

NAMA : QULBI KHUTSI AZZUMI

NIM : 244107020050

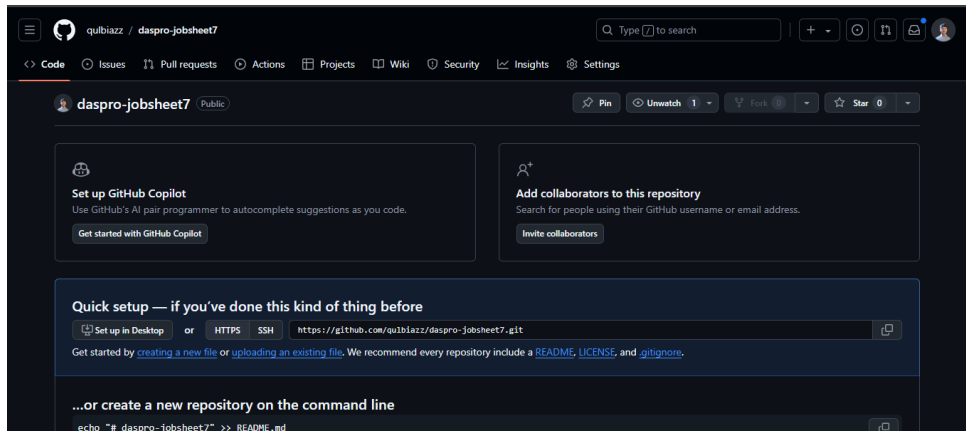
ABSEN : 23

KELAS : TI – 1D

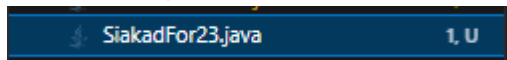
## JOBSHEET 7

### Percobaan 1

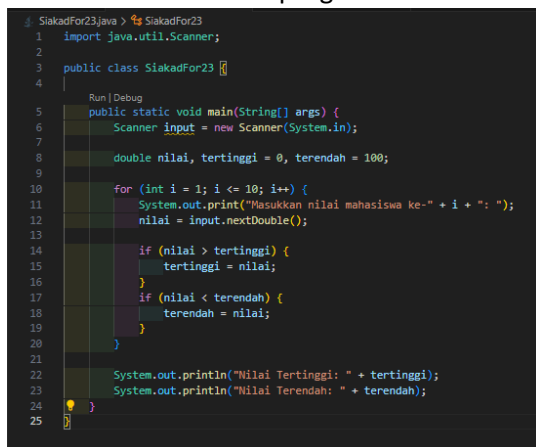
1. Buat repository baru pada akun Github Anda, beri nama daspro-jobsheet7



2. Buat file baru, beri nama SiakadForNoAbsen.java



3. Buatlah struktur dasar program Java



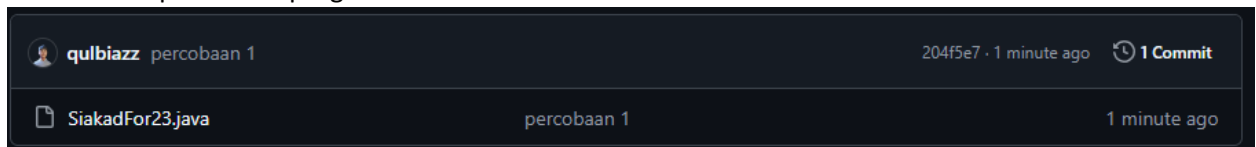
4. Compile dan run program

```

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

```

5. Commit dan push kode program ke Github



## Pertanyaan

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

```

for (int i = 1; i <= 10; i++) {
    System.out.print("Masukkan nilai mahasiswa ke-" + i + ": ");
    nilai = input.nextDouble();
}

```

Int i = inialisasi variabel I dan juga awal nilai perulangan

I <= 10 = nilai kurang dari sepuluh

I++ = update dari nilai perulangan, nilai akan update dan berulang sampai angka ke 10

```

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

```

Gambar diatas adalah statement dari perulangan for:

- Jika nilai yang dimasukkan lebih dari variable tertinggi = 0, maka variable tertinggi akan update nilai yang dimasukkan
  - Jika nilai yang dimasukkan kurang dari variable terendah = 100, maka variable terendah akan update nilai yang dimasukkan
2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Karena, agar output dari nilai yang tertinggi dan terendah bisa diupdate, dan jika variable tertinggi di inialisasi dengan 100 dan terendah 0, maka akan tetap outputnya. Karena, input nilai yang dimasukkan bilangan positif yang kurang dari 100. Jika ingin berubah outputnya nilai yang di inputkan harus kurang dari 0 untuk yang terendah dan lebih dari 100 untuk yang tertinggi

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

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

Mengecek dan membandingkan jika nilai yang dimasukkan lebih dari 0 dan nilai yang paling tinggi maka variable tertinggi akan memperbarui data nilai tertinggi. Dan mengecek dan membandingkan jika nilai yang dimasukkan kurang dari 100 dan nilai paling rendah maka variable terendah akan memperbarui data nilai terendah. Fungsinya, untuk memperbarui inputan nilai tertinggi dan terendah sesuai dengan inputan nilai.

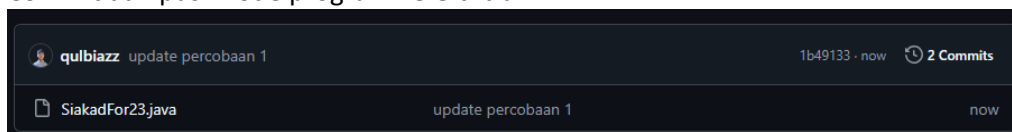
4. Modifikasi kode program sehingga terdapat perhitungan untuk menentukan berapa mahasiswa yang lulus dan yang tidak lulus berdasarkan batas kelulusan (nilai minimal 60). Tampilkan jumlah mahasiswa lulus dan tidak lulus setelah menampilkan nilai tertinggi dan terendah!

```
SiakadFor23.java > SiakadFor23 > main(String[])  
D:\KULIAH\MATRIKULUSMT-FYUGAS\PRAKTEK DASPRO\daspro-jobsheet7\SiakadFor23.java + 1  
problem in this file • Modified  
3 public class SiakadFor23 {  
4  
5     Run [Debug  
6     public static void main(String[] args) {  
7         Scanner input = new Scanner(System.in);  
8  
9         double nilai, tertinggi = 0, terendah = 100;  
10        int lulus = 0, tidakLulus = 0;  
11  
12        for (int i = 1; i <= 10; i++) {  
13            System.out.print("Masukkan nilai mahasiswa ke-" + i + ": ");  
14            nilai = input.nextDouble();  
15  
16            if (nilai > tertinggi) {  
17                tertinggi = nilai;  
18            }  
19            if (nilai < terendah) {  
20                terendah = nilai;  
21            }  
22  
23            if (nilai >= 60) {  
24                lulus++;  
25            }  
26            else {  
27                tidakLulus++;  
28            }  
29  
30            System.out.println("Nilai Tertinggi: " + tertinggi);  
31            System.out.println("Nilai Terendah: " + terendah);  
32            System.out.println("Jumlah Mhs Lulus: " + lulus);  
33            System.out.println("Jumlah Mhs Tidak Lulus: " + tidakLulus);  
34        }  
35    }  
36 }
```

Hasil:

```
45\bin\ "SiakadFor23"  
Masukkan nilai mahasiswa ke-1: 87,5  
Masukkan nilai mahasiswa ke-2: 89  
Masukkan nilai mahasiswa ke-3: 90  
Masukkan nilai mahasiswa ke-4: 80  
Masukkan nilai mahasiswa ke-5: 78  
Masukkan nilai mahasiswa ke-6: 77  
Masukkan nilai mahasiswa ke-7: 34  
Masukkan nilai mahasiswa ke-8: 77  
Masukkan nilai mahasiswa ke-9: 77  
Masukkan nilai mahasiswa ke-10: 0  
Nilai Tertinggi: 90.0  
Nilai Terendah: 0.0  
Jumlah Mhs Lulus: 8  
Jumlah Mhs Tidak Lulus: 2
```

5. Commit dan push kode program ke Github



## Percobaan 2

1. Buat file baru, beri nama SiakadWhileNoAbsen.java







2. Buatlah struktur dasar program Java

```
SiakadWhile23.java > SiakadWhile23 > main(String[])
1  import java.util.Scanner;
2
3  public class SiakadWhile23 {
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner input = new Scanner(System.in);
7
8          int i = 0;
9          int jml, nilai;
10
11         System.out.print(s:"Masukkan Jumlah Mhs: ");
12         jml = input.nextInt();
13
14         while (i < jml) {
15             System.out.print("Masukkan Nilai Mhs ke-" + (i+1) + ": ");
16             nilai = input.nextInt();
17
18             if (nilai < 0 || nilai > 100) {
19                 System.out.println(x:"Nilai Tidak Valid");
20                 continue;
21             }
22             if (nilai > 80 && nilai <= 100) {
23                 System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah A");
24             } else if (nilai > 73 && nilai <= 80) {
25                 System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah B+");
26             } else if (nilai > 65 && nilai <= 73) {
27                 System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah B");
28             } else if (nilai > 60 && nilai <= 65) {
29                 System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah C+");
30             } else if (nilai > 50 && nilai <= 60) {
31                 System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah C");
32             } else if (nilai > 39 && nilai <= 50) {
33                 System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah D");
34             } else {
35                 System.out.println("Nilai mahasiswa ke-" + (i+1) + " adalah E");
36             }
37             i++;
38         }
39     }
40 }
```

3. Compile dan run program

```
Masukkan Jumlah Mhs: 5
Masukkan Nilai Mhs ke-1: 20
Nilai mahasiswa ke-1 adalah E
Masukkan Nilai Mhs ke-2: 90
Nilai mahasiswa ke-2 adalah A
Masukkan Nilai Mhs ke-3: 91
Nilai mahasiswa ke-3 adalah A
Masukkan Nilai Mhs ke-4: 80
Nilai mahasiswa ke-4 adalah B+
Masukkan Nilai Mhs ke-5: 78
Nilai mahasiswa ke-5 adalah B+
```

4. Commit dan push kode program ke Github

 qulbiazz percobaan 2		cb2fe26 · 6 minutes ago	 3 Commits
 SiakadFor23.java	update percobaan 1	9 hours ago	
 SiakadWhile23.java	percobaan 2	6 minutes ago	

## 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`
- a. Berfungsi, untuk membandingkan variabel nilai dengan operator logika or. Maksudnya jika input nilai yang dimasukkan kurang dari 0 atau lebih dari seratus akan muncul outputnya.
  - b. Berfungsi, untuk melompati atau skip atau tidak diperhatikan input nilai yang dimasukkan.
2. Mengapa sintaks `i++` dituliskan di akhir perulangan WHILE? Apa yang terjadi jika posisinya dituliskan di awal perulangan WHILE?  
**Karena, agar perulangan menyelesaikan pengecekan nilai terlebih dulu, lalu baru diupdate nilainya dan jika `i++` ditulis diawal maka akan mengupdate variable `i` terlebih dahulu, dan memulai perulangan dari angka 2**
  3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan WHILE akan berjalan?  
**19, dengan angka awal `i` adalah  $0+1 = 1$**
  4. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya"!

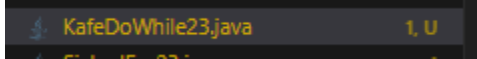
```
Masukkan Jumlah Mhs: 1  
Masukkan Nilai Mhs ke-1: 100  
Nilai mahasiswa ke-1 adalah A  
Bagus, pertahankan nilainya
```

5. Commit dan push kode program ke Github

 <b>qulbiazz</b> update percobaan 2	19cfb8b · 2 minutes ago	 <b>4 Commits</b>
 SiakadFor23.java	update percobaan 1	10 hours ago
 SiakadWhile23.java	update percobaan 2	2 minutes ago

## Percobaan 3

1. Buat file baru, beri nama KafeDoWhileNoAbsen.java



2. Buatlah struktur dasar program Java

```
1  KafeDoWhile23.java > KafeDoWhile23 > main(String[])
2  import java.util.Scanner;
3  public class KafeDoWhile23 {
4      public static void main(String[] args) {
5          Scanner input = new Scanner(System.in);
6
7          int kopi, teh, roti;
8          String namaPelanggan;
9          int hargaKopi = 12000, hargaTeh = 7000, hargaRoti = 20000;
10
11          do {
12              System.out.print(s:"Masukkan nama pelanggan (ketik 'batal' untuk keluar): ");
13              namaPelanggan = input.nextLine();
14
15              if (namaPelanggan.equalsIgnoreCase(anotherString:"batal")) {
16                  System.out.println(x:"Transaksi dibatalkan");
17                  break;
18              }
19
20              System.out.print(s:"Jumlah Kopi: ");
21              kopi = input.nextInt();
22              System.out.print(s:"Jumlah Teh: ");
23              teh = input.nextInt();
24              System.out.print(s:"Jumlah Roti: ");
25              roti = input.nextInt();
26
27              int totalHarga = (kopi * hargaKopi) + (teh * hargaTeh) + (roti * hargaRoti);
28              System.out.println(x:"-----");
29              System.out.println("Total Kopi: " + kopi + " pcs");
30              System.out.println("Total Teh: " + teh + " pcs");
31
32              System.out.println("Total yang harus dibayar: Rp. " + totalHarga);
33              input.nextLine();
34
35          } while (true);
36          System.out.println(x:"Semua transaksi selesai :)");
37      }
38  }
```

3. Compile dan run program

```
\Roaming\Code\User\workspaceStorage\934b3aa6d5772b99f42f122fb5
Masukkan nama pelanggan (ketik 'batal' untuk keluar): qul
Jumlah Kopi: 1
Jumlah Teh: 2
Jumlah Roti: 3
-----
Total Kopi: 1 pcs
Total Teh: 2 pcs
Total Roti: 3 pcs
Total yang harus dibayar: Rp. 86000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): batal
Transaksi dibatalkan
Semua transaksi selesai :)
```

4. Commit dan push kode program ke Github

qulbazz percobaan 3	7928876 · 1 minute ago	🕒 5 Commits
KafeDoWhile23.java	percobaan 3	1 minute ago
SiakadFor23.java	update percobaan 1	10 hours ago
SiakadWhile23.java	update percobaan 2	12 minutes ago

## Pertanyaan

1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah "batal", maka berapa kali perulangan dilakukan?  
**1 kali, program akan menjalankan do terlebih dahulu, sebelum mengecek kondisi while. Jadi, program akan mengecek bagian input nama pelanggan, lalu cek kondisi jika batal program akan berhenti dengan sintaks break.**
2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!  
**Program diatas menggunakan BREAK;. Jadi, ketika input yang dimasukkan "batal" maka program akan berhenti**
3. Apa fungsi dari penggunaan nilai true pada kondisi DO-WHILE?  
**Agar program terus berjalan dengan batas jika input yang dimasukkan adalah "batal"**
4. Mengapa perulangan DO-WHILE tersebut tetap berjalan meskipun tidak ada komponen inisialisasi dan update?  
**Karena, pada perulangan do-while tidak perlu inialisasi dan update. Program akan berhenti jika kondisi salah/batal terpenuhi**

## 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.

```

4: tugas1.java > tugas1 > main(String[])
1  import java.util.Scanner;
2
3  public class tugas1 {
4      Run | Debug
      public static void main(String[] args) {
5          Scanner input = new Scanner(System.in);
6
7          int hargaTiket = 50000, jml_tiketBeli = 0, jml_tiketTerjual = 0;
8          double penjualanSehari = 0, diskon = 0.0, totalDiskon, hargaHarian;
9          int jumlahPelanggan, i = 0;
10
11         System.out.print(s:"Masukkan Jumlah Pelanggan Per Hari: ");
12         jumlahPelanggan = input.nextInt();
13
14         while (i < jumlahPelanggan) {
15             System.out.print("Masukkan Jumlah Tiket Pelanggan ke- " + (i + 1) + ": ");
16             jml_tiketBeli = input.nextInt();
17             jml_tiketTerjual += jml_tiketBeli;
18
19             if (jml_tiketBeli < 0) {
20                 System.out.println(x:"Jumlah Tiket Tidak Valid");
21                 continue;
22             }
23
24             if (jml_tiketBeli > 10) {
25                 diskon = 0.15;
26             }else if (jml_tiketBeli > 4) {
27                 diskon = 0.1;
28             }else{
29                 diskon = 0.0;
30             }
31
32             totalDiskon = jml_tiketBeli * hargaTiket * diskon;
33             hargaHarian = jml_tiketBeli * hargaTiket - totalDiskon;
34             penjualanSehari += hargaHarian;
35
36             i++;
37         }

```

```

      System.out.println("Jumlah Tiket Terjual: " + jml_tiketTerjual);
      System.out.println("Total Penjualan Tiket Sehari: " + penjualanSehari);
  }
}

```

Hasil:

```

PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\KULIAH\MATKUL\SMT 1\TUGAS\PRAKTEK DASPRO\daspro-jobsheet7> & 'C:\Program F
\Roaming\Code\User\workspaceStorage\934b3aa6d5772b99f42f122fb59cb6b7\redhat.java\
Masukkan Jumlah Pelanggan Per Hari: 2
Masukkan Jumlah Tiket Pelanggan ke- 1: 17
Masukkan Jumlah Tiket Pelanggan ke- 2: 3
Jumlah Tiket Terjual: 20
Total Penjualan Tiket Sehari: 872500.0
PS D:\KULIAH\MATKUL\SMT 1\TUGAS\PRAKTEK DASPRO\daspro-jobsheet7>

```



2. 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!

```
3: tugas2.java > ...
1  import java.util.Scanner;
2
3  public class tugas2 {
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner input = new Scanner(System.in);
7
8          int jenis, durasi, total = 0;
9
10         do {
11             System.out.print(s:"Masukkan Jenis Kendaraan (1 Mobil, 2 Motor, 0 Keluar): ");
12             jenis = input.nextInt();
13
14             if (jenis == 1 || jenis == 2) {
15                 System.out.print(s:"Masukkan Durasi: ");
16                 durasi = input.nextInt();
17                 if (durasi > 5) {
18                     total += 12500;
19                 }else if (jenis == 1) {
20                     total += durasi * 3000;
21                 }else if (jenis == 2) {
22                     total += durasi * 2000;
23                 }else{
24                     System.out.println(x:"Jenis Kendaraan Tidak Valid");
25                 }
26             }
27         } while (jenis != 0);
28
29         System.out.println("Total Pembayaran Parkir = " + total);
30     }
31 }
32
33 }
```

Hasil:

```
45 (021) Cag032
Masukkan Jenis Kendaraan (1 Mobil, 2 Motor, 0 Keluar): 1
Masukkan Durasi: 2
Total Pembayaran Parkir = 6000
Masukkan Jenis Kendaraan (1 Mobil, 2 Motor, 0 Keluar): 2
Masukkan Durasi: 2
Total Pembayaran Parkir = 10000
Masukkan Jenis Kendaraan (1 Mobil, 2 Motor, 0 Keluar): 0
Transaksi Batal
Total Pembayaran Parkir = 10000
```