat heading `<h1>` dalam laman web.
- 2 Penetapan CSS `background-color: lightgreen` menjadi warna latar belakang laman web.



APLIKASI HARIAN

Anda boleh melayari goo.gl/27ZdNDM untuk cuba menukar warna latar belakang dokumen HTML.



• Menambah imej atau gambar sebagai latar belakang

Sintaks yang berikut akan menukar latar belakang dokumen HTML dengan menambah imej atau gambar sebagai latar belakang. Secara lalai (*by default*), gambar akan diulang sehingga memenuhi keseluruhan dokumen HTML. Gambar boleh disimpan di tempat yang sama atau tempat yang berlainan dengan dokumen HTML. Gambar juga boleh diperoleh dengan memanggil pautan kepada gambar tersebut.

```
body
{
    background-image: url("gambar.jpg");
}
```

CONTOH 13

Contoh 13 menunjukkan latar belakang dokumen HTML (laman web) boleh diwakili oleh suatu imej. Imej yang dijadikan latar belakang tersebut diperoleh daripada fail "gambar.jpg" yang telah disimpan. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.54.

```
<html>
  <head>
    <title>Latar belakang Imej</title>
    <style>
      body
      {
        background-image: url("gambar.jpg"); 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.54 Contoh HTML, CSS dan Output (imej yang disimpan di tempat yang sama sebagai latar belakang)

Penerangan:

- 1 Terdapat heading `<h1>` dalam laman web.
- 2 Imej yang disimpan di tempat yang sama dengan dokumen HTML, iaitu "gambar.jpg" telah menjadi latar belakang laman web.

Sintaks yang berikut menunjukkan gambar atau imej yang boleh disimpan di dalam tempat yang berlainan dengan dokumen HTML tetapi perlu mencapai imej dengan mengetahui *path* atau direktori imej tersebut.



Background-image
goo.gl/gyV5LB

```
body
{
    background-image: url("D:/web/gambar.jpg");
}
```

CONTOH 14

Contoh 14 menunjukkan latar belakang dokumen HTML (laman web) boleh diwakili oleh suatu imej yang diperoleh daripada fail "bandar.jpg" yang disimpan dalam satu direktori lain, iaitu "D:/web/". Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.55.

```
<html>
  <head>
    <title>Latar belakang Imej</title>
    <style>
      body
      {
        background-image: url("D:/web/bandar.jpg"); 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.55 Contoh HTML, CSS dan Output (imej yang disimpan di tempat yang lain)

Penerangan:

- 1 Terdapat heading `<h1>`.
- 2 Imej yang disimpan di direktori yang lain, iaitu daripada "D:/web/bandar.jpg" menjadi latar belakang laman web.

Sintaks yang berikut pula akan memuatkan gambar atau imej yang terdapat dalam sumber Internet dengan mengambil pautan gambar tersebut.

(Contoh: <https://newevolutiondesigns.com/images/freebies/white-wallpaper-18.jpg>)

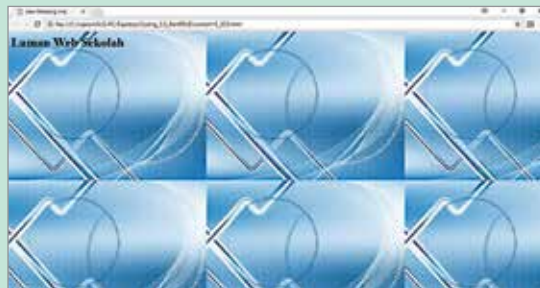
```
body
{
    background-image: url ("https://newevolutiondesigns.com/images/freebies/
        white-wallpaper-18.jpg");
}
```

CONTOH 15

Contoh 15 menunjukkan latar belakang dokumen HTML (laman web) boleh diwakili oleh suatu imej yang diperoleh daripada fail dalam sumber Internet seperti pada "<http://allfreedesigns.com/wp-content/uploads/2014/08/tech-background-6.jpg>". Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.56.

```
<html>
  <head>
    <title>Latar Belakang Imej</title>
    <style>
      body
      {
        background-image: url ("http://allfreedesigns.com/
          wp-content/uploads/2014/08/tech-background-6.jpg"); 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.56 Contoh HTML, CSS dan Output (imej daripada sumber Internet sebagai latar belakang)

Penerangan:

- 1 Terdapat heading `<h1>`.
- 2 Imej daripada sumber Internet, iaitu daripada "<http://allfreedesigns.com/wp-content/uploads/2014/08/tech-background-6.jpg>" menjadi latar belakang laman web.

Membina Laman Web HTML yang Mengandungi CSS untuk Memberikan Gaya kepada *Tables*

Sebelum ini, murid telah mempelajari bagaimana untuk menghasilkan jadual atau dikenali sebagai *table* dalam suatu dokumen HTML. *Table* terdiri daripada “<tr>”, “<th>” dan “<td>” iaitu merujuk setiap baris dalam *table* (*row*), kepala jadual (*header*) dan item jadual (*data*). Terdapat beberapa gaya yang boleh ditambah kepada *table*. Antaranya ialah:

- Meletakkan garisan luar atau *border*

Dalam sintaks yang berikut, seluruh *table* akan ditambah garisan luar atau *border*.

```
table
{
    border: 1px solid;
}
```

CONTOH 16

Contoh 16 menunjukkan jadual (*table*) yang terdapat dalam dokumen HTML (laman web) ditambah garis luar (*border*). Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.57.

```
<html>
  <head>
    <title>Gaya Jadual</title>
    <style>
      table
      {
        border: 1px solid; 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1>
    <table>
      <tr>
        <th>KANDUNGAN LAMAN WEB</th>
      </tr>
      <tr>
        <td>Pengenalan Sekolah</td>
      </tr>
      <tr>
        <td>Senarai Guru</td>
      </tr>
    </table>
  </body>
</html>
```



Output:



Rajah 3.3.57 Contoh HTML, CSS dan Output (*border table*)

Penerangan:




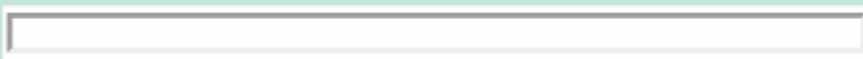
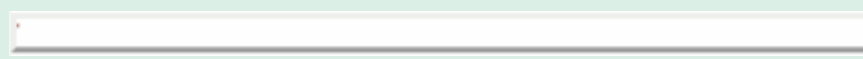
- 1 Terdapat `<table>` pada laman web.
- 2 `<table>` mempunyai garis luar dengan ketetapan CSS jenis garisan solid dan ketebalan sebanyak 1px.

Begitu juga dengan *header* dan data pada jadual yang setiap bahagiannya akan diletakkan *border* masing-masing. Ketebalan garisan atau *border* ditunjukkan melalui 1px, iaitu ketebalan sebanyak 1 piksel. Murid boleh menebalkannya lagi dengan menambah nombor seperti 3px.

Sintaks yang berikut menggunakan jenis garisan `solid`.

```
table, th, td
{
    border: 1px solid;
}
```

Terdapat beberapa jenis garisan, iaitu *Solid*, *Dotted*, *Dashed*, *Inset* dan *Outset* seperti yang ditunjukkan dalam Rajah 3.3.58.

Nama garisan	Garisan
<i>Solid</i>	
<i>Dotted</i>	
<i>Dashed</i>	
<i>Inset</i>	
<i>Outset</i>	

Rajah 3.3.58 Jenis garisan

CONTOH 17

Contoh 17 menunjukkan bahawa *border* hendak ditambah pada jadual (*table*), kepala jadual (*header*) dan data jadual yang diwakili oleh `<table>`, `<th>` dan `<td>`. Setiap *border* mempunyai ketebalan sebanyak 1 piksel dan jenis garisan solid. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.59.

```
<html>
  <head>
    <title>Gaya Jadual</title>
    <style>
      table, th, td
      {
        border: 1px solid; 2
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1>
    <table>
      <tr>
        <th>KANDUNGAN LAMAN WEB</th>
      </tr>
      <tr>
        <td>Pengenalan Sekolah</td>
      </tr>
      <tr>
        <td>Senarai Guru</td>
      </tr>
    </table>
  </body>
</html>
```



Output:



Rajah 3.3.59 Contoh HTML, CSS dan Output (jenis garisan dari tebal *Border Table*)

Penerangan:

- 1 Terdapat `<table>` pada laman web.
- 2 Border `<table>`, `<th>`, `<td>` mempunyai satu garis dengan ketetapan CSS jenis garisan solid dan ketebalan sebanyak 1px.

Sintaks yang berikut menetapkan garisan atau *border* menjadi *single border*.

```
table
{
    border-collapse:collapse;
}
```

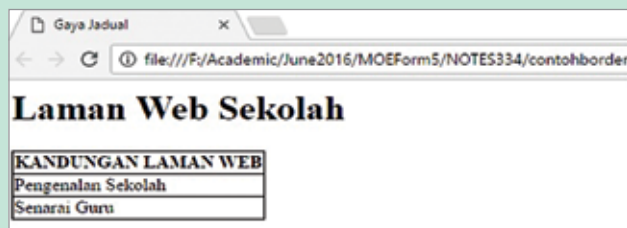
CONTOH 18

Contoh 18 menunjukkan jadual (*table*) mempunyai ketetapan satu garis pada *border* dengan ketebalan sebanyak 1 piksel serta garisan jenis solid. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.60.

```
<html>
  <head>
    <title> Gaya Jadual</title>
    <style>
      table
      {
        border-collapse: collapse; 2
      }

      table, th, td
      {
        border: 1px solid;
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1>
    <table>
      <tr>
        <th>KANDUNGAN LAMAN WEB</th>
      </tr>
      <tr>
        <td>Pengenalan Sekolah</td>
      </tr>
      <tr>
        <td>Senarai Guru</td>
      </tr>
    </table>
  </body>
</html>
```

Output:



Rajah 3.3.60 Contoh HTML, CSS dan Output (*Single Border Table*)

Penerangan:

- 1 Terdapat `<table>` pada laman web.
- 2 Border `<table>` mempunyai satu garis dengan ketetapan CSS `border-collapse: collapse`.

Sintaks yang berikut menebalkan *border* sebanyak 3px.

```
table, th, td
{
    border: 3px solid;
}
```

CONTOH 19

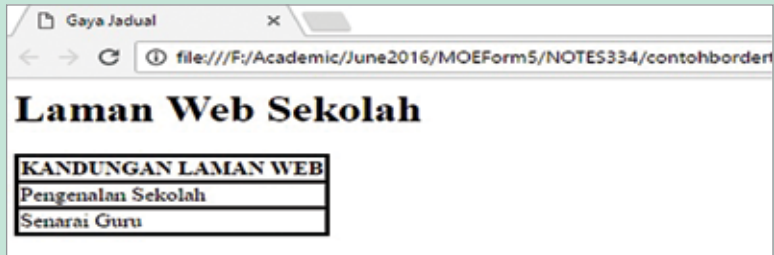
Contoh 19 menunjukkan jadual (*table*) mempunyai satu garis pada garisan *border* dan ketebalan sebanyak 3 piksel serta jenis garisan solid. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.61.

```
<html>
<head>
  <title> Gaya Jadual</title>
  <style>
    table
    {
      border-collapse: collapse;
    }

    table, th, td
    {
      border: 3px solid; 2
    }
  </style>
</head>
<body>
  <h1>Laman Web Sekolah</h1>
  <table>
    <tr>
      <th>KANDUNGAN LAMAN WEB</th>
    </tr>
    <tr>
      <td>Pengenalan Sekolah</td>
    </tr>
    <tr>
      <td>Senarai Guru</td>
    </tr>
  </table>
</body>
</html>
```



Output:



Rajah 3.3.61 Contoh HTML, CSS dan Output (menebalkan *Border Table*)

Penerangan:

- 1 Terdapat `<table>` pada laman web.
- 2 *Border table* ditebalkan dengan ketetapan CSS `border: 3px solid`.

CONTOH 20

Contoh 20 menunjukkan jadual mempunyai satu garis pada garisan *border* dan ketebalan sebanyak 3 piksel serta jenis garisan *solid*, manakala `<td>` mempunyai jenis garisan *dashed*. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.62.

```
<html>
  <head>
    <title> Gaya Jadual</title>
    <style>
      table
      {
        border-collapse: collapse;
      }

      table, th
      {
        border: 2px solid;
      }
      td
      {
        border: 2px dashed;
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1>
```

2



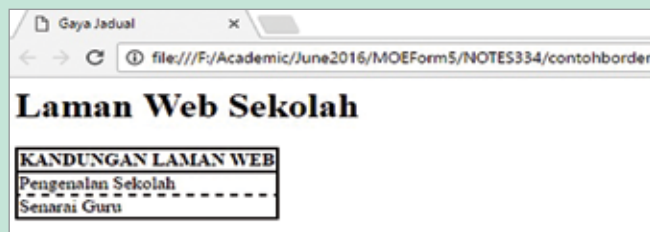
```

<table>
  <tr>
    <th>KANDUNGAN LAMAN WEB</th>
  </tr>
  <tr>
    <td>Pengenalan Sekolah</td>
  </tr>
  <tr>
    <td>Senarai Guru</td>
  </tr>
</table>
</body>
</html>

```

1

Output:



Rajah 3.3.62 Contoh HTML, CSS dan Output (asingkan ketetapan *Border Table*)

Penerangan:

- 1 Terdapat `<table>` pada laman web.
- 2 *Border table* telah ditetapkan dengan mengasingkan ketetapan CSS bagi `<th>`, iaitu `border: 2px solid` dan bagi `<td>` iaitu, `border: 2px dashed`.

• Menetapkan kelebaran dan ketinggian sesuatu table

Sintaks yang berikut menunjukkan lebar dan tinggi sesuatu *table* boleh ditetapkan mengikut kesesuaian sesuatu halaman web.

```

table
{
  width: 100%;
  height: 20%;
}

```

CONTOH 21

Contoh 21 menunjukkan lebar dan tinggi jadual (*table*) ditukar kepada 100% dan 20%, iaitu lebar akan memenuhi keseluruhan dokumen HTML dan tinggi hanya memenuhi 20% daripada dokumen HTML. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.63.



```

<html>
  <head>
    <title> Gaya Jadual</title>
    <style>
      table
      {
        border-collapse: collapse;
        width: 100%; 2
        height: 20%; 3
      }

      table, th, td
      {
        border: 1px solid;
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1>
    <table>
      <tr>
        <th>KANDUNGAN LAMAN WEB</th>
      </tr>
      <tr>
        <td>Pengenalan Sekolah</td>
      </tr>
      <tr>
        <td>Senarai Guru</td>
      </tr>
    </table>
  </body>
</html>

```

Output:



Rajah 3.3.63 Contoh HTML, CSS dan Output (lebar dan tinggi *table*)

Penerangan:

- 1 Terdapat `<table>` pada laman web.
- 2 Lebar `<table>` telah ditetapkan, iaitu `width:100%`.
- 3 Tinggi `<table>` telah ditetapkan, iaitu `height:20%`.

- **Menetapkan jajaran teks di dalam table**

Jajaran teks di dalam *table* juga boleh ditetapkan dan terdapat tiga bentuk jajaran, iaitu:

(a) *left* (b) *center* (c) *right*

```
table
{
  text-align: center;
}
```

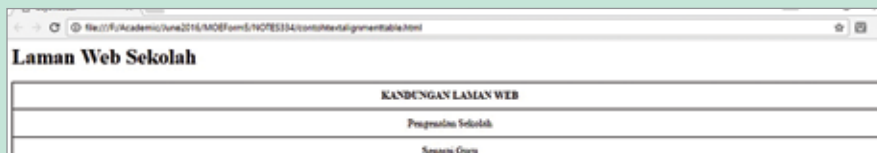
CONTOH 22

Contoh 22 menunjukkan teks dalam jadual (*table*) boleh ditetapkan jajaran kepada center, iaitu berada di tengah-tengah medan <th> atau <td>. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.64.

```
<html>
  <head>
    <title> Gaya Jadual</title>
    <style>
      table
      {
        border-collapse: collapse;
        width: 100%;
        height: 20%;
        text-align: center; 2
      }
      table, th, td
      {
        border: 1px solid;
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1>
    <table>
      <tr>
        <th>KANDUNGAN LAMAN WEB</th>
      </tr>
      <tr>
        <td>Pengenalan Sekolah</td>
      </tr>
      <tr>
        <td>Senarai Guru</td>
      </tr>
    </table>
  </body>
</html>
```

1

Output:



KANDUNGAN LAMAS WEB
Pengenalan Sekolah
Senarai Guru

Rajah 3.3.64 Contoh HTML, CSS dan Output (jajaran teks *table*)

Penerangan:

- 1 Terdapat `<table>` pada laman web.
- 2 Teks dalam `<table>` telah dijajarkan di tangan dengan perletakan CSS `text-align:center`.

- **Menukar warna latar belakang dan warna teks dalam table**

Sintaks yang berikut menukarkan warna latar belakang dan warna teks di dalam *table*.

```
table
{
  background-color: green;
  color: white;
}
```

CONTOH 23

Contoh 23 menunjukkan warna latar belakang jadual (*table*) ditukar kepada warna hijau (*green*) dan warna teks dalam table ditetapkan kepada warna putih. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.65.

```
<html>
<head>
  <title> Gaya Jadual</title>
  <style>
    table
    {
      border-collapse: collapse;
      background-color: green; 2
      color: white; 3
    }
    table, th, td {
      border: 1px solid;
    }
  </style>
</head>
<body>
  <h1>Laman Web Sekolah</h1>
  <table>
    <tr>
      <th>KANDUNGAN LAMAN WEB</th>
    </tr>
    <tr>
      <td>Pengenalan Sekolah</td>
    </tr>
    <tr>
      <td>Senarai Guru</td>
    </tr>
  </table>
</body>
</html>
```

1



Output:



Rajah 3.3.65 Contoh HTML, CSS dan Output (warna latar belakang dan warna teks bagi *Table*)

Penerangan:

- 1 Terdapat `<table>` pada laman web.
- 2 `<table>` mempunyai warna latar belakang hijau dengan ketetapan CSS `background-color: green`.
- 3 `<table>` mempunyai warna teks putih dengan ketetapan CSS `color: white`.

- **Menebalkan jarak di antara medan (*padding*) dalam table**

Jarak di antara satu medan atau satu "`<td>`" di dalam suatu *table* boleh ditambah dengan menetapkan jarak dalam ukuran piksel.

```
table
{
  padding: 20px;
}
```

CONTOH 24

Contoh 24 menunjukkan jarak antara medan jadual (*table*) ditukar atau ditambah kepada 20 piksel. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.66.

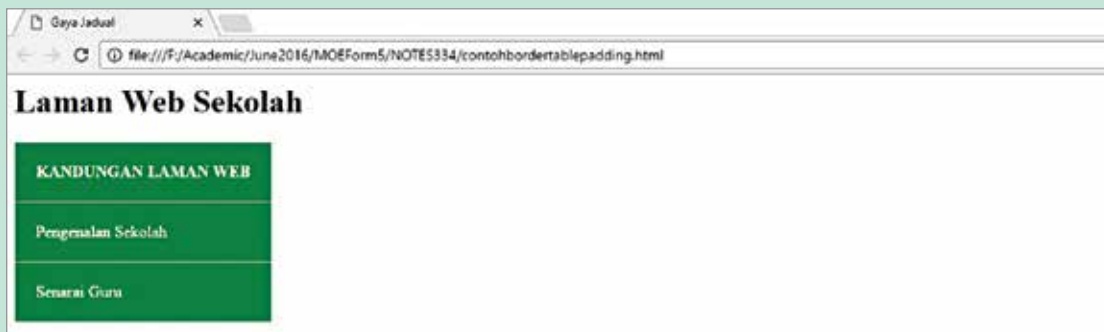
```
<html>
<head>
  <title> Gaya Jadual</title>
  <style>
    table
    {
      border-collapse: collapse;
      background-color: green;
      color: white;
      padding: 20px; 2
    }
    table, th, td
    {
      border: 1px solid;
    }
  </style>
</head>
```

```

<body>
  <h1>Laman Web Sekolah</h1>
  <table>
    <tr>
      <th>KANDUNGAN LAMAN WEB</th>
    </tr>
    <tr>
      <td>Pengenalan Sekolah</td>
    </tr>
    <tr>
      <td>Senarai Guru</td>
    </tr>
  </table>
</body>
</html>

```

Output:



Rajah 3.3.66 Contoh HTML, CSS dan O/P (*Padding Table*)

Penerangan:

- 1 Terdapat `<table>` pada laman web.
- 2 `padding: 20px` digunakan pada jarak di antara medan dalam `<table>`, iaitu dari satu `<td>` ke satu yang lain.

Membina Laman Web HTML yang Mengandungi CSS untuk Memberikan Gaya kepada *Borders*

Bahagian ini akan menunjukkan bagaimana meletakkan garisan luar atau dikenali sebagai *border* bagi suatu bahagian seperti *paragraph* “`<p>`”, *heading* “`<h1>`” dan sebagainya.

Caranya adalah sama sepertimana yang telah dipelajari dalam *table*, iaitu pelajar boleh meletakkan garisan luar atau *border* bagi *table* yang dihasilkan.

```

h1
{
  border-style: solid;
  border-width: 5px;
  border-color: red;
}

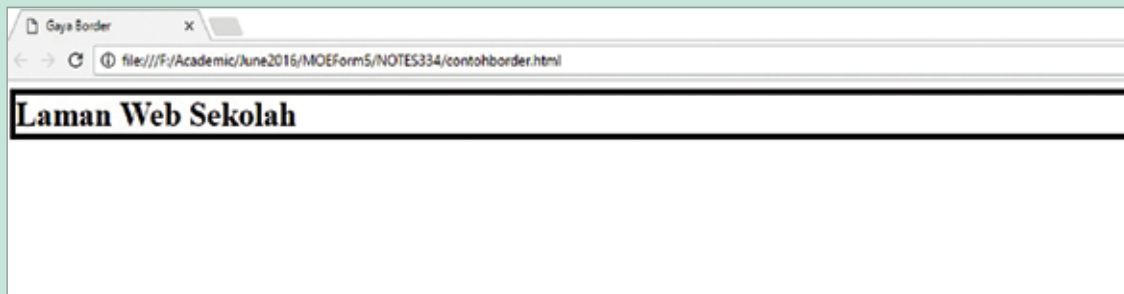
```

CONTOH 25

Contoh 25 menunjukkan heading `<h1>` yang mempunyai border, yaitu garis luar dan jenis garisannya adalah *solid* serta ditetapkan ketebalan border sebanyak 5 piksel. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.67.

```
<html>
  <head>
    <title>Gaya Border</title>
    <style>
      h1
      {
        border-style: solid; 2
        border-width: 5px;
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>
```

Output:



Rajah 3.3.67 Contoh HTML, CSS dan Output (*borders*)

Penerangan:

- 1 Terdapat heading `<h1>` pada laman web.
- 2 `border-style:solid` ialah jenis *border* yang digunakan.

CONTOH 26

Contoh 26 menunjukkan *border* bagi heading `<h1>` yang mempunyai garisan jenis *dashed*. Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.68.

```

<html>
  <head>
    <title>Gaya Border</title>
    <style>
      h1
      {
        border-style: dashed; 2
        border-width: 5px;
        border-color: red;
      }
    </style>
  </head>
  <body>
    <h1>Laman Web Sekolah</h1> 1
  </body>
</html>

```

Output:



Rajah 3.3.68 Contoh HTML, CSS dan Output (jenis *border*)

Penerangan:

- 1 Terdapat heading <h1> pada laman web.
- 2 `border-style: dashed` ialah jenis *border* yang digunakan.

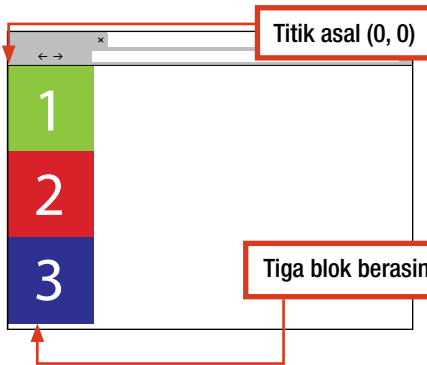
Membina Laman Web HTML yang Mengandungi CSS untuk Memberikan Gaya kepada *Position*

Bahagian ini akan menetapkan kedudukan sesuatu komponen atau tag dalam “<body>” seperti “<h1>”. Kedudukan ditetapkan dengan penetapan komponen-komponen CSS, iaitu bekas atau *container* seperti “<h1>”, jenis kedudukan seperti *relative*, lebar bekas (*width*) dan tinggi bekas (*height*) dalam piksel atau peratus dan kedudukan seperti *top*, *left*, *right* dan *bottom* dalam piksel. Kedudukan bermula daripada nilai-nilai *top* 0, *left* 0, *right* 0 dan *bottom* 0. Terdapat beberapa jenis kedudukan, iaitu:

- **Static**

Kedudukan ini ialah kedudukan lalai setiap elemen HTML yang terdapat dalam dokumen. Kedudukan elemen akan mengikut turutan atau susunan pada laman web. Kedudukan elemen tidak akan berubah atau tidak akan mengikut ketetapan.

Contoh:



Apabila kotak ke-4 ingin ditambah, maka kotak tersebut akan diletakkan di kedudukan yang terbawah, iaitu di bawah kotak ke-3.

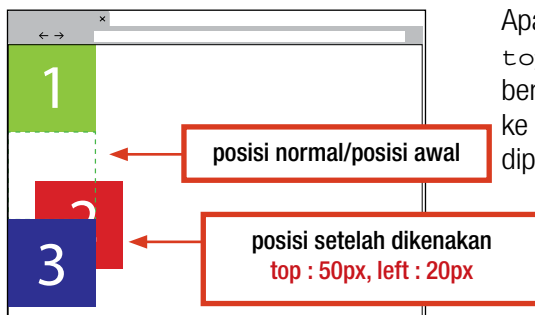
- **Fixed**

Kedudukan elemen bergantung pada *browser viewport*, iaitu pada pandangan pengguna kepada pelayar web. Kedudukan elemen akan tetap di kedudukan yang sama walaupun pengguna *scroll* pada halaman web.

- **Relative**

Kedudukan elemen diubah bergantung atau berkait terhadap kedudukan normalnya atau kedudukan awalnya.

Contoh:

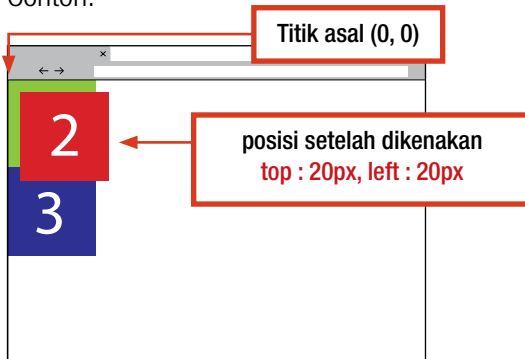


Apabila kedudukan kotak ke-2 ditukar kepada `top : 50px` dan `left : 20px`, maka kotak ke-2 akan berganjak ke bawah sebanyak 50px dan berganjak ke kanan sebanyak 20px. Kotak yang lain tidak akan dipengaruhi oleh perubahan pada kotak ke-2.

- **Absolute**

Kedudukan elemen tidak akan mengikut turutan atau susunan pada laman web lagi dan akan ditempatkan mengikut kedudukan yang dikehendaki.

Contoh:



Kedudukan kotak ke-2 tidak akan mengikut turutan atau susunan dalam laman web dan menuju ke titik asal. Kedudukan kotak ke-2 akan berubah, iaitu berganjak sebanyak 20px ke bawah dan 20px ke kanan yang dikira bermula dari titik asal.

CONTOH 27

Contoh 27 menunjukkan satu dokumen HTML (laman web) yang mempunyai heading `<h1>`, `<h2>`, `<h3>` dan `<h4>` dan boleh berada pada kedudukan yang ditetapkan seperti yang ditunjukkan dalam Rajah 3.3.69.

```
<html>
  <head>
    <title>Tanpa Gaya Position</title>
  </head>
  <body>
    <h1>Laman Web Sekolah (Absolute)</h1>
    <h2>Sekolah 1 (Relative)</h2>
    <h3>Sekolah 2 (Fixed)</h3>
    <h4>Sekolah 3 (Static)</h4>
  </body>
</html>
```

Output:



Rajah 3.3.69 Contoh HTML, CSS dan Output tanpa gaya kedudukan

CONTOH 28

Contoh 28 menunjukkan penukaran kedudukan (*position*) komponen `<h1>`, `<h2>`, `<h3>`, `<h4>` dalam laman web. Atur cara HTML, *JavaScript* dan output ditunjukkan dalam Rajah 3.3.70 dan Rajah 3.3.71.

```
<html>
  <head>
    <title> Gaya Position</title>
    <style>
      h1
      {
        background-color: lightgreen;
        position: absolute;
        width: 40%;
        top: 10px;
        left: 50px;
        height: 100px;
      }
      h2
      {
        background-color: lightpink;
        position: relative;
        top: 70px;
        width: 30%;
        left: 300px;
        height: 100px;
      }
      h3
      {
        background-color: lightblue;
        position: fixed;
        bottom: 70px;
        width: 20%;
        left: 200px;
        height: 100px;
      }
    </style>
  </head>
  <body>
  </body>
</html>
```

2

3

4




```

h4
{
  background-color: lightgrey;
  position: static;
  bottom: 70px;
  width: 10%;
  left: 200px;
  height: 100px;
}
</style>
</head>
<body>
  <h1>Laman Web Sekolah (Absolute)</h1>
  <h2>Sekolah 1 (Relative)</h2>
  <h3>Sekolah 2 (Fixed)</h3>
  <h4>Sekolah 3 (Static)</h4>
</body>
</html>

```

Rajah 3.3.70 Contoh HTML dan CSS (kedudukan)

Output:



Rajah 3.3.71 Contoh Output (kedudukan)

Penerangan:

- 1 Terdapat empat komponen pada laman web, iaitu <h1>, <h2>, <h3>, dan <h4>.
- 2 Gaya *position* <h1> telah ditukar kepada *absolute*, iaitu mengikut penetapan CSS, *top: 10px*, *left: 50px*, *height: 100px* pada kedudukan laman web.
- 3 Gaya *position* <h2> telah ditukar kepada *relative*, iaitu mengikut kedudukan asal dan mengikut penetapan *top: 70px*, *left: 300px*, *height: 100px*.
- 4 Gaya *position* <h3> telah ditukar kepada *fixed*, iaitu kedudukan <h3> sentiasa berada di tempat yang sama walaupun pengguna melihat laman web itu di kawasan yang lain. Ketetapan *position* <h3> ialah *bottom: 70px*, *left: 200px*, *height: 100px*.
- 5 Gaya *position* <h4> telah ditukar kepada *static*, iaitu kedudukan tidak akan berubah atau tidak mengikut ketetapan. Kedudukan itu akan mengikut turutan pada laman web.

Objektif: Murid dapat mengaplikasikan stail CSS yang telah dipelajari.

Lakukan aktiviti ini di dalam makmal komputer.

Bahagian 1

Murid dikehendaki mencipta satu fail HTML yang baharu dan menghasilkan satu laman web tanpa mengaplika sebarang stail dalam CSS. Tag HTML bagi laman web tersebut diberi seperti berikut:

```
<html>
  <head>
    <title>Aplikasi CSS</title>
  </head>
  <body>
    <table>
      <tr>
        <td>Tentang Kami</td>
        <td>Fokus</td>
        <td>Syarat dan Terma</td>
      </tr>
      <tr>
        <td>Peluang Kerjaya</td>
        <td>Halaman Web Berkaitan</td>
        <td>Polisi Privasi</td>
      </tr>
      <tr>
        <td>Hubungi Kami</td>
        <td>eBook</td>
        <td>Maklumat Cookie</td>
      </tr>
      <tr>
        <td>Peta Tapak (Site Map)</td>
        <td>Di Seluruh Dunia</td>
        <td>Hak & Kebenaran</td>
      </tr>
    </table>

    <table>
      <tr>
        <td>© 2016 Aktiviti CSS. (001111-T). Hakcipta Terpelihara.</td>
      </tr>
    </table>
  </body>
</html>
```

Bahagian 2

Seterusnya, murid akan menambah atau mengubah suai fail HTML tersebut dengan menambah stail CSS yang menukar gaya *text*, *font*, *background*, *table* dan *position*.

Membina Laman Web Pengguna yang Interaktif, yang Mengandungi Fitur Pengesahan Data, Butang Pilihan dan Kotak *Popup*

Laman web pengguna interaktif ialah suatu halaman web yang membenarkan pengunjung laman web berinteraksi dengan kandungan dalam laman web. Antara aktiviti yang boleh dilakukan adalah seperti memberikan komen atau meninggalkan nombor untuk dihubungi dalam borang yang disediakan di dalam laman web.

Aktiviti lain yang boleh dilaksanakan adalah seperti pengesahan data pada borang, membuat pilihan daripada pelbagai butang pilihan dan bertindak balas terhadap kotak *popup* atau kotak dialog yang terdapat dalam suatu laman web.

Topik ini akan menerangkan dan menunjukkan bagaimana bahasa penskripan klien, iaitu *JavaScript* dapat menerima pelaksanaan interaksi dengan laman web dan memproses permintaan pada peringkat pengguna.

Pembinaan Laman Web Pengguna yang Interaktif yang Mengandungi Fitur Pengesahan Data

Dalam suatu laman web, data dapat disahkan pada borang atau medan yang disediakan. Contohnya, data untuk nama, alamat dan nombor pengenalan.

CONTOH 29

Contoh 29 menunjukkan atur cara yang akan menyemak sama ada data dalam medan teks telah dimasukkan sebelum menekan butang "Pengesahan Nama". Atur cara HTML, CSS dan output ditunjukkan dalam Rajah 3.3.72.

```
<html>
  <body>
    Nama Anda:
    <input id="n1"> 1
    <button type="button" onclick="semakData()">Pengesahan Nama</button> 2
    <script>
      function semakData() 3
      {
        var teksnama;
        teksnama = document.getElementById("n1").value; 4
        if (teksnama==" ")
        {
          window.alert("Sila masukkan nama anda"); 5
        }
      }
    </script>
  </body>
</html>
```

Output:

Setelah butang "Pengesahan Nama" ditekan tanpa memasukkan sebarang data dalam medan teks "Nama Anda:", yang berikut ialah tindak balas atur cara:



Rajah 3.3.72 Contoh HTML, JavaScript dan Output (pengesahan data)

Penerangan:

- 1 Satu Medan teks disediakan supaya pengguna boleh memasukkan nama.
- 2 Butang `onclick` bagi "Pengesahan Nama" disediakan.
- 3 Jika nama tidak dimasukkan dan butang diklik, subatur cara `semakData ()` dipanggil.
- 4 Nilai nama pada medan teks "Nama Anda:" akan dibaca dan disemak dengan menggunakan pernyataan `if`.
- 5 Jika medan teks kosong, iaitu tiada nama dimasukkan, kotak *popup alert* berserta mesej "Sila masukkan nama anda" akan dipaparkan sebagai peringatan kepada pengguna untuk memasukkan nama.

CONTOH 30

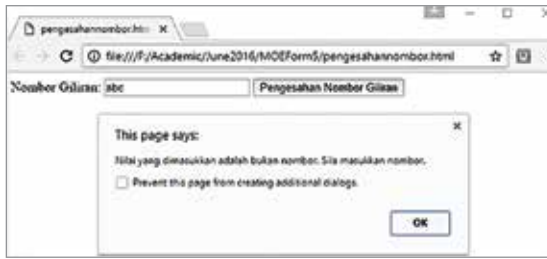
Contoh 30 menunjukkan atur cara yang akan menyemak sama ada data di dalam medan teks adalah suatu nombor atau bukan nombor. Sekiranya pengguna memasukkan data bukan nombor, suatu mesej akan dipaparkan dengan menggunakan fungsi `window.alert ()`. Atur cara HTML, JavaScript dan output ditunjukkan dalam Rajah 3.3.73.

```
<html>
  <body>
    Nombor Giliran:
    <input id="n2"> 1          2
    <button type="button" onclick="semakNombor()">Pengesahan Nombor Giliran
  </button>

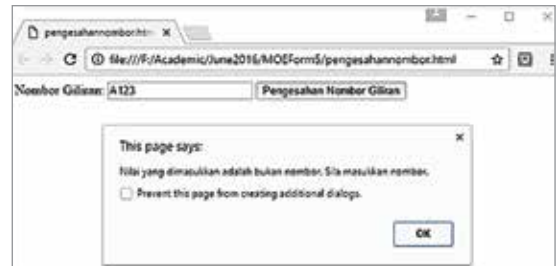
  <script>
    function semakNombor() 3
    {
      var teksnombor;
      teksnombor = document.getElementById("n2").value; } 4
      if (isNaN(teksnombor))
      {
        window.alert("Nilai yang dimasukkan adalah bukan nombor. Sila } 5
          masukkan nombor.");
      }
    }
  </script>
</body>
</html>
```

Output:

Teks sahaja



Gabungan teks dan nombor



Nombor sahaja



Rajah 3.3.73 Contoh HTML, JavaScript dan Output (pengesahan nombor atau bukan nombor)

Penerangan:

- 1 Satu medan teks disediakan supaya pengguna boleh memasukkan nombor giliran.
- 2 Butang `onClick` bagi “Pengesahan Nombor Giliran” disediakan.
- 3 Setelah nombor giliran dimasukkan dan butang diklik, subatur cara `semakNombor()` dinyatakan.
- 4 Nilai nombor giliran pada medan teks dibaca dan disemak dengan menggunakan pernyataan `if`.
- 5 Jika nilai nombor giliran bukan nombor (semakan menggunakan fungsi `isNaN`), kotak *popup alert* berserta mesej “Nilai yang dimasukkan adalah bukan nombor. Sila masukkan nombor.” akan dipaparkan. Sebaliknya jika nilai nombor giliran ialah nombor, tiada arahan lain dilaksanakan.

Membina Laman Web Pengguna yang Interaktif yang mengandungi Fitur Butang Pilihan

Dalam suatu laman web, pengguna boleh juga berinteraksi dengan membuat pilihan yang telah disediakan dalam laman web tersebut. Oleh sebab itu, untuk membangunkan suatu laman web, terdapat pelbagai kaedah bagi menghasilkan butang pilihan yang disediakan oleh suatu laman web. Antara butang pilihan yang disediakan adalah seperti butang radio dan butang *checkbox*.

• Butang radio

Butang radio akan menyediakan lebih daripada satu pilihan dan pengguna perlu memilih salah satu sahaja pilihan tersebut.

Butang radio boleh dihasilkan dalam “`<form>`” dan butang radio ialah sejenis input dalam “form” (`input type = "radio"`). Nama setiap butang radio boleh diberikan, iaitu dengan format `name: "nama_butang_radio"`. Jika nama pada atribut *name* adalah sama, maka kedua-dua butang radio adalah dalam kumpulan yang sama. Hanya salah satu butang yang boleh dipilih pada satu masa. *id* pula merupakan nama unik yang diberikan kepada setiap butang radio

sebagai pengenalan diri dan boleh dipanggil dalam atur cara penskripan untuk tujuan capaian atau semakan.

Contoh 31 menunjukkan bagaimana penghasilan butang radio dalam HTML dan tindak balas terhadap pilihan pada butang radio yang dilaksanakan oleh *JavaScript*, iaitu pada peringkat bahasa penskripan klien.

CONTOH 31


Contoh 31 menunjukkan atur cara yang membuat pengesahan terhadap pilihan daripada butang radio. Terdapat dua butang radio, iaitu "SEPENUH MASA" dan "SEPARUH MASA". Apabila butang radio diklik, atur cara memberikan nilai `true`, iaitu butang tersebut dipilih dan menjadi input kepada atur cara. Terdapat satu butang "Pengesahan Mod Pengajian" yang akan memanggil prosedur `semakMod()` untuk dilaksanakan dan menyemak butang radio yang dipilih. Mesej "Mod Pengajian: SEPENUH MASA" akan dipaparkan jika butang radio pertama dipilih, dan mesej "Mod Pengajian: SEPARUH MASA" akan dipaparkan jika butang radio kedua dipilih. Atur cara HTML, *JavaScript* dan output ditunjukkan pada Rajah 3.3.74.

```
<html>
  <body>
    <b>MOD PENGAJIAN</b>
    <form action="">
      <input type="radio" name="mod" id="sepenuh">Sepenuh Masa } 1
      <input type="radio" name="mod" id="separuh">Separuh Masa } 1
    <p>
      <button type="button" onclick="semakMod()">Pengesahan Mod Pengajian } 2
    </button>
    </p>
    </form>
    <script>
      function semakMod() } 3
      {
        if (document.getElementById('sepenuh').checked == true) } 4
        {
          window.alert("Mod Pengajian: SEPENUH MASA"); } 5
        }
        else
        {
          if (document.getElementById('separuh').checked == true) } 6
          {
            window.alert("Mod Pengajian: SEPARUH MASA"); } 6
          }
        }
        document.getElementById('sepenuh').checked=false;
        document.getElementById('separuh').checked=false;
      }
    </script>
  </body>
</html>
```

Jika nama pada atribut *name* adalah sama, maka kedua-dua butang radio adalah dalam kumpulan yang sama. Hanya salah satu butang yang boleh dipilih pada satu masa.



Output:

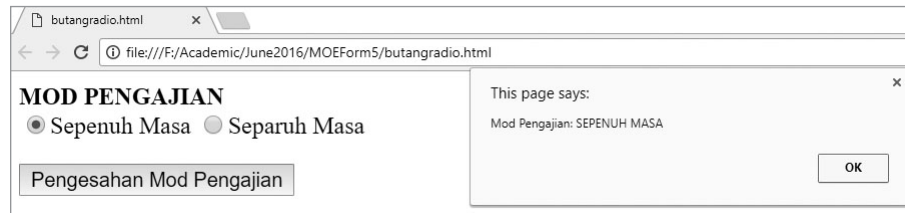


butangradio.html x
file:///F:/Academic/June2016/MOEForm5/bu

MOD PENGAJIAN
 Sepenuh Masa Separuh Masa

Pengesahan Mod Pengajian

Pilihan 1



butangradio.html x
file:///F:/Academic/June2016/MOEForm5/butangradio.html

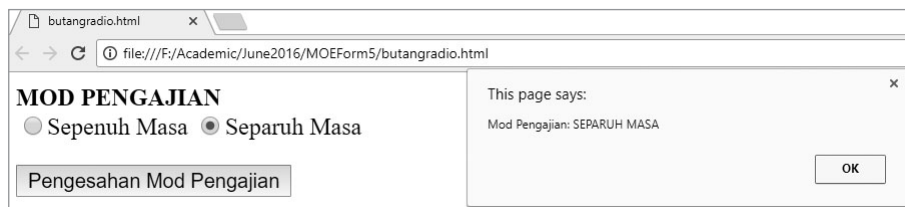
MOD PENGAJIAN
 Sepenuh Masa Separuh Masa

Pengesahan Mod Pengajian

This page says:
Mod Pengajian: SEPENUH MASA

OK

Pilihan 2



butangradio.html x
file:///F:/Academic/June2016/MOEForm5/butangradio.html

MOD PENGAJIAN
 Sepenuh Masa Separuh Masa

Pengesahan Mod Pengajian

This page says:
Mod Pengajian: SEPARUH MASA

OK

Rajah 3.3.74 Contoh HTML, JavaScript dan Output (Butang Radio)

Penerangan:

- 1 Dua butang radio bagi mod Pengajian dipaparkan, iaitu "sepenuh masa" dan "separuh masa".
- 2 Butang `onClick` bagi "Pengesahan Mod Pengajian" disediakan.
- 3 Apabila butang "Pengesahan Mod Pengajian" diklik, subatur cara `semakMod()` dipanggil.
- 4 Pilihan butang radio mod pengajian disemak dengan menggunakan pernyataan `if`.
- 5 Jika pengguna memilih butang radio `id="sepenuh"`, kotak *pop up alert* berserta mesej "Mod Pengajian: SEPENUH MASA" akan dipaparkan sebagai peringatan.
- 6 Sebaliknya, jika pengguna memilih butang radio `id="separuh"`, kotak *pop up alert* berserta mesej "Mod Pengajian: SEPARUH MASA" akan dipaparkan sebagai peringatan.

• Butang *checkbox*

Butang *checkbox* pula akan menyediakan beberapa pilihan dan pengguna boleh memilih lebih daripada satu pilihan.

Butang *checkbox* boleh dihasilkan dalam "`<form>`" dan butang *checkbox* juga merupakan sejenis input dalam form (`input type="checkbox"`). Jika nama pada atribut `name` adalah sama, maka kedua-dua butang *checkbox* adalah dalam kumpulan yang sama. Tidak seperti butang radio, lebih daripada satu butang boleh dipilih pada satu masa bagi butang *checkbox*. `id` pula merupakan nama unik yang diberikan kepada setiap butang *checkbox* sebagai pengenalan diri dan boleh dipanggil dalam atur cara penskripan untuk tujuan capaian atau semakan.

Contoh di bawah menunjukkan bagaimana penghasilan butang *checkbox* dalam HTML dan tindak balas terhadap pilihan pada butang *checkbox* yang dilaksanakan oleh JavaScript, iaitu pada peringkat bahasa penskripan klien.

CONTOH 32

Contoh 32 menunjukkan atur cara menggunakan butang *checkbox* bagi menyediakan senarai pilihan subjek dan setelah butang "Pengesahan Subjek" ditekan, subjek-subjek yang dipilih dimasukkan ke dalam kawasan medan teks yang dipanggil *text area*.

```
<html>
  <body>
    <b>SUBJEK-SUBJEK YANG INGIN DIDAFITAR</b>
    <form action="" name="MyForm">
      <input type="checkbox" name="sub" id="Matematik">Matematik
      <input type="checkbox" name="sub" id="English">English
      <input type="checkbox" name="sub" id="Sains">Sains
      <input type="checkbox" name="sub" id="Bahasa Malaysia">Bahasa Malaysia
      <p><button type="button" onclick="semakSubjek()">Pengesahan Subjek
      </button></p>
      <p><textarea id="SenaraiSubjek" rows="10" cols="30" value="b">
      </textarea></p>
      <p><button type="button" onclick="resetSubjek()">Reset Subjek
      </button></p>
    </form>

    <script>
      function semakSubjek()
      {
        var elLength = document.MyForm.elements.length;
        for (i=0; i<elLength; i++)
        {
          var type = MyForm.elements[i].type;
          if (type=="checkbox" && MyForm.elements[i].checked)
          {
            document.getElementById("SenaraiSubjek").value +=
              MyForm.elements[i].id+'\n';
          }
        }
      }

      function resetSubjek()
      {
        var elLength = document.MyForm.elements.length;
        document.getElementById("SenaraiSubjek").value = " ";
        for (i=0; i<elLength; i++)
        {
          MyForm.elements[i].checked=false;
        }
      }
    </script>
  </body>
</html>
```

1

2

3

4

5



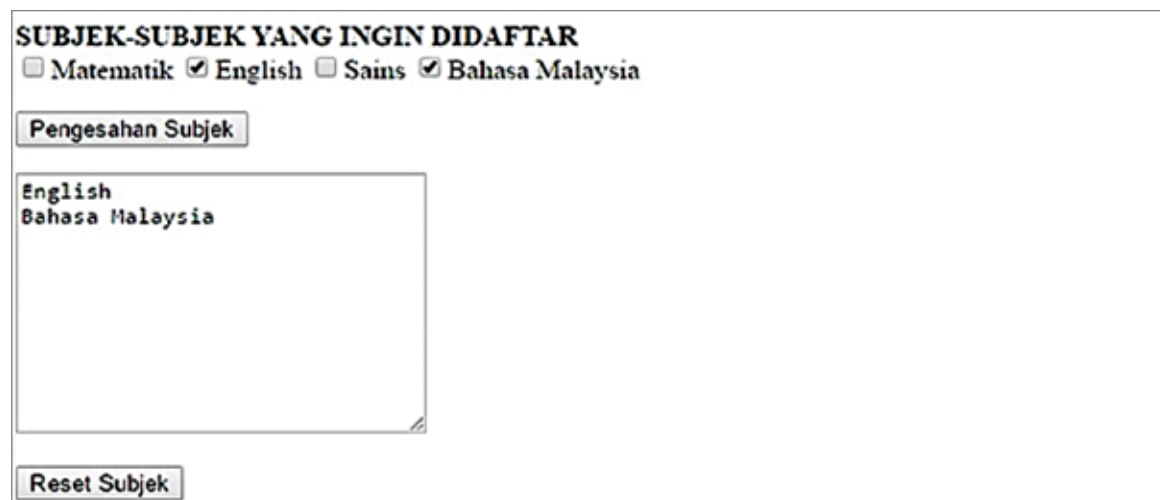
Output:

Sebelum klik pengesahan subjek



The screenshot shows a web form titled "SUBJEK-SUBJEK YANG INGIN DIDAFTAR". It contains four checkboxes: "Matematik", "English", "Sains", and "Bahasa Malaysia", all of which are unchecked. Below the checkboxes is a button labeled "Pengesahan Subjek". Underneath the button is a large, empty text area. At the bottom of the form is a button labeled "Reset Subjek".

Selepas klik pengesahan subjek



The screenshot shows the same web form after the "Pengesahan Subjek" button has been clicked. The checkboxes for "English" and "Bahasa Malaysia" are now checked, while "Matematik" and "Sains" remain unchecked. The text area below the button now contains the text "English" and "Bahasa Malaysia" on separate lines. The "Reset Subjek" button remains at the bottom.

Rajah 3.3.75 Contoh HTML, JavaScript dan Output (Checkbox)

Penerangan:

- 1 Empat *checkbox* bagi pilihan subjek dipaparkan.
- 2 Butang `onclick` bagi "Pengesahan Subjek" dan "Reset Subjek" disediakan.
- 3 Apabila butang "Pengesahan Subjek" diklik, subatur cara `semakSubjek()` dipanggil. Semua subjek yang dipilih pada checkbox akan disimpan dalam borang "MyForm" dan dipaparkan pada `textarea` `id="Senarai Subjek"` yang disediakan.
- 4 Apabila butang "Reset Subjek" diklik, subatur cara `resetSubjek()` dipanggil.
- 5 `resetSubjek()` akan mengosongkan borang "MyForm" dan `textarea`.

Membina Laman Web Pengguna yang Interaktif yang mengandungi Fitur Kotak *Popup*

JavaScript mempunyai tiga jenis kotak *popup*, iaitu kotak *Alert*, kotak *Confirm* dan kotak *Prompt*.

• Kotak *Alert*

Kotak *Alert* ini akan dipaparkan dan digunakan jika ingin mendapatkan kepastian maklumat yang datang daripada pengguna atau memberikan maklum balas kepada pengguna. Sintaks bagi menghasilkan kotak *Alert* ialah `window.alert(" ")`.

CONTOH 33

Contoh 33 menunjukkan atur cara perlu menerima nama pengguna dan mengingatkan pengguna untuk memasukkan nama dengan memaparkan mesej pada kotak *Alert*. Atur cara HTML, *JavaScript* dan output ditunjukkan dalam Rajah 3.3.76.

```
<html>
  <body>
    Nama :
    <input type="text" > 1

    <script>
      alertMasuk(); 2
      function alertMasuk()
      {
        window.alert("Anda perlu memasukkan nama anda."); 3
      }
    </script>
  </body>
</html>
```

Output:



Rajah 3.3.76 Contoh HTML, *JavaScript* dan Output (Kotak *Alert*)

Penerangan:

- 1 Satu medan teks disediakan supaya pengguna boleh memasukkan nama.
- 2 Kotak *popup alert* terus dipaparkan sebaik sahaja laman web dibuka kerana subaturcara `alertMasuk()` dipanggil.
- 3 Mesej "Anda perlu memasukkan nama anda." dipaparkan untuk mengingatkan pengguna supaya memasukkan nama.

- **Kotak Confirm**

Kotak *Confirm* ini akan digunakan jika pengguna ingin mengesahkan sesuatu maklumat. Sintaks bagi menghasilkan kotak *Confirm* ialah `window.confirm(" ")`.

CONTOH 34

Contoh 34 menunjukkan atur cara yang menerima nama pengguna dan mengesahkan nama yang ingin dihantar dengan menggunakan kotak *Confirm*. Atur cara HTML, *JavaScript* dan output ditunjukkan dalam Rajah 3.3.77.

```
<html>
  <body>
    Nama:
    <input type="text" > 1
    <button onclick="pastiHantar()">Hantar</button> 2
    <script>
      function pastiHantar() 3
      {
        if (window.confirm("Anda pasti?") == true) 4
        {
          document.write("Maklumat telah dihantar."); 5
        }
      }
    </script>
  </body>
</html>
```

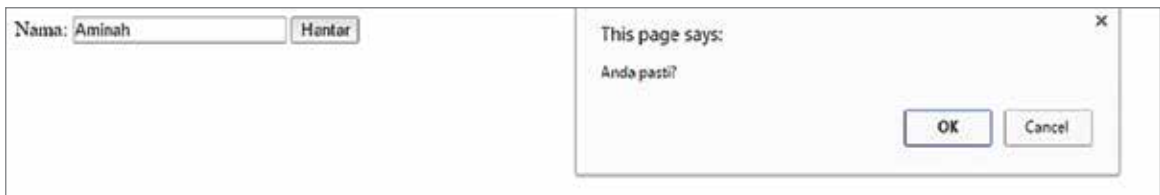
Output:

Pengguna input nama dan klik butang "Hantar "



The screenshot shows a web form with a text input field containing the name 'Aminah' and a button labeled 'Hantar' to its right.

Kotak *Confirm* dengan teks "Anda pasti?" di paparkan untuk pengesahan



The screenshot shows the web form from the previous image, but with a modal dialog box overlaid on top. The dialog box has a title bar that says 'This page says:' and contains the text 'Anda pasti?'. At the bottom of the dialog box are two buttons: 'OK' and 'Cancel'.

Setelah butang "OK" diklik, mesej "Maklumat telah dihantar" dipaparkan. Jika butang "Cancel" diklik, tiada mesej dipaparkan dan paparan sebelum dikekalkan.



The screenshot shows the web form with the text 'Maklumat telah dihantar.' displayed in the input field area.

Rajah 3.3.77 Contoh HTML, *JavaScript* dan Output (Kotak *Confirm*)



Penerangan:

- 1 Satu medan teks disediakan supaya pengguna boleh memasukkan nama.
- 2 Butang `onClick` bagi "Hantar" disediakan.
- 3 Apabila butang "Hantar" diklik, subatur cara `pastiHantar()` dipanggil dan kotak *popup confirm* akan dipaparkan.
- 4 Jika butang "OK", diklik maklumat "Anda Pasti" adalah benar dan disemak dengan menggunakan perkataan `if`.
- 5 Mesej "Maklumat telah dihantar." akan dipaparkan. Jika butang "Cancel" diklik paparan yang sama akan dikekalkan.

• Kotak *Prompt*

Kotak *Prompt* ini akan digunakan jika pengguna ingin memasukkan satu nilai sebelum memasuki suatu laman web. Sintaks bagi menghasilkan kotak *Prompt* ialah `window.prompt(" ")`.

CONTOH 35

Contoh 35 menunjukkan atur cara yang memasukkan nama pengguna melalui penggunaan kotak *Prompt*.

Penyelesaian:

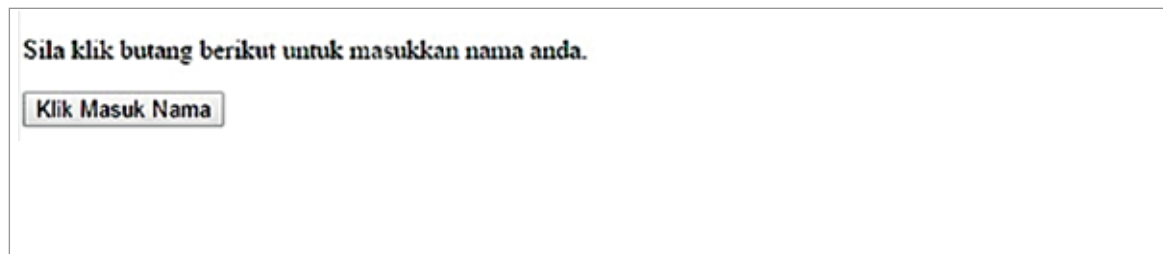
```
<html>
  <body>
    <p>Sila klik butang berikut untuk masukkan nama anda.</p>
    <button onclick="masukNama()">Klik Masuk Nama</button> 1
    <p id="maklumat"> </p>
    <script>
      function masukNama()
      {
        var pengguna = window.prompt("Sila masukkan nama anda:", ""); } 2

        if (pengguna != null) 3
        {
          document.getElementById("maklumat").innerHTML = } 4
          "Hai " + pengguna + "! Apa khabar hari ini?";
        }
      }
    </script>
  </body>
</html>
```



Output:

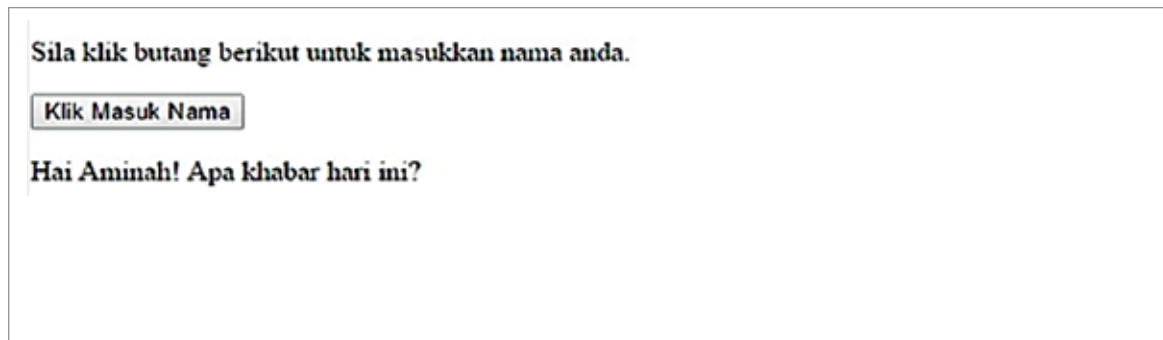
Butang perlu diklik



Paparan kotak Prompt



Setelah butang "OK" diklik, mesej "Hai Aminah! Apa khabar hari ini?" dipaparkan.



Rajah 3.3.78 Contoh HTML, JavaScript dan Output (Kotak Prompt)

Penerangan:

- 1 Satu butang `onclick`, iaitu "Klik Masuk Nama" disediakan.
- 2 Apabila butang diklik, subatur cara `masukNama()` dipanggil dan kotak `pop up prompt` dipaparkan untuk meminta pengguna memasukkan nama.
- 3 Jika klik "OK", medan teks nama akan disemak dengan menggunakan pernyataan `if`. Medan teks disemak sama ada nama dimasukkan ataupun tidak.
- 4 Jika nama dimasukkan atau medan teks tidak kosong, mesej "Hai Aminah! Apa khabar hari ini?" dipaparkan. Jika nama tidak dimasukkan, iaitu medan teks kosong, butang `pop up prompt` kekal dipaparkan kecuali butang "Cancel" diklik.

Penggunaan Bahasa Penskripan Pelayan untuk Membina Laman Web yang Boleh Mencapai dan Mengemas kini Data dalam Pangkalan Data

Laman web yang interaktif bermaksud suatu laman web yang mempunyai kandungan yang dinamik dan sesuai atas sesuatu carian atau permintaan daripada pengguna. Kandungan laman web tersebut boleh diubah daripada laman web tertentu atau hasil daripada pemprosesan komputer pelayan yang dipaparkan berbeza mengikut permintaan dari pengguna. Antara ciri laman web yang interaktif ialah:

- (i) menggunakan bahasa pengaturcaraan web seperti PHP, ASP dan lain-lain.
- (ii) mempunyai capaian kepada pangkalan data.
- (iii) kandungan laman web berubah mengikut permintaan pengguna.
- (iv) kandungan laman web sering dikemaskinikan.

Contoh laman web yang interaktif adalah seperti laman web yang memaparkan maklumat biodata murid. Maklumat murid terlebih dahulu telah disimpan dalam komputer pelayan. Apabila kita ingin memaparkan maklumat murid tertentu, kita hanya perlu memasukkan maklumat carian seperti nombor murid dan seterusnya laman web akan memaparkan maklumat biodata murid tersebut berdasarkan kepada nombor murid yang telah dimasukkan. Rajah 3.3.79 menunjukkan elemen-elemen penting yang terdapat dalam laman web interaktif.

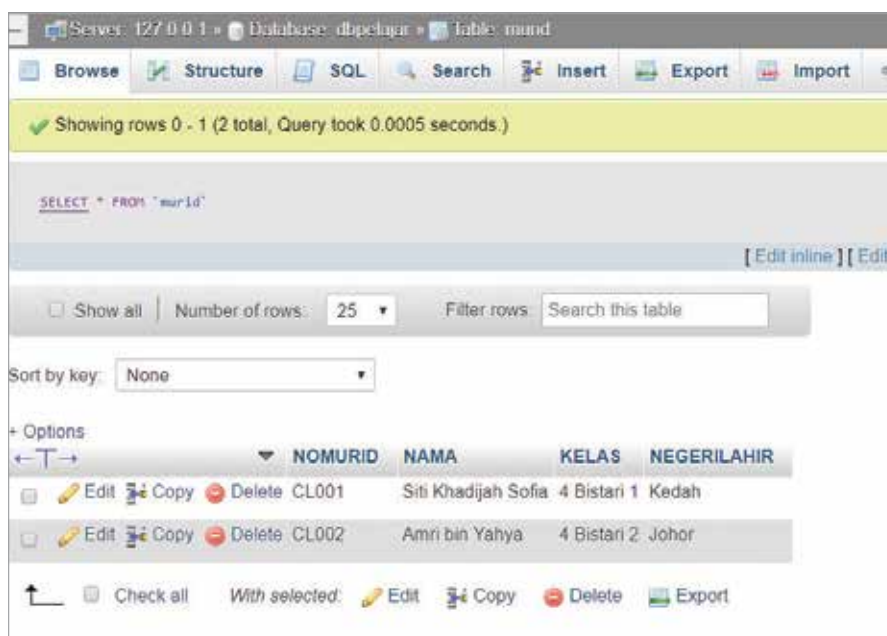


Rajah 3.3.79 Elemen-elemen penting dalam laman web interaktif

Data yang disimpan dalam pangkalan data dengan menggunakan MySQL boleh dicapai dan dikemas kini melalui laman web. Untuk tujuan tersebut, sambungan kepada pangkalan data perlu dilakukan terlebih dahulu sebelum data dapat dicapai dan dikemaskinikan.

Membina Laman Web yang dapat Mencapai Data dalam Pangkalan Data

Bagi membina laman web yang dapat mencapai data dalam pangkalan data, kita memerlukan jadual (*table*) dalam pangkalan data tersebut. Data daripada jadual akan dicapai dengan menggunakan arahan kendalian ke atas jadual. Arahan kendalian SELECT akan digunakan. Yang berikut ialah format arahan kendalian terhadap jadual. SELECT lajur FROM jadual. Katakan kita mempunyai satu jadual yang dinamakan **MURID** dan mempunyai tiga lajur. Lajur ini akan dinamakan **NAMA**, **KELAS** dan **NEGERILAHIR**. Rajah 3.3.80 menunjukkan jadual yang bernama Murid.



The screenshot shows a MySQL database interface with the following details:

- Server: 127.0.0.1, Database: dbpelajar, Table: murid
- Query: `SELECT * FROM 'murid'`
- Showing rows 0 - 1 (2 total, Query took 0.0005 seconds.)
- Number of rows: 25
- Sort by key: None
- Table columns: NOMURID, NAMA, KELAS, NEGERILAHIR
- Table data:

NOMURID	NAMA	KELAS	NEGERILAHIR
CL001	Siti Khadijah Sofia	4 Bistari 1	Kedah
CL002	Amri bin Yahya	4 Bistari 2	Johor

Rajah 3.3.80 Contoh jadual yang bernama Murid

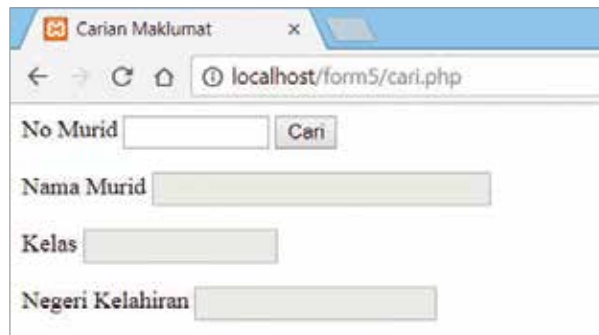
Arahan 'SELECT NAMA, KELAS, NEGERILAHIR FROM MURID' boleh digunakan. Sekiranya terdapat banyak lajur di dalam jadual, kita juga boleh menggunakan arahan seperti 'SELECT * FROM MURID'. Penggunaan simbol '*' merujuk pemilihan semua lajur yang terdapat dalam jadual tersebut. Rajah 3.3.81(a) menunjukkan laman web yang akan mencapai data dalam pangkalan data.

Rajah 3.3.81(a) menunjukkan atur cara bagi laman web Cari.php yang membenarkan kita memasukkan maklumat seperti no murid untuk tujuan pencarian data dalam pangkalan data manakala paparan outputnya seperti yang ditunjukkan pada Rajah 3.3.81(b). Selepas no murid dimasukkan dan butang 'Cari' di klik maklumat pelajar seperti nama, kelas dan negeri kelahiran, atur cara bagi laman web Papar.php akan dilaksanakan. Atur cara bagi laman web Papar.php ini adalah seperti yang ditunjukkan pada Rajah 3.3.82(a) manakala paparan outputnya seperti yang ditunjukkan pada Rajah 3.3.82(b).

Cari.php

```
<html>
<head>
  <title>Carian Maklumat</title>
</head>
<body>
  <form action = "Papar.php" method="POST">
    <p>No Murid <input name = "nomurid" type = "text" size = "10">
      <input Type = "submit" Name = "submit" Value = "Cari"> </p>
    <p>Nama Murid <input name = "nama" type = "text" size = "30"
      disabled = true></p>
    <p>Kelas <input name = "kelas" type = "text" size = "15"
      disabled = true></p>
    <p>Negeri Kelahiran <input name = "negeri" type="text" size = "20"
      disabled = true></p>
  </form>
</body>
</html>
```

Rajah 3.3.81(a) Atur cara laman web "Cari.php"



Rajah 3.3.81(b) Laman web "Cari.php"

Papar.php

```
<html>
<head>
  <title>Papar Rekod</title>
</head>
<body>
<?php
$nomurid = $_POST['nomurid'];
$con = mysqli_connect("localhost","root","");
if (!$con)
{
  die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
}
mysqli_select_db($con,"dbPelajar");
$hasil = mysqli_query($con,"SELECT * FROM MURID
  WHERE NOMURID = '". $nomurid. "'");
$row = mysqli_fetch_array($hasil);
$name = htmlspecialchars($row['NAMA'], ENT_QUOTES);
```



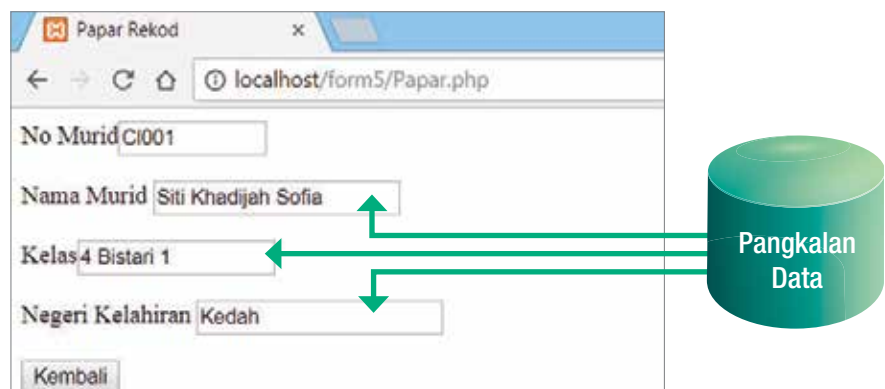

```

$kelas = $row['KELAS'];
$negeri =$row['NEGERILAHIR'];
?>
<form action = "Cari.php" method = "">
  <p>No Murid<input name = "nomurid" readonly value = "
  <?php print $nomurid;?> " type = "text" size = "10"></p>
  <p>Nama Murid
    <input name = "nama" readonly type = "text" value = "
    <?php print $nama;?>">
  </p>
  <p>Kelas<input name = "kelas" readonly type = "text" size = "15"
  value = "<?php print $kelas; ?>"></p>
  <p>Negeri Kelahiran <input name = "negeri" readonly
  type = "text" size = "20" value = "<?php print $negeri;?>">
  </p>
  <p><input Type = "submit" Name = "submit" Value = "Kembali"></p>
</form>

```

Rajah 3.3.82(a) Atur cara bagi laman web “Papar.php”

Dalam laman web ini (Cari.php), kita dikehendaki memasukkan no murid. Selepas butang ‘Cari’ diklik maklumat murid seperti nama, kelas dan negeri kelahiran akan dipaparkan dalam laman web ‘papar.php’. Maklumat murid ini diperoleh daripada pangkalan data yang disimpan dalam jadual MURID.



Rajah 3.3.82(b) Laman web “Papar.php”

Membina Laman Web yang dapat Mengemas kini Data dalam Pangkalan Data

Bagi tujuan mengemas kini data dalam pangkalan data, kita memerlukan kunci dalam jadual yang dapat membezakan setiap rekod dengan rekod yang lain. Katakan dalam jadual MURID terdapat satu lajur yang bernama NOMURID. Arahan kendalian terhadap jadual untuk tujuan kemas kini data ialah ‘UPDATE’

Yang berikut ialah format arahan kendalian terhadap Jadual untuk tujuan kemas kini data.

```

UPDATE medan FROM Jadual
UPDATE Jadual SET medan=nilaibaru1,medan2=nilaibaru2,...WHERE medan_kunci=nilai;

```

Katakan dalam jadual MURID terdapat nama-nama murid. Kita ingin mengemas kini data negeri kelahiran murid yang bernama SITI KHADIJAH kepada negeri JOHOR.

NOMURID	NAMA	KELAS	NEGERILAHIR
CL001	Siti Khadijah Sofia	4 Bistari 1	Kedah
CL002	Amri bin Yahya	4 Bistari 2	Johor

Rajah 3.3.83 Jadual Murid

Arahan yang akan digunakan adalah seperti yang berikut:

```
UPDATE MURID SET NEGERILAHIR = 'JOHOR' WHERE NOMURID = 'CL001';
```

Kita memerlukan beberapa laman web untuk melakukan tugas pengemaskinian data dalam pangkalan data. Jadual 3.3.2 menunjukkan laman web yang perlu kita bina.

Jadual 3.3.2 Laman web yang perlu dibina

Laman web	Tujuan
Senarai.php	Bertujuan untuk menyenaraikan semua maklumat pelajar daripada pangkalan data. Laman web ini akan membenarkan kita memilih rekod yang ingin kita kemaskinikan datanya. Atur cara bagi laman web Senarai.php ini seperti yang ditunjukkan pada Rajah 3.3.83(b) manakala paparan outputnya seperti pada Rajah 3.3.83(a).
Kemaskini.php	Bertujuan untuk memaparkan maklumat dalam bentuk borang, iaitu maklumat yang diperoleh daripada pangkalan data akan dipaparkan untuk tujuan pengemaskinian. Aturcara bagi laman web Kemaskini.php ini adalah seperti yang ditunjukkan pada Rajah 3.3.84(b) manakala paparan outputnya seperti pada Rajah 3.3.84(a).
ProsesKemiskini.php	Bertujuan untuk menyimpan maklumat yang telah dikemas kini kembali ke dalam pangkalan data. Aturcara bagi laman web ProsesKemiskini.php ini adalah seperti yang ditunjukkan pada Rajah 3.3.84(c).

Senarai Maklumat Murid Untuk Dikemas kini

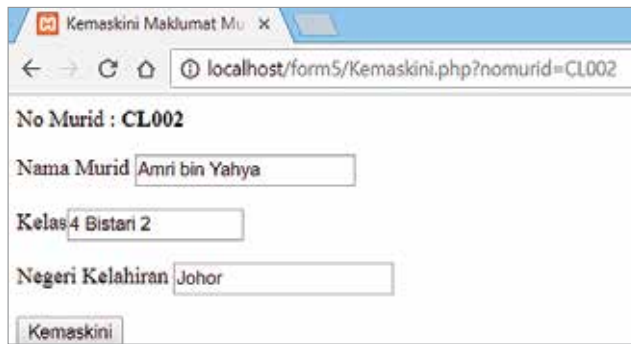
No Murid	Nama	Kelas	Negeri Kelahiran	Tindakan
CL001	Siti Khadijah Sofia	4 Bistari 1	Kedah	Kemaskini
CL002	Amri bin Yahya	4 Bistari 2	Johor	Kemaskini

Rajah 3.3.83(a) Output 'Senarai.php'

Senarai.php

```
<html>
  <head>
    <title>Papar Rekod</title>
  </head>
  <body>
    <?php
      $nomurid = $_POST['nomurid'];
      $con = mysqli_connect("localhost","root","");
      if (!$con)
      {
        die('Sambungan kepada Pangkalan Data Gagal'
          .mysqli_connect_error());
      }
      mysqli_select_db($con,"dbPelajar");
      $hasil = mysqli_query($con,"SELECT * FROM MURID WHERE
        NOMURID ='".$nomurid."'");
      $row = mysqli_fetch_array($hasil);
      $nama = htmlspecialchars($row['NAMA'],ENT_QUOTES);
      $kelas = $row['KELAS'];
      $negeri=$row['NEGERILAHIR'];
    ?>
    <form action = "Cari.php" method = "">
      <p>No Murid<input name = "nomurid" readonly
        value = "<?php print$nomurid;?> " type = "text" size = "10"></p>
      <p>Nama Murid <input name = "nama" readonly type="text"
        value = "<?php print $nama;?>">
      </p>
      <p>Kelas<input name = "kelas" readonly type = "text" size = "15"
        value = "<?php print $kelas; ?>"> </p>
      <p>Negeri Kelahiran <input name = "negeri" readonly type = "text"
        size = "20" value = "<?php print $negeri;?>">
      </p>
      <p><input Type = "submit" Name = "submit" Value = "Kembali"></p>
    </form>
  </body>
</html>
```

Rajah 3.3.83(b) Atur cara bagi "Senarai.php"



Rajah 3.3.84(a) Output 'Kemaskini.php'

Kemaskini.php

```

<html>
<head>
  <title>Kemaskini Maklumat Murid</title>
</head>
<body>
  <?php
    $noP=$_GET['nomurid'];
    $con = mysqli_connect("localhost","root","");
    if (!$con)
    {
      die ('Sambungan kepada Pangkalan Data Gagal'
        .mysqli_connect_error());
    }
    mysqli_select_db($con,"dbPelajar");
    $sql = "SELECT * FROM MURID WHERE NOMURID = '". $noP. "'";
    // contoh $sql = "SELECT * FROM MURID WHERE NOMURID = 'CL001'";
    $result = mysqli_query($con,$sql);
    $row = mysqli_fetch_array($result);
    $nama = htmlspecialchars($row['NAMA'], ENT_QUOTES);
    $kelas = $row['KELAS'];
    $negeri = $row['NEGERILAHIR'];
  ?> //penamat php
  <form action = "ProsesKemaskini.php" method = "POST">
    <p>No Murid : <b><?php print $noP;?></b></p>
    <p><input type = "hidden" name = "nomurid"
      value='<?php print $noP;?>'></p>
    <p>Nama Murid <input name = "nama" type='text'
      value = '<?php print $nama;?>'>
    </p>
    <p>Kelas<input name = "kelas" type="text" size = "15"
      value = '<?php print $kelas;?>'></p>
    <p>Negeri Kelahiran <input name = "negeri" type = "text"
      size = "20" value = '<?php print $negeri;?>'>
    </p>
    <p><input type = "submit" value = "Kemaskini"></p>
  </form>
</body>
</html>

```

Rajah 3.3.84(b) Atur cara bagi "Kemaskini.php"

ProsesKemakini.php

```
<?php
$con = mysqli_connect("localhost","root","");
if (!$con)
{
    die ('Sambungan kepada Pangkalan Data Gagal '.mysqli_connect_error());
}
mysqli_select_db($con,"dbPelajar");
$numurid = $_POST['nomurid'];
$name = $_POST['nama'];
$class = $_POST['kelas'];
$negeri = $_POST['negeri'];
$sql = "update murid set NAMA = '$name', KELAS = '$class',
        NEGERILAHIR ='$negeri' where NOMURID = '$numurid'";
$result = mysqli_query($con,$sql);
header('location:Senarai.php');
?>
```

Rajah 3.3.84(c) Atur cara bagi "ProsesKemakini.php"

3.3.7 Penghasilan Sebuah Laman Web Interaktif untuk Kegunaan Pengguna dan Pentadbir Sistem bagi Menyelesaikan Masalah

Sesetengah laman web mempunyai antara muka yang berbeza, iaitu antara muka untuk kegunaan pengguna biasa dan yang satu lagi antara muka untuk kegunaan pentadbir sistem.

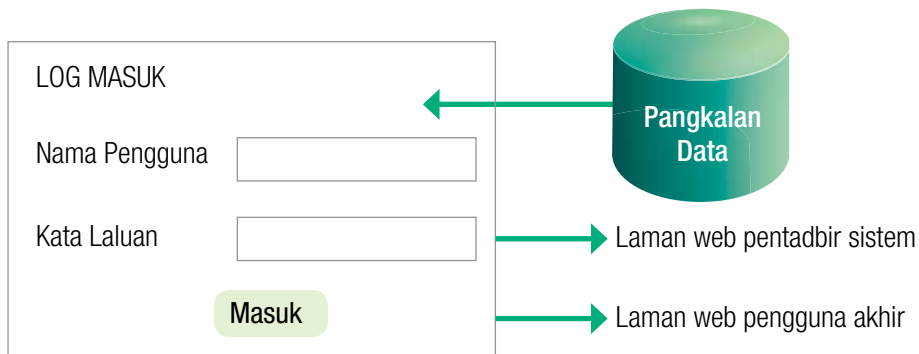
Laman web untuk kegunaan pentadbir sistem contohnya membolehkan pentadbir sistem mendaftarkan nama pengguna yang boleh mencapai data dalam pangkalan data manakala laman web untuk pengguna akhir membenarkan pengguna tersebut melihat maklumat yang telah ditetapkan sahaja.

Cara untuk membezakan jenis pengguna laman web sama ada pengguna biasa atau pentadbir sistem, dapat dilakukan dengan menyediakan laman web log masuk. Dalam laman web ini, pengguna akan memasukkan nama pengguna dan kata laluan.

Berdasarkan nama pengguna dan kata laluan ini, jenis pengguna yang telah disimpan di dalam pangkalan data dapat ditentukan. Seterusnya, laman web yang dipaparkan akan mempunyai pandangan (*view*) yang berbeza.



Pentadbir sistem ialah orang yang menjadi pengendali sesebuah laman web dan menjalankan prosedur tertentu. Laman web untuk pentadbir sistem ialah tempat pentadbir sistem dapat mengatur, mengawal dan mengurus proses yang ada pada suatu sistem laman web.



Rajah 3.3.85 Laman web yang mempunyai pandangan (*view*) berbeza

Membina Sebuah Laman Web Interaktif yang Mempunyai Dua Kategori Pengguna, iaitu Pengguna Biasa dan Pentadbir Sistem

Bagi membina laman web interaktif yang mempunyai dua kategori pengguna, kita memerlukan:

- Laman web log masuk (LogMasuk.php)
- Laman web menentukan jenis pengguna (JenisPengguna.php)
- Laman web Pentadbir sistem (Admin.php)
- Laman web pengguna biasa (Pengguna.php)

Selepas pengguna memasukkan nama pengguna dan kata laluan, jenis pengguna ditentukan daripada data yang telah disimpan dalam pangkalan data. Yang berikut ialah contoh data berkenaan dengan pengguna yang telah disimpan dalam pangkalan data.

+ Options			
NAMAPENGGUNA	KATALALUAN	JENISPENGGUNA	
halim	as0219	Pentadbir	
Khadijah	malaysia	Ahli Biasa	

Rajah 3.3.86(a) Contoh data berkenaan pengguna

The screenshot shows a web browser window with the address bar displaying "localhost/form5/LogMasuk.php". The page title is "Log Masuk Pengguna". Below the title are two input fields: "Nama Pengguna" and "Katalaluan", and a "Masuk" button.

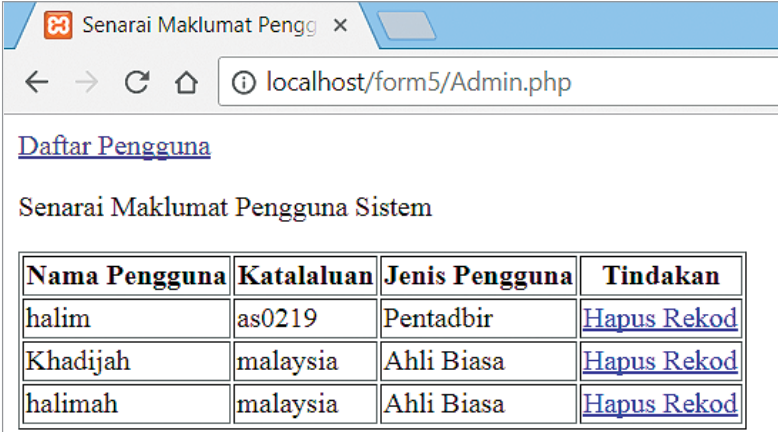
Rajah 3.3.86(b) Log Masuk Pengguna

LogMasuk.php

```
<html>
  <head>
    <title>Log Masuk</title>
  </head>
  <body>
    <h1>Log Masuk Pengguna</h1>
    <form action = "JenisPengguna.php" method="POST">
      <table>
        <tr>
          <td>Nama Pengguna</td>
          <td><input name = "namapengguna" type = "text" size = "10"></td>
        </tr>
        <tr>
          <td>Katalaluan</td>
          <td><input name = "katalaluan" type = "Password" size = "10"></td>
        </tr>
        <tr>
          <td><input Type = "submit" Name = "submit" Value = "Masuk"></td>
        </tr>
      </table>
    </form>
  </body>
</html>
```

Rajah 3.3.86(c) Atur cara bagi "LogMasuk.php"

Atur cara bagi laman web LogMasuk.php seperti yang ditunjukkan dalam Rajah 3.3.86(c) akan membenarkan pengguna memasukkan Nama Pengguna serta kata laluan. Selepas butang 'Masuk' diklik, maklumat tentang jenis pengguna dapat ditentukan dalam laman web yang bernama 'JenisPengguna.php' seperti yang ditunjukkan dalam Rajah 3.3.87(b) berdasarkan data yang telah disimpan dalam pangkalan data. Sekiranya jenis pengguna ialah Pentadbir, maka laman web yang akan dipaparkan ialah laman web (Admin.php), khusus bagi pentadbir sistem sahaja manakala bagi jenis pengguna yang lain, laman web (Pengguna.php) untuk pengguna biasa akan dipaparkan.



The screenshot shows a web browser window with the address bar displaying 'localhost/form5/Admin.php'. The page content includes a link 'Daftar Pengguna' and a heading 'Senarai Maklumat Pengguna Sistem'. Below the heading is a table with four columns: 'Nama Pengguna', 'Katalaluan', 'Jenis Pengguna', and 'Tindakan'. The table contains three rows of user data, each with a 'Hapus Rekod' link in the 'Tindakan' column.

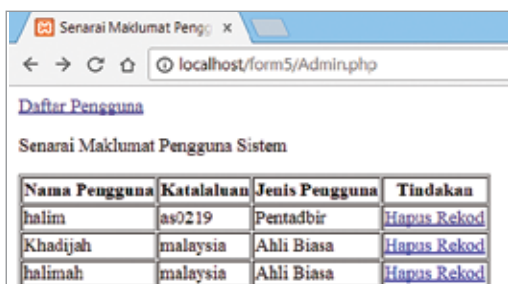
Nama Pengguna	Katalaluan	Jenis Pengguna	Tindakan
halim	as0219	Pentadbir	Hapus Rekod
Khadijah	malaysia	Ahli Biasa	Hapus Rekod
halimah	malaysia	Ahli Biasa	Hapus Rekod

Rajah 3.3.87(a) Output jenis pengguna

JenisPegguna.php

```
<html>
<head>
<title>Log Masuk</title>
</head>
<body>
<?php
    $nama = $_POST['namapengguna'];
    $katalaluan = $_POST['katalaluan'];
    $jumpa = False;
    $con = mysqli_connect("localhost","root","");
    if (!$con)
    {
        die ('Sambungan kepada Pangkalan Data Gagal'
            .mysqli_connect_error());
    }
    mysqli_select_db($con,"dbPelajar");
    $hasil = mysqli_query($con,"SELECT * FROM PENGGUNA WHERE NAMAPENGGUNA =
        '$nama' AND KATALALUAN = '$katalaluan'");
    $bil_baris = mysqli_num_rows($hasil);
    if ($bil_baris>0)
    {
        $row = mysqli_fetch_array($hasil);
        $jenispengguna = $row["JENISPENGGUNA"];
        if ($jenispengguna == 'Pentadbir')
        {
            header('location:Admin.php');
        }
        else
        {
            header('location:Pengguna.php?namapengguna = '.$nama);
        }
    }
    else
    {
        print "<p>Nama Pengguna atau Katalaluan Salah</p>";
    }
?> //penamat php
</body>
</html>
```

Rajah 3.3.87(b) Atur cara bagi “JenisPengguna.php”



Senarai Maklumat Pengg... x
localhost/form5/Admin.php

Daftar Pengguna

Senarai Maklumat Pengguna Sistem

Nama Pengguna	Katalaluan	Jenis Pengguna	Tindakan
halim	as0219	Pentadbir	Hapus Rekod
Khadijah	malaysia	Ahli Biasa	Hapus Rekod
halimah	malaysia	Ahli Biasa	Hapus Rekod

Rajah 3.3.88(a) Paparan Output Admin.php



Rajah 3.3.88(b) Paparan laman web Pengguna.php

Admin.php

```
<html>
  <head>
    <title>Senarai Maklumat Pengguna</title>
  </head>
  <body>
    <a href = 'Daftar.php'> Daftar Pengguna</a>
    <br>
    <p>Senarai Maklumat Pengguna Sistem</p>
    <?php
      $con = mysqli_connect("localhost","root","");
      if (!$con)
      {
        die ('Sambungan kepada Pangkalan Data Gagal'
            .mysqli_connect_error());
      }
      mysqli_select_db($con,"dbPelajar");
      print "<table border='1'>";
      print "<tr>";
      print"<th>Nama Pengguna</th>";
      print"<th>Katalaluan</th>";
      print"<th>Jenis Pengguna</th>";
      print"<th>Tindakan</th>";
      print "</tr>";
      $result = mysqli_query($con,"SELECT * FROM PENGGUNA");
      while ($row = mysqli_fetch_array($result))
      {
        $namapengguna = $row['NAMAPENGGUNA'];
        $katalaluan = $row['KATALALUAN'];
        $jenispengguna = $row['JENISPENGGUNA'];
        $lnk = "<a href = 'Padam.php?namapengguna = $namapengguna'>
        Hapus Rekod</a>";
        print "<tr>";
        print "<td>".$namapengguna."</td>";
        print "<td>".$katalaluan."</td>";
        print "<td>".$jenispengguna."</td>";
        print "<td>".$lnk."</td>";
        print "</tr>";
      } //penamat untuk while
      print "</table>";
    ?> //penamat untuk php
  </body>
</html>
```

Rajah 3.3.88(c) Atur cara bagi "Admin.php"

LATIHAN PENGUKUHAN 3.3

- 1 Secara ringkas, terangkan **tiga** kaedah prinsip penekanan maklumat dalam laman web yang dapat diimplementasikan bagi tujuan menarik perhatian pengunjung laman web.
- 2 Dengan menggunakan pelayar web seperti *Google Chrome* atau *Internet Explorer*, layari satu laman web yang anda sukai. Sediakan satu ringkasan berdasarkan laman web pilihan anda. Ringkasan tersebut perlulah menerangkan perkara-perkara yang berikut:
 - (a) Tujuan web laman itu dibina
 - (b) Sasaran pengguna
 - (c) Teknik navigasi yang digunakan
 - (d) Kaedah penstrukturan laman web
 - (e) Gaya persembahan laman web tersebut dari segi pemilihan latar belakang, jenis teks, grafik dan animasi
- 3 Dengan menggunakan kod HTML, bina satu laman web mengenai diri anda. Laman web tersebut perlulah mengandungi maklumat tentang anda seperti biodata, ringkasan mengenai hobi anda pada masa lapang serta gambar anda. Laman web tersebut mestilah mengandungi elemen-elemen HTML yang berikut:
 - (a) Heading
 - (b) Header
 - (c) Paragraph
 - (d) Break line
 - (e) Image
- 4 Berikan output bagi atur cara di bawah:

```
<html>
  <head><title>Tentang Saya</title>
    <style>
      table
      {
        border-collapse:collapse;
      }
      table, th, td
      {
        border: 3px solid;
      }
    </style>
  </head>
  <body>
    <h3>Tentang Saya</h3>
    <table>
      <tr>
        <th>Butir-butir Mengenai Saya</th>
      </tr>
      <tr>
        <td>Hobi Saya</td>
      </tr>
    </table>
  </body>
</html>
```

- 5 Senaraikan **dua** butang pilihan yang biasa digunakan dalam laman web untuk mendapatkan maklumat daripada pengguna dan berikan **satu** contoh penggunaan yang bersesuaian berdasarkan butang pilihan tersebut.
- 6 Berikan **tiga** contoh aktiviti yang boleh dilaksanakan dalam laman web interaktif.
- 7 Berikan **empat** elemen penting yang terdapat dalam laman web interaktif.

IMBAS SEMULA



LATIHAN PENGAYAAN

3

BAHAGIAN A

- 1 _____ ialah bahasa pengaturcaraan yang merupakan tambahan kepada HTML yang membenarkan perubahan kepada gaya sesuatu halaman web dan menerangkan persembahan bagi sesuatu dokumen HTML.
- 2 Arahan MySQL yang digunakan untuk mengemas kini data ialah _____.
- 3 Teknik navigasi yang paling sesuai untuk golongan kanak-kanak ialah _____.
- 4 Penstrukturan halaman yang sesuai bagi laman web berkonsepkan penceritaan atau kronologi ialah _____.
- 5 *Empty element* ialah elemen HTML yang mempunyai _____ tag sahaja.
- 6 _____ ialah program komputer yang digunakan untuk menyokong proses pengesahan input daripada pengguna apabila mengisi borang atas talian.

BAHAGIAN B

- 1 Jelaskan perbezaan antara laman web statik dengan laman web dinamik.
- 2 Tulis pernyataan `if...else...if` untuk menentukan kategori generasi berdasarkan umur pengguna dalam satu pemboleh ubah yang dipanggil `kumpulanUmur`. Paparkan nama bagi kategori generasi seperti yang ditunjukkan dalam Jadual 1.

Kumpulan Umur	Kategori Generasi
50 dan ke atas	<i>Baby Boomers</i>
39 – 49	Generasi X
18 – 38	Generasi Y
Bawah 18	<i>Millennials</i>

Jadual 1

3 Berdasarkan *function* di bawah, jawab soal-an-soalan yang berikutnya:

```
function KIRA(n)
{
  var i, j, tax;

  i = 0;
  tax = 0;
  for (j=1; j<=n; j++)
  {
    tax = tax + (i + 1);
    i += 2;
    document.write("<br> Hasil:"+tax);
  }
}
```

- (a) Tulis pernyataan *JavaScript* yang dapat memanggil *function* di atas.
 - (b) Apakah output, jika *function* di atas dipanggil dengan memberikan nilai $n=4$?
 - (c) Tulis semula *function* di atas ini, jika nilai akhir "tax" perlu dipulangkan semula kepada *function* yang memanggilnya.
- 4 Tulis keratan atur cara *Javascript* yang dapat menghasilkan satu butang `onclick()` dan menggunakan `window.alert()` untuk memaparkan hasil seperti Rajah 1 selepas butang ditekan.



Rajah 1

5 Kenal pasti **lima** kesalahan berdasarkan atur cara di bawah.

```
<?php
$con = mysqli_connect("root","localhost","");
if (!$con)
{
    die('Sambungan kepada Pangkalan Data Gagal'.mysqli_connect_error());
}
mysqli_select_db("dbPelajar");
$nopelajar=$_POST['nopelajar'];
$nama=$_POST['nama'];
$kelas=$_POST['kelas'];
$negeri=$_POST['negeri'];
$sql="update pelajar change NAMA='$nama', KELAS='kelas',
NEGERILAHIR='$negeri' where NOPELAJAR='$nopelajar'";
$result=mysqli_query($con,$sql);
header('location:Senarai.php');
mysqli_close(&con);
```

6 Nyatakan dan jelaskan cara pengesahsahihan pada input dapat dilakukan untuk objek borang (*Form*) seperti yang ditunjukkan pada Rajah 2.



The image shows a web form with the following elements:

- A text input field labeled "Nama".
- A text input field labeled "Kelas".
- A heading: "Pilih Persatuan/Kelab yang ingin disertai: (Pilih 3 sahaja)".
- A list of five checkboxes with labels:
 - Persatuan Bulan Sabit Merah
 - Kelab Bimbingan dan Kerjaya
 - Kelab Sejarah
 - Kelab Matematik
 - Kelab Bahasa
- A button labeled "Harbar" at the bottom.

Rajah 2

7 Ryana ialah seorang usahawan maya yang menjual bantal kanak-kanak yang ditempah khas. Ryana memulakan perniagaannya melalui laman *Facebook*. Memandangkan perniagaannya mendapat sambutan yang menggalakkan, dia ingin mengembangkan perniagaannya dengan memiliki laman webnya sendiri bagi mempromosikan bantalnya dengan pelbagai rekaan terbaharu. Dia menyinggahi laman web dengan halaman utamanya memaparkan pilihan rekaan bantalnya serta kualiti kain yang digunakan.

Bina rajah bagi penstrukturan laman web dengan berdasarkan senario di atas. Penstrukturan halaman yang dibina perlulah mengandungi halaman utama yang menghubungkan sekurang-kurangnya lima pilihan halaman yang lain.

ACC

Digunakan untuk menyimpan data yang menjadi input dan output semasa operasi UAL.

Asymmetric Cryptosystem

Proses penyulitan yang menggunakan kunci yang berlainan.

Atur cara

Arahan berkod yang berturutan yang dimasukkan ke dalam komputer dan datanya diproses oleh komputer.

Bahasa Pengolahan Data (DML)

Pernyataan SQL yang digunakan untuk memanipulasi data dalam jadual.

Bahasa penskripan klien

Program komputer dalam web yang dilaksanakan di sebelah klien.

Bahasa Takrifan Data (DDL)

Pernyataan SQL yang digunakan untuk mencipta dan mengubah struktur pangkalan data.

Cascading Style Sheet (CSS)

Membenarkan perubahan kepada gaya sesuatu halaman web dan menerangkan persembahan bagi sesuatu dokumen HTML.

Count

Mengira bilangan nilai yang terdapat di dalam suatu senarai.

Daftar

Komponen yang digunakan bagi menyimpan data dan arahan yang dicapai dari ingatan utama sebelum dihantar ke UAL untuk diproses dan menyimpan hasil pemprosesan sementara sebelum dihantar ke ingatan utama dan dipamerkan melalui peranti output.

ERD

Model asas untuk membina pangkalan data. ERD merupakan teknik yang digunakan untuk memodelkan data yang diperlukan dalam suatu organisasi.

Etika dalam pengkomputeran

Standard panduan bagi mengawal perlakuan pengguna komputer supaya lebih bermoral.

Heading

Elemen yang digunakan bagi mentakrifkan "tajuk" dalam laman web.

HTML

Bahasa *markup standard* yang digunakan oleh pereka web profesional bagi mereka bentuk laman web.

Isihan buih

Satu algoritma mudah dalam pengisihan suatu senarai, iaitu proses dan langkahnya akan dibuat satu per satu secara berturutan, membandingkan unsur-unsur bersebelahan dan menukarkan kedudukan (*swap*) sekiranya berlaku turutan yang salah.

Jadual kebenaran

Menyemak output yang dihasilkan daripada get-get logik atau daripada litar logik.

Jenayah siber

Penyalahgunaan teknologi pengkomputeran dan Internet bagi melaksanakan aktiviti jenayah, iaitu merosakkan, mencuri atau mengubah data dalam sistem komputer bagi tujuan tertentu.

Keseimbangan visual

Gabungan elemen visual seperti garisan, imej, teks, bentuk dan warna dalam laman web.

Litar bersepadu

Gabungan komponen get logik yang disambungkan menjadi satu litar yang lengkap atau separa lengkap dan dikesilkan beribu-ribu kali ganda dari saiz asal, kemudian ditempatkan dalam satu bahan kepingan silikon yang dinamakan cip atau cip mikro.

Malware

Program yang direka khas untuk menjangkiti sesebuah sistem komputer tanpa diketahui oleh pemiliknya.

Mean

Purata nilai dalam suatu senarai nilai.

Model Hubungan

Menyimpan data dalam bentuk jadual hubungan, iaitu data dan hubungan dipersembahkan dalam bentuk jadual.

Navigasi

Struktur laman web tersebut untuk dilayari.

Navigasi web

Proses di mana objek-objek dalam laman web yang terdiri dari teks, rajah, audio dan video yang dihubungkan di antara satu sama lain dengan menggunakan pautan hiperteks atau hipermedia.

Nilai maksimum

Nilai terbesar di dalam suatu senarai nilai.

Nilai minimum

Nilai terkecil di dalam suatu senarai nilai.

Nyahsulit

Proses mengembalikan bentuk maklumat yang tidak dapat difahami (*ciphertext*) kepada maklumat asal (*plaintext*).

Pemproses linear

Pemproses yang memproses data dan arahan secara berjujukan.

Penormalan

Satu proses untuk memastikan bahawa model data logikal yang terhasil mempunyai anomali yang paling minimum.

Penstrukturan berjujukan

Pengguna dipandu dari satu halaman ke halaman yang lain secara berjujukan.

Penyulitan

Proses pengkodan maklumat (*plaintext*) dari pengirim kepada bentuk maklumat yang tidak dapat difahami (*ciphertext*).

Prinsip

Asas yang menjadi dasar kepada perkembangan atau pembinaan sesuatu benda.

Prinsip penekanan

Maklumat atau kandungan yang ingin diberikan perhatian dalam laman web tersebut.

Queue

Membuang data lama yang dimasukkan sahaja.

Rangka wayar

Panduan visual yang mewakili rangka kerja keseluruhan laman web.

Reka bentuk

Susunan atau struktur dalam proses binaan sesuatu perkara.

Sistem jam

Peranti cip di dalam pemproses yang menjana denyutan elektronik pada kadar yang tetap bagi menyeragamkan masa aktiviti pemprosesan semasa melaksanakan kitaran mesin.

SQL

Bahasa yang digunakan untuk mencapai dan mengemas kini data dari pangkalan data.

Symmetric Cryptosystem

Proses penyulitan yang menggunakan kunci yang sama.

Talian paip

Teknik beberapa arahan adalah bertindih semasa pelaksanaan.

Tipografi (*typography*)

Penyusunan teks yang dipaparkan dalam laman web.

Undang-undang siber

Peraturan yang ditetapkan oleh pihak berkuasa di sesebuah negara untuk memberikan hukuman kepada pesalah-pesalah yang melakukan kesalahan di dalam talian.

Unit Aritmetik dan Logik (UAL)

Melakukan operasi aritmetik dan logik.

Unit ingatan

Komponen komputer yang akan memegang data atau set arahan sebelum, semasa dan selepas data atau set arahan.

Unit kawalan

Menerima dan menjana isyarat-isyarat bagi mengawal operasi keseluruhan sistem komputer.



A

ASP 126, 129, 226, 342
Atribut 60–80, 113, 119, 125, 127, 196, 230, 291, 295–297,
333–335
Atur cara 5, 24–37, 113–325, 124–353

B

Bahasa pengaturcaraan 24, 33–38, 56, 77, 125–126, 180,
194, 298, 342
Bahasa penskripan 124–126, 129–133, 170, 178, 194–195,
200–201, 205–206
Bas data 28, 57, 360

C

Carta alir 124, 132–136, 139, 145–147, 159, 169, 178, 181,
285, 287
Cascading Style Sheet 125, 274, 298, 370
Count 60, 103, 108–111, 124, 160–163, 170–171, 240, 355

D

Daftar 24, 26, 28–29, 37
Do 191–192

E

Entity relationship diagram 60, 62, 65
Etika dalam pengkomputeran 6

H

Heading 292–293, 299, 302–312, 324–326
HTML 21, 124–135, 170, 197, 211–220, 230, 290–296,
298–353

I

Isihan buih 132–135, 171–173, 207, 214–216
Isihan pilih 132, 138–143, 173–174

J

Jadual kebenaran 42–52
Jenayah siber 2–6, 15–19

K

Keseimbangan visual 281

L

Laman web 124, 126, 128–131, 195–196, 211, 213–214,
217, 224–227
Litar bersepadu 39–40, 51

M

Malware 5, 21
Mean 124, 148, 157–171
Model hubungan 68–70, 72, 119

N

Navigasi web 281, 287
Nilai maksimum 148–151, 170–171
Nilai minimum 153–154, 171
Nyahsulitan 10–11, 32

P

Pemproses linear 31–33
Penormalan 70, 74–76
Penstrukturan kandungan 283–284
Penyulitan 8–11, 20
PHP 21, 34, 126, 129, 226–228
Prinsip penekanan 277

Q

Queue 124, 165–169

R

Rangka wayar 287

S

Sistem jam 29–30
Symmetric encryption 10
SQL 60–116, 257

T

Talian paip 31–32
Typography 278

U

Undang–undang siber 15–19
Unit aritmetik 28–29
Unit ingatan 24–25, 27, 56
Unit kawalan 24, 26, 28–29, 37–38



- "8 Guidelines for Exceptional Web Design, Usability, and User Experience," dicapai pada 20 Januari 2017, <https://blog.hubspot.com/blog/tabid/6307/bid/30557/6-Guidelines-for-Exceptional-Website-Design-and-Usability.aspx#sm.000054ojy8yqsf52zun2qhgy8fns>
- "Bubble Sort," dicapai pada 15 November 2016, <https://www.kidscodex.com/bubble-sorts/>
- "CSS background-image Property," dicapai pada 26 Januari 2017, http://www.w3schools.com/cssref/pr_background-image.asp
- "DML command," dicapai pada 13 Januari 2017, <http://www.studytonight.com/dbms/dml-command>
- "File.CreateText Method (String)," dicapai pada 6 Januari 2017, [https://msdn.microsoft.com/en-us/library/system.io.file.createtext\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.io.file.createtext(v=vs.110).aspx)
- "HTML Style Sheet," dicapai pada 26 Januari 2017, https://www.tutorialspoint.com/html/html_style_sheet.htm
- "Javascript History," dicapai pada 15 Desember 2016, <http://www.howtcreate.co.uk/jshistory.html>
- "Selection Sort: Explained Along with Java Program Code," dicapai pada 18 November 2016, <http://javahungry.blogspot.com/2013/06/java-sorting-program-code-selection-sort.html>
- Dorling, M. dan Rouse, G., 2014. *Compute-IT 2: Computing for KS3*. London: Hodder Education.
- Dorling, M. dan Rouse, G., 2014. *Compute-IT 3: Computing for KS3*. London: Hodder Education.
- Hoffer J., Venkataraman R., dan Topi H., 2015. *Modern Database Management*, Edisi Kedua Belas. Prentice Hall.
- Langfield, S. dan Duddell, D., 2015. *Cambridge International AS and A Level: Computer Science Coursebook*. United Kingdom: University Printing House.
- Deitel, P. J. dan Deitel, H. M., 2014. *Java How To Program (Early Objects)*, Edisi Kesepuluh. England: Pearson Education Limited.
- "How to Recognize Phishing Email Messages, Links, or Phone Calls", dicapai pada 6 Februari 2017, <https://www.microsoft.com/en-us/safety/online-privacy/phishing-symptoms.aspx>
- "Ethics in Computing", dicapai pada 6 Februari 2017, <https://ethics.csc.ncsu.edu/privacy/>
- "Website Design", dicapai pada 19 Januari 2017, <http://www.usabilityfirst.com/about-usability/website-design/>
- "Scripting Language", dicapai pada 7 Februari 2017, <http://www.computerweekly.com/feature/Scripting-languages>
- "Client-side Scripting and HTML", dicapai pada 27 Desember 2016, <https://www.w3.org/TR/WD-script-970314>
- Yew, K. H., 2013. *Computer Science for Matriculation Semester 2*. Shah Alam, Selangor: Oxford Fajar.

Dengan ini **SAYA BERJANJI** akan menjaga buku ini dengan baik dan bertanggungjawab atas kehilangannya serta mengembalikannya kepada pihak sekolah pada tarikh yang ditetapkan

Skim Pinjaman Buku Teks

Sekolah _____

Tahun	Tingkatan	Nama Murid Yang Menerima	Tarikh Terima

Nombor Perolehan: _____

Tarikh Penerimaan: _____

BUKU INI TIDAK BOLEH DIJUAL

Harga: RM24.50

ISBN 978-983-47-2375-0



9 789834 723750

FT305001