

Nama : Sheilandra Zarawiba

Kelas : TI-1D

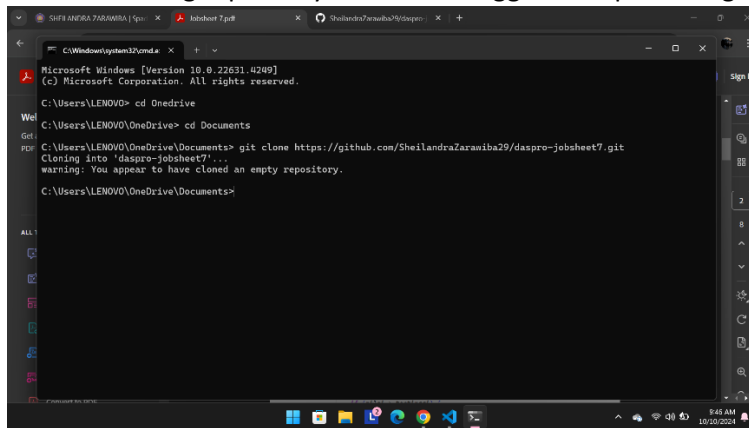
No. Absen : 28

NIM : 244107020110

## JOBSHEET 7

### Percobaan 1

1. Buat repository baru pada akun Github Anda, beri nama daspro-jobsheet7
2. Lakukan cloning repository tersebut menggunakan perintah git clone dari terminal

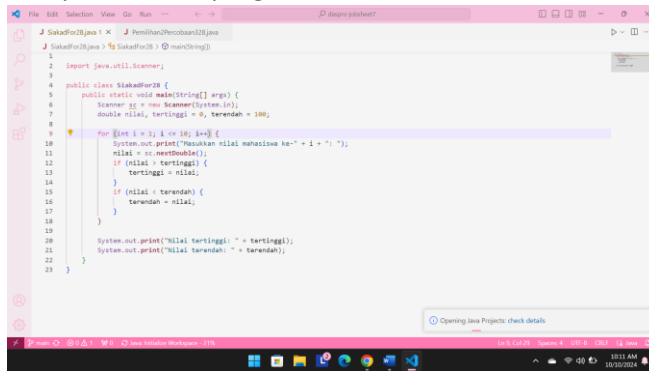


```
Microsoft Windows [Version 10.0.22631.4249]
(c) Microsoft Corporation. All rights reserved.

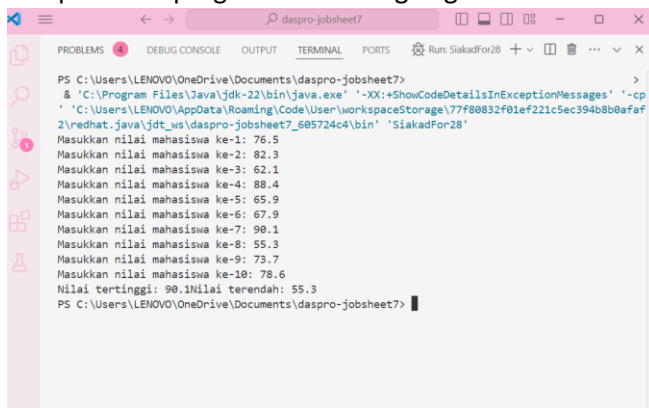
C:\Users\LENOVO> cd OneDrive
C:\Users\LENOVO\OneDrive> cd Documents
C:\Users\LENOVO\OneDrive\Documents> git clone https://github.com/SheilandraZarawiba29/daspro-jobsheet7.git
Cloning into 'daspro-jobsheet7'...
warning: You appear to have cloned an empty repository.
C:\Users\LENOVO\OneDrive\Documents>
```

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
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

## 12. Compile dan run program



## 13. Commit dan push kode program ke Github 2.1.2 Verifikasi Hasil Percobaan Cocokkan hasil compile kode program Anda dengan gambar berikut ini



## Pertanyaan

1. Sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program Percobaan 1!  
Jawab:
  - a) Inisialisasi: `int i = 1` (nilai awal)
  - b) Kondisi: `i <= 10` (perulangan berlangsung selama kondisinya terpenuhi)
  - c) Inkrementasi: `i++` (nilai `i` bertambah 1 setiap kali iterasi)
2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Jawab:

Variabel tertinggi = 0 dan terendah = 100 adalah inisialisasi yang benar, karena mereka memastikan bahwa nilai tertinggi dan terendah dapat diperbarui secara tepat berdasarkan nilai yang dimasukkan pengguna. Jika tertinggi = 100 dan terendah = 0, hasil yang diperoleh dari program akan salah, karena nilai tertinggi tidak akan pernah berubah dari 100, dan terendah tidak akan pernah berubah dari 0, kecuali jika input ekstrim dimasukkan.

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

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

Jawab:

- Fungsi potongan kode diatas adalah untuk membandingkan nilai yang baru dimasukkan oleh pengguna (nilai) dengan nilai tertinggi atau terendah yang tersimpan saat ini dalam variabel tertinggi atau terendah.
  - Alur kerja dari potongan kode diatas:
    - Pada setiap iterasi perulangan, pengguna memasukkan nilai mahasiswa.
    - Setiap kali nilai baru dimasukkan, kedua potongan kode di atas dijalankan.
    - Pertama**, program mengecek apakah nilai yang baru dimasukkan lebih besar dari **tertinggi**. Jika ya, nilai tersebut diperbarui menjadi nilai tertinggi baru.
    - Kedua**, program mengecek apakah nilai yang baru dimasukkan lebih kecil dari **terendah**. Jika ya, nilai tersebut diperbarui menjadi nilai terendah baru.
    - Setelah 10 iterasi (dalam contoh program), nilai tertinggi dan terendah yang dicatat adalah nilai maksimal dan minimal dari semua nilai yang dimasukkan.
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!

```
4 public class SiswaFord {  
5     public static void main(String[] args) {  
6         Scanner sc = new Scanner(System.in);  
7         double nilai, tertinggi = 0, terendah = 100;  
8         int jmlLulus = 0, jmlTidakLulus = 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                jmlLulus++;  
21            } else {  
22                jmlTidakLulus++;  
23            }  
24        }  
25  
26        System.out.println("Nilai tertinggi: " + tertinggi);  
27        System.out.println("Nilai terendah: " + terendah);  
28        System.out.println("Jumlah mahasiswa yang lulus: " + jmlLulus);  
29        System.out.println("Jumlah mahasiswa yang tidak lulus: " + jmlTidakLulus);  
30    }  
31}
```

Output of the program:

```
16/05/2016 10:08 AM  
Masukkan nilai mahasiswa ke-1: 76.9  
Masukkan nilai mahasiswa ke-2: 60.3  
Masukkan nilai mahasiswa ke-3: 62.1  
Masukkan nilai mahasiswa ke-4: 66.4  
Masukkan nilai mahasiswa ke-5: 60.3  
Masukkan nilai mahasiswa ke-6: 67.9  
Masukkan nilai mahasiswa ke-7: 68.5  
Masukkan nilai mahasiswa ke-8: 55.2  
Masukkan nilai mahasiswa ke-9: 75.7  
Masukkan nilai mahasiswa ke-10: 78.6  
Nilai tertinggi: 78.6  
Nilai terendah: 55.2  
Jumlah mahasiswa yang lulus: 9  
Jumlah mahasiswa yang tidak lulus: 1  
% c:\Users\JAGHOV\Documents\Jagpro-johwa7>
```

5. Commit dan push code program ke Github

## Percobaan 2

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.
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

```
SiakadWhile28.java > Language Support for Java(TM) by Red Hat > SiakadWhile28 > main(String[])
1
2 import java.util.Scanner;
3
4 public class SiakadWhile28{
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         int nilai, jml, i=0;
8
9         System.out.print("Masukkan jumlah mahasiswa: ");
10        jml = sc.nextInt();
11
12        while (i < jml){
13            System.out.print("Masukkan nilai mahasiswa ke-" + (i+1) + ": ");
14            nilai = sc.nextInt();
```

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

```
15        if (nilai < 0 || nilai > 100){
16            System.out.println("Nilai tidak valid. Masukkan lagi nilai yang valid!");
17            continue;
18        }
19        if (nilai < 60 || nilai > 100){
20            System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah A");
21        } else if (nilai > 70 || nilai < 80){
22            System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah B");
23        } else if (nilai > 80 || nilai < 90){
24            System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah C");
25        } else if (nilai > 90 || nilai < 100){
26            System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah D");
27        } else {
28            System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah E");
29        }
30        i++;
31    }
32 }
```

9. Compile dan run program

```
PS C:\Users\LEONARD\Documents\depan-jahnet> javac SiakadWhile28.java
PS C:\Users\LEONARD\Documents\depan-jahnet> java SiakadWhile28
Masukkan jumlah mahasiswa: 5
Masukkan nilai mahasiswa ke-1: 85
Nilai mahasiswa ke-1 adalah C
Masukkan nilai mahasiswa ke-2: 65
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: 75
Nilai mahasiswa ke-3 adalah B
Masukkan nilai mahasiswa ke-4: 110
Nilai tidak valid. Masukkan lagi nilai yang valid!
Masukkan nilai mahasiswa ke-4: 70
Nilai mahasiswa ke-4 adalah C
Masukkan nilai mahasiswa ke-5: 90
Nilai mahasiswa ke-5 adalah D
PS C:\Users\LEONARD\Documents\depan-jahnet>
```

10. Commit dan push kode program ke Github

## Pertanyaan

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

Jawab:

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

Ini adalah ekspresi logika yang memeriksa dua kondisi menggunakan operator **OR** (`||`):

- `nilai < 0`: Mengecek apakah nilai yang dimasukkan lebih kecil dari 0 (artinya nilai negatif).
- `nilai > 100`: Mengecek apakah nilai yang dimasukkan lebih besar dari 100 (artinya melebihi nilai maksimal yang diperbolehkan, misalnya dalam skala nilai 0-100).

- b. `Continue`

Digunakan ketika kita ingin melewati eksekusi kode tertentu dalam suatu iterasi perulangan, tetapi tetap ingin melanjutkan perulangan untuk iterasi berikutnya.

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

Jawab:

Ketika `i++` diletakkan di akhir perulangan, artinya kita menggunakan nilai `i` terlebih dahulu dalam perulangan, kemudian meningkatkan nilainya setelah semua instruksi di dalam blok perulangan dijalankan. Jika `i++` diletakkan di awal perulangan, maka nilai `i` akan ditingkatkan sebelum instruksi lain dieksekusi dalam setiap iterasi. Artinya, nilai `i` akan meningkat dulu sebelum digunakan dalam logika perulangan.

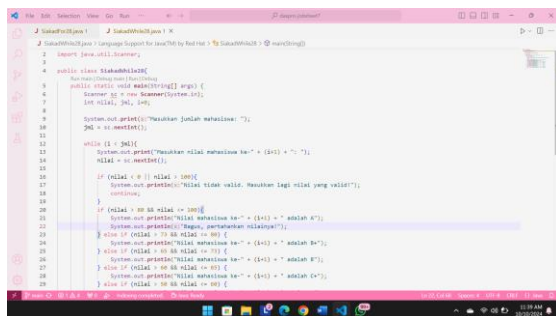
3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan `WHILE` akan berjalan?

Jawab:

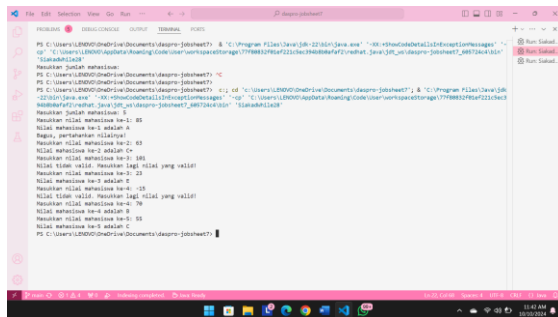
Karena perulangan **WHILE** akan terus berjalan selama nilai `i` lebih kecil dari `jml` (jumlah mahasiswa yang dimasukkan), dan nilai `i` akan bertambah 1 setiap kali perulangan selesai, kecuali ada **continue** yang dilewatkan jadi perulangan `WHILE` akan berjalan sebanyak 19 kali.

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

Jawab:



```
1 // Sifat: Sederhana, Mudah Dipahami
2 // Nama: Sifat: Sederhana, Mudah Dipahami
3 // Tujuan: Mengetahui cara menggunakan operator logika dan perulangan
4 // Langkah:
5 // 1. Import java.util.Scanner
6 // 2. Buat objek Scanner
7 // 3. Buat variabel untuk menyimpan skor
8 // 4. Buat variabel untuk menyimpan jumlah skor
9 // 5. Buat variabel untuk menyimpan jumlah mahasiswa
10 // 6. Buat variabel untuk menyimpan rata-rata skor
11 // 7. Buat variabel untuk menyimpan pesan tambahan
12 // 8. Buat variabel untuk menyimpan pesan tambahan
13 // 9. Buat variabel untuk menyimpan pesan tambahan
14 // 10. Buat variabel untuk menyimpan pesan tambahan
15 // 11. Buat variabel untuk menyimpan pesan tambahan
16 // 12. Buat variabel untuk menyimpan pesan tambahan
17 // 13. Buat variabel untuk menyimpan pesan tambahan
18 // 14. Buat variabel untuk menyimpan pesan tambahan
19 // 15. Buat variabel untuk menyimpan pesan tambahan
20 // 16. Buat variabel untuk menyimpan pesan tambahan
21 // 17. Buat variabel untuk menyimpan pesan tambahan
22 // 18. Buat variabel untuk menyimpan pesan tambahan
23 // 19. Buat variabel untuk menyimpan pesan tambahan
24 // 20. Buat variabel untuk menyimpan pesan tambahan
25 // 21. Buat variabel untuk menyimpan pesan tambahan
26 // 22. Buat variabel untuk menyimpan pesan tambahan
27 // 23. Buat variabel untuk menyimpan pesan tambahan
28 // 24. Buat variabel untuk menyimpan pesan tambahan
29 // 25. Buat variabel untuk menyimpan pesan tambahan
30 // 26. Buat variabel untuk menyimpan pesan tambahan
31 // 27. Buat variabel untuk menyimpan pesan tambahan
32 // 28. Buat variabel untuk menyimpan pesan tambahan
33 // 29. Buat variabel untuk menyimpan pesan tambahan
34 // 30. Buat variabel untuk menyimpan pesan tambahan
35 // 31. Buat variabel untuk menyimpan pesan tambahan
36 // 32. Buat variabel untuk menyimpan pesan tambahan
37 // 33. Buat variabel untuk menyimpan pesan tambahan
38 // 34. Buat variabel untuk menyimpan pesan tambahan
39 // 35. Buat variabel untuk menyimpan pesan tambahan
40 // 36. Buat variabel untuk menyimpan pesan tambahan
41 // 37. Buat variabel untuk menyimpan pesan tambahan
42 // 38. Buat variabel untuk menyimpan pesan tambahan
43 // 39. Buat variabel untuk menyimpan pesan tambahan
44 // 40. Buat variabel untuk menyimpan pesan tambahan
45 // 41. Buat variabel untuk menyimpan pesan tambahan
46 // 42. Buat variabel untuk menyimpan pesan tambahan
47 // 43. Buat variabel untuk menyimpan pesan tambahan
48 // 44. Buat variabel untuk menyimpan pesan tambahan
49 // 45. Buat variabel untuk menyimpan pesan tambahan
50 // 46. Buat variabel untuk menyimpan pesan tambahan
51 // 47. Buat variabel untuk menyimpan pesan tambahan
52 // 48. Buat variabel untuk menyimpan pesan tambahan
53 // 49. Buat variabel untuk menyimpan pesan tambahan
54 // 50. Buat variabel untuk menyimpan pesan tambahan
55 // 51. Buat variabel untuk menyimpan pesan tambahan
56 // 52. Buat variabel untuk menyimpan pesan tambahan
57 // 53. Buat variabel untuk menyimpan pesan tambahan
58 // 54. Buat variabel untuk menyimpan pesan tambahan
59 // 55. Buat variabel untuk menyimpan pesan tambahan
60 // 56. Buat variabel untuk menyimpan pesan tambahan
61 // 57. Buat variabel untuk menyimpan pesan tambahan
62 // 58. Buat variabel untuk menyimpan pesan tambahan
63 // 59. Buat variabel untuk menyimpan pesan tambahan
64 // 60. Buat variabel untuk menyimpan pesan tambahan
65 // 61. Buat variabel untuk menyimpan pesan tambahan
66 // 62. Buat variabel untuk menyimpan pesan tambahan
67 // 63. Buat variabel untuk menyimpan pesan tambahan
68 // 64. Buat variabel untuk menyimpan pesan tambahan
69 // 65. Buat variabel untuk menyimpan pesan tambahan
70 // 66. Buat variabel untuk menyimpan pesan tambahan
71 // 67. Buat variabel untuk menyimpan pesan tambahan
72 // 68. Buat variabel untuk menyimpan pesan tambahan
73 // 69. Buat variabel untuk menyimpan pesan tambahan
74 // 70. Buat variabel untuk menyimpan pesan tambahan
75 // 71. Buat variabel untuk menyimpan pesan tambahan
76 // 72. Buat variabel untuk menyimpan pesan tambahan
77 // 73. Buat variabel untuk menyimpan pesan tambahan
78 // 74. Buat variabel untuk menyimpan pesan tambahan
79 // 75. Buat variabel untuk menyimpan pesan tambahan
80 // 76. Buat variabel untuk menyimpan pesan tambahan
81 // 77. Buat variabel untuk menyimpan pesan tambahan
82 // 78. Buat variabel untuk menyimpan pesan tambahan
83 // 79. Buat variabel untuk menyimpan pesan tambahan
84 // 80. Buat variabel untuk menyimpan pesan tambahan
85 // 81. Buat variabel untuk menyimpan pesan tambahan
86 // 82. Buat variabel untuk menyimpan pesan tambahan
87 // 83. Buat variabel untuk menyimpan pesan tambahan
88 // 84. Buat variabel untuk menyimpan pesan tambahan
89 // 85. Buat variabel untuk menyimpan pesan tambahan
90 // 86. Buat variabel untuk menyimpan pesan tambahan
91 // 87. Buat variabel untuk menyimpan pesan tambahan
92 // 88. Buat variabel untuk menyimpan pesan tambahan
93 // 89. Buat variabel untuk menyimpan pesan tambahan
94 // 90. Buat variabel untuk menyimpan pesan tambahan
95 // 91. Buat variabel untuk menyimpan pesan tambahan
96 // 92. Buat variabel untuk menyimpan pesan tambahan
97 // 93. Buat variabel untuk menyimpan pesan tambahan
98 // 94. Buat variabel untuk menyimpan pesan tambahan
99 // 95. Buat variabel untuk menyimpan pesan tambahan
100 // 96. Buat variabel untuk menyimpan pesan tambahan
101 // 97. Buat variabel untuk menyimpan pesan tambahan
102 // 98. Buat variabel untuk menyimpan pesan tambahan
103 // 99. Buat variabel untuk menyimpan pesan tambahan
104 // 100. Buat variabel untuk menyimpan pesan tambahan
```



## 5. Commit dan push kode program ke Github

### Percobaan 3

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()
4. 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.

```
J SiakadFor28.java 1 J SiakadWhile28.java 1 J CafeDoWhile28.java 1
J CafeDoWhile28.java > CafeDoWhile28 > main(String[] args)
1
2 import java.util.Scanner;
3
4 public class CafeDoWhile28{
    Run main | Debug main | Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner (System.in);
7
8         int kopi, teh, roti, totalHarga;
9         String namaPelanggan;
10        int hargaKopi = 12000;
11        int hargaTeh = 7000;
12        int hargaRoti = 20000;
13    }
```

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

```

1  public class CafeDohhito2 {
2      public static void main(String[] args) {
3          Scanner sc = new Scanner(System.in);
4          System.out.println("Masukkan nama pelanggan (ketik 'batal' untuk keluar): ");
5          namaPelanggan = sc.nextLine();
6          if (namaPelanggan.equals("batal")) {
7              System.out.println("Transaksi dibatalkan.");
8              break;
9          }
10         System.out.println("Jumlah kopi: ");
11         kopi = sc.nextInt();
12         System.out.println("Jumlah teh: ");
13         teh = sc.nextInt();
14         System.out.println("Jumlah roti: ");
15         roti = sc.nextInt();
16         totalHarga = (kopi * hargaKopi) + (teh * hargaTeh) + (roti * hargaRoti);
17         System.out.println("Total yang harus dibayar: Rp. " + totalHarga);
18         sc.nextLine();
19     } while (true);
20     System.out.println("Semua transaksi selesai.");
21 }

```

## 7. Compile dan run program

```

PS C:\Users\LENDU\OneDrive\Documents> java -cp . CafeDohhito2.java
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Rana
Jumlah kopi: 5
Jumlah teh: 1
Jumlah roti: 2
Total yang harus dibayar: Rp. 50000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Yudi
Jumlah kopi: 1
Jumlah teh: 2
Jumlah roti: 2
Total yang harus dibayar: Rp. 80000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): BATAL
Transaksi dibatalkan.
Semua transaksi selesai.
PS C:\Users\LENDU\OneDrive\Documents>

```

## 8. Commit dan push code program ke Github

## TUGAS

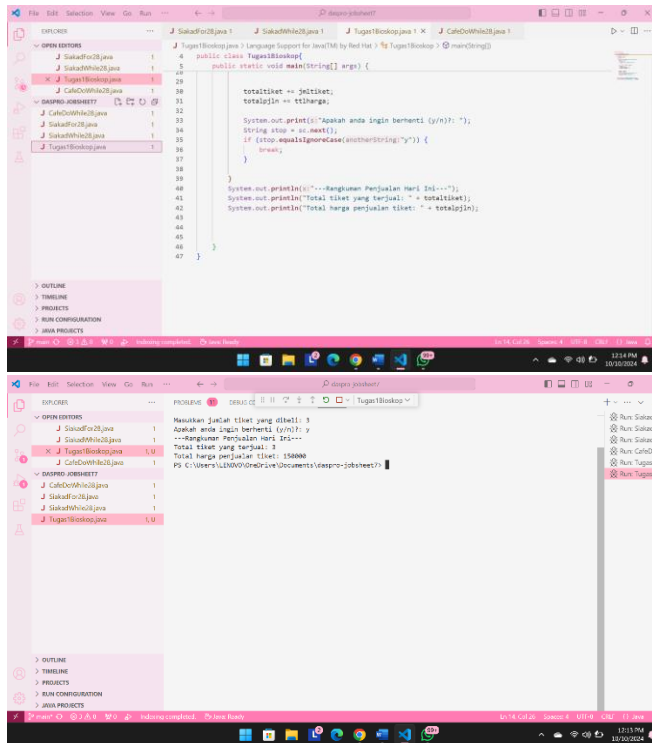
- Seorang pengelola bioskop ingin membuat program untuk menghitung total penjualan tiket dalam satu hari. Tiket dijual dengan harga Rp 50.000 per tiket. Program harus menghitung total tiket yang terjual dan total harga penjualan tiket selama satu hari dengan ketentuan sebagai berikut:
  - Jika pelanggan 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

Jawab:

```

1  import java.util.Scanner;
2
3  public class Tugaskiosk {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6          int jumlahTiket, totalTiket = 0;
7          int totalJml = 0;
8          int hargaTiket = 50000;
9
10         while (true) {
11             System.out.println("Masukkan jumlah tiket yang dibeli: ");
12             jumlahTiket = sc.nextInt();
13
14             if (jumlahTiket < 0) {
15                 System.out.println("Input tidak valid");
16                 continue;
17             }
18
19             double ttiharga = jumlahTiket * hargaTiket;
20
21             if (jumlahTiket > 10) {
22                 ttiharga -= (ttiharga * 0.15);
23             } else if (jumlahTiket > 4) {
24                 ttiharga -= (ttiharga * 0.10);
25             }
26         }
27     }
28 }

```



- 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:

