

LAPORAN TUGAS 2
PEMROGRAMAN BERORIENTASI OBJEK



NURSYAMSI

B4

13020220273

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS MUSLIM INDONESIA

MAKASSAR

2025

1. BacaString

Program ini bertujuan untuk membaca input string dari pengguna. Dengan menggunakan `BufferedReader`, program meminta pengguna untuk memasukkan sebuah string. Setelah input diterima, program akan mencetak string tersebut ke layar. Ini adalah contoh dasar penggunaan input dan output di Java.

```
D:\Tugas PBO\Tugas 2>java BacaString

Baca string dan Integer:
masukkan sebuah string: Mahasiswa
String yang dibaca : Mahasiswa
```

2. ForEver

Program ini mencetak "Print satu baris" secara terus-menerus tanpa henti dalam sebuah loop `while (true)`. Loop ini tidak memiliki kondisi berhenti, sehingga program akan terus berjalan hingga dihentikan secara paksa (misalnya dengan menekan `Ctrl+C` pada terminal). Ini menunjukkan contoh penggunaan loop yang tidak terbatas.

```
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
```

3. If1

Dalam program ini, pengguna diminta untuk memasukkan sebuah nilai integer. Program kemudian memeriksa apakah nilai tersebut positif. Jika nilai yang dimasukkan lebih besar atau sama dengan nol, maka program akan mencetak nilai tersebut. Ini adalah contoh penggunaan pernyataan `if` untuk melakukan kondisi sederhana.

```
D:\Tugas PBO\Tugas 2>javac If1.java  
  
D:\Tugas PBO\Tugas 2>java If1  
Contoh IF satu kasus  
Ketikkan suatu nilai integer : 2
```

4.If2

Program ini lebih kompleks karena menggunakan pernyataan if-else untuk memeriksa nilai integer yang dimasukkan oleh pengguna. Jika nilai tersebut lebih besar atau sama dengan nol, program mencetak bahwa nilai tersebut positif; jika tidak, program mencetak bahwa nilai tersebut negatif. Ini menunjukkan penggunaan pengkondisian ganda dalam Java.

```
D:\Tugas PBO\Tugas 2>javac If2.java  
  
D:\Tugas PBO\Tugas 2>java If2  
Contoh IF dua kasus  
Ketikkan suatu nilai integer :5  
Nilai a positif 5
```

5.If3

Program ini meminta pengguna untuk memasukkan sebuah nilai integer dan menggunakan pernyataan if, else if, dan else untuk menentukan kategori nilai tersebut: positif, nol, atau negatif. Ini memberikan contoh penggunaan beberapa kondisi dalam satu blok kode yang sama.

```
D:\Tugas PBO\Tugas 2>javac If3.java  
  
D:\Tugas PBO\Tugas 2>java If3  
Contoh IF tiga kasus  
Ketikkan suatu nilai integer :12  
Nilai a positif 12
```

6.KasusBoolean

Program ini menunjukkan dasar penggunaan tipe data boolean. Variabel boolean diatur ke true, dan program mencetak "true" jika kondisi tersebut benar. Jika kondisi dibalik dengan !bool, program akan mencetak "benar" karena nilai

boolean awalnya adalah true. Ini adalah contoh dasar untuk memahami logika boolean.

```
D:\Tugas PBO\Tugas 2>javac KasusBoolean.java

D:\Tugas PBO\Tugas 2>java KasusBoolean
true
benar
```

7.KasusSwitch

Program ini meminta pengguna untuk memasukkan sebuah karakter dan menggunakan pernyataan switch untuk mencetak pesan sesuai dengan huruf yang dimasukkan. Jika huruf yang dimasukkan adalah vokal ('a', 'e', 'i', 'o', 'u'), program mencetak huruf yang dimasukkan; jika tidak, program mencetak bahwa karakter tersebut adalah huruf mati. Ini menunjukkan cara alternatif untuk menangani banyak kondisi.

```
D:\Tugas PBO\Tugas 2>javac KasusSwitch.java

D:\Tugas PBO\Tugas 2>java KasusSwitch
Ketikkan sebuah huruf, akhiri dengan RETURN:
s
Yang anda ketik adalah huruf mati

D:\Tugas PBO\Tugas 2>java KasusSwitch
Ketikkan sebuah huruf, akhiri dengan RETURN:
i
Yang anda ketik adalah i
```

8.Konstant

Program ini menghitung luas lingkaran berdasarkan jari-jari yang dimasukkan pengguna. Dengan menggunakan konstanta PHI (3.1415), program mengambil input jari-jari dari pengguna dan menghitung luas lingkaran dengan rumus πr^2 . Ini adalah contoh penggunaan konstanta dalam program Java.

```
D:\Tugas PBO\Tugas 2>javac Konstant.java

D:\Tugas PBO\Tugas 2>java Konstant
Jari-jari lingkaran =1
Luas lingkaran = 3.1415
Akhir program
```

9.Max2

Dalam program ini, pengguna diminta untuk memasukkan dua bilangan. Program kemudian membandingkan kedua bilangan tersebut untuk menentukan mana yang lebih besar dan mencetak nilai maksimum. Ini adalah contoh implementasi logika perbandingan sederhana di Java.

```
D:\Tugas PBO\Tugas 2>javac Max2.java

D:\Tugas PBO\Tugas 2>java Max2
Maksimum dua bilangan:
Ketikkan dua bilangan, pisahkan dengan RETURN:
68
12
Kedua bilangan: a = 68, b = 12
Nilai a yang maksimum: 68
```

10.PriFor

Program ini meminta pengguna untuk memasukkan nilai N, lalu mencetak angka dari 1 hingga N menggunakan loop for. Setiap angka dicetak dalam baris baru. Program ini menunjukkan penggunaan loop dengan batas tertentu.

```
D:\Tugas PBO\Tugas 2>javac PriFor.java

D:\Tugas PBO\Tugas 2>java PriFor
Baca N, print 1 s/d N N = n
Exception in thread "main" java.util.InputMismatchException
    at java.base/java.util.Scanner.throwFor(Scanner.java:964)
    at java.base/java.util.Scanner.next(Scanner.java:1619)
    at java.base/java.util.Scanner.nextInt(Scanner.java:2284)
    at java.base/java.util.Scanner.nextInt(Scanner.java:2238)
    at PriFor.main(PriFor.java:21)
```

11.PrintIterasi

Program ini mencetak angka dari 1 hingga N menggunakan loop tanpa batas (for (;;)), yang berakhir ketika kondisi tertentu terpenuhi. Dalam hal ini, program mencetak angka dan menghentikan loop saat mencapai N. Ini memberikan contoh penggunaan loop yang tidak biasa.

```
D:\Tugas PBO\Tugas 2>javac PrintIterasi.java

D:\Tugas PBO\Tugas 2>java PrintIterasi
Nilai N >0 = 2
Print i dengan ITERATE :
1
2
```

12.PrintRepeat

Dalam program ini, pengguna diminta untuk memasukkan nilai N, dan program mencetak angka dari 1 hingga N menggunakan loop do-while. Loop ini memastikan bahwa proses pencetakan dilakukan setidaknya sekali, sebelum memeriksa kondisi untuk melanjutkan.

```
D:\Tugas PBO\Tugas 2>javac PrintRepeat.java

D:\Tugas PBO\Tugas 2>java PrintRepeat
Nilai N >0 = 2
Print i dengan REPEAT:
1
2
```

13.PrintWhile

Program ini mencetak angka dari 1 hingga N menggunakan loop while. Program ini menginisialisasi i dengan 1 dan terus mencetak angka hingga i lebih besar dari N. Ini adalah contoh penggunaan loop yang lebih umum.

```
D:\Tugas PBO\Tugas 2>javac PrintWhile.java

D:\Tugas PBO\Tugas 2>java PrintWhile
Nilai N >0 = 0
Print i dengan WHILE:
```

14.PrintWhile1

Program ini mirip dengan program sebelumnya, tetapi lebih ringkas. Program ini langsung mencetak angka di dalam kondisi while, menggunakan `i++` untuk meningkatkan nilai `i` setelah setiap iterasi. Ini menunjukkan cara penulisan yang lebih efisien.

```
D:\Tugas PBO\Tugas 2>java PrintWhile1
Nilai N >0 = 8
Print i dengan WHILE (ringkas):
1
2
3
4
5
6
7
8
```

15.PrintXinterasi

Program ini meminta pengguna untuk memasukkan nilai integer dan menjumlahkan semua nilai yang dimasukkan hingga pengguna memasukkan 999, yang berfungsi sebagai sinyal untuk berhenti. Ini menunjukkan penggunaan loop untuk pengulangan berdasarkan input pengguna.

```
D:\Tugas PBO\Tugas 2>javac PrintXinterasi.java

D:\Tugas PBO\Tugas 2>java PrintXinterasi
Masukkan nilai x (int), akhiri dengan 999: 65
Masukkan nilai x (int), akhiri dengan 999: 24
Masukkan nilai x (int), akhiri dengan 999: 12
Masukkan nilai x (int), akhiri dengan 999: 999
Hasil penjumlahan = 101
```

16.PrintXRepeat

Program ini mirip dengan program sebelumnya tetapi menggunakan loop do-while. Pengguna memasukkan nilai hingga 999, dan setiap nilai yang dimasukkan dijumlahkan. Program ini memberikan contoh penggunaan loop yang memastikan setidaknya satu iterasi dilakukan.

```
D:\Tugas PBO\Tugas 2>javac PrintXRepeat.java

D:\Tugas PBO\Tugas 2>java PrintXRepeat
Masukkan nilai x (int), akhiri dg 999 : 89
Masukkan nilai x (int), akhiri dg 999 : 56
Masukkan nilai x (int), akhiri dg 999 : 79
Masukkan nilai x (int), akhiri dg 999 : 99
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 323
```

17.PrintXWhile

Program ini juga menjumlahkan nilai yang dimasukkan hingga 999, tetapi menggunakan loop while. Ini menunjukkan cara lain untuk melakukan pengulangan berdasarkan input, dengan logika serupa pada program sebelumnya.

```
D:\Tugas PBO\Tugas 2>javac PrintXWhile.java

D:\Tugas PBO\Tugas 2>java PrintXWhile
Masukkan nilai x (int), akhiri dengan 999: 23
Masukkan nilai x (int), akhiri dengan 999: 24
Masukkan nilai x (int), akhiri dengan 999: 999
Hasil penjumlahan = 47
```

18.SubProgram

Dalam program ini, terdapat definisi fungsi untuk mencari maksimum dari dua bilangan bulat dan prosedur untuk menukar nilai dua bilangan. Program utama meminta pengguna memasukkan dua bilangan, menampilkan maksimum menggunakan fungsi, dan menukar nilai menggunakan prosedur. Ini memberikan pemahaman tentang penggunaan fungsi dan prosedur dalam Java.


```

D:\Tugas PBO\Tugas 2>javac SubProgram.java

D:\Tugas PBO\Tugas 2>java SubProgram
Maksimum dua bilangan
Ketikkan dua bilangan, pisahkan dengan RETURN:
2
6
Kedua bilangan: a = 2, b = 6
Maksimum = 6
Tukar kedua bilangan...
Kedua bilangan setelah tukar: a = 6, b = 2

```

19.Tempair

Program ini meminta pengguna untuk memasukkan temperatur dalam derajat Celsius dan menentukan wujud air berdasarkan nilai temperatur tersebut. Jika temperatur di bawah 0, air beku; antara 0 dan 100, cair; dan di atas 100, menjadi uap. Ini menunjukkan penggunaan pengkondisian untuk menentukan kategorisasi berdasarkan input yang diberikan.

```

D:\Tugas PBO\Tugas 2>javac Tempair.java

D:\Tugas PBO\Tugas 2>java Tempair
Contoh IF tiga kasus
Temperatur (der. C) = Wujud air beku
Exception in thread "main" java.util.InputMismatchException
    at java.base/java.util.Scanner.throwFor(Scanner.java:964)
    at java.base/java.util.Scanner.next(Scanner.java:1619)
    at java.base/java.util.Scanner.nextInt(Scanner.java:2284)
    at java.base/java.util.Scanner.nextInt(Scanner.java:2238)
    at Tempair.main(Tempair.java:18)

```