

TUGAS 3 PEMROGRAMAN BERORIENTASI OBJEK

Nama : Rahman Ramadhan

Nim : 13020220128

Kelas : A4

Output Code Program 1

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -javaagent:
Baca string dan Integer:
masukkan sebuah string: Halo
String yang dibaca : Halo
Process finished with exit code 0
```

Penjelasan Mengenai Program :

1. Import statement: Import beberapa kelas yang diperlukan.
2. Deklarasi kelas utama **BacaString**.
3. Deklarasi variabel **str** untuk menyimpan string yang akan dibaca.
4. Membuat objek **BufferedReader** untuk membaca input dari **System.in**.
5. Mencetak pesan ke layar untuk meminta pengguna memasukkan string.
6. Memanggil metode **readLine()** dari objek **BufferedReader** untuk membaca string yang dimasukkan oleh pengguna.
7. Mencetak kembali string yang telah dibaca ke layar.

Output Code Program 2

[illegible]

Penjelasan Mengenai Program :

1. Deklarasi kelas **Forever** yang berisi metode **main**.
2. Di dalam metode **main**, program mencetak pesan "Program akan looping, akhiri dengan ^c" menggunakan **System.out.println**.
3. Program memasuki loop **while (true)** yang akan terus berjalan selama kondisinya adalah **true**. Ini berarti bahwa loop ini akan terus berjalan tanpa henti.
4. Di dalam loop, program mencetak pesan "Print satu baris" menggunakan **System.out.print**.
5. Karena tidak ada kode di dalam loop yang mengubah nilai kondisi **true**, maka loop akan berjalan tanpa henti, mencetak pesan yang sama berulang kali ke konsol.
6. Program akan terus berjalan sampai dihentikan secara paksa oleh pengguna dengan menekan kombinasi tombol ^c (Ctrl + C) di terminal (di lingkungan UNIX/Linux) atau dengan menekan tombol "Terminate" di IDE atau lingkungan pengembangan yang digunakan.

Output Kode Program 3

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -jd
Contoh IF satu kasus
Ketikkan suatu nilai integer : 6

Nilai a positif 6
Process finished with exit code 0
```

Penjelasan Mengenai Program :

1. Import statement: Mengimpor kelas **Scanner** yang digunakan untuk membaca input.
2. Deklarasi kelas utama **If1**.
3. Deklarasi variabel **masukan** untuk membaca input dari pengguna.
4. Deklarasi variabel **a** untuk menyimpan nilai integer yang dimasukkan.
5. Meminta pengguna untuk memasukkan nilai integer.
6. Memanggil metode **nextInt()** dari objek **masukan** untuk membaca nilai integer yang dimasukkan oleh pengguna.

7. Menggunakan struktur kondisional **if** untuk memeriksa apakah nilai **a** lebih besar atau sama dengan 0.
8. Jika kondisi terpenuhi (nilai **a** positif), maka program mencetak pesan yang menyatakan bahwa nilai **a** adalah positif.

Output Kode Program 4

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -ja
Contoh IF dua kasus
Ketikkan suatu nilai integer :5
Nilai a positif 5

Process finished with exit code 0
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **If2** yang berisi metode **main**.
2. Di dalam metode **main**, sebuah objek **Scanner** bernama **masukan** dibuat untuk membaca input dari **System.in**.
3. Sebuah variabel **a** dengan tipe data integer dideklarasikan untuk menyimpan nilai yang dimasukkan oleh pengguna.
4. Program mencetak pesan "Contoh IF dua kasus" dan "Ketikkan suatu nilai integer :"
menggunakan **System.out.print**.
5. Program menggunakan **masukan.nextInt()** untuk membaca nilai integer yang dimasukkan oleh pengguna dan menyimpannya ke dalam variabel **a**.
6. Program memeriksa apakah nilai **a** lebih besar dari atau sama dengan 0 dengan menggunakan pernyataan kondisional **if (a >= 0)**.
7. Jika kondisi tersebut benar (nilai **a** positif atau nol), program mencetak pesan "Nilai a positif" diikuti dengan nilai **a** menggunakan **System.out.println**.
8. Jika kondisi tersebut salah (nilai **a** negatif), program menggunakan blok **else** dan mencetak pesan "Nilai a negatif" diikuti dengan nilai **a** menggunakan **System.out.println**.

Output Kode Program 5

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents/Home/bin/java -j
Contoh IF tiga kasus
Ketikkan suatu nilai integer :3
Nilai a positif 3

Process finished with exit code 0
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **If3** dengan metode **main**.
2. Membuat objek **Scanner** untuk membaca input dari pengguna.
3. Mendeklarasikan variabel **a** untuk menyimpan nilai yang dimasukkan oleh pengguna.
4. Mencetak pesan permintaan input dan membaca nilai integer dari pengguna.
5. Memeriksa tiga kasus nilai **a** menggunakan pernyataan kondisional **if-else if-else**.
6. Menampilkan hasil sesuai dengan kasus nilai **a**.

Output Kode Program 6

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents
true
benar

Process finished with exit code 0
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **KasusBoolean**.
2. Deklarasi variabel **bool** dengan tipe data boolean.
3. Menginisialisasi variabel **bool** dengan nilai **true**.
4. Menggunakan struktur kondisional if-else untuk mengecek nilai **bool**.
 - o Jika **bool** adalah **true**, maka program akan mencetak "true".

- Jika **bool** adalah **false**, maka program akan mencetak "false".
- 5. Menggunakan operator logika NOT (!) untuk membalikkan nilai **bool**.
- 6. Menggunakan struktur kondisional if-else untuk mengecek nilai **bool** yang telah dibalik.
 - Jika **bool** adalah **true** setelah dibalik, maka program akan mencetak "benar".
 - Jika **bool** adalah **false** setelah dibalik, maka program akan mencetak "salah".

Output Kode Program 7

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents
Ketikkan sebuah huruf, akhiri dengan RETURN
0
Yang anda ketik adalah huruf mati
Process finished with exit code 0
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **KasusSwitch** dengan metode **main**.
2. Membuat objek **Scanner** untuk membaca input dari pengguna.
3. Mendeklarasikan variabel **cc** untuk menyimpan karakter yang dimasukkan oleh pengguna.
4. Mencetak pesan permintaan input karakter.
5. Membaca satu karakter dari pengguna dan menyimpannya dalam variabel **cc**.
6. Menggunakan pernyataan **switch** untuk memeriksa nilai karakter yang dimasukkan oleh pengguna.
7. Setiap **case** pada **switch** mencocokkan karakter yang dimasukkan oleh pengguna dengan karakter tertentu, dan mencetak pesan sesuai.
8. Jika tidak ada kasus yang cocok, akan dieksekusi bagian **default**, mencetak pesan bahwa karakter yang dimasukkan adalah "huruf mati".

Output Kode Program 8

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents
Jari-jari lingkaran =25
Luas lingkaran = 1963.4375
Akhir program

Process finished with exit code 0
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **Konstant**.
2. Deklarasi konstanta **PHI** dengan nilai 3.1415f menggunakan kata kunci **final**, yang menandakan bahwa nilai konstanta ini tidak dapat diubah setelah diberikan nilai.
3. Deklarasi variabel **r** untuk menyimpan nilai jari-jari lingkaran.
4. Membuat objek **Scanner** untuk membaca input dari pengguna.
5. Meminta pengguna untuk memasukkan nilai jari-jari lingkaran.
6. Menghitung luas lingkaran dengan rumus $PI * r * r$ dan mencetak hasilnya.
7. Menggunakan konstanta **PHI** untuk menggantikan nilai π dalam perhitungan luas lingkaran.
8. Mencetak pesan "Akhir program" sebagai penanda akhir dari program.

Output Kode Program 9

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents
Maksimum dua bilangan :
Ketikkan dua bilangan, pisahkan dg RETURN :
3
8
Ke dua bilangan : a = 3 b = 8
Nilai b yang maksimum: 8
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **Max2** dengan metode **main**.
2. Membuat dua variabel **a** dan **b** untuk menyimpan dua bilangan yang akan dibaca.

3. Membuat objek **Scanner** untuk membaca input dari pengguna.
4. Mencetak pesan permintaan untuk memasukkan dua bilangan.
5. Membaca dua bilangan yang dimasukkan oleh pengguna.
6. Menampilkan kembali kedua bilangan yang dimasukkan.
7. Menggunakan pernyataan **if** untuk memeriksa bilangan mana yang lebih besar.
8. Menampilkan bilangan yang lebih besar.

Output Kode Program 10

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents
Baca N, print 1 s/d N N = 10
1
2
3
4
5
6
7
8
9
10
Akhir program
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **PriFor** yang berisi metode **main**.
2. Di dalam metode **main**, dua variabel **i** dan **N** dengan tipe data integer dideklarasikan untuk melakukan iterasi dan menyimpan nilai **N** yang akan dibaca dari pengguna.
3. Program membuat objek **Scanner** bernama **masukan** untuk membaca input dari **System.in**.
4. Program mencetak pesan "Baca N, print 1 s/d N" dan "N = " menggunakan **System.out.print**.
5. Program menggunakan **masukan.nextInt()** untuk membaca nilai **N** yang dimasukkan oleh pengguna dari konsol dan menyimpannya dalam variabel **N**.
6. Program menggunakan loop **for** untuk mencetak angka dari 1 hingga **N**:
 - Variabel **i** diinisialisasi dengan nilai 1.
 - Loop akan terus berjalan selama nilai **i** kurang dari atau sama dengan **N**.

- Pada setiap iterasi, nilai **i** dicetak menggunakan **System.out.println(i)**.
 - Setelah mencetak nilai **i**, nilai **i** akan ditambahkan dengan 1 untuk iterasi selanjutnya.
7. Setelah loop selesai, program mencetak pesan "Akhir program" menggunakan **System.out.println**.

Output Kode Program 11

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents  
Nilai N >0 = 5  
Print i dengan ITERATE :  
1  
2  
3  
4  
5
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **PrintIterasi**.
2. Deklarasi variabel **N** dan **i** untuk digunakan dalam program.
3. Membuat objek **Scanner** untuk membaca input dari pengguna.
4. Meminta pengguna untuk memasukkan nilai **N**.
5. Menginisialisasi nilai **i** dengan 1 sebagai elemen pertama.
6. Menggunakan loop **for(;;)** untuk melakukan iterasi tanpa kondisi berhenti yang jelas.
 - Pada setiap iterasi, nilai **i** dicetak ke layar.
 - Jika nilai **i** sama dengan **N**, program akan menggunakan **break** untuk keluar dari loop.
 - Jika nilai **i** tidak sama dengan **N**, nilai **i** akan diincrement untuk mencetak nilai berikutnya.
7. Setelah keluar dari loop, program selesai dan mencetak pesan akhir.

Output Kode Program 12

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents
Nilai N >0 = 6
Print i dengan REPEAT:
1
2
3
4
5
6
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **PrintRepeat** dengan metode **main**.
2. Membuat dua variabel **N** dan **i** untuk menyimpan nilai yang akan dibaca dan nilai iterasi.
3. Membuat objek **Scanner** untuk membaca input dari pengguna.
4. Mencetak pesan permintaan untuk memasukkan nilai **N**.
5. Membaca nilai **N** yang dimasukkan oleh pengguna.
6. Menginisialisasi nilai iterasi **i** dengan 1.
7. Mencetak pesan sebelum melakukan iterasi.
8. Menggunakan loop **do-while** untuk mencetak nilai **i** dari 1 hingga **N**.
9. Di dalam loop, mencetak nilai **i** dan menambahkan **i** dengan 1.
10. Kondisi untuk menjalankan loop adalah selama **i** kurang dari atau sama dengan **N**.

Output Kode Program 13

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents
Nilai N >0 = 7
Print i dengan WHILE:
1
2
3
4
5
6
7
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **PrintWhile**.
2. Deklarasi variabel **N** dan **i** untuk digunakan dalam program.
3. Membuat objek **Scanner** untuk membaca input dari pengguna.
4. Meminta pengguna untuk memasukkan nilai **N**.
5. Menginisialisasi nilai **i** dengan 1 sebagai elemen pertama.
6. Menggunakan loop **while** dengan kondisi **i <= N** untuk mencetak nilai dari 1 hingga **N**.
 - o Pada setiap iterasi, nilai **i** dicetak ke layar.
 - o Setelah dicetak, nilai **i** diincrement untuk mencetak nilai berikutnya.
 - o Loop akan berhenti ketika nilai **i** lebih besar dari **N**.
7. Setelah loop selesai, program mencetak pesan akhir.

Output Kode Program 14

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents
Nilai N >0 = 7
Print i dengan WHILE (ringkas):
1
2
3
4
5
6
7
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **PrintWhile1** dengan metode **main**.
2. Membuat dua variabel **N** dan **i** untuk menyimpan nilai yang akan dibaca dan nilai iterasi. Variabel **i** diinisialisasi dengan nilai 1.
3. Membuat objek **Scanner** untuk membaca input dari pengguna.
4. Mencetak pesan permintaan untuk memasukkan nilai **N**.
5. Membaca nilai **N** yang dimasukkan oleh pengguna.
6. Mencetak pesan sebelum melakukan iterasi.
7. Menggunakan loop **while** untuk mencetak nilai **i** dari 1 hingga **N** dengan operator penambahan **++** untuk nilai **i**.
8. Kondisi untuk menjalankan loop adalah selama **i** kurang dari atau sama dengan **N**.

Output Kode Program 15

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents
Masukkan nilai x (int), akhiri dg 999 : 1
Masukkan nilai x (int), akhiri dg 999 : 2
Masukkan nilai x (int), akhiri dg 999 : 3
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 6
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **PrintXinterasi** dengan metode **main**.
2. Membuat dua variabel **Sum** dan **x** untuk menyimpan total nilai yang dibaca dan nilai yang dibaca saat ini.
3. Membuat objek **Scanner** untuk membaca input dari pengguna.
4. Mencetak pesan permintaan untuk memasukkan nilai **x**.
5. Membaca nilai **x** yang dimasukkan oleh pengguna.
6. Jika nilai **x** sama dengan 999, mencetak pesan "Kasus kosong".
7. Jika tidak, inisialisasi total **Sum** dengan nilai **x**.
8. Menggunakan loop **for** tanpa kondisi awal, kondisi, atau pernyataan iterasi.
9. Di dalam loop, mencetak pesan untuk memasukkan nilai **x** lagi.
10. Membaca nilai **x** berikutnya.
11. Jika nilai **x** sama dengan 999, keluar dari loop.
12. Jika tidak, menambahkan nilai **x** ke total **Sum**.
13. Setelah loop selesai, mencetak total **Sum**.
14. Program selesai.

Output Kode Program 16

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents  
Masukkan nilai x (int), akhiri dg 999 : 2  
Masukkan nilai x (int), akhiri dg 999 : 3  
Masukkan nilai x (int), akhiri dg 999 : 4  
Masukkan nilai x (int), akhiri dg 999 : 999  
Hasil penjumlahan = 9
```

Penjelasan Mengenai Program :

1. Deklarasi kelas **PrintXRepeat**.
2. Deklarasi variabel **Sum** untuk menyimpan hasil penjumlahan dan variabel **x** untuk menyimpan nilai yang dimasukkan oleh pengguna.
3. Membuat objek **Scanner** untuk membaca input dari pengguna.
4. Meminta pengguna untuk memasukkan nilai x (integer) dan menyimpannya dalam variabel x.
5. Jika pengguna langsung memasukkan 999, maka program akan mencetak "Kasus kosong" karena tidak ada data yang dijumlahkan.
6. Jika pengguna memasukkan nilai selain 999, maka program akan menggunakan loop **do-while** untuk terus meminta pengguna memasukkan nilai x dan menambahkannya ke dalam variabel Sum.
 - o Setelah pengguna memasukkan nilai 999, loop akan berhenti.
7. Setelah loop selesai, program akan mencetak hasil penjumlahan dari semua nilai x yang dimasukkan pengguna kecuali 999.

Output Kode Program 17

```

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents,
Masukkan nilai x (int), akhiri dg 999 : 5
Masukkan nilai x (int), akhiri dg 999 : 6
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 11

```

Penjelasan Mengenai Program :

1. Program membaca nilai-nilai dari pengguna dan menjumlahkannya menggunakan loop **while**.
2. Variabel **Sum** dan **x** digunakan untuk menyimpan total jumlah dan nilai yang dibaca.
3. Objek **Scanner** digunakan untuk membaca input dari pengguna.
4. Program mencetak pesan permintaan untuk memasukkan nilai **x**.
5. Loop **while** digunakan untuk membaca nilai **x** dan menambahkannya ke total **Sum** sampai pengguna memasukkan nilai 999.
6. Setelah loop selesai, program mencetak total jumlah nilai yang dimasukkan.

Output Kode Program 18

```

/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents,
Maksimum dua bilangan
Ketikkan dua bilangan, pisahkan dg RETURN :
2
8
Ke dua bilangan : a = 2 b = 8
Maksimum = 8
Tukar kedua bilangan...
Ke dua bilangan setelah tukar: a = 8 b = 2

```

Penjelasan Mengenai Program :

1. **Prosedur tukar(int a, int b):** Prosedur ini bertujuan untuk menukar nilai dari dua variabel **a** dan **b**. Pada prosedur ini, nilai variabel **a** dan **b** ditukar menggunakan variabel sementara **temp**.
2. **Fungsi maxab(int a, int b):** Fungsi ini digunakan untuk mencari nilai maksimum dari dua bilangan bulat **a** dan **b**. Fungsi ini mengembalikan nilai **a** jika **a** lebih besar dari atau sama dengan **b**, dan mengembalikan nilai **b** jika sebaliknya.
3. **Program Utama (main):** Bagian ini merupakan program utama yang akan dieksekusi pertama kali. Pada bagian ini, program akan membaca dua bilangan dari input pengguna menggunakan kelas **Scanner**. Setelah itu, program akan menampilkan kedua bilangan tersebut, mencari nilai

maksimum dari kedua bilangan menggunakan fungsi **maxab**, dan menampilkan hasilnya. Selain itu, program juga akan memanggil prosedur **tukar** untuk menukar nilai kedua bilangan tersebut, dan menampilkan hasilnya setelah ditukar.

Output Kode Program 19

```
/Library/Java/JavaVirtualMachines/jdk-21.jdk/Contents,  
Contoh IF tiga kasus  
Temperatur (der. C) = 13  
Wujud air cair  
13  
Process finished with exit code 0
```

Penjelasan Mengenai Program :

1. **Input Suhu:** Program ini meminta pengguna untuk memasukkan suhu dalam derajat Celsius menggunakan kelas **Scanner**.
2. **Pemilihan Tiga Kasus:**
 - Jika suhu kurang dari 0 derajat Celsius ($T < 0$), program akan mencetak bahwa air berada dalam wujud beku.
 - Jika suhu antara 0 sampai 100 derajat Celsius ($0 \leq T \leq 100$), program akan mencetak bahwa air berada dalam wujud cair.
 - Jika suhu lebih dari 100 derajat Celsius ($T > 100$), program akan mencetak bahwa air berada dalam wujud uap atau gas.
3. **Output:** Setelah melakukan pengecekan, program akan mencetak hasil wujud air berdasarkan suhu yang diinputkan oleh pengguna.