|  |  |  |
| --- | --- | --- |
| **BAHAGIAN PENDIDIKAN DAN LATIHAN TEKNIK VOKASIONAL**  **KEMENTERIAN PENDIDIKAN MALAYSIA**  **ARAS 5 & 6, BLOK E14, KOMPLEKS E,**  **PUSAT PENTADBIRAN KERAJAAN PERSEKUTUAN**  **KERTAS PENERANGAN**  ***(INFORMATION SHEET)***  C:\Users\norsu\Desktop\jata negara.png | | |
| **KOD DAN NAMA PROGRAM NOSS** | IT-010-3:2016 PEMBANGUNAN APLIKASI | |
| **TAHAP NOSS** | 3 | |
| **KOD, NAMA CU DAN WA NOSS** | **CU05/WA3 - DEBUG APPLICATION CODE** | |
| **NAMA PROGRAM KV** | SISTEM PENGURUSAN PANGKALAN DATA DAN APLIKASI WEB | |
| **KOD DAN NAMA KURSUS KSKV** | KPD2033 FUNDAMENTAL OF PROGRAMMING | |
| **NO DAN TAJUK STANDARD KANDUNGAN KSKV** | K3 WRITE MODULE CODE | |
| **NO. KOD NOSS** | IT-010-3:2016-C05 / P(24/27) | Muka Surat : 1 Drp : 11 |
| **NO. KOD KSKV** | KPD2033/P(10/13) |

**TAJUK/***TITLE***:**

**CHECK NAMING CONVENTION**

**TUJUAN/***PURPOSE***:**

Kertas penerangan ini adalah bertujuan menerangkan mengenai :

* Jenis Data
* Pemalar dan pembolehubah
* Deklarasi
* Operator
* Struktur kawalan

**PENERANGAN:**

**Data**

**Setiap bahasa pengaturcaraan mempunyai jenis data yang hampir sama**. Dalam Java terdapat dua jenis data

• data jenis primitif

• data jenis bukan primitif

**Data Jenis Primitif**

**Java boleh menyokong beberapa jenis data primitif. Jenis data primitif adalah data yang telah ditakrifkan dalam suatu bahasa pengaturcaraan dan dinamakan sebagai kata simpanan**.

Integer (int) – nombor bulat tanpa perpuluhan:

• Jenis data *int* menggunakan 4 bait ingatan.

• Nilai minimumnya ialah - 2,147,483,648

• Nilai maksimumnya ialah 2,147,483,647

• Lazimnya int digunakan sebagai jenis data lalai.(default value).

• Nilai lalai ialah 0

Contoh 1 :

int markah;

• Contoh 2:

*int* bilPelajar = 50;

*int* suhu = - 5;

*double – nombor perpuluhan:*

• Jenis data *double* menggunakan 8 bait ingatan.

• Nilai lalai ialah 0.0

Contoh 1 :

double gaji;

• Contoh 2 :

*double* gaji = 2340.5;

*Boolean* – nilai true dan false.

• Jenis data boolean ialah satu bit.

• Jenis data ini digunakan sebagai *flag* untuk mengesan syarat benar/palsu.

• Nilai lalai ialah *false*.

Contoh 1:

boolean statusPerkahwinan ;

• Contoh 2:

*boolean* statusPerkahwinan = *true;*

*char* – aksara:

• Jenis data *char* menggunakan 2 bait ingatan.

• Nilai minimumnya ialah '\u0000' (atau 0).

• Nilai maksimumnya ialah '\uffff' (atau 65,535 termasuk).

• Jenis data *char* digunakan untuk menyimpan sebarang aksara tunggal.

Nilai lalai ialah '\u0000' – aksara null.

Contoh :

*char abjad;*

• Contoh:

*char* abjad= 'A' ;

**Data jenis Bukan Primitif**

**String** – mewakili gabungan aksara.

Contoh 1 :

String nama; *// tidak diterima oleh compiler*

**sepatutnya**

String nama = “ “; *//perlu diletakkan nilai awalan*

Contoh 2 :

String nama = “Hazzra”;

**Pemboleh ubah**

Nilai pembolehubah berubah-ubah sepanjang pelaksanaan aturcara. Mengikut amalan pengaturcaraan yang baik pembolehubah bermula dengan huruf kecil dan jika terdapat dua perkataan atau lebih awalan perkataan seterusnya menggunakan huruf besar.(Camel Case)

**Contoh Pengisytiharan Pembolehubah**

*int* panjang,lebar,luas;

*double gajiBapa;*

*char* aks = 'a';

*boolean sudahBayarYuran;*

**Pemalar**

Nilai pemalar adalah kekal sepanjang pelaksanaan atur cara. Pemalar digunakan untuk nilai yang sama digunakan berulang kali dalam satu pengaturcaraan. Pemalar diisytiharkan menggunakan arahan final. Setelah diisytiharkan, nilai final tidak boleh diubah lagi. Mengikut amalan pengaturcaraan yang baik pemalar menggunakan semua huruf besar.

Contoh

*final double* PI = 3.1415;

*final int* SAIZ = 100;

*final* int DISKAUN = 10;

**Operator**

Operator digunakan untuk memanipulasikan pemboleh ubah. Operator dalam bahasa Java boleh dikategorikan seperti berikut:

Operator Aritmetik

Operator Perbandingan

Operator Logikal

Operator Umpukan

**Operator Aritmetik**

Operator aritmetik digunakan dalam ungkapan matematik seperti dalam algebra.

Anggap pemboleh ubah integer A menyimpan nilai 10 dan pemboleh ubah B menyimpan nilai 20

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator Aritmetik** | | **Ungkapan aritmetik** | **Nilai** |
| **Nama** | **Simbol** |
| Tambah | + | A + B | 30 |
| Tolak | - | A - B | -10 |
| Darab | \* | A \* B | 200 |
| Bahagi | / | B / A | 2 |
| Modulus | % | B % A | 0 |
| Tokok (Increment) | ++ | B++ | 21 |
| Susut (Decrement) | -- | B-- | 19 |

**Operator Umpukan (Assignment Operator)**

Operator umpukan yang disokong oleh bahasa Java adalah seperti berikut:

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator Umpukan** | | **Ungkapan** | **Huraian** |
| **Nama** | **Simbol** |
| simple | = | C = A + B | A + B → C |
| tambah DAN | += | C += A | C + A → C |
| Tolak DAN | -= | C -= A | C - A → C |
| Darab DAN | \*= | C \*= A | C \* A → C |
| Bahagi DAN | /= | C /= A | C / A → C |
| Modulus DAN | %= | C %= A | C % A → C |

**Pernyataan Umpukan Dan Pernyataan Aritmetik**

Dalam pernyataan umpukan, ungkapan wujud di sebelah kanan simbol umpukan. Formatnya ialah:

<pemboleh ubah> = <ungkapan>;

Contoh pernyataan umpukan

luas = 3.142 \* jejari ^ 2;

gajiBersih = gajiKasar + gajiLebihMasa – potongan;

purata= (nombor1 + nombor2 + nombor3) / 3;

baki = nombor1 % 6;

**STRUKTUR KAWALAN.**

* Struktur kawalan berfungsi untuk menentukan aliran operasi yang akan dilakukan dalam atur cara. Terdapat penggunaan struk kawalan iaitu:

1. **If-else**
2. **Switch-case**
3. **Repetition**

* Tiga jenis struktur kawalan :
  + Struktur jujukan
  + Struktur pilihan
  + Struktur gelung

Struktur jujukan

Struktur jujukan sesuai untuk selesaikan masalah yang mudah dan tidak berulang.

Struktur pilihan

* Penggunaan struktur pilihan membenarkan satu atau lebih arahan disediakan.
* Ungkapan syarat memainkan peranan penting kerana pilihan akan dibuat berdasarkan kepada syarat yang diberikan.
* Terdapat tiga jenis struktur pilihan :
  + Satu pilihan
  + Dua pilihan
  + Pelbagai pilihan
* Struktur pilihan menggunakan :-

1. pernyataan if

2. pernyataan else

3.pernyataan else if

4. pernyataan switch

Menulis pernyataan if…else berdasarkan peraturan standard pengaturcaran

web.

**4.7.1 Pernyataan if**

Pernyataan if digunakan bagi kes yang mempunyai satu pilihan sahaja iaitu samaada dilaksanakan ataupun tidak.

Format pernyataan if

if (ungkapan syarat)

Kod dilaksanakan jika benar;

Carta alir pernyataan if

Syarat

Blok 1

Ya

Tidak

Pernyataan boleh terdiri daripada satu pernyataan sahaja atau satu blok pernyataan. Blok pernyataan mesti diletakkan di dalam { }.

Contoh

Markah

> 80 ?

Gred A

Ya

Tidak

**4.7.2 Pernyataan else**

* Pernyataan else merupakan pernyataan yang tidak boleh berdiri sendiri.
* Pernyataan ini mesti digunakan bersama dengan pernyataan if.
* Pernyataan else digunakan untuk kes yang menyediakan dua pilihan , iaitu satu pilihan akan dilaksanakan jika syarat benar dan pilihan lain ialah palsu.

Format pernyataan else

if (syarat)

Kod dilaksanakan jika syarat if benar;

else

Kod dilaksanakan jika syarat tidak dipenuhi;

Carta alir pernyataan if else

Syarat

Blok 1

Arahan seterusnya

Ya

Tidak

Blok 2

Menulis pernyataan if…else if…else berdasarkan peraturan standard pengaturcaran web.

**4.7.3 Pernyataan else if**

Pernyataan else if digunakan untuk kes yang menyediakan lebih daripada dua pilihan.

Syarat akan diuji dari atas ke bawah.

Format pernyataan else if

if (syarat)

Kod dilaksanakan jika syarat if benar;

else if (syarat)

Kod dilaksanakan jika syarat else if benar;

……

else

Kod dilaksanakan jika semua syarat tidak dipenuhi ;

Carta alir if else if

Syarat

Menulis pernyataan switch berdasarkan peraturan standard pengaturcaran web.

**Pernyataan Switch**

* Pernyataan switch adalah kaedah alternatif untuk multipilihan.
* untuk mengelakkan kekeliruan dalam multipilihan
* Pernyataan switch ini memiliki kegunaan sama seperti if else.
* Batasan Switch :data yang diperiksa haruslah bertype Integer (int) atau Character (char).

Format pernyataan switch

**switch (ungkapan kawalan)**

**{**

**case pemalar 1 : kod dilaksanakan jika pernyataan = pemalar 1;**

**break;**

**case pemalar 2 : kod dilaksanakan jika pernyataan = pemalar 2;**

**break;**

**case pemalar 3 : kod dilaksanakan jika pernyataan = pemalar 3;**

**break;**

**default : kod dilaksanakan jika pernyataan tidak sama dengan semua pemalar;**

**}**

Anda boleh menggunakan beberapa pernyataan dalam pernyataan switch untuk

memastikan penggunaan kod lebih efisien. Iaitu :-

* Pernyataan break
* Pernyataan default
* Pernyataan continue

**Pernyataan break**

Pernyataan break mengelakkan kod pada pernyataan case yang berikutnya dilaksanakan selepas padanan dilakukan.

Pelaksanaan switch akan tamat apabila pernyataan break dilaksanakan.

Perlaksanaan aturcara seterusnya akan beralih ke pernyataan selepas switch.

**Pernyataan default**

Pernyataan default dilaksanakan jika tiada pernyataan case yang memenuhi syarat. Selalunya diletak sebagai pilihan terakhir dalam pernyataan switch.

**Pernyataan continue**

Pelaksanaan pernyataan ini akan menyebabkan pernyataan-pernyataan selepas pernyataan continue dalam suatu gelung tidak dilaksanakan tetapi kawalan akan terus melaksanakan gelung tersebut

**SOALAN/***QUESTION***:**

1. Berikan dua jenis data yang ada?
2. Berikan contoh pengishtiharan bagi pemalar.
3. Berikan contoh bagi data Bukan Primitif
4. Senaraikan jenis struktur kawalan.

**RUJUKAN/***REFERENCE* **:**

1. Keith Cooper, Linda Torczon(2011), Engineering a Compiler (2nd Edition), Morgan Kaufmann, ISBN: 978-0120884780
2. Todd Zaki Warfel (2009), Prototyping: A Practitioner's Guide, Louis Rosenfeld Media, LLC, ISBN I-933820-21-7
3. Steve McConnell (2004), Code Complete: A Practical Handbook of Software Construction, (2nd Edition),Microsoft Press, ISBN: 978-0-7356-1967-8
4. Michael L. Scot (2009) , Programming Language Pragmatics, Third Edition, Morgan Kaufmann, ISBN-13: 978-0123745149
5. Simon Marlow (2013), Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming (1st Edition), O'Reilly Media, ISBN-13: 978-1449335946