

# LAPORAN PRAKTIKUM PEMROGRAMAN DASAR

## BAB 3 Pengulangan

Nama : Kusindra Aji Rabbany

Program Studi : Teknologi Informasi

Kelas : B

Asisten 1 : Dionisius Seraf Saputra

Asisten 2 : Malika Zahro Nadia Aulia

---

### JAWABAN

#### 1. A. Flowchart dan/atau Pseudocode

BEGIN

CREATE scanner for input

PROMPT "Masukkan nilai n = "

READ n

FOR i FROM 0 TO n-1 DO

SET spaces = string of (n - i) spaces

SET stars = string of (i + 1) asterisks

SET line = spaces + stars

DISPLAY line

END FOR

CLOSE scanner

END

#### B. Implementasi

```
1  import java.util.Scanner;
2
3  public class Star {
4      public static void main(String[] args) {
5          Scanner scan = new Scanner(System.in);
6
7          System.out.print("Masukkan nilai n = ");
8          int n = scan.nextInt();
9
10         for (int i = 0; i < n; i++) {
11             String line = (" ".repeat(n - i)) + ("*".repeat(i + 1));
12             System.out.println(line);
13         }
14
15         scan.close();
```

```
16 }
17 }
```

### C. Tampilan Keluaran (Screenshot)

```
ben@kusindras-macbook ~/kuliah/Semester 1/Pemrograman Dasar/Praktikum 3 java Star.java
Masukkan nilai n = 4
*
**
***
****

ben@kusindras-macbook ~/kuliah/Semester 1/Pemrograman Dasar/Praktikum 3 java Star.java
Masukkan nilai n = 6
*
**
***
****
*****
*****
```

### D. Pembahasan dan Analisis

Kode ini digunakan untuk menampilkan pola bintang (\*) secara terurut dari atas ke bawah. Program meminta masukan berupa sebuah bilangan bulat  $n$  yang menunjukkan tinggi pola. Setiap baris yang dicetak terdiri atas kombinasi spasi dan tanda bintang sesuai perhitungan yang ditentukan.

- Baris ke-1 digunakan untuk membuat objek Scanner yang berfungsi membaca input dari pengguna.
- Baris ke-2 menampilkan pesan ke layar agar pengguna memasukkan nilai  $n$ .
- Baris ke-3 menyimpan input bilangan bulat dari pengguna ke dalam variabel  $n$ .
- Baris ke-4 hingga baris ke-8 digunakan untuk melakukan perulangan dari  $i = 0$  sampai  $i < n$ . Di dalam perulangan, program membentuk string yang terdiri atas  $n - i$  spasi diikuti  $i + 1$  tanda bintang. Setelah itu, baris tersebut ditampilkan ke layar.
- Baris terakhir menutup objek Scanner untuk menghindari kebocoran sumber daya.

Dari percobaan dan analisis yang telah dilakukan, apabila baris pembentukan spasi (" ".repeat( $n - i$ )) dihilangkan, maka pola yang dicetak hanya berupa deretan bintang tanpa ada pergeseran ke kanan. Sedangkan apabila baris pembentukan bintang ("\*".repeat( $i + 1$ )) dihilangkan, maka output yang muncul hanya berupa spasi kosong tanpa pola bintang. Oleh karena itu, supaya pola bintang dapat terbentuk dengan benar, program harus menggabungkan antara spasi dan bintang pada setiap baris dengan mempertimbangkan posisi baris ke- $i$ .

### E. Refleksi Diri

Saya dapat mengetahui cara untuk menuliskan code ini dengan lebih efektif, yaitu mengubah 3 for loop menjadi 1 for loop. Saya melakukannya dengan mengubah looping bintang dan spasi menggunakan fungsi repeat yang dimiliki oleh string.

## 2. A. Flowchart dan/atau Pseudocode

BEGIN

CREATE scanner for input

DECLARE variable choice as integer

REPEAT

DISPLAY "MENU"

DISPLAY "0. KELUAR"

DISPLAY "1. HITUNG VOLUME BALOK"

DISPLAY "2. HITUNG VOLUME BOLA"

DISPLAY "3. HITUNG VOLUME KERUCUT"

DISPLAY "4. HITUNG VOLUME SILINDER"

DISPLAY "5. HITUNG VOLUME LIMAS SEGITIGA"

PROMPT "MASUKKAN PILIHAN ANDA: "

READ choice

SWITCH (choice)

CASE 1:

PROMPT "Masukkan panjang: "

READ length

PROMPT "Masukkan lebar: "

READ width

PROMPT "Masukkan tinggi: "

READ height

SET volumeBlock = length \* width \* height

DISPLAY "Volum Balok = " + volumeBlock

BREAK

CASE 2:

PROMPT "Masukkan jar-jari: "

READ radiusSphere

SET volumeSphere =  $(4/3) * \pi * (\text{radiusSphere}^3)$

DISPLAY "Volume Bola = " + volumeSphere

BREAK

CASE 3:

PROMPT "Masukkan jari-jari alas: "

READ radiusCone

PROMPT "Masukkan tinggi: "

READ heightCone

SET volumeCone =  $(1/3) * \pi * (\text{radiusCone}^2) * \text{heightCone}$

DISPLAY "Volume Kerucut = " + volumeCone

BREAK

CASE 4:

```

PROMPT "Masukkan jari-jari alas: "
READ radiusCylinder
PROMPT "Masukkan tinggi: "
READ heightCylinder
SET volumeCylinder =  $\pi$  * (radiusCylinder^2) * heightCylinder
DISPLAY "Volume Silinder = " + volumeCylinder
BREAK

```

CASE 5:

```

PROMPT "Masukkan luas alas segitiga: "
READ baseArea
PROMPT "Masukkan tinggi limas: "
READ heightPyramid
SET volumePyramid = (1/3) * baseArea * heightPyramid
DISPLAY "Volume Luas Segitiga = " + volumePyramid
BREAK

```

CASE 0:

```

DISPLAY "Keluar."
BREAK

```

DEFAULT:

```

DISPLAY "Pilihan tidak valid. Silakan coba lagi."
END SWITCH

```

UNTIL choice = 0

CLOSE scanner

END

## B. Implementasi

```

1  import java.util.Scanner;
2
3  public class VolumeCalculator {
4      public static void main(String[] args) {
5          Scanner input = new Scanner(System.in);
6          int choice;
7
8          do {
9              System.out.println("MENU");
10             System.out.println("0. KELUAR");
11             System.out.println("1. HITUNG VOLUME BALOK");
12             System.out.println("2. HITUNG VOLUME BOLA");
13             System.out.println("3. HITUNG VOLUME KERUCUT");
14             System.out.println("4. HITUNG VOLUME SILINDER");
15             System.out.println("5. HITUNG VOLUME LIMAS SEGITIGA");
16             System.out.print("\nMASUKKAN PILIHAN ANDA: ");
17             choice = input.nextInt();
18
19             switch (choice) {
20                 case 1:
21                     System.out.print("Masukkan panjang: ");

```

```

22         double length = input.nextDouble();
23         System.out.print("Masukkan lebar: ");
24         double width = input.nextDouble();
25         System.out.print("Masukkan tinggi: ");
26         double height = input.nextDouble();
27         double volumeBlock = length * width * height;
28         System.out.println("Volume Balok = " + volumeBlock + "\n");
29         break;
30
31         case 2:
32             System.out.print("Masukkan jari-jari: ");
33             double radiusSphere = input.nextDouble();
34             double volumeSphere = (4.0 / 3.0) * Math.PI *
Math.pow(radiusSphere, 3);
35             System.out.println("Volume Bola = " + volumeSphere + "\n");
36             break;
37
38         case 3:
39             System.out.print("Masukkan jari-jari alas: ");
40             double radiusCone = input.nextDouble();
41             System.out.print("Masukkan tinggi: ");
42             double heightCone = input.nextDouble();
43             double volumeCone = (1.0 / 3.0) * Math.PI * Math.pow(radiusCone,
2) * heightCone;
44             System.out.println("Volume Kerucut = " + volumeCone + "\n");
45             break;
46
47         case 4:
48             System.out.print("Masukkan jari-jari alas: ");
49             double radiusCylinder = input.nextDouble();
50             System.out.print("Masukkan tinggi: ");
51             double heightCylinder = input.nextDouble();
52             double volumeCylinder = Math.PI * Math.pow(radiusCylinder, 2) *
heightCylinder;
53             System.out.println("Volume Silinder = " + volumeCylinder + "\n");
54             break;
55
56         case 5:
57             System.out.print("Masukkan luas alas segitiga: ");
58             double baseArea = input.nextDouble();
59             System.out.print("Masukkan tinggi limas: ");
60             double heightPyramid = input.nextDouble();
61             double volumePyramid = (1.0 / 3.0) * baseArea * heightPyramid;
62             System.out.println("Volume Limas Segitiga = " + volumePyramid +
"\n");
63             break;
64
65         case 0:
66             System.out.println("Keluar.");
67             break;
68
69         default:
70             System.out.println("Pilihan tidak valid. Silakan coba lagi.\n");
71     }
72     while (choice != 0);
73     input.close();
74 }
75 }
76 }

```

### C. Tampilan Keluaran (Screenshot)

```
MENU
0. KELUAR
1. HITUNG VOLUME BALOK
2. HITUNG VOLUME BOLA
3. HITUNG VOLUME KERUCUT
4. HITUNG VOLUME SILINDER
5. HITUNG VOLUME LIMAS SEGITIGA

MASUKKAN PILIHAN ANDA: 1
Masukkan panjang: 10
Masukkan lebar: 10
Masukkan tinggi: 10
Volume Balok = 1000.0

MENU
0. KELUAR
1. HITUNG VOLUME BALOK
2. HITUNG VOLUME BOLA
3. HITUNG VOLUME KERUCUT
4. HITUNG VOLUME SILINDER
5. HITUNG VOLUME LIMAS SEGITIGA

MASUKKAN PILIHAN ANDA: 2
Masukkan jari-jari: 16
Volume Bola = 17157.284678805056
```

#### D. Pembahasan dan Analisis

Kode ini digunakan untuk melakukan perhitungan volume berbagai bangun ruang secara interaktif berdasarkan pilihan pengguna. Perhitungan dilakukan dengan menampilkan menu utama, kemudian menerima input pilihan dari pengguna, dan selanjutnya menghitung volume bangun ruang yang dipilih menggunakan rumus masing-masing.

- Baris ke-1 digunakan untuk mengimpor kelas Scanner agar program dapat membaca input dari pengguna.
- Baris ke-2 sampai baris ke-3 mendefinisikan kelas VolumeCalculator serta metode main sebagai titik awal eksekusi program.
- Baris ke-4 membuat objek Scanner bernama input untuk membaca data dari keyboard.
- Baris ke-5 mendeklarasikan variabel choice yang berfungsi menyimpan pilihan menu dari pengguna.
- Baris ke-6 hingga baris ke-52 berisi struktur perulangan do-while yang memastikan menu akan terus ditampilkan sampai pengguna memilih keluar (choice = 0).
- Di dalam perulangan, baris ke-7 sampai baris ke-13 digunakan untuk menampilkan menu pilihan bangun ruang.
- Baris ke-14 menerima input pilihan dari pengguna lalu menyimpannya pada variabel choice.
- Baris ke-15 sampai baris ke-51 merupakan struktur switch-case yang menentukan jalannya program sesuai dengan pilihan pengguna.

Jika dianalisis lebih lanjut:

- Apabila bagian case 1 (perhitungan volume balok) dihapus, maka program tidak dapat menghitung volume balok dan akan menampilkan pesan "Pilihan tidak valid" ketika pengguna memilih opsi 1.
- Apabila bagian case 2 dihapus, maka perhitungan volume bola tidak bisa dilakukan, sehingga program hanya bisa menghitung bangun ruang lainnya.

- Jika baris yang menutup perulangan do-while dihilangkan, maka program hanya akan menampilkan menu sekali dan langsung selesai setelah satu kali perhitungan.
- Apabila baris penutupan Scanner (input.close()) dihapus, program tetap berjalan tetapi berpotensi menimbulkan kebocoran sumber daya pada penggunaan input.

#### **E. Refleksi Diri**

Saya dapat mengetahui perbedaan penggunaan while dan do while. Pada kasus ini, lebih baik menggunakan do while dikarenakan program pasti harus dijalankan terlebih dahulu sebelum menentukan kondisi perulangan kondisi selanjutnya.

#### **KESIMPULAN**

Percobaan ini menunjukkan bahwa konsep looping memiliki peran penting dalam menyelesaikan masalah pemrograman. Pada program pola bintang, perulangan digunakan untuk mengatur jumlah spasi dan bintang sehingga terbentuk pola sesuai nilai input. Pada program perhitungan volume, perulangan do-while memastikan menu selalu ditampilkan dan pengguna dapat melakukan perhitungan berulang kali hingga memilih keluar.

#### **DAFTAR REFERENSI**

-