



MODUL 02

Pengantar Bahasa Pemrograman Java (lanjutan)

Kompetensi

1. Mengenal bahasa pemrograman JAVA.
2. Mampu membuat kelas sederhana dengan bahasa pemrograman JAVA.
3. Mengerti penggunaan komentar dalam bahasa pemrograman JAVA.
4. Mengetahui cara menggunakan Netbeans.
5. Mengetahui Penggunaan Character Literal.

Percobaan 1

Buat kode berikut ini kemudian masukkan character literals yang ada didalam table dibawahnya:

```
1 package modul2.percobaan1;
2 // @author Rizdania
3 public class Modul2Percobaan1 {
4     public static void main(String[] args) {
5         System.out.println("Nama Buah: ");
6         System.out.println("1. Manggis");
7         System.out.println("2. Semangka");
8         System.out.println("3. Durian");
9         System.out.println("4. Salak");
10    }
11 }
```

Hasil running program:

```
run:
Nama Buah:
1. Manggis
2. Semangka
3. Durian
4. Salak
BUILD SUCCESSFUL (total time: 4 seconds)
```

Kemudian sisipkan character literals yang ada pada tabel ke dalam akhir string literalsnya, seperti kode dibawah ini:

| Kode | Nama |
|------|-----------|
| \b | Backspace |
| \n | Line feed |

| | |
|----|----------|
| \t | Tabulasi |
|----|----------|

```

1  package modul2.percobaan1;
2  //@author Rizdania
3  public class Modul2Percobaan1 {
4      public static void main(String[] args) {
5          System.out.println("Nama \t Buah: ");
6          System.out.println("1.\b Manggis");
7          System.out.println("2. Semangka \n");
8          System.out.println("3. Durian");
9          System.out.println("4. Salak");
10     }
11 }

```

Hasil running program:

```

run:
Nama      Buah:
1 Manggis
2. Semangka

3. Durian
4. Salak
BUILD SUCCESSFUL (total time: 0 seconds)

```

Instruksi yang harus dijalankan:

- Compile dan jalankan program tersebut!
- Apakah yang dapat dijelaskan dari program diatas?

Percobaan 2

Dengan menggunakan character literals yang ada pada tabel diatas, coba buat program dengan hasil keluaran seperti di bawah ini:

```

run:
=====
Daftar Pemenang Kuis Bulan Oktober
=====
No      Nama      Kota
1       Dian      Jakarta
2       Herdi      Medan
3       Randy      Surabaya

=====
Selamat kepada pemenang
=====
BUILD SUCCESSFUL (total time: 0 seconds)

```

Kode Programnya:

```
1 package modul2;
2
3 public class percobaan2 {
4     public static void main(String[] args) {
5         System.out.println("=====");
6         System.out.println("Daftar Pemenang Kuis Bulan Oktober");
7         System.out.println("=====");
8         System.out.println("No \t Nama \t Kota ");
9         System.out.println("1 \t Dian \t Jakarta");
10        System.out.println("2 \t Herdi \t Medan");
11        System.out.println("3 \t Randy \t Surabaya \n");
12        System.out.println("=====");
13        System.out.println("\t Selamat kepada pemenang");
14        System.out.println("=====");
15    }
16 }
```

Character Literal dalam Pemrograman Java

Literal adalah nilai yang ditulis dalam kode program, misalnya nilai yang ditulis untuk memberikan (*assign*) nilai terhadap sebuah variabel. Ada beberapa jenis literal yaitu integer literal, floating-point literal, boolean literal, character literal, dan String literal. Masing-masing tergantung pada jenis atau tipe data.

Literal karakter seperti huruf, angka, dan simbol. Literal karakter digunakan untuk memberi nilai pada variabel dengan tipe data char.

- Nilai diapit dengan tanda kutip satu ('). Contoh, 'A', '#', '+', '1'.
- Penulisan nilai hanya terdiri dari satu karakter, kecuali escape character dan unicode notation. Whitespace (spasi) terhitung satu karakter. Anda tidak bisa menulis seperti, 'a', 'ab', '10', '-1'.
- Untuk escape character, seperti pindah baris, simbol slash (\), kutip satu (') dan dua ("), tambahkan awalan tanda slash (\), sebelum simbol tersebut. Contoh, '\n', '\n', '\n', '\\'.
- Unicode notation dengan menggunakan awalan \u diikuti dengan nilai bilangan bulat dari 0 sampai 65536 dalam bentuk heksadesimal yaitu 0000 sampai FFFF. Contoh, '\u0041', '\u0059', '\u003F'. Nilai heksadesimal tidak case sensitive, sehingga b dengan B dianggap sama.



- Bisa diberikan dalam bentuk bilangan bulat antara 0 sampai 65536, untuk karakter yang biasa kita gunakan bisa dilihat nilai bilangan bulatnya pada tabel ASCII.

Percobaan 3

Salinlah program berikut ini:

```
1  package modul2;
2
3  public class CharacterLiterals {
4      public static void main(String[] args)
5      {
6          /*
7           * Literal karakter A dalam bentuk
8           * karakter, unicode dan bilangan bulat.
9           */
10         char k1 = 'A';
11         char k2 = '\u0041';
12         char k3 = 65;
13
14         // Literal escape character untuk pindah baris.
15         char pb = '\n';
16         System.out.println(k1);
17         System.out.println(k2);
18         System.out.println(pb);
19         System.out.println(k3);
20     }
21 }
```

Hasil running program:

```
run:
A
A

A
BUILD SUCCESSFUL (total time: 0 seconds)
```

Unicode adalah suatu standar teknis yang dirancang untuk mengizinkan teks dan simbol dari semua sistem tulisan di dunia untuk ditampilkan dan dimanipulasi secara konsisten oleh komputer. Dikembangkan secara tandem dengan standar Universal Character Set dan dipublikasikan dalam bentuk buku The Unicode Standard. Bisa dicek di <https://unicode-table.com/en/>

Berikut adalah beberapa contoh dari tabel unicode:

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|------|---|---|---|---|----|---|---|---|---|---|---|---|---|---|---|---|
| 0020 | | ! | " | # | \$ | % | & | ' | (|) | * | + | , | - | . | / |
| 0030 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | : | ; | < | = | > | ? |
| 0040 | @ | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
| 0050 | P | Q | R | S | T | U | V | W | X | Y | Z | [| \ |] | ^ | _ |
| 0060 | ` | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o |
| 0070 | p | q | r | s | t | u | v | w | x | y | z | { | | } | ~ | ␣ |

Kemudian koreksi program diatas menjadi seperti berikut ini:

```

1  package modul2;
2
3  public class CharacterLiterals {
4      public static void main(String[] args)
5      {
6          /*
7           * Literal karakter A dalam bentuk
8           * karakter, unicode dan bilangan bulat.
9           */
10         char k1 = 'A';
11         char k2 = '\u0041';
12         char k3 = 65;
13
14         // Literal escape character untuk pindah baris.
15         char pb = '\n';
16         System.out.print(k1);
17         System.out.print(k2);
18         System.out.print(pb);
19         System.out.print(k3);
20     }
21 }

```

Hasil running programnya:

```

run:
AA
ABUILD SUCCESSFUL (total time: 0 seconds)

```

System.out.print(), tidak akan pindah baris setelah menampilkan output, sebaliknya System.out.println(), selalu pindah baris setelah menampilkan nilai.