

JOBSCHEET 7

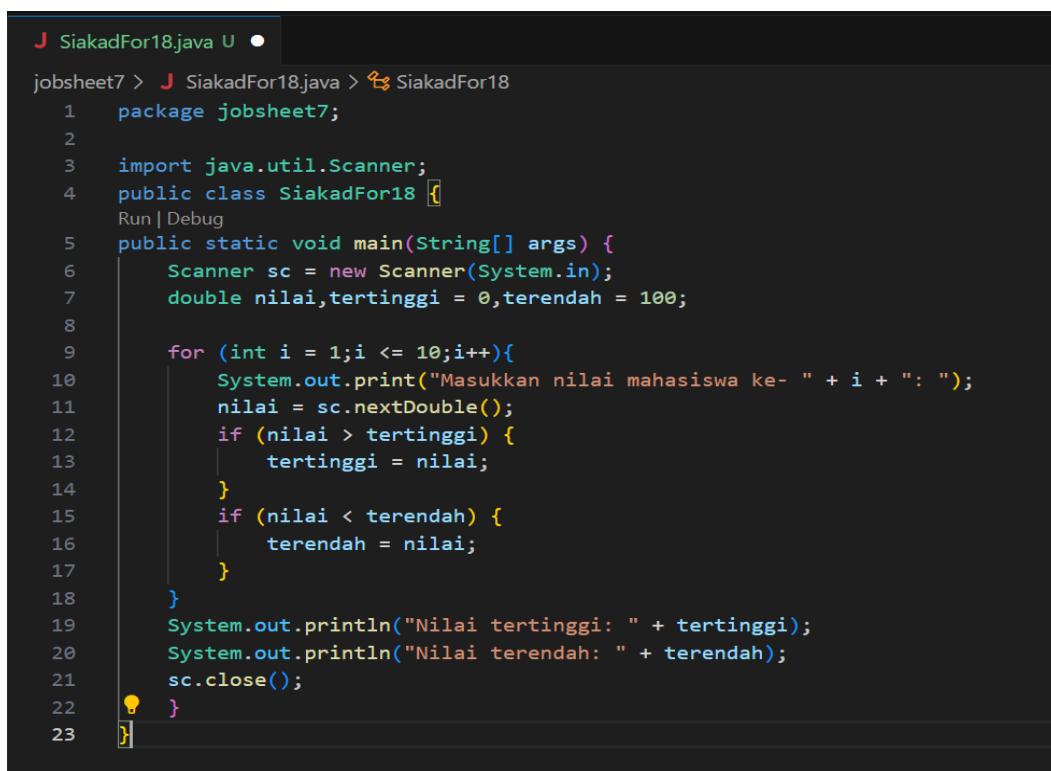
NAMA : M. Javier Thufail

NIM : 254107020019

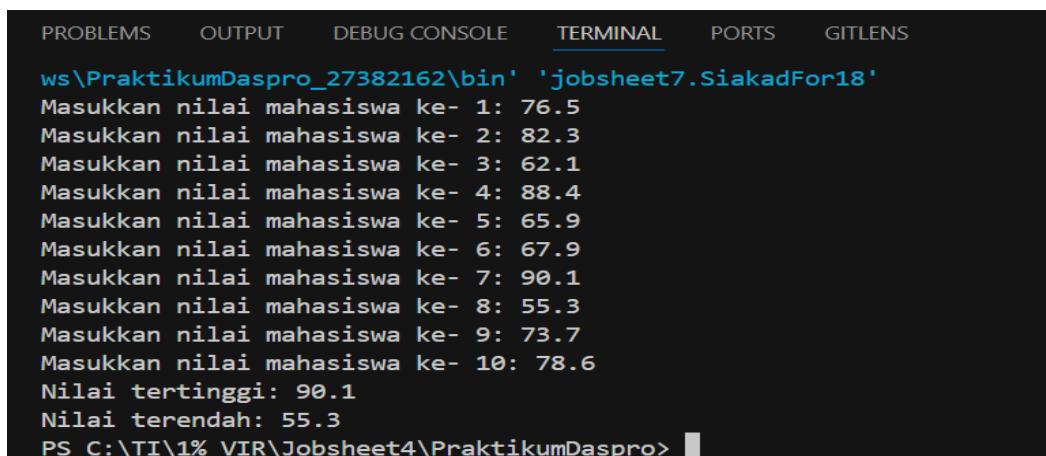
KELAS : TI 1-G

NO. ABSEN : 18

PERCOBAAN 1 (Studi Kasus Nilai Mahasiswa di SIAKAD – Perulangan FOR).



```
SiakadFor18.java
1 package jobsheet7;
2
3 import java.util.Scanner;
4 public class SiakadFor18 {
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
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         System.out.println("Nilai tertinggi: " + tertinggi);
20         System.out.println("Nilai terendah: " + terendah);
21         sc.close();
22     }
23 }
```



PROBLEMS	OUTPUT	DEBUG CONSOLE	TERMINAL	PORTS	GITLENS
			ws\PraktikumDaspro_27382162\bin' 'jobsheet7.SiakadFor18'		
	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				
	PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro>				

Pertanyaan!

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

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

- a. Inisialisasi → int i = 1, digunakan untuk nilai sementara.
 - b. Kondisi → i <= 10, digunakan untuk mengecek kondisi dalam perulangan, jika kondisinya tidak sesuai program akan keluar dari perulangan.
 - c. Increment → i++, digunakan untuk menambahkan nilai i setiap perulangan berakhir.
2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Agar, nilai tertinggi yang ditampilkan lebih dari 0 dan nilai terendah yang ditampilkan kurang dari 100.

```
6 | double nilai,tertinggi = 100, terendah = 0; You
7 |
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS
\bin' 'jobsheet7.SiakadFor18'
Masukkan nilai mahasiswa ke- 1: 90
Masukkan nilai mahasiswa ke- 2: 90
Masukkan nilai mahasiswa ke- 3: 89
Masukkan nilai mahasiswa ke- 4: 78
Masukkan nilai mahasiswa ke- 5: 67
Masukkan nilai mahasiswa ke- 6: 89
Masukkan nilai mahasiswa ke- 7: 78
Masukkan nilai mahasiswa ke- 8: 89
Masukkan nilai mahasiswa ke- 9: 97
Masukkan nilai mahasiswa ke- 10: 75
Nilai tertinggi: 100.0
Nilai terendah: 0.0
PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro>
```

Yang terjadi adalah output nilai tertinggi hanya akan menampilkan nilai yang diatas 100 sebaliknya nilai terendah hanya akan menampilkan nilai yang dibawah 0

```
Masukkan nilai mahasiswa ke- 1: 200
Masukkan nilai mahasiswa ke- 4: -49
Masukkan nilai mahasiswa ke- 5: -1
Masukkan nilai mahasiswa ke- 6: -50
Masukkan nilai mahasiswa ke- 7: 556
Masukkan nilai mahasiswa ke- 8: 78
Masukkan nilai mahasiswa ke- 9: 39
Masukkan nilai mahasiswa ke- 10: 38
Nilai tertinggi: 556.0
Nilai terendah: -50.0
PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro>
```

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

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

potongan kode ini digunakan untuk mencari angka terbesar dan terkecil dari nilai yang di input. Jadi Setiap kali pengguna memasukan nilai baru, program akan membandingkannya dengan nilai tertinggi dan terendah yang sudah disimpan sebelumnya. Jika nilai baru lebih besar dari nilai tertinggi, maka nilai itu akan menggantikannya sebagai nilai tertinggi yang baru. Sebaliknya, jika nilai baru lebih kecil dari nilai terendah, maka nilai itu akan menjadi nilai terendah yang baru. Habis itu semua nilai dimasukkan, program dapat menentukan angka paling tinggi dan paling rendah .

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!

Code:

The screenshot shows a Java code editor with the following code:

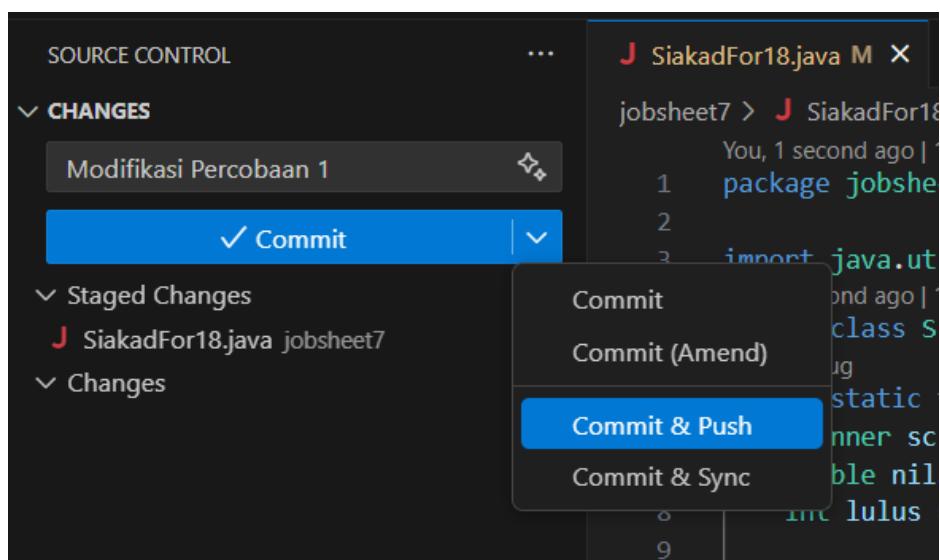
```
J SiakadFor18.java M
jobsheet7 > J SiakadFor18.java > ↗ SiakadFor18
You, 1 minute ago | 1 author (You)
1 package jobsheet7;
2
3 import java.util.Scanner;
4 public class SiakadFor18 {
Run|Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         double nilai, tertinggi = 0, terendah = 100;
8         int lulus = 0, tidakLulus = 0;
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 += 1;
21            }else if (nilai < 60) {
22                tidakLulus +=1;
23            }
24        }
25        System.out.println("Nilai tertinggi\t: " + tertinggi);
26        System.out.println("Nilai terendah\t: " + terendah);
27        System.out.println("Jumlah mahasiswa lulus\t\t: " + lulus);
28        System.out.println("Jumlah mahasiswa tidak lulus\t: " + tidakLulus);
29        sc.close();
30    }
31 }
```

The code defines a class SiakadFor18 that reads 10 student scores from the user. It keeps track of the highest and lowest scores. It also counts the number of students who passed (lulus) and failed (tidakLulus) based on a passing grade of 60. Finally, it prints out the highest score, lowest score, total number of students, and the count of each.

Output:

```
C:\Users\user\AppData\Roaming\Code\User\workspaces\storage
Masukkan nilai mahasiswa ke- 1: 56
Masukkan nilai mahasiswa ke- 2: 69
Masukkan nilai mahasiswa ke- 3: 90
Masukkan nilai mahasiswa ke- 4: 67
Masukkan nilai mahasiswa ke- 5: 35
Masukkan nilai mahasiswa ke- 6: 89
Masukkan nilai mahasiswa ke- 7: 90
Masukkan nilai mahasiswa ke- 8: 10
Masukkan nilai mahasiswa ke- 9: 40
Masukkan nilai mahasiswa ke- 10: 85
Nilai tertinggi : 90.0
Nilai terendah : 10.0
Jumlah mahasiswa lulus      : 6
Jumlah mahasiswa tidak lulus : 4
PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro>
```

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



PERCOBAAN 2 (Studi Kasus Nilai Mahasiswa di SIAKAD – Perulangan WHILE).

```
SiakadFor18.java SiakadWhile18.java

jobsheet7 > SiakadWhile18.java > SiakadWhile18

1 package jobsheet7;
2
3 import java.util.Scanner;
4 public class SiakadWhile18 {
5     Run | Debug
6     public static void main(String[] args) {
7         Scanner sc =new Scanner(System.in);
8         int jml,nilai,i = 0;
9
10        System.out.print("Masukkan jumlah mahasiswa: ");
11        jml = sc.nextInt();
12        while (i <= jml) {
13            System.out.print("Masukkan nilai mahasiswa ke-" + (i+1) +": ");
14            nilai = sc.nextInt();
15
16            if (nilai < 0 || nilai > 100) {
17                System.out.println("Nilai tidak valid. Masukkan lagi nilai yang valid!");
18                continue;
19            }
20            if (nilai > 80 && nilai <= 100) {
21                System.out.println("Nilai mahasiswa ke-" + (i + 1) +" adalah A");
22            }else if (nilai > 73 && nilai <= 80) {
23                System.out.println("Nilai mahasiswa ke-" + (i + 1) +" adalah B+");
24            }else if (nilai > 65 && nilai <= 73) {
25                System.out.println("Nilai mahasiswa ke-" + (i + 1) +" adalah B");
26            }else if (nilai > 60 && nilai <= 65) {
27                System.out.println("Nilai mahasiswa ke-" + (i + 1) +" adalah C+");
28            }else if (nilai > 50 && nilai <= 60) {
29                System.out.println("Nilai mahasiswa ke-" + (i + 1) +" adalah C");
30            }else if (nilai > 39 && nilai <= 50) {
31                System.out.println("Nilai mahasiswa ke-" + (i + 1) +" adalah D");
32            }else{
33                System.out.println("Nilai mahasiswa ke-" + (i + 1) +" adalah E");
34            }
35            i++;
36        }
37        sc.close();
38    }
}
```

```
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
```

Pertanyaan!

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

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

- a. nilai < 0 || nilai > 100 .

Bagian ini dipakai untuk mengecek apakah angka yang dimasukkan salah. Lalu ada Tanda logika operator || artinya “atau”. Jadi, kalau nilai yang dimasukkan kurang dari 0 atau lebih dari 100, program akan tahu kalau angka itu tidak benar. Nilai yang dianggap benar hanya yang berada di antara 0 sampai 100 saja. Jadi, kegunaannya untuk mengecek kondisi.

- b. continue

Kata continue dipakai supaya program lompat ke pengulangan berikutnya kalau ada kesalahan. Jadi, kalau pengguna memasukkan nilai yang tidak benar, program akan menulis pesan “Nilai tidak valid. Masukan lagi nilai yang valid!” dan langsung meminta pengguna untuk mencoba lagi, tanpa melanjutkan langkah yang salah tadi.

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

Agar i nya bertambah sebelum akhirnya diulang.

```
Masukkan jumlah mahasiswa  
1  
Masukkan nilai mahasiswa ke-2:10  
Nilai mahasiswa ke-2 adalah E  
Masukkan nilai mahasiswa ke-3:10  
Nilai mahasiswa ke-3 adalah E
```

Program akan tetap berjalan namun terdapat masalah pada urutan mahasiswa yang langsung loncat ke 2.

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

```
Nilai mahasiswa ke-19 adalah E  
Masukkan nilai mahasiswa ke-20: 10  
Nilai mahasiswa ke-20 adalah E  
PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro>
```

Perulangan akan terjadi sebanyak 20 kali, karena perulangan dimulai dari 0. Ialu di variable mahasiswa pertama output nya tidak mahasiswa ke 0 karena i nya ditambah 1 sehingga tampilan di outputnya juga “masukan nilai mahasiswa ke-20”.

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

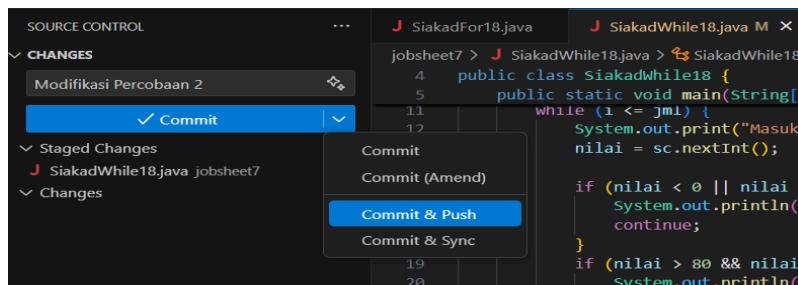
Code:

```
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) {
```

Output:

```
Masukkan nilai mahasiswa ke-1: 90
Nilai mahasiswa ke-1 adalah A
Bagus, pertahankan nilainya
Masukkan nilai mahasiswa ke-2: ■
```

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”



PERCOBAAN 3 (Studi Kasus Transaksi di Kafe – Perulangan DO-WHILE).

```
jobsheet7 > J KafeDoWhile18.java > KafeDoWhile18
1 package jobsheet7;
2
3 import java.util.Scanner;
4 public class KafeDoWhile18 {
    Run|Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int kopi, teh, roti;
        String namaPelanggan;
        int hargaKopi = 12000, hargaTeh = 7000, hargaRoti = 20000, totalHarga;
10
11    do {
12        System.out.print(s:"Masukkan nama pelanggan (Ketik 'batal' untuk keluar): ");
13        namaPelanggan = sc.nextLine();
14        if (namaPelanggan.equalsIgnoreCase("Batal")) {
15            System.out.println(x:"Transaksi dibatalkan.");
16            break;
17        }
18        System.out.print(s:"Jumlah kopi: ");
19        kopi = sc.nextInt();
20        System.out.print(s:"Jumlah teh: ");
21        teh = sc.nextInt();
22        System.out.print(s:"Jumlah roti: ");
23        roti = sc.nextInt();
24
25        totalHarga = (kopi * hargaKopi) + (teh * hargaTeh) + (roti * hargaRoti);
26        System.out.println("Total harga yang harus dibayarkan :" + totalHarga);
27        sc.nextLine();
28    } while (true);
29
30    System.out.println(x:"Semua transaksi selesai");
31    sc.close();
32
33 }
```

The code is a Java program named KafeDoWhile18. It uses a do-while loop to repeatedly ask the user for the name of a customer (until they type 'batal'). For each customer, it asks for the quantity of coffee, tea, and bread. It calculates the total price based on the prices defined at the top (12000 for coffee, 7000 for tea, 20000 for bread). Finally, it prints a message indicating all transactions are completed and closes the scanner.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro> & 'C:\Program Files\Java\aceStorage\14c66d3581d5c197b3e5a48dbfa5d9db\redhat.java\jdt_ws\Prak
Masukkan nama pelanggan (Ketik 'batal' untuk keluar): Javier
Jumlah kopi: 3
Jumlah teh: 0
Jumlah roti: 1
Total harga yang harus dibayarkan :56000
Masukkan nama pelanggan (Ketik 'batal' untuk keluar): Lindhu
Jumlah kopi: 1
Jumlah teh: 4
Jumlah roti: 2
Total harga yang harus dibayarkan :80000
Masukkan nama pelanggan (Ketik 'batal' untuk keluar): batal
Transaksi dibatalkan.
Semua transaksi selesai
PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro>
```

Pertanyaan!

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

```
' 'C:\Users\user\AppData\Roaming\Code\User\workspaceStorage\1
Masukkan nama pelanggan (Ketik 'batal' untuk keluar): batal
Transaksi dibatalkan.
Semua transaksi selesai
PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro>
```

Dalam perulangan DO-WHILE, program pasti berjalan minimal satu kali, walaupun inputnya langsung “batal”. Jadi, kalau kita langsung mengetik “batal” program tetap dijalankan sekali untuk membaca kata itu. Setelah itu, karena sudah tertulis “batal”, program langsung berhenti dan menampilkan pesan bahwa “Transaksi dibatalkan”.

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

```
if (namaPelanggan.equalsIgnoreCase(anotherString:"Batal")) {
    System.out.println(x:"Transaksi dibatalkan.");
    break;
```

Kondisi yang digunakan untuk memberhentikan perulangan tersebut adalah kondisi dimana jika user menginput “batal” dan perulangan akan dihentikan karena ada kode break.

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

Fungsi dari penggunaan nilai true pada kondisi DO-WHILE adalah untuk membuat perulangan terus berjalan tanpa henti, karena kondisinya selalu benar. dan cara untuk menghentikannya adalah menggunakan kode break.

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

Sama seperti perulangan while, perulangan do-while tetap bisa berjalan tanpa inisialisasi dan update, karena yang dibutuhkan hanya kondisi untuk menentukan apakah perulangan akan dilanjutkan atau dihentikan.

Tugas!

- 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:
 - Jika pelanggan membeli lebih dari 4 tiket, pelanggan mendapatkan diskon 10%.
 - Jika pelanggan membeli lebih dari 10 tiket, pelanggan mendapatkan diskon 15%.
 - Jika input jumlah tiket tidak valid (negatif), program akan mengabaikan input tersebut dan meminta input ulang.

Commit dan push program Anda ke Github dengan pesan “Tugas 1”

Catatan: Perulangan dapat menggunakan for, while, atau do-while. Penambahan break atau continue jika diperlukan

Code:

```
jobsheet7 > J Tugas1PenjualanTiket18.java > ...
You, 2 minutes ago | 1 author (You)
1 package jobsheet7;
2
3 import java.util.Scanner;
4
5 You, 2 minutes ago | 1 author (You)
6 public class Tugas1PenjualanTiket18 {
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9         double hargaTiket = 50000, totalHarga = 0;
10        double totalHargaPelanggan;
11        int jumlahTiketYangDibeli = 0, pelanggan, i = 1, totalTiketTerjual = 0;
12
13        System.out.println("==== PENJUALAN TIKET BIOSKOP ====");
14        System.out.print("Masukkan jumlah pelanggan: ");
15        pelanggan = sc.nextInt();
16        while (i <= pelanggan) {
17            System.out.print("Masukkan jumlah tiket yang dibeli pelanggan ke- " + i + ": ");
18            jumlahTiketYangDibeli = sc.nextInt();
19
20            if (jumlahTiketYangDibeli < 0) {
21                System.out.println("Masukkan jumlah tiket sampai valid!");
22                continue;
23            }
24            if (jumlahTiketYangDibeli > 10) {
25                System.out.println("Selamat anda mendapatkan diskon sebesar 15%");
26                hargaTiket -= hargaTiket * 0.15;
27            } else if (jumlahTiketYangDibeli > 4) {
28                System.out.println("Selamat anda mendapatkan diskon sebesar 10%");
29                hargaTiket -= hargaTiket * 0.10;
30            }
31
32            totalHargaPelanggan = jumlahTiketYangDibeli * hargaTiket;
33            totalTiketTerjual += jumlahTiketYangDibeli;
34            totalHarga += totalHargaPelanggan;
35            i++;
36        }
37        System.out.println("=====");
38        System.out.println("Total tiket terjual : " + totalTiketTerjual);
39        System.out.println("Total harga tiket : " + totalHarga);
40        System.out.println("=====");
41        sc.close();
42    }
43}
44
```

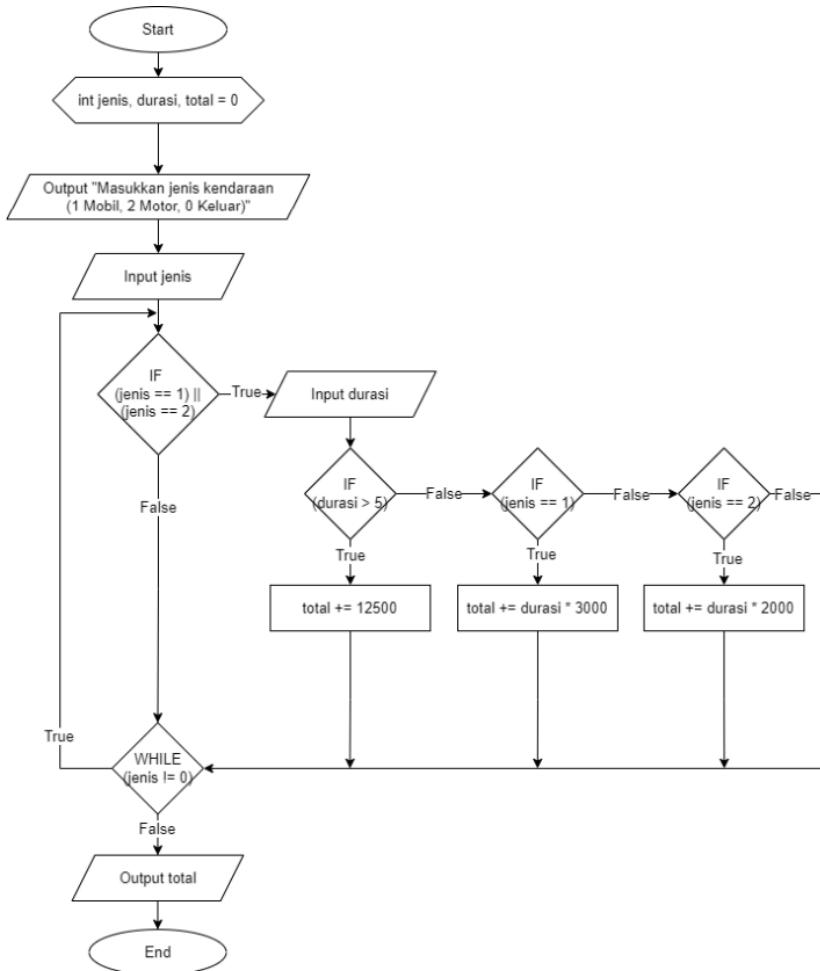
Output:

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS    GITLENS

jdt_ws\PraktikumDaspro_27382162\bin' 'jobsheet7.Tugas1PenjualanTiket18'
== PENJUALAN TIKET BIOSKOP ==
Masukkan jumlah pelanggan: 3
Masukkan jumlah tiket yang dibeli pelanggan ke-1: -1
Masukkan jumlah tiket sampai valid!
Masukkan jumlah tiket yang dibeli pelanggan ke-1: 2
Masukkan jumlah tiket yang dibeli pelanggan ke-2: 6
Selamat anda mendapatkan diskon sebesar 10%
Masukkan jumlah tiket yang dibeli pelanggan ke-3: 12
Selamat anda mendapatkan diskon sebesar 15%
=====
Total tiket terjual : 20
Total harga tiket   : 829000.0
=====

PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro>
```

2. Perhatikan flowchart berikut!



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! Commit dan push program Anda ke Github dengan pesan “Tugas 2”

Code:

```
J SiakadFor18.java J SiakadWhile18.java J KafeDoWhile18.java J Tugas1PenjualanTiket18.java M J Tugas2Tempat
jobsheet7 > J Tugas2TempatParkir18.java > Tugas2TempatParkir18 > main(String[])
You, 5 minutes ago | 1 author (You)
1 package jobsheet7;
2
3 import java.util.Scanner;
4
5 public class Tugas2TempatParkir18 {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9         int jenis, durasi, total = 0;
10
11     do {
12         System.out.println("\nSISTEM PARKIR KENDARAAN");
13         System.out.println("1. Mobil");
14         System.out.println("2. Motor");
15         System.out.println("0. Keluar");
16         System.out.print("Masukkan jenis kendaraan: ");
17         jenis = sc.nextInt();
18
19         if (jenis == 1 || jenis == 2) {
20             System.out.print("Masukkan durasi parkir (jam): ");
21             durasi = sc.nextInt();
22
23             if (durasi > 5) {
24                 total += 12500;
25             } else if (jenis == 1) {
26                 total += durasi * 3000;
27             } else if (jenis == 2) {
28                 total += durasi * 2000;
29             }
30
31             System.out.println("-----");
32             System.out.println("Durasi Parkir : " + durasi + " jam");
33             System.out.println("Biaya Sementara : Rp " + total);
34             System.out.println("-----");
35
36         } else if (jenis != 0) {
37             System.out.println("Jenis Kendaraan tidak valid.");
38             System.out.println("Masukkan 1 (Mobil), 2 (Motor), atau 0 untuk keluar.");
39         }
40     } while (jenis != 0);
41
42     System.out.println("\n=====");
43     System.out.println("Total Biaya Parkir : Rp " + total);
44     System.out.println("Terima kasih telah menggunakan layanan kami!");
45     System.out.println("=====");
46
47     sc.close();
48 }
49
50
```

Output:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS
-----
Durasi Parkir : 2 jam
Biaya Sementara : Rp 4000
-----

SISTEM PARKIR KENDARAAN
1. Mobil
2. Motor
0. Keluar
Masukkan jenis kendaraan: 1
Masukkan durasi parkir (jam): 2
-----
Durasi Parkir : 2 jam
Biaya Sementara : Rp 10000
-----

SISTEM PARKIR KENDARAAN
1. Mobil
2. Motor
0. Keluar
Masukkan jenis kendaraan: 0
-----
Total Biaya Parkir : Rp 10000
Terima kasih telah menggunakan layanan kami!
=====
PS C:\TI\1% VIR\Jobsheet4\PraktikumDaspro>
```