

Nama : Siti Mutmainah

Kelas : 1B

Absen : 21

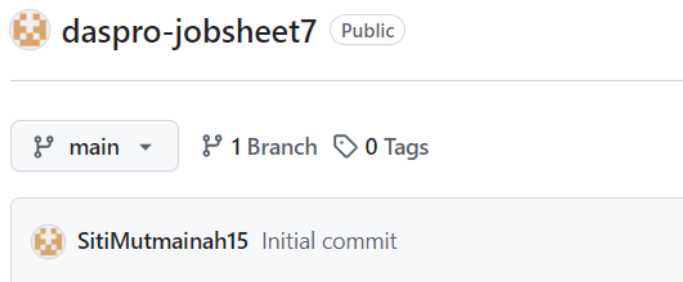
NIM : 244107020143

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

Di dalam Sistem Informasi Akademik (SIAKAD), dosen mengisi nilai mata kuliah Praktikum Dasar Pemrograman yang ditempuh oleh mahasiswa. Dosen tersebut ingin mencari nilai tertinggi dan terendah Kuis dari 10 mahasiswa di dalam satu kelas. Dosen tersebut harus memasukkan nilai dari setiap siswa, kemudian menentukan dan menampilkan nilai tertinggi dan terendah.

Langkah-langkah Percobaan :

1. Buat Repository di GitHub, penamaan disesuaikan.



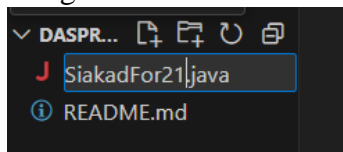
2. Lakukan cloning di terminal.

Dengan ketik **git.clone** (code repository), lalu **cd** (penaruhan file).

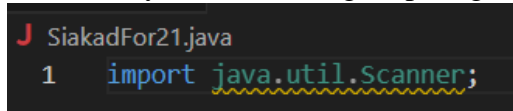
```
PS D:\DASPRO> git clone https://github.com/SitiMutmainah15/daspro-jobsheet7.git
Cloning into 'daspro-jobsheet7'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
PS D:\DASPRO> cd .\daspro-jobsheet7\
```

3. Buka file di VSCode.
4. Buat file baru dengan nama file disesuaikan.

Dengan klik +



5. Buat library Scanner di bagian paling atas (luar) class.



6. Buat struktur dasar program java yang terdiri dari fungsi main().

```

3 public class SiakadFor21 {
    Run | Debug
4     public static void main(String[] args){
5
6     }
7
8 }

```

7. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main().

```

Scanner sc = new Scanner(System.in);
}

```

8. Deklarasikan variabel nilai, tertinggi, dan terendah tipe double. Inisialisasi tertinggi dengan 0 dan terendah dengan 100.

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

9. Buat struktur perulangan for dengan batas kondisi sesuai jumlah mahasiswa yaitu 10. Kode menggunakan perulangan for untuk meminta masukan nilai sebanyak 10 kali.

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

10. Tuliskan kode program untuk menerima input nilai mahasiswa. (tanpa pindah baris)

```
System.out.print("Masukkan nilai mahasiswa ke-" +i+ ": ");
```

11. Pada perulangan for tersebut, tambahkan perintah untuk memasukkan nilai mahasiswa. Kemudian, buat dua kondisi if terpisah untuk mengecek nilai tertinggi dan terendah. Caranya, bandingkan nilai yang dimasukkan dengan variabel nilai tertinggi dan nilai terendah.

```

nilai = sc.nextDouble();

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

```

12. Tampilkan hasil nilai tertinggi dan terendah ke layar dengan perintah (baris baru).

```

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

```

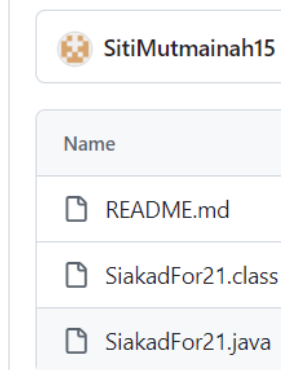
13. Hasil run program.

```

at Coba.main(Coba.java:11)
PS D:\DASPRO\daspro-jobsheet7> cd "d:\DA
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

```

14. Commit dan push ke GitHub.



PERTANYAAN

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

Jawab :

- Inisialisasi (int i = 1)
Bagian ini dijalankan sekali di awal perulangan.
- Kondisi (i <= 10)
Kondisi akan di cek setiap perulangan di jalankan, jika kondisi benar maka kode akan dijalankan, jika kondisi salah maka kode akan berhenti.
- Update/Increment (i++)
Bagian ini digunakan untuk mengubah nilai variable i yang ada dalam blok perulangan, update dilakukan dari awal-akhir.

2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Jawab : Inisialisasi digunakan agar variable mudah diperbarui saat membandingkan dengan nilai baru.

- Inisialisasi tertinggi dengan nilai (0) dan terendah dengan nilai (100) maka hasil akan akurat.
Contoh nya : nilai awal tertinggi = 0, jika ada yang lebih besar dari 0 (misal 80), maka variable akan diperbarui (hasil nilai tersebut 80).

- Inisialisasi tertinggi dengan nilai (100) dan terendah dengan nilai (0) maka hasil akan tidak akurat, karena nilai tertinggi akan tetap (100) dan terendah akan tetap (0), meski masukan data berbeda. Perbandingan tidak akan berfungsi.

Contoh nya : nilai awal tertinggi = 100, dan masukkan nilai (misal 80), maka variable tidak akan diperbarui karena 80 tidak akan pernah lebih besar dari 100 (hasil akan tetap 100)

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

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

Jawab :

- Fungsi kode
Untuk cek (membandingkan & memperbarui) nilai tertinggi dan terendah dari nilai yang dimasukan (program akan mencari nilai tertinggi dan terendah dari semua nilai yang dimasukkan).
 - Alur kerja kode
 - Pengecekan nilai tertinggi
nilai > tertinggi (program akan cek nilai yang baru dimasukkan, jika nilai lebih besar akan disimpan di variable tertinggi), jika kondisi benar (nilai yang baru lebih besar maka akan diperbarui/disimpan menjadi nilai tertinggi), jika kondisi salah (maka tidak akan ada perubahan)
 - Pengecekan nilai terendah
nilai < terendah (program akan cek nilai yang baru dimasukkan, jika nilai lebih kecil akan disimpan di variable terendah), jika kondisi benar (nilai yang baru lebih kecil maka akan diperbarui/disimpan menjadi nilai terendah), jika kondisi salah (maka tidak akan ada perubahan)
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 :

- Modifikasi program

```

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

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;
    }
    if (nilai >= 60) {
        lulus++;
    }
    if (nilai < 60) {
        tidaklulus++;
    }
}
System.out.println("Nilai tertinggi: " + tertinggi);
System.out.println("Nilai terendah: " + terendah);
System.out.println("Jumlah mahasiswa yang lulus: " + lulus);
System.out.println("Jumlah mahasiswa yang tidak lulus: " + tidaklulus);

```

- Tampilkan hasil

```

PS D:\DASPRO\daspro-jobsheet7> cd "d:\l
Masukkan nilai mahasiswa ke-1: 67,5
Masukkan nilai mahasiswa ke-2: 80
Masukkan nilai mahasiswa ke-3: 79,8
Masukkan nilai mahasiswa ke-4: 40
Masukkan nilai mahasiswa ke-5: 92,8
Masukkan nilai mahasiswa ke-6: 100
Masukkan nilai mahasiswa ke-7: 74
Masukkan nilai mahasiswa ke-8: 87
Masukkan nilai mahasiswa ke-9: 81
Masukkan nilai mahasiswa ke-10: 60
Nilai tertinggi: 100.0
Nilai terendah: 40.0
Jumlah mahasiswa yang lulus: 9
Jumlah mahasiswa yang tidak lulus: 1

```

5. Commit dan push ke GitHub.

Jawab :

SitiMutmainah15 Merge branch 'main' of https://github.com/SitiMutmainah15	
README.md	Initial commit
SiakadFor21.class	lulus&tdk
SiakadFor21.java	lulus&tdk

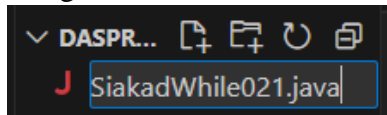
PERCOBAAN 2 : Studi Kasus Nilai Mahasiswa di SIAKAD – Perulangan WHILE

Seorang dosen ingin memasukkan nilai beberapa mahasiswa ke dalam SIAKAD untuk ditentukan kategori nilai hurufnya. Program harus meminta dosen untuk memasukkan nilai setiap mahasiswa. Jika dosen memasukkan nilai yang tidak valid (negatif atau lebih dari 100), program harus mengabaikan input tersebut dan meminta dosen untuk melakukan input ulang. Selanjutnya, nilai yang valid dikelompokkan ke dalam kategori huruf A ($80 < \text{nilai} \leq 100$), B+ ($73 < \text{nilai} \leq 80$), B ($65 < \text{nilai} \leq 73$), C+ ($60 < \text{nilai} \leq 65$), C ($50 < \text{nilai} \leq 60$), D ($39 < \text{nilai} \leq 50$), dan E ($\text{nilai} \leq 39$).

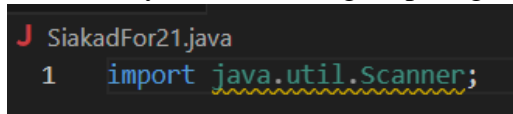
Langkah-langkah Percobaan :

1. Buka file di VSCode.
2. Buat file baru dengan nama file disesuaikan.

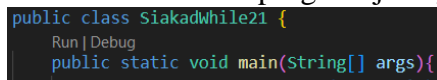
Dengan klik +



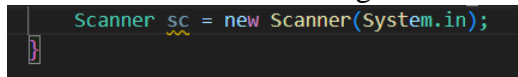
3. Buat library Scanner di bagian paling atas (luar) class.



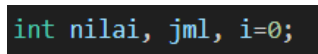
4. Buat struktur dasar program java yang terdiri dari fungsi main().



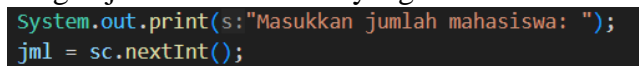
5. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main().



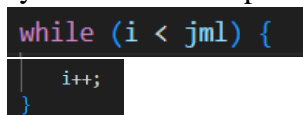
6. Deklarasikan variabel nilai, jml, dan i=0 (untuk nilai awal perulangan) tipe integer.



7. Buatlah kode program untuk meminta masukan jumlah mahasiswa dari pengguna dan simpan ke dalam variabel jml. Dengan begitu, batas perulangan akan dinamis sesuai dengan jumlah mahasiswa yang dimasukkan oleh pengguna melalui keyboard.



8. Buatlah struktur perulangan while dengan batas kondisi sebanyak 5 mahasiswa. Gunakan symbol < karena perulangan dimulai dari nilai 0, bukan 1.



9. Di dalam perulangan while tersebut, tambahkan perintah untuk memasukkan nilai mahasiswa. Selanjutnya, buatlah kondisi if untuk memeriksa apakah nilai yang dimasukkan valid atau tidak, dengan syarat nilai harus berada dalam rentang 0 hingga 100. Kemudian, tambahkan juga kondisi if-else if-else untuk menampilkan kategori nilai

huruf sesuai dengan ketentuan yang ada.



```
System.out.print("Masukkan nilai mahasiswa ke-" + (i + 1) + " : ");
nilai = sc.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");
}
```

10. Hasil run program.

```
Masukkan nilai mahasiswa ke-5 : cd "d:\DASPRO\daspro-j
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
```

11. Commit dan push ke GitHub.

 SiakadWhile21.class	coba1
 SiakadWhile21.java	coba1

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. $nilai < 0 \parallel nilai > 100$
- b. continue

Jawab :

- a. $Nilai < 0 \parallel nilai > 100$

Sintaks ini digunakan untuk memeriksa apakah nilai yang dimasukkan pengguna tidak valid. Nilai < 0 berarti nilai tersebut kurang dari 0, dan nilai > 100 berarti nilai

tersebut lebih dari 100. Tanda || adalah operator "atau" yang berarti jika salah satu dari dua kondisi ini benar, maka keseluruhan kondisi akan dianggap benar (true).

b. Continue

Perintah continue digunakan untuk melewati langkah yang sedang berlangsung dan langsung ke langkah berikutnya dalam perulangan. Ketika program menemukan perintah continue, semua kode yang tersisa untuk iterasi saat ini akan dihentikan, dan program akan kembali ke kondisi perulangan untuk memulai iterasi berikutnya.

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

Jawab : Penulisan i++ di akhir perulangan while memastikan bahwa semua nilai dalam rentang yang diinginkan akan diperiksa dan diproses dengan benar (nilai i akan bertambah setiap kali perulangan). Namun, jika i++ ditulis di awal, hal ini bisa menyebabkan iterasi pertama tidak diperiksa atau perulangan menjadi tidak terhingga.

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

Jawab : 19 kali

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

Jawab :

- Modifikasi Program

```
if (nilai > 80 && nilai <= 100) {  
    System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A");  
    System.out.println(x:"Bagus, pertahankan nilainya!");  
}
```

- Hasil

```
Masukkan nilai mahasiswa ke-2 : 98  
Nilai mahasiswa ke-2 adalah A  
Bagus, pertahankan nilainya!
```

5. Commit dan push kode program ke Github

Jawab :

 SiakadWhile21.class	nilai A
 SiakadWhile21.java	nilai A

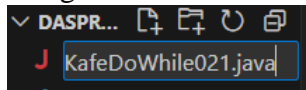
PERCOBAAN 3 : Studi Kasus Transaksi di Kafe – Perulangan DO-WHILE

Di sebuah kafe, kasir ingin memproses transaksi beberapa pelanggan. Pelanggan dapat membeli lebih dari satu item (kopi dengan harga Rp 12.000, teh dengan harga Rp 7.000, dan roti dengan harga Rp 20.000), dan kasir akan terus memasukkan jumlah pembelian untuk setiap pelanggan. Jika ada pelanggan yang memutuskan untuk membatalkan transaksi (dengan memasukkan "batal"), maka kasir akan menghentikan input transaksi dan program berhenti.

Langkah-langkah Percobaan :

1. Buka file di VSCode.
2. Buat file baru dengan nama file disesuaikan.

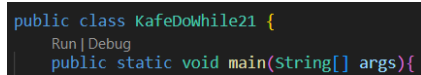
Dengan klik +



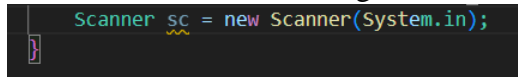
3. Buat library Scanner di bagian paling atas (luar) class.



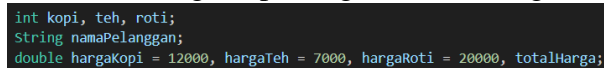
4. Buat struktur dasar program java yang terdiri dari fungsi main().



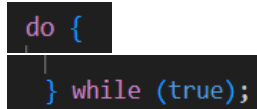
5. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main().



6. Deklarasikan variabel kopi, teh, roti tipe integer (untuk menampung banyaknya item yang dibeli pelanggan). Deklarasikan nama pelanggan tipe string. Deklarasi dan inisialisasi hargaKopi dengan 12000, hargaTeh dengan 7000, hargaRoti dengan 20000.



7. Buat struktur perulangan DO-WHILE dengan kondisi true



8. Di dalam perulangan DO-WHILE tersebut, tambahkan perintah untuk memasukkan nama pelanggan. Kemudian tambahkan kondisi if untuk mengecek isi variabel nama pelanggan. Selanjutnya, tambahkan perintah untuk memasukkan banyaknya item yang dibeli pelanggan untuk setiap menu, apabila masukan nama pelanggan bukan "batal".

```

System.out.print(s:"Masukkan nama pelanggan (ketik 'batal' untuk keluar): ");
namaPelanggan = sc.nextLine();
if (namaPelanggan.equalsIgnoreCase(anotherString:"batal")) {
    System.out.println(x:"Transaksi 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.");
}
}

```

Keterangan: sc.nextLine(); setelah print totalHarga merupakan sintaks untuk membersihkan newline dari buffer



9. Hitung total harga pembelian dan tampilkan hasil run program.

```

PS D:\DASPRO\daspro-jobsheet7> cd "d:\DASPRO\daspro-jobsheet7\" ;
Masukkan nama pelanggan (ketik 'batal' untuk keluar): rena
Jumlah kopi: 3
Jumlah teh: 0
Jumlah roti: 1
Total yang harus dibayar: Rp 56000.0
Masukkan nama pelanggan (ketik 'batal' untuk keluar): yuni
Jumlah kopi: 1
Jumlah teh: 4
Total yang harus dibayar: Rp 80000.0
Masukkan nama pelanggan (ketik 'batal' untuk keluar): batal
Transaksi dibatalkan.
Semua transaksi selesai.

```

10. Commit dan push ke GitHub.

 SiakadWhile21.class	coba1
 SiakadWhile21.java	coba1

PERTANYAAN

1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah “batal”, maka berapa kali perulangan dilakukan?
Jawab : Perulangan akan selalu dijalankan satu kali, karena perintah dalam blok DO akan dieksekusi terlebih dahulu sebelum mengecek kondisi WHILE.
2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!
Jawab : Perulangan berhenti jika nama pelanggan yang dimasukkan adalah batal.
3. Apa fungsi dari penggunaan nilai true pada kondisi DO-WHILE?
Jawab :

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

Jawab : Penggunaan true berfungsi untuk memastikan perulangan tetap berjalan tanpa henti sampai ada kondisi tertentu (seperti break) yang menghentikannya.

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.

Jawab :

- Kode Program.

```
J bioskopTugas.java > bioskopTugas
1 import java.util.Scanner;
2
3 public class bioskopTugas {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7
8         int hargaPertiket = 50000, totalTiket = 0, jumlahTiket;
9         double diskon, totalHarga, totalDiskon, hargaBayar, totalPenjualan = 0;
10
11         while (true) {
12             System.out.print("Masukkan jumlah tiket (ketik 0 untuk selesai): ");
13             jumlahTiket = sc.nextInt();
14
15             if (jumlahTiket == 0) {
16                 break;
17             }
18             if (jumlahTiket < 0) {
19                 System.out.println("Input tidak valid");
20                 continue;
21             }
22             if (jumlahTiket > 10) {
23                 diskon = 0.15;
24             } else if (jumlahTiket > 4) {
25                 diskon = 0.1;
26             } else {
27                 diskon = 0;
28             }
29
30             totalHarga = hargaPertiket * jumlahTiket;
31             totalDiskon = totalHarga * diskon;
32             hargaBayar = totalHarga - totalDiskon;
33
34             totalTiket += jumlahTiket;
35             totalPenjualan += hargaBayar;
36
37             System.out.println("Total harga pengunjung: Rp " + totalHarga);
38             System.out.println("Diskon yang didapat pengunjung: Rp " + totalDiskon);
39             System.out.println("Harga bayar pengunjung: Rp " + hargaBayar);
40
41         }
42
43         System.out.println("Total Tiket Terjual Hari Ini: " + totalTiket);
44         System.out.println("Total Penjualan Tiket Hari Ini : Rp " + totalPenjualan);
45     }
46 }
```

- Hasil run

```

PS D:\DASPRO\daspro-jobsheet> cd "d:\DASPRO\daspro-jobsheet"
Masukkan jumlah tiket (ketik 0 untuk selesai): 4
Total harga pengunjung: Rp 200000.0
Diskon yang didapat pengunjung: Rp 0.0
Harga bayar pengunjung: Rp 200000.0
Masukkan jumlah tiket (ketik 0 untuk selesai): 12
Total harga pengunjung: Rp 600000.0
Diskon yang didapat pengunjung: Rp 90000.0
Harga bayar pengunjung: Rp 510000.0
Masukkan jumlah tiket (ketik 0 untuk selesai): 6
Total harga pengunjung: Rp 300000.0
Diskon yang didapat pengunjung: Rp 30000.0
Harga bayar pengunjung: Rp 270000.0
Masukkan jumlah tiket (ketik 0 untuk selesai): 0
Total Tiket Terjual Hari Ini: 22
Total Penjualan Tiket Hari Ini : Rp 980000.0
PS D:\DASPRO\daspro-jobsheet>

```

- 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!

Jawab :

- Kode Program.

```

pembayaranParkir.java > pembayaranParkir > main(String[])
1  import java.util.Scanner;
2
3  public class pembayaranParkir {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6
7          int durasiParkir, jeniskendaraan, totalPembayaran = 0;
8
9          System.out.println(x:"== SISTEM PEMBAYARAN PARKIR ==");
10
11          while (true) {
12              System.out.println(x:"Masukkan jenis kendaraan anda: ");
13              System.out.println(x:"ketik 1 untuk motor, 2 untuk mobil, 0 untuk keluar");
14              jeniskendaraan = sc.nextInt();
15
16              if (jeniskendaraan == 0) {
17                  break;
18              }
19
20              System.out.print(s:"Masukkan berapa jam parkir anda: ");
21              durasiParkir = sc.nextInt();
22
23              if (jeniskendaraan == 1 || jeniskendaraan == 2) {
24                  if (durasiParkir > 5) {
25                      totalPembayaran += 12500;
26                  } else if (jeniskendaraan == 1) {
27                      totalPembayaran += 2000 * durasiParkir;
28                  } else {
29                      totalPembayaran += 3000 * durasiParkir;
30                  }
31              } else {
32                  System.out.println(x:"Input jenis kendaraan tidak valid. Masukkan lagi.");
33              }
34          }
35          System.out.println(x:"== RINCIAN PEMBAYARAN PARKIR ==");
36          System.out.println("Total pembayaran parkir: Rp " + totalPembayaran);

```

- Hasil run

```

PS D:\DASPRO\daspro-jobsheet> cd "d:\DASPRO\daspro-jobsheet"
== SISTEM PEMBAYARAN PARKIR ==
Masukkan jenis kendaraan anda:
ketik 1 untuk motor, 2 untuk mobil, 0 untuk keluar
1
Masukkan berapa jam parkir anda: 7
Masukkan jenis kendaraan anda:
ketik 1 untuk motor, 2 untuk mobil, 0 untuk keluar
0
== RINCIAN PEMBAYARAN PARKIR ==
Total pembayaran parkir: Rp 12500

```

