

Nama : Fiza Rahmatus Sholikha
Kelas : SIB 1B
No. absen : 09

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

```
import java.util.Scanner;

public class SiakadFor09 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        double nilai, tertinggi = 0, terendah = 100;

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

```
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
```

https://github.com/Fizzrss/Daspro_Jobsheet-7/commit/446e175e9efa42321dc5b9fdb349b8a84f297c2

Pertanyaan

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

Jawab:

- `int i = 1` sebagai inisialisasi dimana perulangan akan dimulai dari nilai 1
 - `i <= 10` sebagai kondisi dimana perulangan akan berjalan selama `i` kurang dari atau sama dengan 10 dan akan berhenti apabila kondisinya salah
 - `++` increment ini digunakan agar nilai `i` bertambah satu setiap perulangan
2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Jawab:

- Inisialisasi tertinggi dengan 0 karena program menganggap bahwa nilai tertinggi yang paling rendah adalah 0 pada awalnya dan jika nilai yang diinput lebih besar daripada 0 maka nilai 0 digantikan nilai yang diinput tersebut
 - Inisialisasi terendah dengan 100 karena program menganggap nilai terendah yang paling tinggi adalah 100 dan apabila ada nilai yang lebih kecil dari 100 maka nilai 100 digantikan dengan input tersebut.
3. Jelaskan fungsi dan alur kerja dari potongan kode berikut!

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

Jawab:

Fungsinya untuk memperbarui nilai tertinggi dan terendah berdasarkan input

Alur kerja

- Program melakukan pengecekan apakah nilai yang di input lebih besar daripada nilai tertinggi
 - Jika iya maka nilai tertinggi diperbarui. Jika tidak maka nilai tertinggi tetap
 - Program melakukan pengecekan apakah nilai yang diinput lebih kecil daripada nilai terendah
 - Jika iya maka nilai terendah diperbarui. Jika tidak nilai terendah tetap
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!

```
import java.util.Scanner;  
  
public class SiakadFor09 {  
  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);
```

```

double nilai, tertinggi = 0, terendah = 100;
int lulus = 0, tidakLulus = 0;

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

    if (nilai >= 60){
        lulus++;
    }else {
        tidakLulus++;
    }
}
System.out.println("Nilai tertinggi: " + tertinggi);
System.out.println("Nilai terendah: " + terendah);
System.out.println("Jumlah mahasiwa yang lulus: " + lulus);
System.out.println("Jumlah mahasiwa yang tidak lulus: " +
tidakLulus);
sc.close();
}
}

```

```

Masukkan nilai mahasiswa ke- 1:
90
Masukkan nilai mahasiswa ke- 2:
98
Masukkan nilai mahasiswa ke- 3:
87
Masukkan nilai mahasiswa ke- 4:
86
Masukkan nilai mahasiswa ke- 5:
83
Masukkan nilai mahasiswa ke- 6:
67
Masukkan nilai mahasiswa ke- 7:
54
Masukkan nilai mahasiswa ke- 8:
90
Masukkan nilai mahasiswa ke- 9:
95
Masukkan nilai mahasiswa ke- 10:
70
Nilai tertinggi: 98.0
Nilai terendah: 54.0
Jumlah mahasiwa yang lulus: 9
Jumlah mahasiwa yang tidak lulus: 1

```

5. Commit dan push kode program ke Github

https://github.com/Fizzrss/Daspro_Jobsheet-7/blob/main/SiakadFor09.java

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

```
import java.util.Scanner;

public class SiakadWhile09 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int nilai, jml;

        System.out.println("Masukkan jumlah mahasiswa: ");
        jml = sc.nextInt();

        int i = 0;
        while (i < jml){
            System.out.println("Masukkan nilai mahasiswa ke- " + (i + 1) +
":");
            nilai = sc.nextInt();

            if (nilai < 0 || nilai > 100){
                System.out.println("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++;
        }
    }
}
```

```
}  
}
```

```
Masukkan jumlah mahasiswa:  
7  
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  
Masukkan nilai mahasiswa ke- 6:  
32  
nilai mahasiswa ke- 6 adalah E  
Masukkan nilai mahasiswa ke- 7:  
75  
nilai mahasiswa ke- 7 adalah B+
```

https://github.com/Fizzrss/Daspro_Jobsheet-7/commit/4982b02877fe5d695482e5c3d8cd655b1eeb2853

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. `nilai < 0 || nilai > 100`

Jawab:

Digunakan untuk pengecekan jika nilai kurang dari 0 (angka negatif) atau nilai lebih dari 100 maka inputnya salah karena rentang nilainya antara 0 hingga 100

- b. `continue`

untuk melewati literasi saat ini jika input yang dimasukkan salah sehingga program melanjutkan literasi selanjutnya

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

Jawab:

- Fungsi `i++` berada di akhir agar program hanya akan menambah nilai `i` setelah nilai mahasiswa yang valid dimasukkan dan diproses. Jika input nilai tidak valid program akan melakukan `continue` sehingga `i++` tidak dijalankan.
 - Apabila `i++` ditulis di awal perulangan, program akan melewati mahasiswa jika nilai tidak valid dan mengacaukan urutan mahasiswa dalam output
3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan `WHILE` akan berjalan?

Jawab:

Jika input yang dimasukkan 19 maka perulangan `while` akan berjalan sebanyak 19 kali selama kondisi memenuhi syaratnya

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

```
import java.util.Scanner;

public class SiakadWhile09 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int nilai, jml;

        System.out.println("Masukkan jumlah mahasiswa: ");
        jml = sc.nextInt();

        int i = 0;
        while (i < jml){
            System.out.println("Masukkan nilai mahasiswa ke- " + (i + 1) + ":");
            nilai = sc.nextInt();

            if (nilai < 0 || nilai > 100){
                System.out.println("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("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++;
}
sc.close();
}
}

```

```

Masukkan jumlah mahasiswa:
7
Masukkan nilai mahasiswa ke- 1:
90
nilai mahasiswa ke- 1 adalah A
Bagus, pertahankan nilainya
Masukkan nilai mahasiswa ke- 2:
75
nilai mahasiswa ke- 2 adalah B+
Masukkan nilai mahasiswa ke- 3:
71
63
nilai mahasiswa ke- 4 adalah C+
58
nilai mahasiswa ke- 5 adalah C
Masukkan nilai mahasiswa ke- 6:
45
nilai mahasiswa ke- 6 adalah D
Masukkan nilai mahasiswa ke- 7:
35
nilai mahasiswa ke- 7 adalah E

```

5. Commit dan push kode program ke Github

https://github.com/Fizzrss/Daspro_Jobsheet-7/blob/main/SiakadWhile09.java

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

```
import java.util.Scanner;

public class KafeDoWhile09 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int kopi, teh, roti, totalHarga = 0;
        int hargaKopi = 12000;
        int hargaTeh = 7000;
        int hargaRoti = 20000;
        String namaPelanggan;

        do {
            System.out.print("Masukkan nama pelanggan (ketik 'batal' untuk
keluar): ");
            namaPelanggan = sc.nextLine();
            if (namaPelanggan.equalsIgnoreCase("batal")){
                System.out.println("Transaksi dibatalkan");
                break;
            }
            System.out.print("jumlah kopi: ");
            kopi = sc.nextInt();
            System.out.print("jumlah teh: ");
            teh = sc.nextInt();
            System.out.print("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("Semua transaksi selesai");
        sc.close();
    }
}
```



```
Masukkan nama pelanggan (ketik 'batal' untuk keluar): fiza  
jumlah kopi: 2  
jumlah teh: 3  
jumlah roti: 1  
Total yang harus dibayar: Rp 65000  
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Tania  
jumlah kopi: 2  
jumlah teh: 1  
jumlah roti: 1  
Total yang harus dibayar: Rp 51000  
Masukkan nama pelanggan (ketik 'batal' untuk keluar): batal  
Transaksi dibatalkan  
Semua transaksi selesai
```

https://github.com/Fizzrss/Daspro_Jobsheet-7/blob/main/KafeDoWhile09.java

Pertanyaan

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

Jawab:

Hanya satu kali karena input yang dimasukkan batal maka program akan otomatis berhenti dan tidak lagi melakukan looping

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

Jawab:

```
if (namaPelanggan.equalsIgnoreCase("batal"))
```

jika pengguna mengetik "batal", "Batal", "BATALL" untuk input nama pelanggan maka program langsung berhenti

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

Jawab:

Fungsinya untuk membuat kondisi perulangan selalu benar sehingga menciptakan perulangan tak terbatas (infinite loop) hingga syarat terpenuhi

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

Jawab:

Karena perulangan do selalu dijalankan setidaknya sekali sebelum pengecekan while setelah itu karena kondisi while adalah true maka perulangan akan berjalan terus kecuali jika pengguna memasukkan input "batal" sehingga perintah break akan dieksekusi untuk menghentikan program tersebut

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:
 - 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.

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

```
import java.util.Scanner;

public class Tugas109 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        int pelanggan, jmlTiket = 0, hargaTiket = 50000,
totalTiketTerjual = 0;
        double diskon, totalPenjualan = 0.0, totalBayar = 0.0;

        System.out.println("=====");
        System.out.println("-LAPORAN PENJUALAN TIKET DALAM SEHARI-");
        System.out.println("=====");
        System.out.println("Masukkan banyak pelanggan dalam sehari: ");
        pelanggan = input.nextInt();
        System.out.println("=====");

        int i = 1;
        while(i <= pelanggan){
            System.out.println("pelanggan ke- " + i);
            System.out.println("Jumlah tiket yang dibeli: ");
            jmlTiket = input.nextInt();

            if (jmlTiket < 0){
                System.out.println("angka tidak boleh negatif");
                continue;
            }

            totalBayar = jmlTiket * hargaTiket;
            if (jmlTiket > 4 && jmlTiket <= 10){
                diskon = (1.0 - 0.1);
                totalBayar *= diskon;
            } else if (jmlTiket > 10 ){
                diskon = (1.0 - 0.15);
                totalBayar *= diskon;
            }
        }
    }
}
```

```

        } else {
            diskon = (1.0 - 0.0);
            totalBayar *= diskon;
        }
        i++;
        System.out.println("Harga tiket yang dibayar: " +
totalBayar);
        totalTiketTerjual += jmlTiket;
        totalPenjualan += totalBayar;
    }
    System.out.println("=====");
    System.out.println("Total tiket yang terjual      : " +
totalTiketTerjual);
    System.out.println("Total harga penjualan      : " +
totalPenjualan);
    System.out.println("=====");
    input.close();
}
}

```

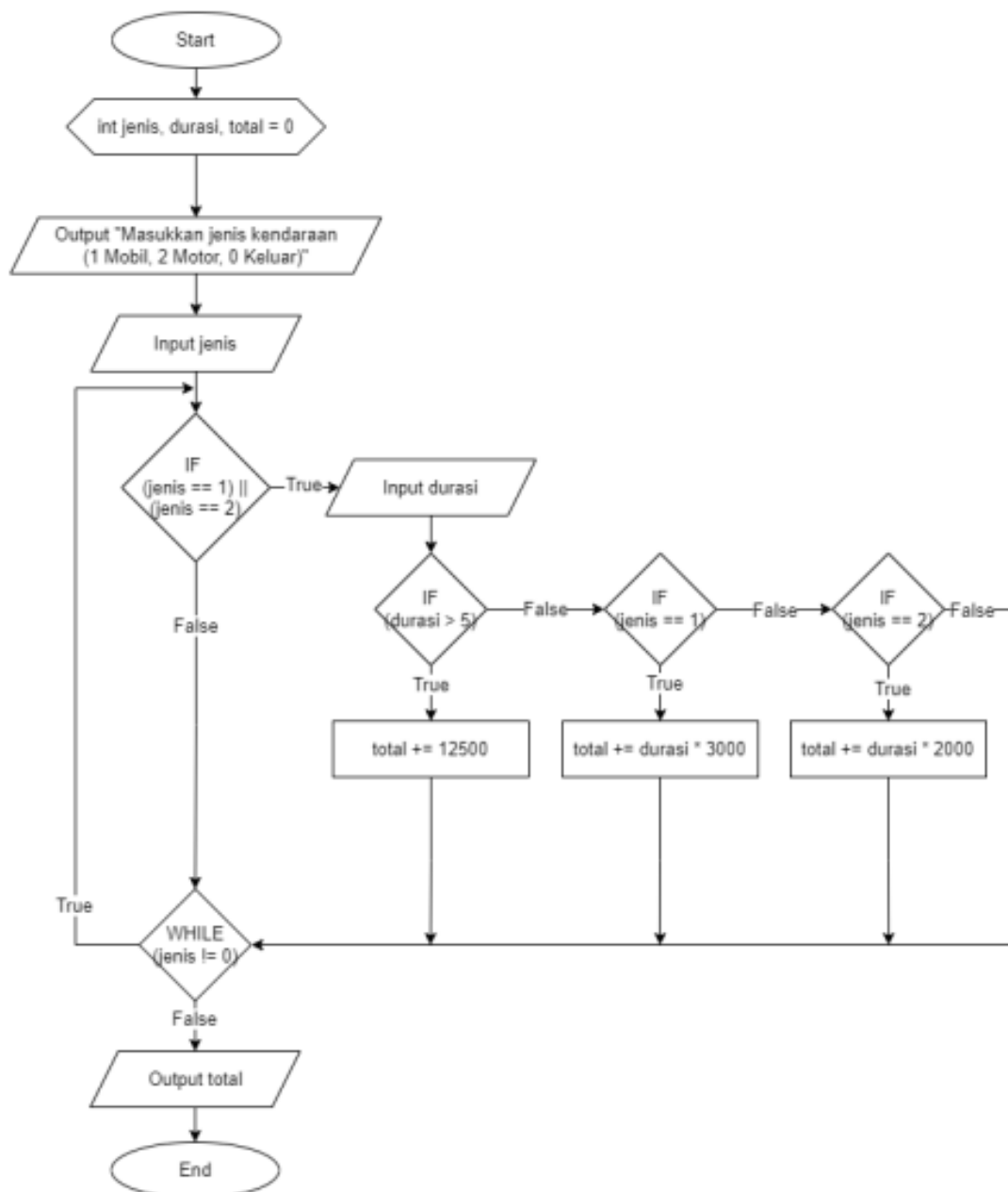
```

=====
-LAPORAN PENJUALAN TIKET DALAM SEHARI-
=====
Masukkan banyak pelanggan dalam sehari:
7
=====
pelanggan ke- 1
Jumlah tiket yang dibeli:
4
Harga tiket yang dibayar: 200000.0
pelanggan ke- 2
Jumlah tiket yang dibeli:
5
Harga tiket yang dibayar: 225000.0
pelanggan ke- 3
Jumlah tiket yang dibeli:
11
Harga tiket yang dibayar: 467500.0
pelanggan ke- 4
Jumlah tiket yang dibeli:
10
Harga tiket yang dibayar: 450000.0
pelanggan ke- 5
Jumlah tiket yang dibeli:
2
pelanggan ke- 6
Jumlah tiket yang dibeli:
3
Harga tiket yang dibayar: 150000.0
pelanggan ke- 7
Jumlah tiket yang dibeli:
4
Harga tiket yang dibayar: 200000.0
=====
Total harga penjualan      : 1792500.0
=====

```

https://github.com/Fizzrrs/Daspro_Jobsheet-7/blob/main/Tugas109.java

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!

```

import java.util.Scanner;

public class Tugas209 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        int jenis, durasi, total = 0;

        do {
            System.out.println("Masukkan jenis kendaraan (1 mobil, 2
motor, 0 keluar)");
            jenis = input.nextInt();

            if (jenis == 1 || jenis == 2){
                System.out.println("Masukkan durasi parkir (jam): ");
                durasi = input.nextInt();

                if (durasi > 5){
                    total += 12500;
                } else {
                    if (jenis == 1){
                        total += durasi * 3000;
                    } else if (jenis == 2){
                        total += durasi*2000;
                    }
                }
            }
        } while (jenis != 0);
        System.out.println("Total pembayaran parkir dari beberapa
kedaraan: " + total);
        input.close();
    }
}

```

```

Masukkan jenis kendaraan (1 mobil, 2 motor, 0 keluar)
1
Masukkan durasi parkir (jam):
4
Masukkan jenis kendaraan (1 mobil, 2 motor, 0 keluar)
2
Masukkan durasi parkir (jam):
2
Masukkan jenis kendaraan (1 mobil, 2 motor, 0 keluar)
1
Masukkan durasi parkir (jam):
7
Masukkan jenis kendaraan (1 mobil, 2 motor, 0 keluar)
0
Total pembayaran parkir dari beberapa kedaraan: 28500

```

https://github.com/Fizzrss/Daspro_Jobsheet-7/blob/main/Tugas209.java