

KAMIS 23 OKTOBER 2025

LAPORAN HASIL PRAKTIKUM

Jobsheet 7



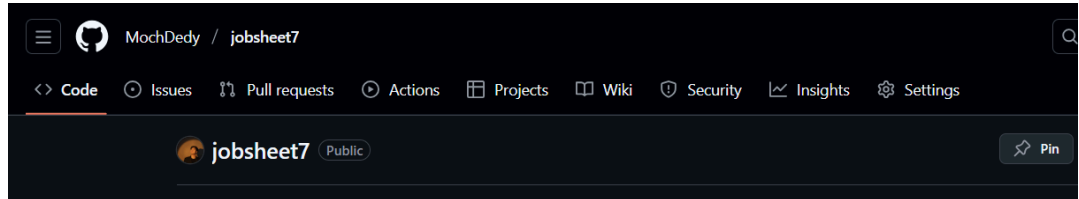
Disusun oleh:
Moch Dedy Triagwi
Kelas 1H/IT
254107020233

**PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2025**

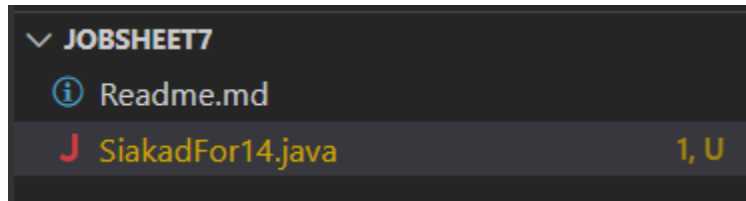
1. Praktikum

1.1.1 Percobaan 1: Studi Kasus Nilai Mahasiswa di SIAKAD – Perulangan FOR

1. Buat folder baru pada repositori lokal Anda, beri nama **jobsheet7**



2. Buat file baru, beri nama **SiakadForNoPresensi.java**



3. Buatlah struktur dasar program Java yang terdiri dari fungsi **main()**.
4. Tambahkan library Scanner di bagian atas (luar) class
5. Buat deklarasi Scanner dengan nama variabel **sc** di dalam fungsi **main()**
6. Deklarasikan variabel **nilai**, **tertinggi**, dan **terendah** bertipe double. Inisialisasi **tertinggi** dengan 0 dan **terendah** dengan 100

```
double nilai, tertinggi = 0, terendah = 100;
```

7. Buat struktur perulangan FOR dengan batas kondisi sesuai jumlah mahasiswa yaitu 10

```
for (int i = 1; i <= 10; i++) {
```

8. Di dalam perulangan FOR tersebut, tambahkan perintah untuk memasukkan nilai mahasiswa. Setelah itu, buat dua kondisi pemilihan secara terpisah untuk mengecek nilai tertinggi dan terendah dengan membandingkan **nilai** masukan dengan variabel

tertinggi dan variabel terendah

```
for (int i = 1; i <= 10; i++) {  
    System.out.print("Masukkan nilai mahasiswa ke-" + i + ": ");  
    nilai = sc.nextDouble();  
    if (nilai > tertinggi) {  
        tertinggi = nilai;  
    }  
    if (nilai < terendah) {  
        terendah = nilai;  
    }  
}
```

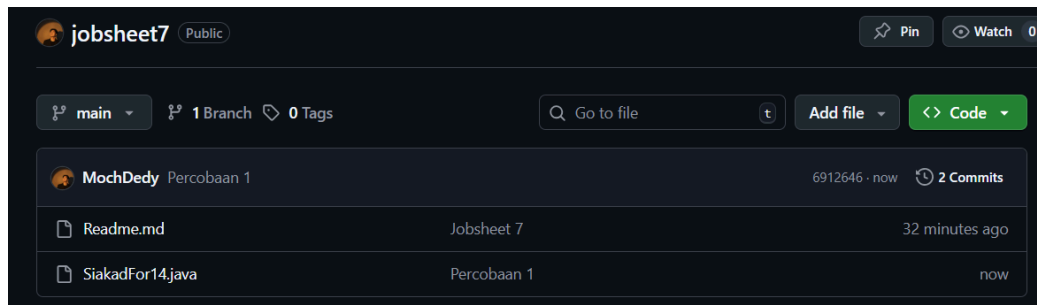
9. Di luar perulangan FOR, tampilkan nilai tertinggi dan terendah

```
System.out.println("Nilai tertinggi: " + tertinggi);  
System.out.println("Nilai terendah: " + terendah);
```

10. Compile dan run program

```
J SiakadFor14.java > ...  
1  import java.util.Scanner;  
2  
3  public class SiakadFor14 {  
    Run | Debug | Run main | Debug main  
4      public static void main(String[] args) {  
5          Scanner sc = new Scanner(System.in);  
6  
7          double nilai, tertinggi = 0, terendah = 100;  
8  
9          for (int i = 1; i <= 10; i++) {  
10             System.out.print("Masukkan nilai mahasiswa ke-" + i + ": ");  
11             nilai = sc.nextDouble();  
12             if (nilai > tertinggi) {  
13                 tertinggi = nilai;  
14             }  
15             if (nilai < terendah) {  
16                 terendah = nilai;  
17             }  
18         }  
19  
20         System.out.println("Nilai tertinggi: " + tertinggi);  
21         System.out.println("Nilai terendah: " + terendah);  
22     }  
23  
24 }
```

11. Commit program Anda ke Github dengan pesan “Percobaan 1”



1.1.2 Verifikasi Hasil Percobaan

Cocokkan hasil compile kode program Anda dengan gambar berikut ini:

```
Masukkan nilai mahasiswa ke-1: 76.5
Masukkan nilai mahasiswa ke-2: 82.3
Masukkan nilai mahasiswa ke-3: 62.1
Masukkan nilai mahasiswa ke-4: 88.4
Masukkan nilai mahasiswa ke-5: 65.9
Masukkan nilai mahasiswa ke-6: 67.9
Masukkan nilai mahasiswa ke-7: 90.1
Masukkan nilai mahasiswa ke-8: 55.3
Masukkan nilai mahasiswa ke-9: 73.7
Masukkan nilai mahasiswa ke-10: 78.6
Nilai tertinggi: 90.1
Nilai terendah: 55.3

C:\Users\Moch Dedy Triagwi\Documents\Kuliah\DEDY
```

1.1.3 Pertanyaan

1. Sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program Percobaan 1!

Jawab:

```
for (int i = 1; i <= 10; i++) {
```

- a. Inisialisasi = int i yaitu menentukan nilai awal variabel yaitu dimulai dari 1

- b. Kondisi: $i \leq 10$ yaitu menentukan syarat agar perulangan terus berjalan yang mana jika else atau program tidak sesuai syarat maka program akan berhenti
 - c. Update: $i++$ yaitu menambah nilai i setiap satu kali perulangan selesai
2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Jawab:

- a. tertinggi = 0 agar nilai yg diinput akan lebih besar dari 0
- b. terendah = 100 agar nilai yg diinput akan lebih kecil dari 100

Apabila dibalik maka

- a. tertinggi = 100 maka nilai akan tetap sama karena tidak ada nilai yang lebih tinggi dari 100
- b. terendah = 0 maka nilai akan tetap sama karena tidak ada nilai yang lebih kecil dari 0

Maka output yang dimunculkan akan tetap yaitu **tertinggi = 100** dan **terendah = 0**

3. Jelaskan fungsi dan alur kerja dari potongan kode berikut!

```
if (nilai > tertinggi) {  
    tertinggi = nilai;  
}  
if (nilai < terendah) {  
    terendah = nilai;  
}
```

Jawab:

- a. Kode ini berfungsi untuk menentukan nilai tertinggi dan terendah dari sekumpulan data nilai mahasiswa yang dimasukkan melalui perulangan.
- b. Alur kerja
Tertinggi: Mengecek apakah nilai mahasiswa saat ini lebih

besar dari nilai tertinggi yang sudah tersimpan sebelumnya. Kalau iya, berarti mahasiswa ini punya nilai yang lebih tinggi, jadi variabel tertinggi diperbarui menjadi nilai mahasiswa itu.

Terendah: Mengecek apakah nilai mahasiswa saat ini lebih kecil dari nilai terendah yang sudah tersimpan. Kalau iya, berarti mahasiswa ini punya nilai yang paling rendah sejauh ini, jadi variabel terendah diperbarui.

4. Modifikasi kode program sehingga terdapat perhitungan untuk menentukan berapa mahasiswa yang lulus dan yang tidak lulus berdasarkan batas kelulusan (nilai minimal 60). Tampilkan jumlah mahasiswa lulus dan tidak lulus setelah menampilkan nilai tertinggi dan terendah!

Jawab:

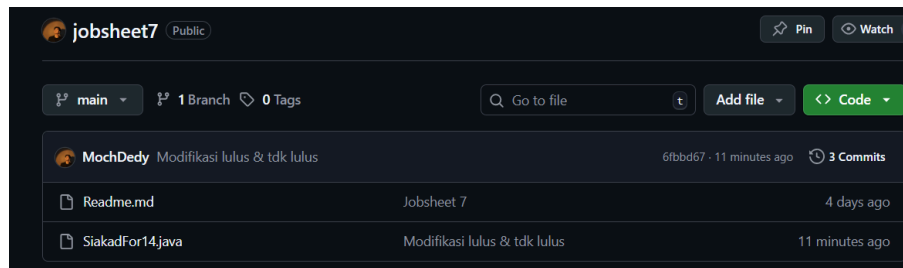
Kode Program

```
J SiakadFor14.java > Java > SiakadFor14 > main(String[] args)
1  import java.util.Scanner;
2
3  public class SiakadFor14 {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6
7          int lulus = 0, tdkLulus = 0;
8          double nilai, tertinggi = 0, terendah = 100;
9
10         for (int i = 1; i <= 10; i++) {
11             System.out.print("Masukkan nilai mahasiswa ke-" + i + ": ");
12             nilai = sc.nextDouble();
13             if (nilai > tertinggi) {
14                 tertinggi = nilai;
15             }
16             if (nilai < terendah) {
17                 terendah = nilai;
18             }
19             if (nilai >= 60) {
20                 lulus++;
21             } else {
22                 tdkLulus++;
23             }
24         }
25
26         System.out.println("Nilai tertinggi: " + tertinggi);
27         System.out.println("Nilai terendah: " + terendah);
28         System.out.println("Banyak mahasiswa yang lulus: " + lulus);
29         System.out.println("Banyak mahasiswa yang tidak lulus: " + tdkLulus);
30     }
31 }
32
33 }
```

Hasil Program

```
Masukkan nilai mahasiswa ke-1: 66
Masukkan nilai mahasiswa ke-2: 92
Masukkan nilai mahasiswa ke-3: 45.5
Masukkan nilai mahasiswa ke-4: 55
Masukkan nilai mahasiswa ke-5: 70
Masukkan nilai mahasiswa ke-6: 42
Masukkan nilai mahasiswa ke-7: 60
Masukkan nilai mahasiswa ke-8: 87
Masukkan nilai mahasiswa ke-9: 99
Masukkan nilai mahasiswa ke-10: 100
Nilai tertinggi: 100.0
Nilai terendah: 42.0
Banyak mahasiswa yang lulus: 7
Banyak mahasiswa yang tidak lulus: 3
```

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 1”



1.2.1 Percobaan 2: Studi Kasus Nilai Mahasiswa di SIAKAD – Perulangan WHILE

1. Buat file baru, beri nama **SiakadWhileNoAbsen.java**

J SiakadWhile14.java 3, U

2. Buatlah struktur dasar program Java yang terdiri dari fungsi **main()**.
3. Tambahkan library Scanner di bagian atas (luar) class
4. Buat deklarasi Scanner dengan nama variabel **sc** di dalam fungsi **main()**
5. Deklarasikan variabel **nilai**, **jml**, dan **i** (untuk perulangan) bertipe integer. Inisialisasi **i** dengan 0 sebagai nilai awal perulangan

6. Tuliskan kode program untuk menerima input banyaknya mahasiswa yang disimpan ke variabel jml. Dengan demikian, batas perulangan akan dinamis sesuai masukan dari pengguna melalui keyboard.

```
int nilai, jml, i = 0;

System.out.print(s:"Masukkan jumlah mahasiswa: ");
jml = input.nextInt();
```

7. Buat struktur perulangan WHILE dengan batas kondisi sesuai jumlah mahasiswa yaitu 5. Perhatikan simbol yang digunakan adalah < karena perulangan variabel i dimulai dari 0, bukan 1

```
while (i < jml) {

    i++;
}
```

8. Di dalam perulangan WHILE tersebut, tambahkan perintah untuk memasukkan nilai mahasiswa. Setelah itu, buat kondisi pemilihan IF untuk mengecek valid atau tidaknya nilai yang dimasukkan, dengan syarat nilai harus berada pada rentang 0 hingga 100. Kemudian tambahkan kondisi pemilihan IF-ELSE IF-ELSE untuk menampilkan kategori nilai huruf berdasarkan ketentuan.

```
while (i < jml) {
    System.out.print("Masukkan nilai mahasiswa ke-" + (i + 1) + ": ");
    nilai = input.nextInt();

    if (nilai < 0 || nilai > 100){
        System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid!");
        continue;
    }

    if (nilai > 80 && nilai <=100){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A");
    } else if (nilai > 73 && nilai <= 80){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B+");
    } else if (nilai > 65 && nilai <= 73){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B");
    } else if (nilai > 60 && nilai <= 65){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C+");
    } else if (nilai > 50 && nilai <= 60){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C");
    } else if (nilai > 39 && nilai <= 50){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah D");
    } else {
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah E");
    }
    i++;
}
```


9. Compile dan run program

```
J SiakadWhile14.java > Java > SiakadWhile14 > main(String[] args)
1  import java.util.Scanner;
2
3  public class SiakadWhile14 {
4      Run | Debug | Run main | Debug main
5      public static void main(String[] args) {
6          Scanner input = new Scanner(System.in);
7
8          int nilai, jml, i = 0;
9
10         System.out.print(s:"Masukkan jumlah mahasiswa: ");
11         jml = input.nextInt();
12
13         while (i < jml) {
14             System.out.print("Masukkan nilai mahasiswa ke-" + (i + 1) + ": ");
15             nilai = input.nextInt();
16
17             if (nilai < 0 || nilai > 100){
18                 System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid!");
19                 continue;
20             }
21
22             if (nilai > 80 && nilai <=100){
23                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + "adalah A");
24             } else if (nilai > 73 && nilai <= 80){
25                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + "adalah B+");
26             } else if (nilai > 65 && nilai <= 73){
27                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + "adalah B");
28             } else if (nilai > 60 && nilai <= 65){
29                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + "adalah C+");
30             } else if (nilai > 50 && nilai <= 60){
31                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + "adalah C");
32             } else if (nilai > 39 && nilai <= 50){
33                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + "adalah D");
34             } else {
35                 System.out.println("Nilai mahasiswa ke-" + (i + 1) + "adalah E");
36             }
37             i++;
38         }
39     }
40 }
```

10. Commit dan push kode program ke Github

 SiakadFor14.java	Modifikasi lulus & tdk lulus	38 minutes ago
 SiakadWhile14.java	1.2.1 While	now

1.2.2 Verifikasi Hasil Percobaan

Cocokkan hasil compile kode program Anda dengan gambar berikut ini.

```
Masukkan jumlah mahasiswa: 5
Masukkan nilai mahasiswa ke-1: 85
Nilai mahasiswa ke-1 adalah A
Masukkan nilai mahasiswa ke-2: 63
Nilai mahasiswa ke-2 adalah C+
Masukkan nilai mahasiswa ke-3: 101
Nilai tidak valid. Masukkan lagi nilai yang valid!
Masukkan nilai mahasiswa ke-3: 23
Nilai mahasiswa ke-3 adalah E
Masukkan nilai mahasiswa ke-4: -15
Nilai tidak valid. Masukkan lagi nilai yang valid!
Masukkan nilai mahasiswa ke-4: 70
Nilai mahasiswa ke-4 adalah B
Masukkan nilai mahasiswa ke-5: 55
Nilai mahasiswa ke-5 adalah C
c:\Users\Moch Dedy Triagwi\Documents\Kuliah\DEDY NITIP\PRAKTEKDASPRO\Jobsheet7\jobsheet7-1>
```

1.2.3 Pertanyaan

1. Pada potongan kode berikut, tentukan maksud dan kegunaan dari sintaks berikut:

```
if (nilai < 0 || nilai > 100){
    System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid!");
    continue;
}
```

a. $\text{nilai} < 0 \parallel \text{nilai} > 100$

Jawab:

Digunakan untuk memeriksa apakah nilai yang dimasukkan berada di luar rentang valid, yaitu kurang dari 0 atau lebih dari 100. Jika kondisi ini bernilai benar, maka program akan menampilkan pesan bahwa nilai tidak valid.

b. `continue`

Jawab:

Perintah `continue` dijalankan untuk melewati sisa perintah dalam iterasi saat ini dan langsung melanjutkan ke iterasi berikutnya pada perulangan. Dengan demikian, program akan meminta pengguna untuk memasukkan kembali nilai yang benar tanpa menghentikan proses perulangan secara keseluruhan.

2. Mengapa sintaks `i++` dituliskan di akhir perulangan WHILE? Apa yang terjadi jika posisinya dituliskan di awal perulangan WHILE?

Jawab:

Sintaks `i++` ditulis di akhir perulangan while agar kenaikan nilai penghitung terjadi setelah satu kali proses perulangan selesai. Jika ditulis di awal, nilai akan bertambah duluan sehingga perulangan bisa berkurang satu atau logikanya menjadi salah.

3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan WHILE akan berjalan?

Jawab:

Jika jumlah mahasiswa yang dimasukkan adalah 19, maka perulangan WHILE akan berjalan sebanyak 19 kali iterasi, karena setiap iterasi digunakan untuk memproses satu data mahasiswa.

4. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya"!

Kode Program

```
while (i < jml) {
    System.out.print("Masukkan nilai mahasiswa ke-" + (i + 1) + ": ");
    nilai = input.nextInt();

    if (nilai < 0 || nilai > 100){
        System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid!");
        continue;
    }

    if (nilai > 80 && nilai <=100){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A");
        System.out.println(x:"Bagus, pertahankan nilainya!");
    } else if (nilai > 73 && nilai <= 80){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B+");
    } else if (nilai > 65 && nilai <= 73){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B");
    } else if (nilai > 60 && nilai <= 65){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C+");
    } else if (nilai > 50 && nilai <= 60){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C");
    } else if (nilai > 39 && nilai <= 50){
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah D");
    } else {
        System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah E");
    }
    i++;
}
```

Hasil Program

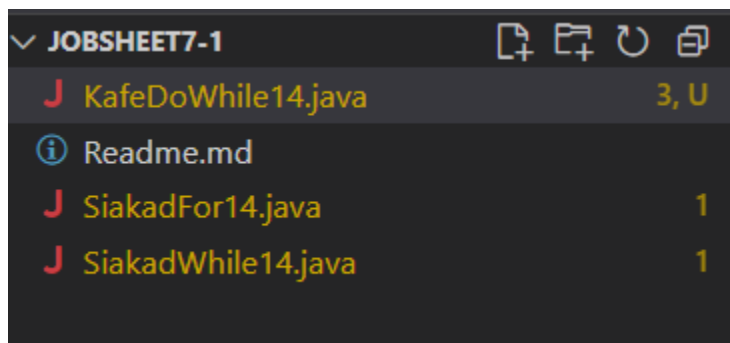
```
Masukkan jumlah mahasiswa: 5
Masukkan nilai mahasiswa ke-1: 100
Nilai mahasiswa ke-1 adalah A
Bagus, pertahankan nilainya!
Masukkan nilai mahasiswa ke-2: 92
Nilai mahasiswa ke-2 adalah A
Bagus, pertahankan nilainya!
Masukkan nilai mahasiswa ke-3: 65
Nilai mahasiswa ke-3 adalah C+
Masukkan nilai mahasiswa ke-4: 40
Nilai mahasiswa ke-4 adalah D
Masukkan nilai mahasiswa ke-5: 50
Nilai mahasiswa ke-5 adalah D
c:\Users\Moch Dedy Triagwi\Documents\Kuliah\DEDY NITIP\PRAKTEKDASPRO\Jobsheet7\jobsheet7-1>
```

5. Commit dan push kode program ke Github

 SiakadFor14.java	Modifikasi lulus & tdk lulus	1 hour ago
 SiakadWhile14.java	Modifikasi 1.2.1 While	now

1.3.1 Percobaan 3: Studi Kasus Transaksi di Kafe – Perulangan DO-WHILE

1. Buat file baru, beri nama **KafeDoWhileNoAbsen.java**



2. Buatlah struktur dasar program Java yang terdiri dari fungsi **main()**.
3. Tambahkan library Scanner di bagian atas (luar) class
4. Buat deklarasi Scanner dengan nama variabel **sc** di dalam fungsi **main()**
5. Deklarasikan variabel **kopi**, **teh**, dan **roti** bertipe integer untuk menampung banyaknya item yang dibeli pelanggan, serta **namaPelanggan** bertipe String. Deklarasi dan inisialisasi **hargaKopi** dengan 12000, **hargaTeh** dengan 7000, **hargaRoti** dengan 20000.

6. Buat struktur perulangan DO-WHILE dengan kondisi **true**

```
do {  
      
} while (true);
```

7. Di dalam perulangan DO-WHILE tersebut, tambahkan perintah untuk memasukkan **namaPelanggan**. Kemudian tambahkan kondisi IF untuk mengecek isi variabel **namaPelanggan**. Selanjutnya, tambahkan perintah untuk memasukkan banyaknya item yang dibeli pelanggan untuk setiap menu, apabila masukan nama pelanggan bukan “batal”. Hitung total harga pembelian dan tampilkan hasilnya.

```
do {  
    System.out.print(s:"Masukkan nama pelanggan (ketik 'batal' untuk keluar): ");  
    namaPelanggan = sc.nextLine();  
    if (namaPelanggan.equalsIgnoreCase(anotherString:"batal")) {  
        System.out.println(x:"Sistem dibatalkan!");  
        break;  
    }  
    System.out.print(s:"Jumlah Kopi: ");  
    kopi = sc.nextInt();  
    System.out.print(s:"Jumlah Teh: ");  
    teh = sc.nextInt();  
    System.out.print(s:"Jumlah Roti: ");  
    roti = sc.nextInt();  
  
    totalHarga = (kopi * hargaKopi) + (teh * hargaTeh) + (roti * hargaRoti);  
    System.out.println("Total yang harus dibayar: Rp." +totalHarga);  
    sc.nextLine();  
} while (true);  
System.out.println(x:"Semua transaksi selesai!");
```

8. Compile dan run program

```
J KafeDoWhile14.java > ...
1  import java.util.Scanner;
2
3  public class KafeDoWhile14 {
4      Run | Debug | Run main | Debug main
5      Scanner sc = new Scanner (System.in);
6
7      int kopi, teh, roti, hargaKopi = 12000, hargaTeh = 7000, hargaRoti = 20000, totalHarga;
8      String namaPelanggan;
9
10     do {
11         System.out.print(s:"Masukkan nama pelanggan (ketik 'batal' untuk keluar): ");
12         namaPelanggan = sc.nextLine();
13         if (namaPelanggan.equalsIgnoreCase(anotherString:"batal")) {
14             System.out.println(x:"Sistem dibatalkan!");
15             break;
16         }
17         System.out.print(s:"Jumlah Kopi: ");
18         kopi = sc.nextInt();
19         System.out.print(s:"Jumlah Teh: ");
20         teh = sc.nextInt();
21         System.out.print(s:"Jumlah Roti: ");
22         roti = sc.nextInt();
23
24         totalHarga = (kopi * hargaKopi) + (teh * hargaTeh) + (roti * hargaRoti);
25         System.out.println("Total yang harus dibayar: Rp." +totalHarga);
26         sc.nextLine();
27     } while (true);
28     System.out.println(x:"Semua transaksi selesai!");
29 }
30 }
```

9. Commit dan push kode program ke Github



The screenshot shows a GitHub repository named 'jobsheet7' (Public) by user 'MochDedy'. The repository has 1 branch (main) and 0 tags. The file list includes:

File Name	Commit Message	Time
KafeDoWhile14.java	1.3.1 Do While	now
Readme.md	Jobsheet 7	4 days ago
SiakadFor14.java	Modifikasi lulus & tdk lulus	1 hour ago
SiakadWhile14.java	Modifikasi 1.2.1 While	18 minutes ago

1.3.2 Verifikasi Hasil Percobaan

Cocokkan hasil compile kode program Anda dengan gambar berikut ini.

```
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Rena
Jumlah Kopi: 3
Jumlah Teh: 0
Jumlah Roti: 1
Total yang harus dibayar: Rp.56000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Yuni
Jumlah Kopi: 1
Jumlah Teh: 4
Jumlah Roti: 2
Total yang harus dibayar: Rp.80000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): BATAL
Sistem dibatalkan!
Semua transaksi selesai!

C:\Users\Moch Dedy Triagwi\Documents\Kuliah\DEDY NITIP\PRAKTEKDASPRO\Jobsheet7\jobsheet7-1>
```

1.3.3 Pertanyaan

1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah “batal”, maka berapa kali perulangan dilakukan?

Jawab:

Perulangan DO-WHILE akan tetap berjalan satu kali.

Hal ini karena pada perulangan do-while, blok perintah di dalam “do” selalu dijalankan minimal satu kali terlebih dahulu sebelum kondisi pada while dicek. Jadi meskipun input pertama langsung “batal”, perulangan tetap terjadi sekali.

2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!

Jawab:

Kondisi perulangan berhenti saat user menginput string berupa “batal” dan break yang memberhentikan perulangan tersebut.

```
if (namaPelanggan.equalsIgnoreCase(anotherString:"batal")) {
    System.out.println(x:"Sistem dibatalkan!");
    break;
}
```

3. Apa fungsi dari penggunaan nilai true pada kondisi DO-WHILE?

Jawab:

Nilai true digunakan agar perulangan berjalan terus (infinite loop) sampai ada perintah yang secara eksplisit menghentikannya, yaitu break. Dengan kata lain, while (true) membuat program terus mengulang input pelanggan sampai pengguna memilih berhenti.

4. Mengapa perulangan DO-WHILE tersebut tetap berjalan meskipun tidak ada komponen inisialisasi dan update?

Jawab:

Perulangan DO-WHILE ini tetap berjalan karena kendali perulangannya menggunakan kondisi logika true dan perintah break di dalam blok kode, bukan dari nilai variabel penghitung seperti i++.

Jadi meskipun tidak ada komponen inisialisasi atau update, perulangan akan tetap berulang sampai kondisi if terpenuhi dan break menghentikan proses.

Tugas

1. Seorang pengelola bioskop ingin membuat program untuk menghitung total penjualan tiket dalam satu hari. Tiket dijual dengan harga Rp 50.000 per tiket. Program harus menghitung total tiket yang terjual dan total harga penjualan tiket selama satu hari dengan ketentuan sebagai berikut:
 - a. Jika pelanggan membeli lebih dari 4 tiket, pelanggan mendapatkan diskon 10%.
 - b. Jika pelanggan membeli lebih dari 10 tiket, pelanggan mendapatkan diskon 15%.
 - c. Jika input jumlah tiket tidak valid (negatif), program akan mengabaikan input tersebut dan meminta input ulang.

Kode Program

```
J Tugas1jobsheet7/java > ...
1  import java.util.Scanner;
2
3  public class Tugas1jobsheet7 {
4      Run | Debug | Run main | Debug main
5      public static void main(String[] args) {
6          Scanner sc = new Scanner(System.in);
7
8          int jumlahTiket, totTiket = 0, totPenjualan = 0;
9          double hargaTiket = 50000, totHarga;
10         String lanjut;
11
12         do {
13             System.out.print(s:"Masukkan tiket yang dibeli: ");
14             jumlahTiket = sc.nextInt();
15
16             if (jumlahTiket <= 0) {
17                 System.out.println(x:"Input tidak valid. Masukkan ulang jumlah tiket!");
18                 continue;
19             }
20             totHarga = jumlahTiket * hargaTiket;
21             if (jumlahTiket > 10) {
22                 totHarga = totHarga - (totHarga * 0.15);
23             } else if (jumlahTiket > 4) {
24                 totHarga = totHarga - (totHarga * 0.1);
25             } else {
26             }
27
28             totTiket += jumlahTiket;
29             totPenjualan += totHarga;
30
31             System.out.println("Total harga pembelian ini:: Rp " + totHarga);
32             sc.nextLine();
33             System.out.println(x:"Apakah masih ada pelanggan? (Ya/Tidak) ");
34             lanjut = sc.nextLine();
35
36             if (lanjut.equalsIgnoreCase(anotherString:"Tidak")) {
37                 break;
38             }
39         } while (true);
40
41         System.out.println("\nTotal tiket terjual: " + totTiket);
42         System.out.println("Total penjualan: Rp" + totPenjualan);
43     }
44 }
```

Hasil Program

```
Masukkan tiket yang dibeli: 11
Total harga pembelian ini:: Rp 467500.0
Apakah masih ada pelanggan? (Ya/Tidak)
ya
Masukkan tiket yang dibeli: 4
Total harga pembelian ini:: Rp 200000.0
Apakah masih ada pelanggan? (Ya/Tidak)
ya
Masukkan tiket yang dibeli: 10
Total harga pembelian ini:: Rp 450000.0
Apakah masih ada pelanggan? (Ya/Tidak)
tidak

Total tiket terjual: 25
Total penjualan: Rp1117500
```

```
c:\Users\Moch Dedy Triagwi\Documents\Kuliah\DEDY NITIP\PRAKTEKDasPRO\Jobsheet7\jobsheet7-1>
```

2. Sebuah tempat parkir ingin membuat program untuk menghitung total pembayaran parkir dari beberapa kendaraan. Tarif parkir adalah Rp 3.000 per jam untuk mobil dan Rp 2.000 per jam untuk motor. Namun, jika durasi parkir lebih dari 5 jam, diberikan tarif tetap sebesar Rp 12.500 untuk semua kendaraan. Program akan terus meminta masukan selama input bukan 0. Implementasikan flowchart tersebut ke dalam bentuk kode program Java!

Kode Program

```
J Tugas2Jobsheet7.java > Java > Tugas2Jobsheet7 > main(String[] args)
1  import java.util.Scanner;
2
3  public class Tugas2Jobsheet7 {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6
7          int jenis, durasi, total = 0;
8
9          do {
10             System.out.println(x:"Masukan Jenis Kendaraan (1.Mobil = Rp.3000 2.Motor = Rp.2000 0 = keluar)");
11             jenis = sc.nextInt();
12
13             if (jenis == 0) {
14                 break;
15             }
16
17             if (jenis != 1 && jenis != 2) {
18                 System.out.println(x:"Jenis kendaraan tidak valid! Coba lagi.");
19                 continue;
20             }
21
22             System.out.println(x:"Masukan Durasi Parkir: ");
23             durasi = sc.nextInt();
24
25             if (durasi <= 0) {
26                 System.out.println(x:"Durasi tidak valid! Coba lagi.");
27                 continue;
28             }
29             if (durasi > 5) {
30                 total += 12500;
31             } else if (jenis == 1) {
32                 total += durasi * 3000;
33             } else {
34                 total += durasi * 2000;
35             }
36         } while (true);
37         System.out.println("Total Pendapatan Parkir Hari Ini: Rp " + total);
38     }
39 }
```

Hasil Program

```
Masukan Jenis Kendaraan (1.Mobil = Rp.3000 2.Motor = Rp.2000 0 = keluar)
1
Masukan Durasi Parkir:
10
Masukan Jenis Kendaraan (1.Mobil = Rp.3000 2.Motor = Rp.2000 0 = keluar)
3
Jenis kendaraan tidak valid! Coba lagi.
Masukan Jenis Kendaraan (1.Mobil = Rp.3000 2.Motor = Rp.2000 0 = keluar)
2
Masukan Durasi Parkir:
15
Masukan Jenis Kendaraan (1.Mobil = Rp.3000 2.Motor = Rp.2000 0 = keluar)
0
Total Pendapatan Parkir Hari Ini: Rp 25000
```

```
C:\Users\Moch Dedy Triagwi\Documents\Kuliah\DEDY NITIP\PRAKTEKDASPRO\Jobsheet7\jobsheet7-1>
```

