

LAPORAN PRAKTIKUM

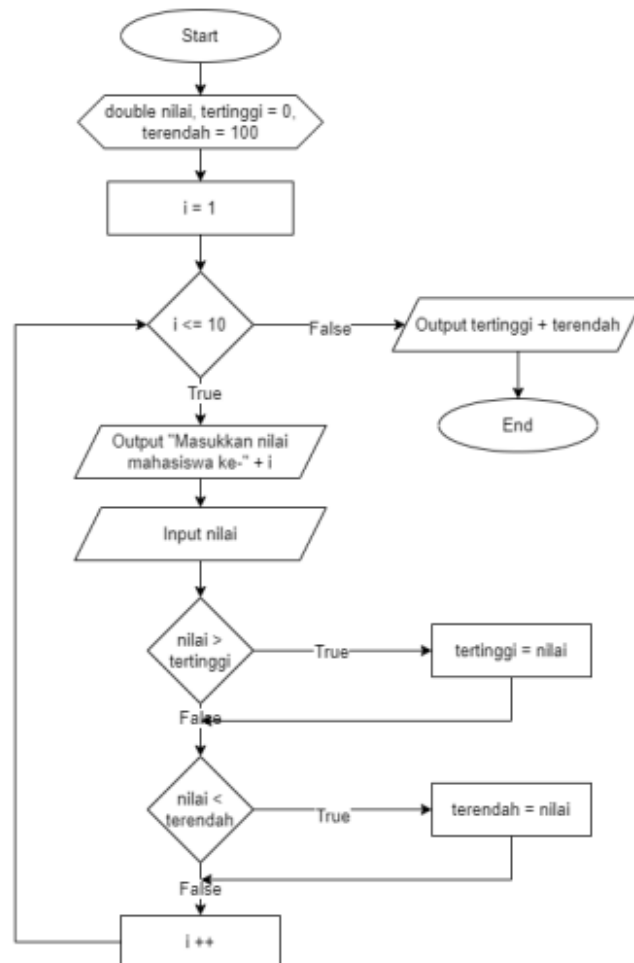
Jobsheet 7

Nama : Sandy kurniawan

NIM : 244107020029

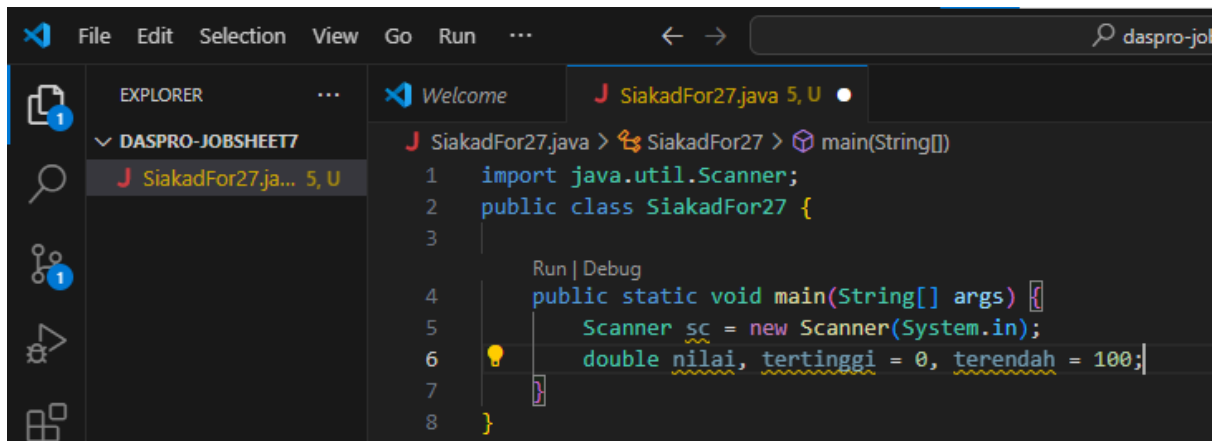
Kelas : 1D (27)

PERCOBAAN1



Berdasarkan flowchart tersebut, buat program menggunakan bahasa pemrograman Java.

1. Buat repository baru pada akun Github Anda, beri nama daspro-jobsheet7
2. Lakukan cloning repository tersebut menggunakan perintah git clone dari terminal
3. Buka folder repository tersebut menggunakan Visual Studio Code
4. Buat file baru, beri nama SiakadForNoAbsen.java
5. Buatlah struktur dasar program Java yang terdiri dari fungsi main()
6. Tambahkan library Scanner di bagian atas (luar) class
7. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main()
8. Deklarasikan variabel nilai, tertinggi, dan terendah bertipe double. Inisialisasi tertinggi dengan 0 dan terendah dengan 100

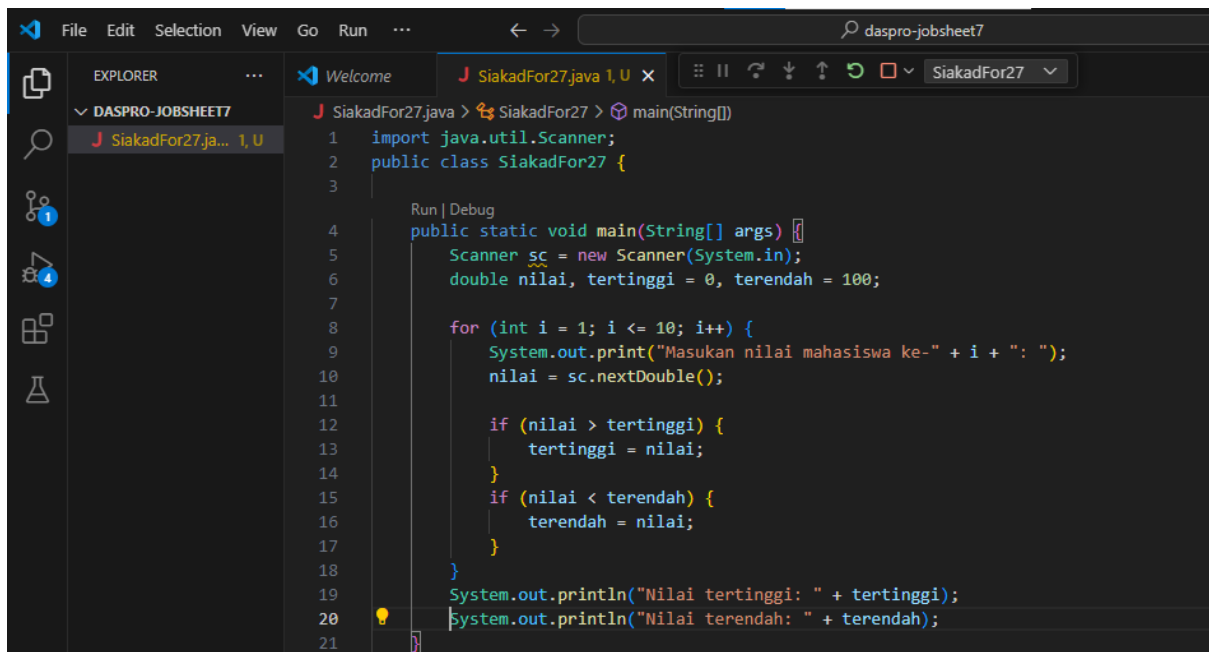


```
1 import java.util.Scanner;
2 public class SiakadFor27 {
3
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         double nilai, tertinggi = 0, terendah = 100;
7     }
8 }
```

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

10. 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

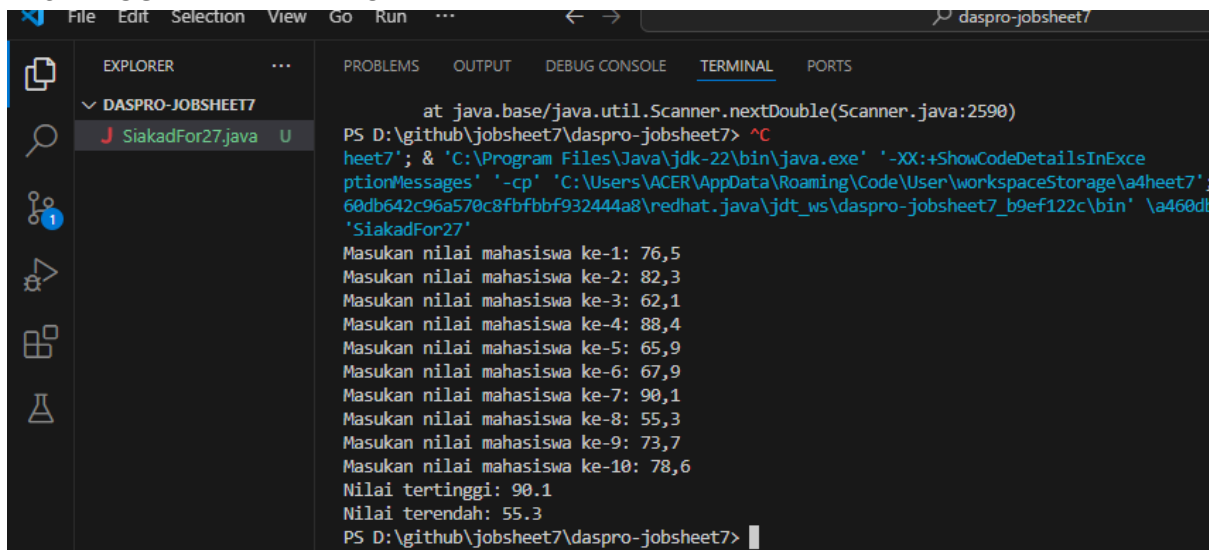
11. Di luar perulangan FOR, tampilkan nilai tertinggi dan terendah



```
1 import java.util.Scanner;
2 public class SiakadFor27 {
3
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         double nilai, tertinggi = 0, terendah = 100;
7
8         for (int i = 1; i <= 10; i++) {
9             System.out.print("Masukan nilai mahasiswa ke-" + i + ": ");
10            nilai = sc.nextDouble();
11
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    }
22 }
```

12. Compile dan run program

HASIL PROGRAM KETIKA DI RUN



```
at java.base/java.util.Scanner.nextDouble(Scanner.java:2590)
PS D:\github\jobsheet7\daspro-jobsheet7> ^C
heet7'; & 'C:\Program Files\Java\jdk-22\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\ACER\AppData\Roaming\Code\User\workspaceStorage\44heet7\60db642c96a570c8fbfbf932444a8\redhat.java\jdt_ws\daspro-jobsheet7_b9ef122c\bin' \a460db642c96a570c8fbfbf932444a8\redhat.java\jdt_ws\daspro-jobsheet7_b9ef122c\bin' 'SiakadFor27'
Masukan nilai mahasiswa ke-1: 76,5
Masukan nilai mahasiswa ke-2: 82,3
Masukan nilai mahasiswa ke-3: 62,1
Masukan nilai mahasiswa ke-4: 88,4
Masukan nilai mahasiswa ke-5: 65,9
Masukan nilai mahasiswa ke-6: 67,9
Masukan nilai mahasiswa ke-7: 90,1
Masukan nilai mahasiswa ke-8: 55,3
Masukan nilai mahasiswa ke-9: 73,7
Masukan nilai mahasiswa ke-10: 78,6
Nilai tertinggi: 90.1
Nilai terendah: 55.3
PS D:\github\jobsheet7\daspro-jobsheet7>
```

13. Commit dan push kode program ke Github

PERTANYAAN

1. Sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program Percobaan 1!
2. Mengapa variabel **tertinggi** diinisialisasi 0 dan **terendah** diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?
3. Jelaskan fungsi dan alur kerja dari potongan kode berikut!

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

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!
5. **Commit dan push kode program ke Github**

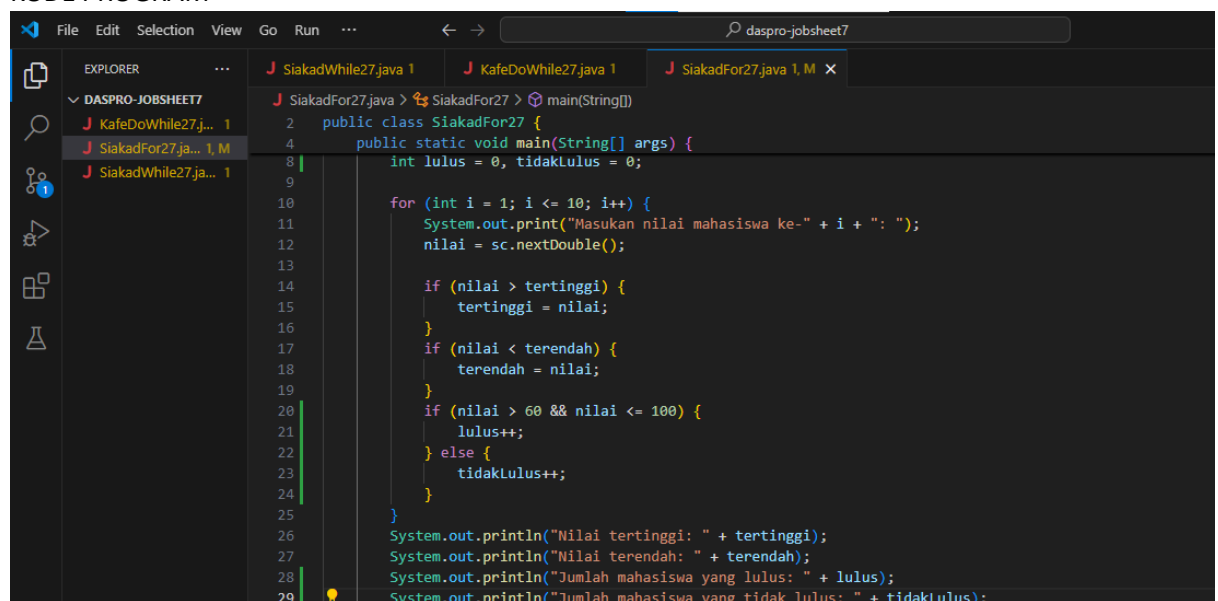
Jawaban:

1. Adapun komponen perulangan FOR pada kode program percobaan 1, yaitu:
 1. Inisialisasi
for (int i = 1; i <= 10; i++) {
 2. Kondisi
i <= 10
 3. Increment/Decrement
i++
2. **Nilai tertinggi diinisialisasi dengan 0** karena kita ingin menemukan nilai maksimum dari masukan yang akan diberikan, Sedangkan Variabel terendah diinisialisasi dengan 100 sehingga setiap nilai yang dimasukkan akan lebih kecil dari terendah, dan karenanya variabel ini akan diperbarui dengan nilai yang dimasukkan.

Dan jika nilai tertinggi di inisialisasi dengan 100, semua nilai yang dimasukkan adalah di bawah 100, maka tertinggi tidak akan pernah diperbarui. Artinya, program tidak akan memberikan nilai tertinggi yang benar. Sedangkan jika nilai terendah di inisialisasi dengan 0, semua nilai yang dimasukkan adalah positif, maka terendah tidak akan pernah diperbarui. Ini berarti bahwa program tidak akan dapat menunjukkan nilai terendah yang benar jika semua nilai positif.

3. **Kode program if (nilai > tertinggi) { tertinggi = nilai;** Berfungsi Memeriksa apakah nilai yang baru dimasukkan (nilai) lebih besar dari nilai tertinggi yang telah ditemukan sejauh ini (tertinggi). **Alur kerjanya** Jika kondisi nilai > tertinggi bernilai true, artinya nilai yang dimasukkan lebih besar dari nilai tertinggi yang sebelumnya. Maka, program akan memperbarui nilai tertinggi dengan nilai baru ini, jika tidak maka tertinggi tetap tidak berubah. **Sedangkan kode program if (nilai < terendah) { terendah = nilai;** Berfungsi Memeriksa apakah nilai yang baru dimasukkan (nilai) lebih kecil dari nilai terendah yang telah ditemukan sejauh ini (terendah). **Alur kerjanya** Jika kondisi nilai < terendah bernilai true, artinya nilai yang dimasukkan lebih kecil dari nilai terendah yang sebelumnya. Maka, program akan memperbarui nilai terendah dengan nilai baru ini, jika tidak maka terendah tetap tidak berubah.

4. KODE PROGRAM



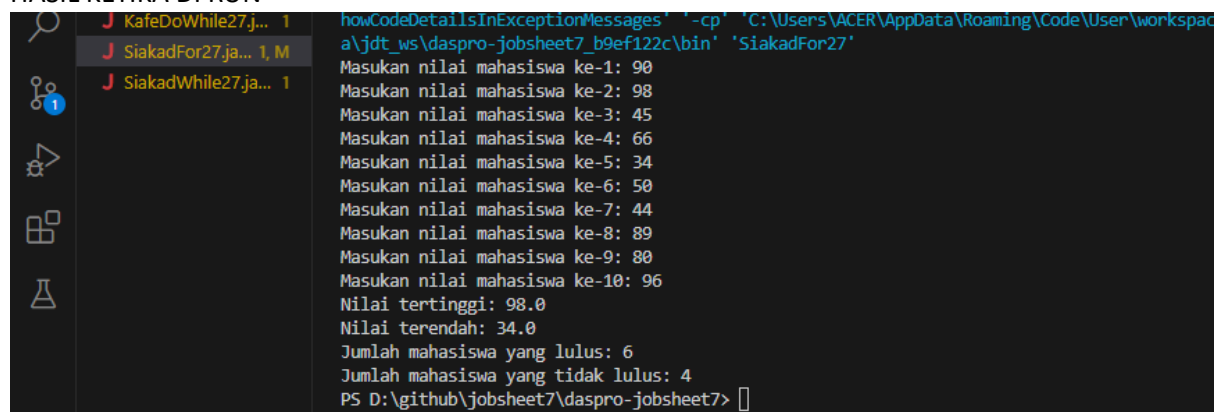
```

File Edit Selection View Go Run ...
daspro-jobsheet7

EXPLORER
DASPRO-JOBSHEET7
  KafedWhile27.j... 1
  SiakadFor27.ja... 1, M
  SiakadWhile27.ja... 1

SiakadFor27.java > SiakadFor27 > main(String[])
1 public class SiakadFor27 {
2     public static void main(String[] args) {
3         int lulus = 0, tidakLulus = 0;
4
5         for (int i = 1; i <= 10; i++) {
6             System.out.print("Masukan nilai mahasiswa ke-" + i + ": ");
7             nilai = sc.nextDouble();
8
9             if (nilai > tertinggi) {
10                 tertinggi = nilai;
11             }
12             if (nilai < terendah) {
13                 terendah = nilai;
14             }
15             if (nilai > 60 && nilai <= 100) {
16                 lulus++;
17             } else {
18                 tidakLulus++;
19             }
20         }
21         System.out.println("Nilai tertinggi: " + tertinggi);
22         System.out.println("Nilai terendah: " + terendah);
23         System.out.println("Jumlah mahasiswa yang lulus: " + lulus);
24         System.out.println("Jumlah mahasiswa yang tidak lulus: " + tidakLulus);
25     }
26 }
  
```

HASIL KETIKA DI RUN

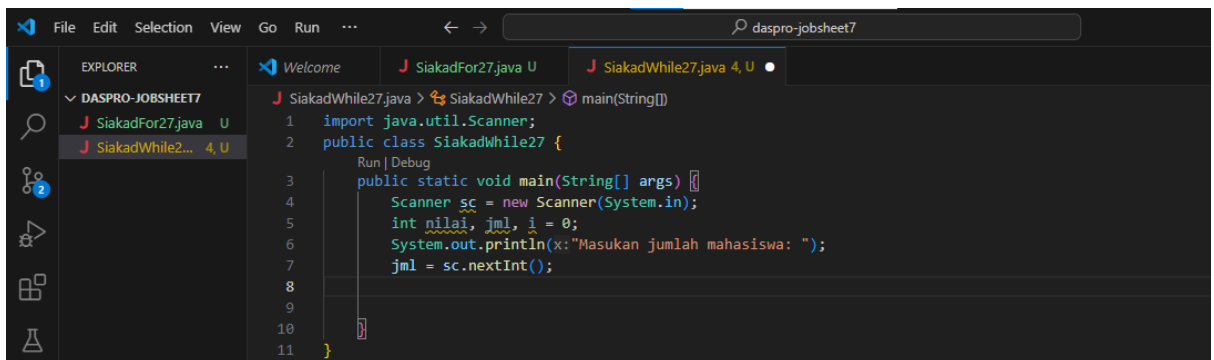


```

howCodeDetailsInExceptionMessages' -cp 'C:\Users\ACER\AppData\Roaming\Code\User\workspac
a\jdt_ws\daspro-jobsheet7_b9ef122c\bin' 'SiakadFor27'
Masukan nilai mahasiswa ke-1: 90
Masukan nilai mahasiswa ke-2: 98
Masukan nilai mahasiswa ke-3: 45
Masukan nilai mahasiswa ke-4: 66
Masukan nilai mahasiswa ke-5: 34
Masukan nilai mahasiswa ke-6: 50
Masukan nilai mahasiswa ke-7: 44
Masukan nilai mahasiswa ke-8: 89
Masukan nilai mahasiswa ke-9: 80
Masukan nilai mahasiswa ke-10: 96
Nilai tertinggi: 98.0
Nilai terendah: 34.0
Jumlah mahasiswa yang lulus: 6
Jumlah mahasiswa yang tidak lulus: 4
PS D:\github\jobsheet7\daspro-jobsheet7>
  
```

PERCOBAAN2

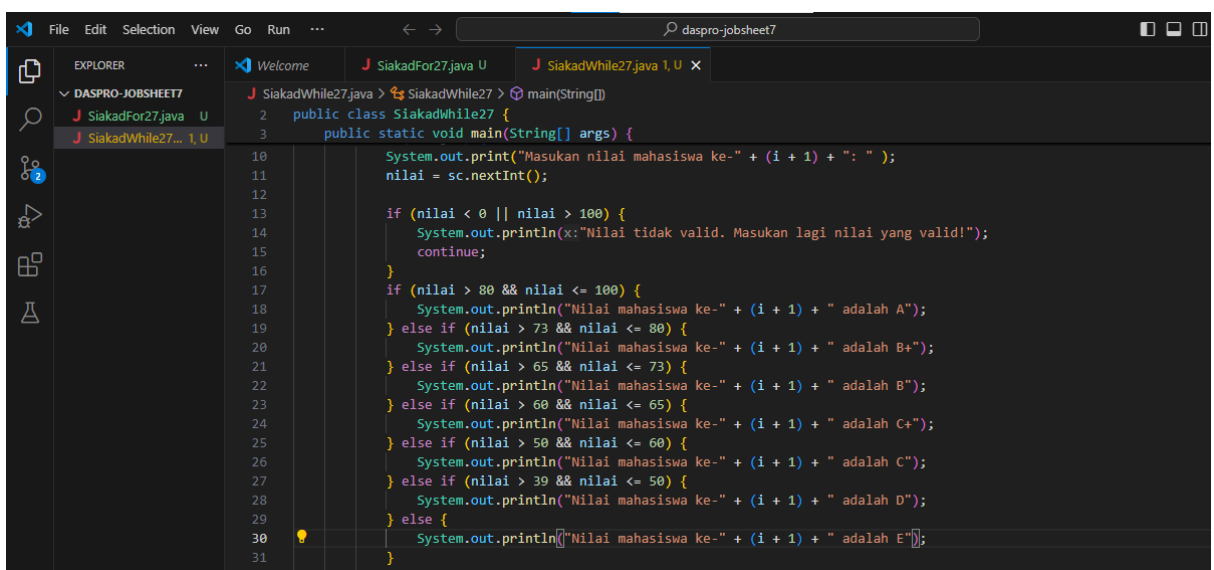
1. Buat file baru, beri nama SiakadWhileNoAbsen.java
2. Buatlah struktur dasar program Java yang terdiri dari fungsi main().
3. Tambahkan library Scanner di bagian atas (luar) class
4. Buat deklarasi Scanner dengan nama variabel sc di dalam fungsi main()
5. Deklarasikan variabel nilai, jml, dan i (untuk 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.



```
1 import java.util.Scanner;
2 public class SiakadWhile27 {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         int nilai, jml, i = 0;
6         System.out.println("Masukan jumlah mahasiswa: ");
7         jml = sc.nextInt();
8     }
9 }
10
11 }
```

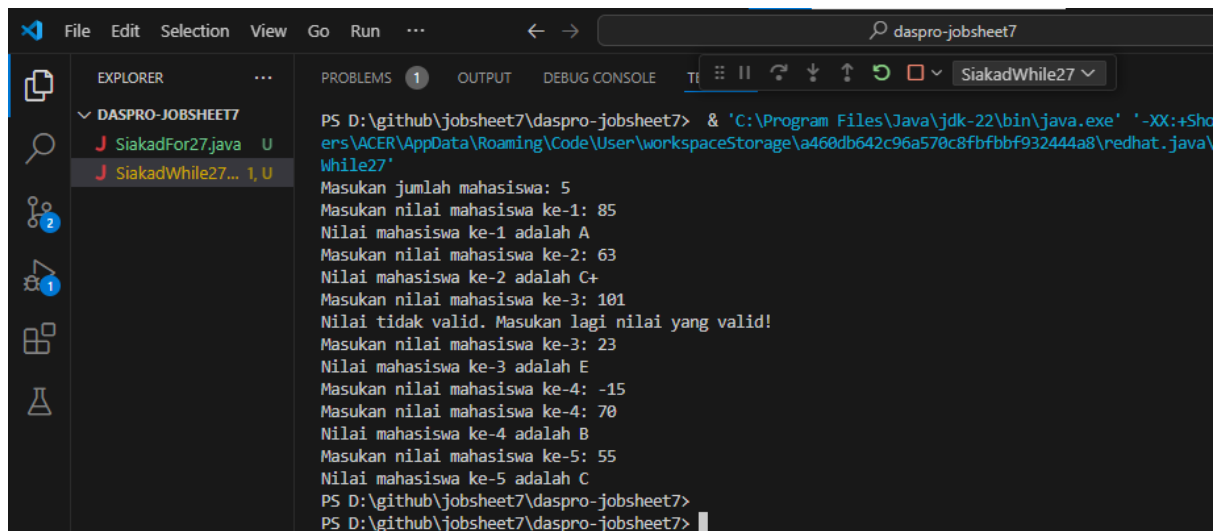
7. Buat struktur perulangan WHILE dengan batas kondisi sesuai jumlah mahasiswa yaitu 5. Perhatikan simbol yang digunakan adalah < karena perulangan variabel i dimulai dari 0, bukan 1

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.



```
1 public class SiakadWhile27 {
2     public static void main(String[] args) {
3         System.out.print("Masukan nilai mahasiswa ke-" + (i + 1) + ": ");
4         nilai = sc.nextInt();
5
6         if (nilai < 0 || nilai > 100) {
7             System.out.println("Nilai tidak valid. Masukan lagi nilai yang valid!");
8             continue;
9         }
10        if (nilai > 80 && nilai <= 100) {
11            System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A");
12        } else if (nilai > 73 && nilai <= 80) {
13            System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B+");
14        } else if (nilai > 65 && nilai <= 73) {
15            System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B");
16        } else if (nilai > 60 && nilai <= 65) {
17            System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C+");
18        } else if (nilai > 50 && nilai <= 60) {
19            System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C");
20        } else if (nilai > 39 && nilai <= 50) {
21            System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah D");
22        } else {
23            System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah E");
24        }
25        i++;
26    }
27 }
```

HASIL PROGRAM KETIKA DI RUN



```
PS D:\github\jobsheet7\daspro-jobsheet7> & 'C:\Program Files\Java\jdk-22\bin\java.exe' '-XX:+Sho
ers\ACER\AppData\Roaming\Code\User\workspaceStorage\460db642c96a570c8fbfbf932444a8\redhat.java\
While27'
Masukan jumlah mahasiswa: 5
Masukan nilai mahasiswa ke-1: 85
Nilai mahasiswa ke-1 adalah A
Masukan nilai mahasiswa ke-2: 63
Nilai mahasiswa ke-2 adalah C+
Masukan nilai mahasiswa ke-3: 101
Nilai tidak valid. Masukkan lagi nilai yang valid!
Masukan nilai mahasiswa ke-3: 23
Nilai mahasiswa ke-3 adalah E
Masukan nilai mahasiswa ke-4: -15
Masukan nilai mahasiswa ke-4: 70
Nilai mahasiswa ke-4 adalah B
Masukan nilai mahasiswa ke-5: 55
Nilai mahasiswa ke-5 adalah C
PS D:\github\jobsheet7\daspro-jobsheet7>
PS D:\github\jobsheet7\daspro-jobsheet7>
```

9. Compile dan run program

10. Commit dan push kode program ke Github

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`
 - b. `continue`
2. Mengapa sintaks `i++` dituliskan di akhir perulangan WHILE? Apa yang terjadi jika posisinya dituliskan di awal perulangan WHILE?
3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan WHILE akan berjalan?
4. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya"!
5. **Commit dan push kode program ke Github**

Jawaban:

1.
 - a. **Maksud** dari Sintaks `if (nilai < 0 || nilai > 100)` memeriksa apakah variabel nilai berada di luar rentang yang valid, yaitu kurang dari 0 atau lebih dari 100 dan juga Operator `||` adalah operator logika "atau", yang berarti jika salah satu dari kondisi tersebut benar, maka blok kode di dalam if akan dieksekusi. **Kegunaan** dari kode tersebut untuk memvalidasi input dari pengguna. Jika nilai tidak valid (kurang dari 0 atau lebih dari 100), maka program akan menampilkan pesan peringatan.
 - b. **Dengan menggunakan continue**, program akan melanjutkan ke iterasi berikutnya dalam loop (misalnya, loop while atau for). **Kegunaannya** memungkinkan pengguna untuk memasukkan nilai yang valid tanpa melanjutkan ke bagian program yang lain.
2. **Kode `i++` berada di akhir perulangan** ini berarti bahwa indeks `i` hanya akan bertambah setelah input nilai yang valid telah diterima dan diproses. Jadi, jika nilai yang dimasukkan

tidak valid (kurang dari 0 atau lebih dari 100), perulangan akan meminta input lagi tanpa mengubah *i*, **jika memindahkan *i++* ke awal**, bisa terjadi situasi di mana program terus meminta input untuk mahasiswa yang sama (karena nilai tidak valid) tetapi tidak pernah maju ke mahasiswa berikutnya.

3. Jika jumlah mahasiswa yang dimasukkan adalah 19, perulangan while akan berjalan sebanyak 19 kali, asalkan setiap input nilai yang dimasukkan untuk setiap mahasiswa adalah valid, dalam percobaan 2 ini tidak ada Batasan jumlah mahasiswa untuk di input

4. KODE PROGRAM

```
SiakadWhile27.java > SiakadWhile27 > main(String[])
2 public class SiakadWhile27 {
3     public static void main(String[] args) {
11         nilai = sc.nextInt();
12
13         if (nilai < 0 || nilai > 100) {
14             System.out.println("Nilai tidak valid. Masukkan lagi nilai yang valid!");
15             continue;
16         }
17         if (nilai > 80 && nilai <= 100) {
18             System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A");
19             System.out.println("Bagus, pertahankan nilainya!");
20         } else if (nilai > 73 && nilai <= 80) {
21             System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B+");
22         } else if (nilai > 65 && nilai <= 73) {
23             System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B");
24         } else if (nilai > 60 && nilai <= 65) {
25             System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C");
26         }
```

HASIL PROGRAM KETIKA DI RUN

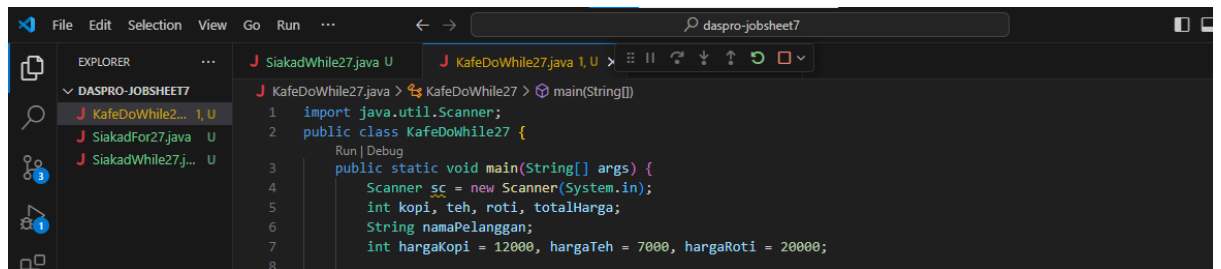
```
File Edit Selection View Go Run ...
EXPLORER
DASPRO-JOBSHEET7
SiakadFor27.java U
SiakadWhile27... 1, U

PROBLEMS 1 OUTPUT DEBUG CONSOLE
PS D:\github\jobsheet7\daspro-jobsheet7> & 'C:\Program Files\Java\jdk-22\bin\java.exe' '-XX:+ShowCodeDetailsIn
ers\ACER\AppData\Roaming\Code\User\workspaceStorage\4660db642c96a570c8fbfbf932444a8\redhat.java\jdt_ws\daspro-
While27'
Masukan jumlah mahasiswa: 90
Masukan nilai mahasiswa ke-1: 90
Nilai mahasiswa ke-1 adalah A
Bagus, pertahankan nilainya!
Masukan nilai mahasiswa ke-2: 23
Nilai mahasiswa ke-2 adalah E
Masukan nilai mahasiswa ke-3: 89
Nilai mahasiswa ke-3 adalah A
Bagus, pertahankan nilainya!
Masukan nilai mahasiswa ke-4: 80
Nilai mahasiswa ke-4 adalah B+
Masukan nilai mahasiswa ke-5: 81
Nilai mahasiswa ke-5 adalah A
Bagus, pertahankan nilainya!
Masukan nilai mahasiswa ke-6: []
```

5. Commit dan push kode program ke GitHub!

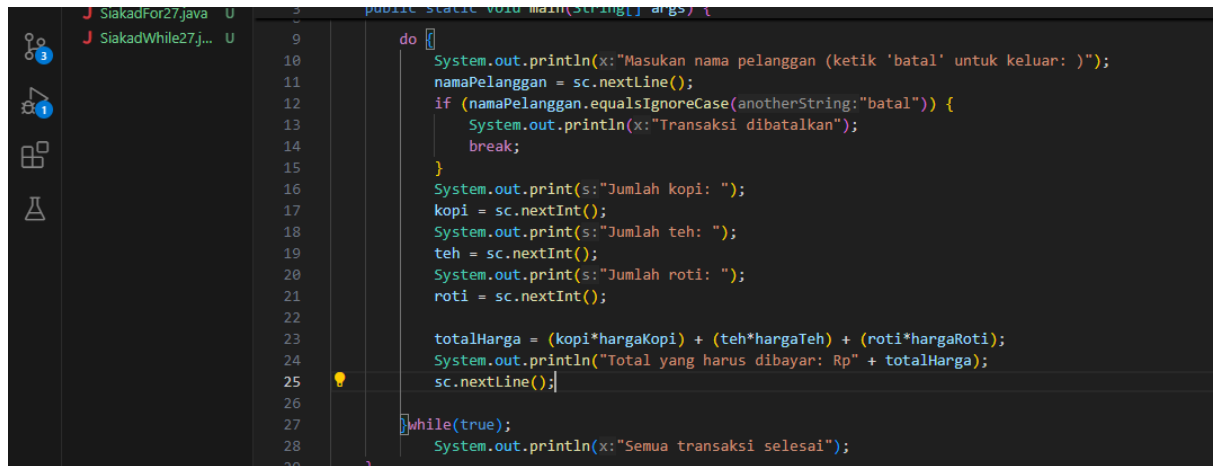
PERCOBAAN3

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



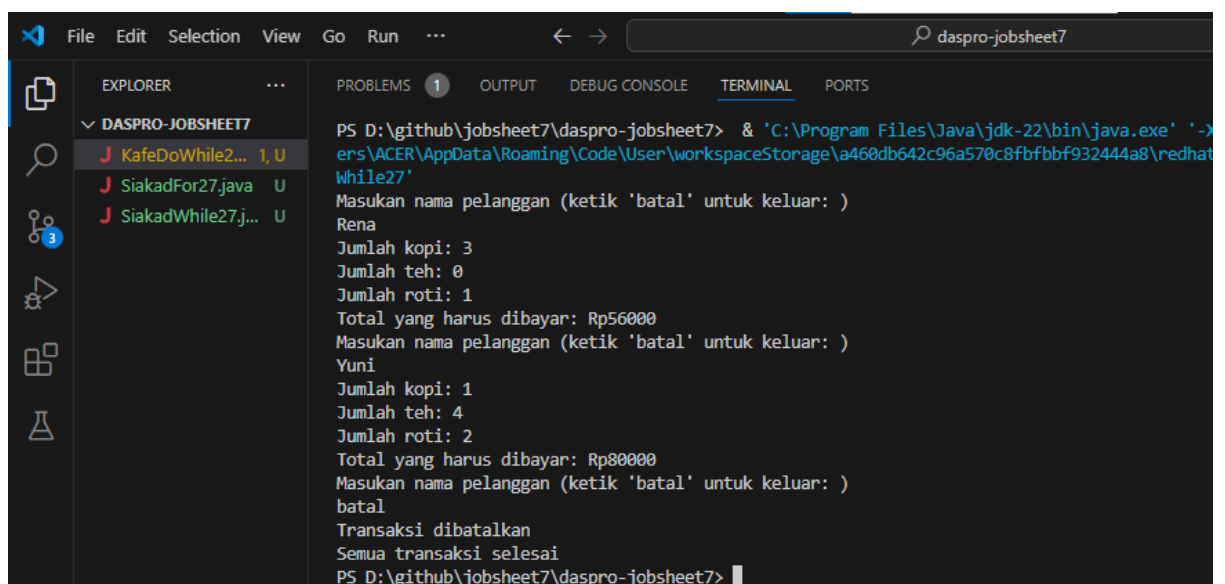
```
1 import java.util.Scanner;
2 public class KafeDoWhile27 {
3     Run | Debug
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         int kopi, teh, roti, totalHarga;
7         String namaPelanggan;
8         int hargaKopi = 12000, hargaTeh = 7000, hargaRoti = 20000;
```

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



```
9     do {
10         System.out.println(x:"Masukan nama pelanggan (ketik 'batal' untuk keluar: )");
11         namaPelanggan = sc.nextLine();
12         if (namaPelanggan.equalsIgnoreCase(anotherString:"batal")) {
13             System.out.println(x:"Transaksi dibatalkan");
14             break;
15         }
16         System.out.print(s:"Jumlah kopi: ");
17         kopi = sc.nextInt();
18         System.out.print(s:"Jumlah teh: ");
19         teh = sc.nextInt();
20         System.out.print(s:"Jumlah roti: ");
21         roti = sc.nextInt();
22
23         totalHarga = (kopi*hargaKopi) + (teh*hargaTeh) + (roti*hargaRoti);
24         System.out.println("Total yang harus dibayar: Rp" + totalHarga);
25         sc.nextLine();
26     } while(true);
27     System.out.println(x:"Semua transaksi selesai");
28 }
```

HASIL PROGRAM KETIKA DI RUN



```
PS D:\github\jobsheet7\daspro-jobsheet7> & 'C:\Program Files\Java\jdk-22\bin\java.exe' '-X
ers\ACER\AppData\Roaming\Code\User\workspaceStorage\4a60db642c96a570c8fbfbf932444a8\redhat
While27'
Masukan nama pelanggan (ketik 'batal' untuk keluar: )
Rena
Jumlah kopi: 3
Jumlah teh: 0
Jumlah roti: 1
Total yang harus dibayar: Rp56000
Masukan nama pelanggan (ketik 'batal' untuk keluar: )
Yuni
Jumlah kopi: 1
Jumlah teh: 4
Jumlah roti: 2
Total yang harus dibayar: Rp80000
Masukan nama pelanggan (ketik 'batal' untuk keluar: )
batal
Transaksi dibatalkan
Semua transaksi selesai
PS D:\github\jobsheet7\daspro-jobsheet7>
```


PERTANYAAN

1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah "batal", maka berapa kali perulangan dilakukan?
2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!
3. Apa fungsi dari penggunaan nilai **true** pada kondisi DO-WHILE?
4. Mengapa perulangan DO-WHILE tersebut tetap berjalan meskipun tidak ada komponen inisialisasi dan update?

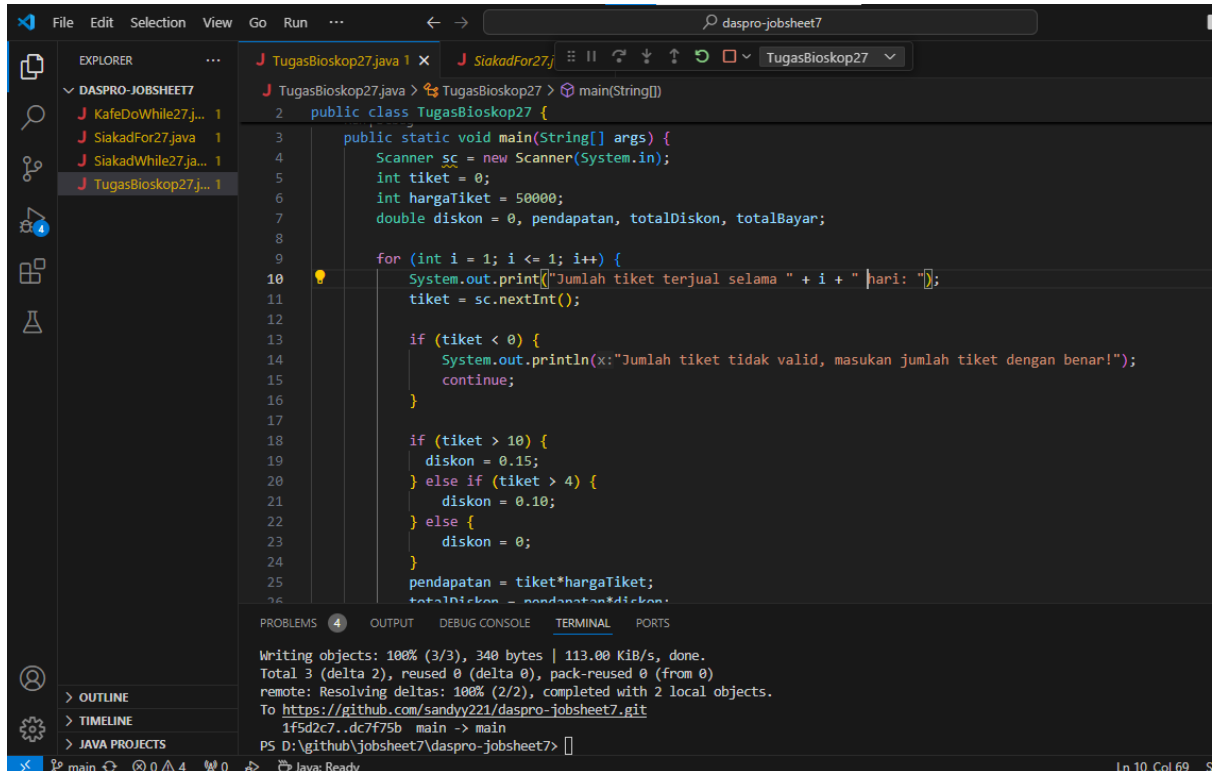
Jawaban:

1. Dalam penggunaan do while, jika nama pelanggan yang dimasukkan pertama kali adalah "batal", maka perulangan akan dilakukan satu kali, Ini karena pada do while, blok kode di dalamnya akan dijalankan setidaknya sekali sebelum memeriksa kondisi. meskipun "batal" adalah input yang tidak diinginkan, pernyataan di dalam do tetap akan dieksekusi sebelum kondisi di while
2. Kondisi dari kode program **if (namaPelanggan.equalsIgnoreCase("batal")) {
System.out.println("Transaksi dibatalkan");
break;**
Ini berarti perulangan akan terus berlangsung selama nilai namaPelanggan tidak sama dengan "batal". Perulangan akan berhenti ketika pengguna memasukkan "batal" sebagai nama pelanggan
3. Menggunakan nilai true pada kondisi do while membuat perulangan menjadi infinite loop (perulangan tak terbatas)
4. Perulangan do while tetap berjalan meskipun tidak ada komponen inisialisasi dan update karena cara kerja perulangan tersebut berfokus pada eksekusi blok kode terlebih dahulu sebelum memeriksa kondisi

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.

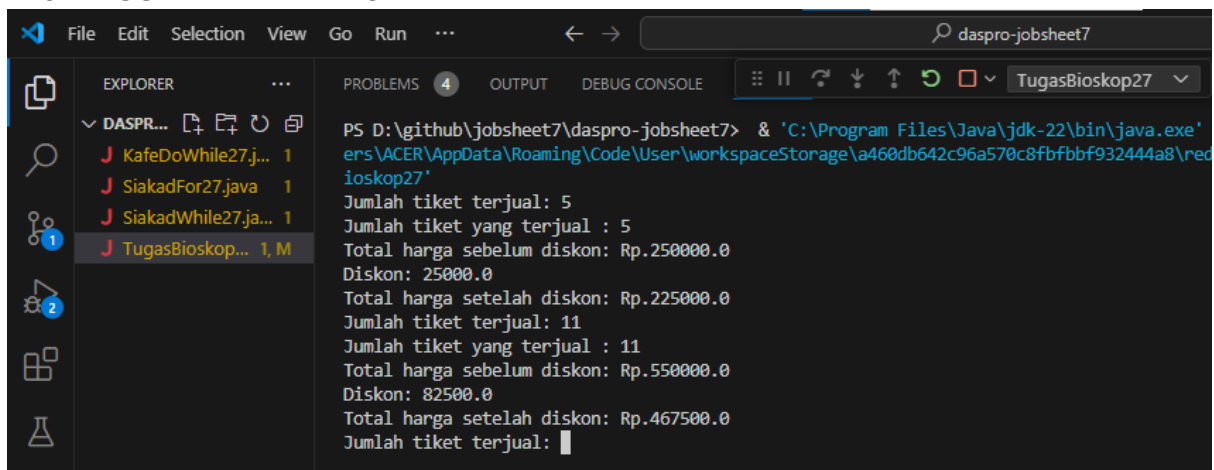
KODE PROGRAM



```
1  TugasBioskop27.java > Tugaskop27 > main(String[])
2  public class Tugaskop27 {
3
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6          int tiket = 0;
7          int hargaTiket = 50000;
8          double diskon = 0, pendapatan, totalDiskon, totalBayar;
9
10         for (int i = 1; i <= 1; i++) {
11             System.out.print("Jumlah tiket terjual selama " + i + " hari: ");
12             tiket = sc.nextInt();
13
14             if (tiket < 0) {
15                 System.out.println(x:"Jumlah tiket tidak valid, masukan jumlah tiket dengan benar!");
16                 continue;
17             }
18
19             if (tiket > 10) {
20                 diskon = 0.15;
21             } else if (tiket > 4) {
22                 diskon = 0.10;
23             } else {
24                 diskon = 0;
25             }
26             pendapatan = tiket*hargaTiket;
27             totalDiskon = pendapatan*diskon;
```

Writing objects: 100% (3/3), 340 bytes | 113.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To <https://github.com/sandyv221/daspro-jobsheet7.git>
1f5d2c7..dc7f75b main -> main
PS D:\github\jobsheet7\daspro-jobsheet7>

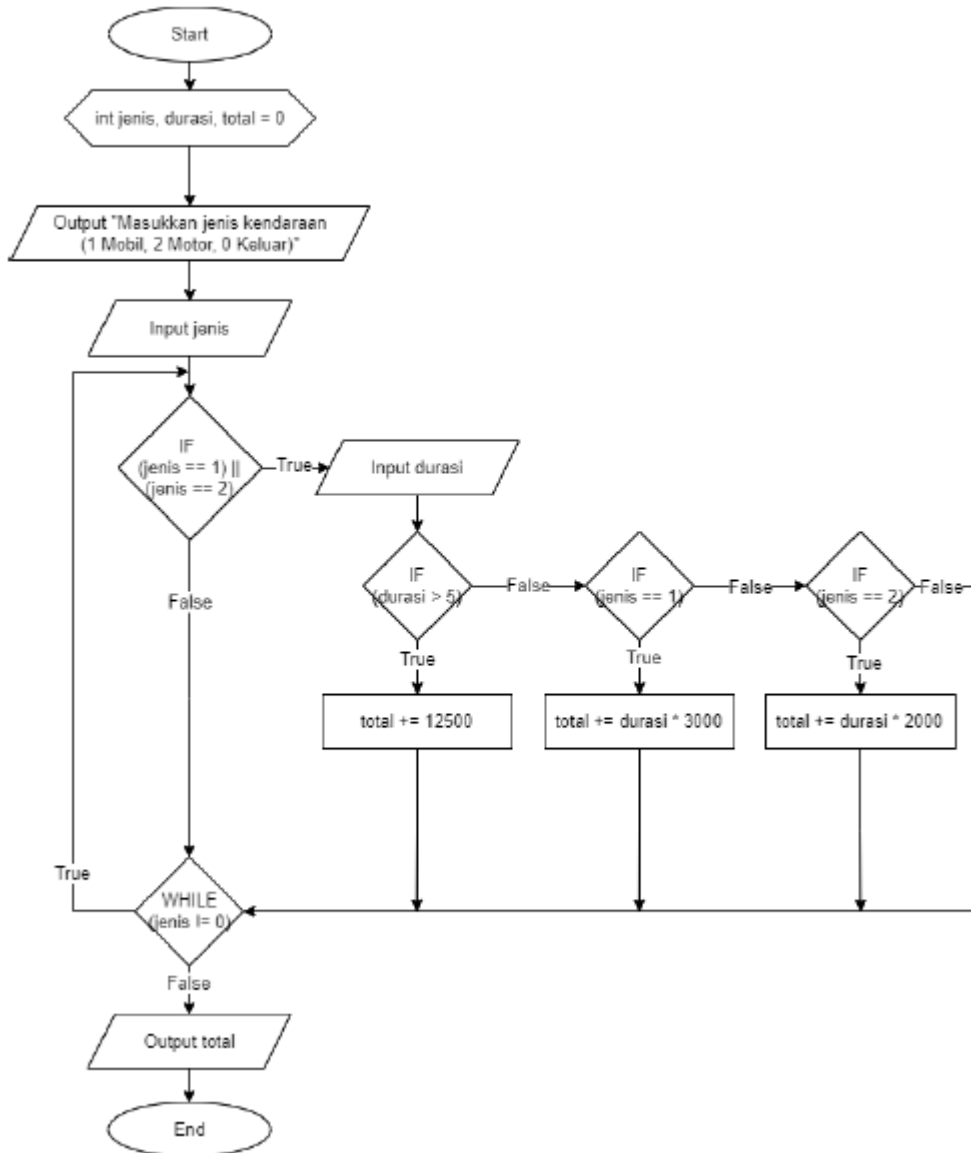
HASIL PROGRAM KETIKA DI RUN



```
PS D:\github\jobsheet7\daspro-jobsheet7> & 'C:\Program Files\Java\jdk-22\bin\java.exe'
ers\ACER\AppData\Roaming\Code\User\workspaceStorage\460db642c96a570c8fbfbf932444a8\red
ioskop27'
Jumlah tiket terjual: 5
Jumlah tiket yang terjual : 5
Total harga sebelum diskon: Rp.250000.0
Diskon: 25000.0
Total harga setelah diskon: Rp.225000.0
Jumlah tiket terjual: 11
Jumlah tiket yang terjual : 11
Total harga sebelum diskon: Rp.550000.0
Diskon: 82500.0
Total harga setelah diskon: Rp.467500.0
Jumlah tiket terjual: 
```

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!

The screenshot shows an IDE with a file explorer on the left and a code editor on the right. The file explorer lists several files, including 'DASPRO-JOBSHEET7' and 'TugasParkir27.java'. The code editor displays the following Java code:

```

1  TugasParkir27.java > 1  KafeDoWhile27.java 1
2  TugasParkir27.java > 2  main(String[])
3  public class TugasParkir27 {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6          int jenis;
7          int durasi;
8          int total;
9          jenis = sc.nextInt();
10         while (jenis != 0) {
11             if (jenis == 1 || jenis == 2) {
12                 System.out.print(s:"Masukan durasi parkir(jam): ");
13                 durasi = sc.nextInt();
14                 if (durasi > 5) {
15                     total += 12500;
16                     System.out.println("Total pembayaran: " + total);
17                 } else {
18                     if (jenis == 1) {
19                         total = durasi*3000;
20                     } else if (jenis == 2) {
21                         total = durasi*2000;
22                     }
23                     System.out.println("Total pembayaran: " + total);
24                 }
25             } else if (jenis != 0) {
26                 System.out.println(x:"Jenis kendaraan tidak valid, masukan jenis kendaraan dengan benar!");
27             }
28         } while (jenis != 0);
29         System.out.println("Total pembayaran: " + total);
30     }
}

```

File Edit Selection View Go Run ...

daspro-jobsheet7

EXPLORER

DASPRO-JOBSHEET7

- J KafeDoWhile27.j... 1
- J SiakadFor27.java 1
- J SiakadWhile27.ja... 1
- J TugasBioskop27.j... 1
- J TugasParkir27.java 1

PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\github\jobsheet7\daspro-jobsheet7> ^C
PS D:\github\jobsheet7\daspro-jobsheet7>
PS D:\github\jobsheet7\daspro-jobsheet7> d; cd 'd:\github\jobsheet7\daspro-jobsheet7'; & 'C:\Program Files\Java\jdk-11.0.10\bin\java.exe' -cp 'C:\Users\ACER\AppData\Roaming\Code\User\workspaceStorage\4a460db642c96a570c8f\TugasParkir27'
Masukan jenis kendaraan(1:mobil, 2:motor, 0:keluar): 2
Masukan durasi parkir(jam): 6
Total pembayaran: 12500
Masukan jenis kendaraan(1:mobil, 2:motor, 0:keluar): 6
Jenis kendaraan tidak valid, masukan jenis kendaraan dengan benar!
Masukan jenis kendaraan(1:mobil, 2:motor, 0:keluar): 2
Masukan durasi parkir(jam): 6
Total pembayaran: 25000
Total pembayaran: 37500
Masukan jenis kendaraan(1:mobil, 2:motor, 0:keluar): 0
Total pembayaran: 37500
```