

MODUL PEMOGRAMAN JAVA

Oleh:

Chalifa Chazar

Pertemuan 2

TOKEN & LITERAL

Tujuan:

- » Mahasiswa mampu memahami konsep-konsep dasar Token dan Literal dalam bahasa pemograman Java.
- » Mahasiswa mampu menerapkan Token dan Literal ke dalam pengembangan aplikasi Java

Pustaka:

- » **Bambang Hariyanto (2014): Esensi-Esensi Bahasa Pemograman Java (Revisi Keempat). Informatika. Bandung.**

Kompilator Java melakukan pemindaian file kode sumber menjadi barisan token. **Token** adalah elemen terkecil di program yang memiliki arti bagi kompilator. Token-token tersebut mendefinisikan struktur bahasa Java.

Dalam Java terdapat 5 kategori token, yaitu:

- » Identifier
- » Keyword
- » Literal
- » Operator
- » Separator

1. IDENTIFIER

Identifier adalah token yang merepresentasikan nama, baik untuk variabel, metode, kelas, paket, dan **interface** untuk secara unik mengidentifikasi bagi kompilator dan memberikan nama yang berarti bagi pemogram.

Berikut ini beberapa aturan penamaan untuk identifier, antara lain:

- » Case sensitive.

- » Identifier harus dimulai dengan huruf, atau garis bawah (_), atau dollar (\$).
- » Simbol-simbol seperti '+', spasi, '@' tidak dapat digunakan.
- » Terdapat beberapa nama yang merupakan reserved word atau keyword dalam Java tidak boleh digunakan

Variabel

Variabel adalah suatu tempat untuk menampung data yang nilainya selalu berubah. Data yang disimpan dalam variabel akan hilang setelah program selesai dieksekusi. Pada pemograman Java, setiap variabel harus mempunyai tipe. Anda dapat mendeklarasikan variabel dengan menempatkan tipe data terlebih dahulu, diikuti nama variabel. Perhatikan contoh mendeklarasikan suatu variabel di bawah ini.

```
double salary;
```

```
int number;
```

```
int i, j;
```

```
String name;
```

Catatan:

Tanda titik koma (;) digunakan untuk mengakhiri masing-masing deklarasi. Sedangkan tanda koma (,) digunakan untuk mendeklarasikan suatu variabel yang merupakan satu pernyataan lengkap.

Berikut ini merupakan contoh penggunaan variabel dalam program Java.

```
public class Variabel {  
  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        // TODO code application logic here  
        String nama, jurusan;  
        int nim;  
  
        nama = "Chalifa";  
        nim = 23512127;  
        jurusan = "informatika";  
  
        System.out.println(nama);  
        System.out.println(nim);  
        System.out.println(jurusan);  
  
    }  
}
```

Konstanta

Konstanta adalah suatu tempat untuk menampung data yang nilainya tetap dan tidak pernah berubah. Pada pemograman Java, untuk mendeklarasikan konstanta digunakan kata kunci (keyword) **final**. Perhatikan contoh mendeklarasikan suatu variabel di bawah ini.

```
final double phi = 3.14;
```

Pendefinisian konstanta dapat dilakukan di luar main(). Dengan demikian konstanta dapat digunakan di metode-motode lain di kelas yang sama.

Berikut ini merupakan contoh penggunaan konstanta dalam program menghitung luas lingkaran.

```
public class Luas_lingkaran {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        final double phi = 3.14;
        double jari2, luas;

        jari2 = 10;
        luas = phi * jari2 * 2;
        System.out.println("Luas lingkaran adalah = " + luas);

    }

}
```

2. KEYWORD

Keyword (kata kunci) adalah identifier yang digunakan Java untuk suatu tujuan khusus. Berikut ini adalah daftar keyword pada bahasa pemograman Java.

abstract	continue	for	new	switch
assert***	default	goto*	package	synchronized
boolean	do	if	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum****	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp**	volatile
const*	float	native	super	while

* not used
 ** added in 1.2
 *** added in 1.4

Keyword tidak dapat digunakan sebagai identifier.

3. LITERAL

Penulisan besaran untuk variabel adalah penting, literal pada Java terdiri dari:

- » Angka
- » Karakter
- » String

Angka terdiri dari bilangan bulat (integer), bilangan mengambang (floating-point), dan boolean. Boolean dianggap angka. Nilai boolean untuk true dan false direpresentasikan sebagai 1 dan 0. **Karakter** selalu mengacu pada karakter Unicode. **String** berisi rangkaian karakter.

Literal Integer

Bilangan integer dapat mempunyai beberapa format berikut:

- » Decimal (basis 10), sebagai bilangan biasa tanpa notasi khusus.
- » Hexadecimal (basis 16), dimulai dengan 0x atau 0X.
- » Octal (basis 8), dimulai dengan 0.

Penulisan default untuk bilangan integer adalah bertipe int yaitu nilai 32-bit.

Literal Floating-Point

Floating-point merepresentasikan angka desimal dengan bagian pecahan seperti 3.14. bilangan floating-point dapat diekspresikan dalam notasi standar atau ilmiah. Secara default bilangan floating-point bertipe double 64-bit.

Literal Boolean

Java menyediakan tipe boolean dengan dua nilai yaitu true dan false. Masing-masing merepresentasikan salah satu dari dua kondisi yang mungkin.

Literal Character

Karakter direpresentasikan satu karakter Unicode tunggal dan dituliskan diantara tanda petik satu (' ... '). Karakter spesial (seperti karakter kendali dan karakter yang tidak dapat dicetak) direpresentasikan dengan backslash (\) diikuti kode karakter.

Berikut ini adalah tabel karakter spesial dalam bahasa Java.

Deskripsi	Representasi
Backslash	\\
Continuation	\
Backspace	\b
Carriage return	\r
Form feed	\f
Horizontal tab	\t
Newline	\n
Single quote	\'
Double quote	\"
Unicode character	\udddd
Octal character	\ddd

Literal String

String literal merepresentasikan banyak karakter dan ditulis diantara tanda petik dua (" "). Di Java, penamaan literal string berbeda dengan literal-literal lain. Untuk literal string akan menciptakan objek kelas string. Java menggunakan pendekatan ini karena cara terbaik menangani string adalah sebagai kelas.

----- GOOD LUCK -----