

# PENGANTAR PBO

(IDENTIFIER, KEYWORD, LITERAL DAN  
TIPE DATA, OPERATOR, SEPARATOR)

*Pemrograman Berorientasi Objek*

## JAVA

- Asal mula Java :
  - Tim sun Microsystems th 1991 *code name* Oak  
→ th 1995 Java
- Bahasa pemrograman java yang telah dirilis adalah :
  1. Produk Sun Microsystem : J2SE, J2ME, J2EE
  2. Produk Borland : Borland JBuilder
  3. Produk Microsoft : Microsoft Visual J/J++

# JAVA

- Kelebihan bahasa Java :
  - *Portable* dan *platform independent*
  - *Garbage collection*
  - Menghilangkan pewarisan berganda pada C++
  - Mengurangi pointer aritmatik → java : reference
  - Array sejati
  - Mengurangi kerancuan antara pemberian nilai pada statemen kondisional. Contoh : '=' dgn '==' pada kondisi **if**

# JAVA

- Karakteristik Java :
  - Berorientasi objek
  - Robust
  - Portable
  - Multitreading
  - Dinamis
  - Sederhana
  - Terdistribusi
  - Aman
  - Netral secara arsitektur
  - Intrepreted
  - Berkinerja tinggi

# JAVA

- Java/Standard Development Kit (JDK/SDK) merupakan alat-alat utama bagi programmer untuk membuat dan menjalankan java.
- Development Kit dapat didownload di :  
<http://java.sun.com/product/JDK/index.html> atau  
<http://java.sun.com/j2se/1.5/index.html> atau,  
<http://java.sun.com/cgi-bin/java-ports.cgi>

## Komponen JDK

- compiler(javac),
- interpreter(java) disebut juga java virtual machine atau java runtime environment,
- applet viewer(appletviewer),
- debugger(jdb),
- java class library(jcl),
- header dan stub generator(javah),
- java documentation(javadoc).

# Komponen JDK

## 1. Kompilator (javac)

Bertugas untuk melaksanakan kompilasi \*.java menjadi \*.class

Syntax umum : javac nama\_file.java

## 2. Interpreter (java)

Bertugas untuk menjalankan bytecode (\*.class)

Syntax umum : java nama\_file.class

## 3. Applet Viewer

Digunakan untuk menjalankan applet viewer, namun sekarang sudah digantikan browser.

Syntax umum : appletviewer nama\_file.html

## 4. Java Debugger

Bertugas untuk melakukan debugging aplikasi java.

Syntax umum : jdb option

# Komponen JDK

## 5. Java Class File Diassembler (javap)

Bertugas membuat daftar method dan attribute public dari suatu kelas.

Syntax : javap namaKelas

## 6. Java Header and Stub Generator

Bertugas menerjemahkan bahasa yang ditulis dalam bahasa Java menjadi bahasa pemrograman C.

Syntax umum : javah namaKelas

## 7. Java Documentation Generator

Menampilkan pustaka kelas, interface, constructor, dan method standard yang telah dibuat vendor.

Dari hasil instalasi, dokumentasi ini dapat dilihat di pada C:\java\docs\api\index.html dan dari hasil instalasi

Netbeans, dapat dilihat pada

C:\Program Files\NetBeans3.6\doc\junit\index.html

## 8. Source Code Java API

Source code ini dapat diperoleh dari file src.zip.

Untuk pemrogram pemula, lingkungan pemrograman java dapat diringkas menjadi:

- *Editing source code* menggunakan editor teks, seperti Notepad atau TextPad
- *Compiling* menggunakan keyword javac melalui command prompt (dapat juga dari editor teks seperti TextPad).
- *Executing* menggunakan :
  1. *Command prompt* untuk *java application* (atau dari editor teks TextPad)
  2. *Browser* atau *appletviewer* untuk *java applet*.

## Pemrograman JAVA

- Baris-baris program dalam java harus ditulis dalam lingkup *class*
- Contoh program pendek untuk menampilkan tulisan “Belajar Java” di layar monitor :

```
class Latihan1 {  
    public static void main(String args[]) {  
        System.out.println("Belajar Java");  
    }  
}
```

- Program di atas disimpan dengan nama file Latihan1.java. Penamaan ini mengikuti aturan bahwa nama file = nama *class*.

## Token

- adalah elemen terkecil di program yang masih memiliki arti
- Lima token dalam bahasa java yaitu :
  - 1.identifier
  - 2.keyword
  - 3.literal dan tipe data
  - 4.operator
  - 5.separator

## Identifier

adalah token yang merepresentasikan nama sesuatu (variabel, atau konstanta, atau method, atau kelas, atau package, atau interface)

## Keyword dalam Java

abstract	continue	for	new	switch
boolean	default	goto	null	synchroniz ed
break	do	if	package	this
byte	double	implements	private	threadsafe
byvalue	else	import	protected	throw
case	extends	instanceof	public	throws
catch	false	int	return	transient
car	final	interface	short	true
class	finally	long	static	try
const	float	native	super	void
				while

University  
<http://www.gunadarma.ac.id>

## Literal dan Tipe Data Primitif

- adalah nilai variabel/attribute atau nilai konstanta atau nilai objek data.
- tiga besaran literal dalam java yaitu angka, karakter, dan string.
- angka terdiri dari byte, short, int, long, float, double, dan boolean (dianggap angka true = 1 atau false = 0).

# deklarasi variable dan konstanta

Bentuk umum :

```
TipeData namaVar = ungkapan_atau_nilai;
```

```
TipeData namaVar1, namaVar2, ...;
```

```
[modifier] static final TipeData NAMA KONSTANTA = nilai;
```

Contoh deklarasi :

```
float f1= 0.13f;
```

```
double a=3, b=4;
```

```
double c = Math.sqrt(a*a+b*b);
```

```
static final PHI=3.14;
```

```
static final double CM_PER_INC = 2.54;
```

Tabel jangkauan dan ukuran dari semua tipe data sederhana dalam Java

Type Data Primitif	Jangkauan	Ukuran (bit)
byte	-128 s/d 127	8
short	-32767 s/d 32767	16
int	-2147483648 s/d 2147483647	32
long	-9223372036854775808 s/d 9223372036854775807	64
char	sebuah unicode	16
float	3.4e-038 s/d 3.4e+038	32
double	1.7e-308 s/d 1.7e+308	54
boolean	false = 0 atau true = 1	8



## Casting

- teknik konversi secara eksplisit, yaitu menerangkan kepada kompiler bahwa terjadi perubahan tipe variabel, shg hasilnya harus dikonversi ke tipe variabel baru tersebut.

- Contoh :

```
int i;  
double d;  
byte b;  
i = 250;  
d = i; //tidak error,penyesuaian otomatis  
b = i; //error, tipe tidak kompatibel  
Seharusnya:b = (byte) i; ← eksplisit cast
```

## Promosi

- Promosi tipe pada ekspresi dilakukan oleh Java secara otomatis bila konversi dari satu tipe variabel dgn kapasitas byte yg lebih kecil ke tipe variabel lain dgn penempatan byte (kapasitas) yg lebih besar

- Contoh :

```
int i;  
byte b = 100;  
i = b + 50; // otomatis tipe b+50  
// menjadi integer
```

# Operator

Operator melakukan komputasi terhadap satu/dua objek data (operan).

Operan yang dioperasikan dapat berupa literal, variabel, atau nilai yang dikirim method.



Tabel Hirarki Operator

Prioritas	Kelompok Operator	Keterangan
1	. [] ()	sekaligus
2	++var, --var, ~, instanceof	preinc, predec, unary, instance dari kelas ...
3	(type) (casting)	
4	!	not
5	*, /, %	perkalian, pembagian, modulus
6	+, -	penjumlahan, pengurangan
7	<<, >>, >>>	geser untuk bil biner
8	<, >, <=, >=	pembandingan
9	==, !=	kesamaan, ketidaksamaan
10	&	and
11	^	exclusive or
12		unconditional or
13	&&	conditional and
14		conditional or
15	? :	shorthand untuk if..then...else...
16	=, +=, -=, *=, /=, %=, ^=	operator penugasan
17	&=,  =, <<=, >>=, >>>=	operator penugasan
18	var++, var--	postincrement, postdecrement

# Separator

Separator menginformasikan ke compiler java mengenai adanya kelompok kode program

## Daftar Separator di Java

Notasi	Nama	Deskripsi
(...)	kurung	mengelompokkan parameter method.
{...}	kurung kurawal	mengelompokkan nilai-nilai suatu array, mendefinisikan blok kode kelas ataupun kode method.
[...]	kurung siku	mendeklarasikan tipe array
:	titik koma	mengakhiri pernyataan
,	koma	memisahkan identifiier-identifiier di bagian deklarsi variable, merangkai pernyataan-pernyataan di dalam for.
.	titik	memisahkan nama-nama package, memisahkan kelas dari objek, dan objek dari method.

Gunadarma University  
<http://www.gunadarma.ac.id>

## Keyword break, continue, dan return

merupakan kendali jump (variasi lain dari goto), sehingga berakibat pengabaian sisa instruksi

## Keyword break

untuk keluar dari kendali percabangan switch, dan  
untuk keluar dari kendali perulangan

**Contoh :**

```
public class contohBreak {
    public static void main(String args[]) {
        int i = 0;
        do {
            i++;
            System.out.println(i);
            if (i==5) break;
        } while (i <= 9)
    }
}
```



## Keyword continue

- untuk segera lompat ke perulangan berikutnya.
- pernyataan-pernyataan setelah keyword continue dalam blok perulangan saat itu berarti diabaikan.

**Contoh :**

```
public class contohContinue {
    public static void main(String args[]) {
        int i=0;
        do {
            i++;
            if (i==3) continue;
            System.out.println("iterasi ke : "+i);
            if (i==5) break;
        } while(i <= 9);
    }
}
```



## Keyword return

untuk keluar dari suatu method

### Contoh cuplikan program :

```
int abs(int x) {  
    if (x >= 0)  
        return x;  
    else  
        return(-x)  
    . . .  
}  
}
```

## Soal Latihan

1. Sebutkan komponen-komponen yang terdapat dalam Java Development Kit!
2. Apa yang dimaksud dengan token dan identifier?
3. Sebutkan kegunaan dari operator , separator, keyword break dan keyword continue.
4. Buatlah program yang menampilkan tulisan : *Belajar java memang mudah dengan tekun*   *Jika dilakukan*
5. Buatlah contoh program lain yang menggunakan keyword **break** dan **continue**.