

**PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK**



Nama : Virkayanti Andani Putri
Stambuk : 13020220298
Dosen : Mardiyah Hasnawi, S.Kom.,M.T.,MTA.
Asisten 1 :
Asisten 2 :

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MUSLIM INDONESIA
MAKASSAR**

2025

1. BacaString.java

➤ Penjelasan Kode Program

- Import Library → Untuk membaca input (BufferedReader, InputStreamReader).
- Deklarasi Variabel → String str; untuk menyimpan input.
- Membaca Input → datAIn.readLine(); membaca string yang diketik pengguna.
- Menampilkan Output → System.out.print("String yang dibaca : " + str); menampilkan hasil input.

➤ Penjelasan Gambar Output

- javac BacaString.java → Mengompilasi kode program.
- java BacaString → Menjalankan program.
- Masukkan "Virkayanti Andani Putri" sebagai input.
- Output: String yang dibaca : Virkayanti Andani Putri ditampilkan sesuai input pengguna.
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac BacaString.java
PS D:\Tugas 2\Source Code> java BacaString

Baca string dan Integer:
masukkan sebuah string: Virkayanti Andani Putri
String yang dibaca : Virkayanti Andani Putri
PS D:\Tugas 2\Source Code> |
```

2. ForEver.java

➤ Penjelasan Kode Program

Program ini menjalankan perulangan tanpa henti (infinite loop) menggunakan while (true).

- Menampilkan Pesan Awal → "Program akan looping, akhiri dengan ^c".
- Loop Tak Terbatas → while (true) membuat program terus mencetak "Print satu baris" selamanya.
- Penghentian Program → Harus dihentikan secara manual dengan Ctrl + C.

➤ Penjelasan Gambar Output

- javac ForEver.java → Mengompilasi program.
- java ForEver → Menjalankan program.
- Pesan awal muncul: "Program akan looping, akhiri dengan ^c".
- Program terus mencetak "Print satu baris" tanpa henti.
- Pengguna harus menekan Ctrl + C untuk menghentikan eksekusi.
- Output akhirnya ditampilkan sebagai:

```

PS D:\Tugas 2\Source Code> javac ForEver.java
PS D:\Tugas 2\Source Code> java ForEver
Program akan looping, akhiri dengan ^c
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 ....
Print satu baris ....
Print satu baris ....
Print satu baris ....

```

3. If1.java

➤ Penjelasan Kode Program

Program ini menggunakan struktur IF satu kasus untuk mengecek apakah angka yang dimasukkan pengguna bernilai positif.

- Menggunakan Scanner → `masukan.nextInt();` untuk membaca input dari pengguna.
- Memeriksa Bilangan Positif → `if (a >= 0)`, jika angka ≥ 0 , maka program mencetak "Nilai a positif" diikuti dengan nilai tersebut.
- Tidak Ada Else → Jika input negatif, program tidak menampilkan output tambahan.

➤ Penjelasan Gambar Output

- `javac If1.java` → Mengompilasi program.
- `java If1` → Menjalankan program.
- Program meminta input integer → Pengguna memasukkan 10.
- Karena $10 \geq 0$, program mencetak "Nilai a positif 10".
- Output akhirnya ditampilkan sebagai:

```

PS D:\Tugas 2\Source Code> javac If1.java
PS D:\Tugas 2\Source Code> java If1
Contoh IF satu kasus
Ketikkan suatu nilai integer : 10

Nilai a positif 10

```

4. If2.java

➤ Penjelasan Kode Program

Program ini menggunakan struktur IF dua kasus untuk mengecek apakah angka yang dimasukkan pengguna positif atau negatif.

1. Menggunakan Scanner → `masukan.nextInt();` untuk membaca input pengguna.
2. Memeriksa Nilai:
 - Jika $a \geq 0$, mencetak "Nilai a positif [a]".
 - Jika $a < 0$, mencetak "Nilai a negatif [a]".

➤ Penjelasan Gambar Output

- `javac If2.java` → Mengompilasi program.
- `java If2` → Menjalankan program.
- Program meminta input integer → Pengguna memasukkan 10.
- Karena $10 \geq 0$, program mencetak "Nilai a positif 10".
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac If2.java
PS D:\Tugas 2\Source Code> java If2
Contoh IF dua kasus
Ketikkan suatu nilai integer :10
Nilai a positif 10
```

5. If3.java

➤ Penjelasan Kode Program

Program ini menggunakan struktur IF tiga kasus untuk menentukan apakah angka yang dimasukkan pengguna positif, nol, atau negatif.

1. Menggunakan Scanner → `masukan.nextInt();` untuk membaca input pengguna.
2. Memeriksa Nilai:
 - Jika $a > 0$, mencetak "Nilai a positif [a]".
 - Jika $a == 0$, mencetak "Nilai Nol [a]".
 - Jika $a < 0$, mencetak "Nilai a negatif [a]".

➤ Penjelasan Gambar Output

- `javac If3.java` → Mengompilasi program.
- `java If3` → Menjalankan program.
- Program meminta input integer → Pengguna memasukkan 10.
- Karena $10 > 0$, program mencetak "Nilai a positif 10".
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac If3.java
PS D:\Tugas 2\Source Code> java If3
Contoh IF tiga kasus
Ketikkan suatu nilai integer :10
Nilai a positif 10
```

6. KasusBoolean.java

➤ Penjelasan Kode Program

Program ini menggunakan boolean untuk mengevaluasi kondisi true atau false.

1. Variabel bool diatur ke true.
2. Struktur IF mengecek bool:
 - Jika bool == true, cetak "true".
 - Jika bool == false, cetak "false" (tidak akan terjadi).
3. Struktur IF mengecek !bool (negasi):
 - Jika !bool (false), cetak "salah".
 - Jika !bool == false, cetak "benar".

➤ Penjelasan Gambar Output

- javac KasusBoolean.java → Mengompilasi program.
- java KasusBoolean → Menjalankan program.
- Karena bool = true, maka mencetak seperti gambar dibawah ini
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac KasusBoolean.java
PS D:\Tugas 2\Source Code> java KasusBoolean
true
benar
```

7. KasusSwitch.java

➤ Penjelasan Kode Program

Program ini membaca satu huruf dari input pengguna dan mencetak hasil berdasarkan huruf yang diketik menggunakan switch-case.

1. Pengguna memasukkan satu karakter (char cc).
2. Switch-case mengecek karakter:
 - Jika 'a', 'u', 'e', 'i', atau 'o', cetak bahwa huruf tersebut diketik.
 - Jika bukan vokal, masuk ke default, mencetak "Yang anda ketik adalah huruf mati".

➤ Penjelasan Gambar Output

- javac KasusSwitch.java → Mengompilasi program.
- java KasusSwitch → Menjalankan program.
- Pengguna mengetik 'a', maka outputnya pada gambar dibawah ini
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac KasusSwitch.java
PS D:\Tugas 2\Source Code> java KasusSwitch
Ketikkan sebuah huruf, akhiri dengan RETURN
a
Yang anda ketik adalah a
```

8. Konstant.java

➤ Penjelasan Kode Program

Program ini menghitung luas lingkaran dengan menggunakan konstanta PHI.

- `final float PHI = 3.1415f;` → Mendefinisikan konstanta PHI.
- Mengambil input jari-jari (r) dari pengguna.
- Menghitung luas lingkaran dengan rumus: $\text{Luas} = \pi \times r \times r$
- Menampilkan hasil perhitungan.

➤ Penjelasan Gambar Output

- `javac Konstant.java` → Mengompilasi program.
- `java Konstant` → Menjalankan program.
- Pengguna memasukkan `r = 2`, maka perhitungan: $3.1415 \times 2 \times 2 = 12.5663.1415$
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac Konstant.java
PS D:\Tugas 2\Source Code> java Konstant
Jari-jari lingkaran =2
Luas lingkaran = 12.566
Akhir program
```

9. Max2.java

➤ Penjelasan Kode Program

Program ini digunakan untuk menentukan bilangan terbesar dari dua angka yang dimasukkan pengguna.

1. Menggunakan Scanner untuk menerima input dua bilangan integer (a dan b).
2. Menampilkan kembali nilai yang dimasukkan pengguna.
3. Membandingkan a dan b:
 - Jika $a \geq b$, maka a adalah maksimum.
 - Jika $b > a$, maka b adalah maksimum.
4. Menampilkan hasil perbandingan.

➤ Penjelasan Gambar Output

- `javac Max2.java` → Mengompilasi program.
- `java Max2` → Menjalankan program.
- Pengguna memasukkan `a = 2` dan `b = 9`, maka:
 - Perbandingan a dan b → $9 > 2$, sehingga b lebih besar.
- Output akhirnya ditampilkan sebagai:

```

PS D:\Tugas 2\Source Code> javac Max2.java
PS D:\Tugas 2\Source Code> java Max2
Maksimum dua bilangan :
Ketikkan dua bilangan, pisahkan dg RETURN :
2
9
Ke dua bilangan : a = 2 b = 9
Nilai b yang maksimum: 9

```

10. PriFor.java

➤ Penjelasan Kode Program

Program ini digunakan untuk mencetak angka dari 1 hingga N menggunakan perulangan for.

1. Menggunakan Scanner untuk menerima input N dari pengguna.
2. Menggunakan perulangan for dengan:
 - $i = 1$ sebagai nilai awal.
 - $i \leq N$ sebagai kondisi perulangan.
 - $i++$ untuk menambah nilai i setiap iterasi.
3. Mencetak angka dari 1 hingga N satu per satu.
4. Menampilkan "Akhir program" setelah selesai.

➤ Penjelasan Gambar Output

- `javac PriFor.java` → Mengompilasi program.
- `java PriFor` → Menjalankan program.
- Pengguna memasukkan $N = 5$,
 - Angka 1 hingga 5 dicetak karena $N = 5$.
 - Setelah selesai, program mencetak "Akhir program".
- Output akhirnya ditampilkan sebagai:

```

PS D:\Tugas 2\Source Code> javac PriFor.java
PS D:\Tugas 2\Source Code> java PriFor
Baca N, print 1 s/d N N = 5
1
2
3
4
5
Akhir program

```

11. PrintIterasi.java

➤ Penjelasan Kode Program

Program ini digunakan untuk mencetak angka dari 1 hingga N menggunakan perulangan tak terbatas (`for(;;)`) dengan kondisi `break` untuk berhenti.

Langkah-langkah program:

1. Menggunakan Scanner untuk menerima input N dari pengguna.
2. Inisialisasi $i = 1$ sebagai nilai awal.
3. Perulangan `for(;;)` digunakan sebagai loop tak terbatas, yang hanya berhenti jika $i == N$.
4. Mencetak angka i setiap iterasi.
5. Jika $i == N$, maka `break`; keluar dari loop.
6. Jika belum mencapai N, $i++$ untuk menambah nilai i dan mengulangi proses.

➤ Penjelasan Gambar Output

- `javac PrintIterasi.java` → Mengompilasi program.
- `java PrintIterasi` → Menjalankan program.
- Pengguna memasukkan $N = 5$,
 - Angka 1 hingga 5 dicetak sesuai input $N = 5$.
 - Perulangan berhenti setelah $i == N$ karena `break`;
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac PrintIterasi.java
PS D:\Tugas 2\Source Code> java PrintIterasi
Nilai N >0 = 5
Print i dengan ITERATE :
1
2
3
4
5
```

12. PrintRepeat.java

➤ Penjelasan Kode Program

Program ini digunakan untuk mencetak angka dari 1 hingga N menggunakan perulangan `do-while`.

Langkah-langkah program:

1. Menggunakan Scanner untuk menerima input N dari pengguna.
2. Inisialisasi $i = 1$ sebagai nilai awal.
3. Perulangan `do-while` digunakan untuk memastikan setidaknya satu kali eksekusi sebelum mengecek kondisi.
4. Di dalam `do`, program mencetak i , lalu $i++$ untuk menambah nilai i .
5. Loop berhenti ketika $i > N$.

➤ Penjelasan Gambar Output

- `javac PrintRepeat.java` → Mengompilasi program.
- `java PrintRepeat` → Menjalankan program.
- Pengguna memasukkan $N = 6$,
 - Angka 1 hingga 6 dicetak karena perulangan `do-while` terus berjalan sampai $i > N$.
 - Perbedaan utama dengan `while` atau `for` adalah bahwa `do-while` selalu dieksekusi setidaknya sekali.

- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac PrintRepeat.java
PS D:\Tugas 2\Source Code> java PrintRepeat
Nilai N >0 = 6
Print i dengan REPEAT:
1
2
3
4
5
6
```

13. PrintWhile.java

➤ Penjelasan Kode Program

Program ini digunakan untuk mencetak angka dari 1 hingga N menggunakan perulangan while.

Langkah-langkah program:

1. Menggunakan Scanner untuk menerima input N dari pengguna.
2. Inisialisasi i = 1 sebagai nilai awal.
3. Perulangan while dijalankan selama i <= N.
4. Di dalam while, program mencetak i, lalu i++ untuk menambah nilai i.
5. Loop berhenti ketika i > N.

➤ Penjelasan Gambar Output

- javac PrintWhile.java → Mengompilasi program.
- java PrintWhile → Menjalankan program.
- Pengguna memasukkan N = 7,
 - Angka 1 hingga 7 dicetak karena perulangan while berjalan selama i <= N.
 - Berbeda dengan do-while, while akan langsung mengecek kondisi sebelum menjalankan perulangan.

- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac PrintWhile.java
PS D:\Tugas 2\Source Code> java PrintWhile
Nilai N >0 = 7
Print i dengan WHILE:
1
2
3
4
5
6
7
```

14. PrintWhile1.java

➤ **Penjelasan Kode Program**

Program ini digunakan untuk mencetak angka dari 1 hingga N menggunakan perulangan while, namun dengan versi yang lebih ringkas.

Perbedaan dengan versi sebelumnya:

1. Inisialisasi $i = 1$ dilakukan langsung saat deklarasi.
2. Dalam while, nilai i langsung bertambah dengan $i++$ dalam `System.out.println(i++)`.
 - Ini lebih ringkas dibandingkan versi sebelumnya yang menambah i dalam baris terpisah.

➤ **Penjelasan Gambar Output**

- `javac PrintWhile1.java` → Mengompilasi program.
- `java PrintWhile1` → Menjalankan program.
- Pengguna memasukkan $N = 8$,
 - Perulangan while ($i \leq N$) mencetak nilai i sebelum menambahnya ($i++$).
- Output akhirnya ditampilkan sebagai:

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

15. PrintXinterasi.java

➤ **Penjelasan Kode Program**

Program ini membaca beberapa bilangan integer dari pengguna dan menjumlahkan semua bilangan yang dimasukkan, hingga pengguna memasukkan 999 sebagai tanda berhenti.

Cara Kerja Program

1. Inisialisasi `Sum = 0` sebagai tempat menyimpan hasil penjumlahan.
2. Meminta input pertama (x) dari pengguna.
3. Jika pengguna langsung memasukkan 999, program mencetak "Kasus kosong" dan selesai.
4. Jika input pertama bukan 999, program:
 - Menyimpan nilai pertama ke dalam `Sum`.
 - Memulai perulangan tak terbatas (`for(;;)`) untuk membaca nilai berikutnya.
 - Jika input adalah 999, perulangan berhenti (`break`).

- Jika bukan, nilai x ditambahkan ke Sum.
- 5. Setelah perulangan selesai, program mencetak hasil penjumlahan Sum.

➤ Penjelasan Gambar Output

Perhitungan

$$10 + 20 + 25 = 55$$

- Nilai nilai yang dimasukkan sampai pengguna memasukkan nilai 999 (sebagai tanda berhenti) itu nilainya akan dijumlahkan
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac PrintXinterasi.java
PS D:\Tugas 2\Source Code> java PrintXinterasi
Masukkan nilai x (int), akhiri dg 999: 10
Masukkan nilai x (int),akhiri dg 999 : 20
Masukkan nilai x (int),akhiri dg 999 : 25
Masukkan nilai x (int),akhiri dg 999 : 999
Hasil penjumlahan = 55
```

16. PrintXRepeat.java

➤ Penjelasan Kode Program

Program ini membaca beberapa bilangan integer dari pengguna dan menjumlahkan semua bilangan yang dimasukkan, hingga pengguna memasukkan 999 sebagai tanda berhenti.

Kode ini menggunakan perulangan do-while, sehingga program akan menjalankan setidaknya satu iterasi sebelum mengecek kondisi penghentian.

Cara Kerja Program

1. Meminta input pertama (x) dari pengguna.
2. Jika pengguna langsung memasukkan 999, program mencetak "Kasus kosong" dan berhenti.
3. Jika input pertama bukan 999, program:
 - Menginisialisasi Sum = 0.
 - Memulai perulangan do-while untuk:
 - Menambahkan nilai x ke dalam Sum.
 - Meminta input berikutnya (x).
 - Jika $x \neq 999$, pengulangan berlanjut.
4. Setelah perulangan selesai, program mencetak hasil penjumlahan Sum.

➤ Penjelasan Gambar Output

Perhitungan

$$10 + 20 + 30 = 60$$

- Nilai nilai yang dimasukkan sampai pengguna memasukkan nilai 999 (sebagai tanda berhenti) itu nilainya akan dijumlahkan.

- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac PrintXRepeat.java
PS D:\Tugas 2\Source Code> java PrintXRepeat
Masukkan nilai x (int), akhiri dg 999 : 10
Masukkan nilai x (int), akhiri dg 999 : 20
Masukkan nilai x (int), akhiri dg 999 : 30
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 60
```

17. PrintXWhile.java

➤ Penjelasan Kode Program

Program ini membaca beberapa bilangan integer dari pengguna dan menjumlahkan semua bilangan yang dimasukkan hingga pengguna memasukkan 999 sebagai tanda berhenti.

Kode ini menggunakan perulangan while, yang berarti program akan memeriksa kondisi terlebih dahulu sebelum menjalankan perulangan.

Cara Kerja Program

1. Menginisialisasi Sum = 0 untuk menyimpan hasil penjumlahan.
2. Meminta input pertama (x) dari pengguna.
3. Memeriksa kondisi while (x != 999):
 - Jika x != 999, program:
 - Menambahkan nilai x ke dalam Sum.
 - Meminta input berikutnya (x).
 - Jika x == 999, perulangan berhenti.
4. Setelah perulangan selesai, program mencetak hasil penjumlahan Sum.

➤ Penjelasan Gambar Output

Perhitungan

$$10 + 20 + 35 = 65$$

- Nilai nilai yang dimasukkan sampai pengguna memasukkan nilai 999 (sebagai tanda berhenti) itu nilainya akan dijumlahkan.
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac PrintXWhile.java
PS D:\Tugas 2\Source Code> java PrintXWhile
Masukkan nilai x (int), akhiri dg 999 : 10
Masukkan nilai x (int), akhiri dg 999 : 20
Masukkan nilai x (int), akhiri dg 999 : 35
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 65
```

18. SubProgram.java

➤ Penjelasan Kode Program

Program ini menunjukkan penggunaan fungsi dan prosedur dalam Java untuk Menentukan bilangan terbesar dari dua bilangan yang dimasukkan dan Menukar nilai dua bilangan dengan prosedur.

Fungsi dan Prosedur dalam Program

1. Fungsi maxab(int a, int b)
 - Mengembalikan nilai terbesar dari dua bilangan yang diberikan.
 - Menggunakan operator ternary untuk memilih nilai maksimum.
2. Prosedur tukar(int a, int b)
 - Menukar nilai a dan b menggunakan variabel sementara (temp).
 - Namun, prosedur ini tidak benar-benar menukar nilai di main(), karena Java menggunakan pass-by-value untuk tipe data primitif.

Cara Kerja Program

1. Meminta memasukkan dua bilangan integer.
 2. Menampilkan kedua bilangan sebelum diproses.
 3. Menggunakan fungsi maxab() untuk mencari bilangan terbesar dan mencetaknya.
 4. Memanggil prosedur tukar(), tetapi nilai di main() tidak berubah karena pass-by-value.
- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac SubProgram.java
PS D:\Tugas 2\Source Code> java SubProgram
Maksimum dua bilangan
Ketikkan dua bilangan, pisahkan dg RETURN :
10
20
Ke dua bilangan : a = 10 b = 20
Maksimum = 20
Tukar kedua bilangan...
Ke dua bilangan setelah tukar: a = 20 b = 10
```

19. Tempair.java

➤ Penjelasan Kode Program

Program ini menggunakan struktur kendali if-else untuk menentukan wujud air berdasarkan suhu dalam derajat Celcius.

Logika Program

1. Pengguna memasukkan suhu T (dalam derajat Celcius).
2. Program menentukan wujud air berdasarkan aturan berikut:
 - Jika $T < 0$, air dalam wujud beku.
 - Jika $0 \leq T \leq 100$, air dalam wujud cair.
 - Jika $T > 100$, air dalam wujud uap/gas.

- Output akhirnya ditampilkan sebagai:

```
PS D:\Tugas 2\Source Code> javac Tempair.java
PS D:\Tugas 2\Source Code> java Tempair
Contoh IF tiga kasus
Temperatur (der. C) = -5
Wujud air beku
-5
PS D:\Tugas 2\Source Code> java Tempair
Contoh IF tiga kasus
Temperatur (der. C) = 20
Wujud air cair
20
PS D:\Tugas 2\Source Code> java Tempair
Contoh IF tiga kasus
Temperatur (der. C) = 105
Wujud air uap/gas
105
PS D:\Tugas 2\Source Code>
```