

Praktek Daspro Ke-7

Nama : Aqila Herfian Fa'aizah Safwa
NIM : 254107020041
Mata Kuliah : Prakt_Daspro
Pertemuan Ke- : 7

2.1 Pertanyaan

```
3 public class SiakadFor03 {  
    Run | Debug | Run main | Debug main  
4 public static void main(String[] args) {  
5     Scanner sc = new Scanner(System.in);  
6  
7     double nilai, tertinggi = 0, terendah = 100;  
8     for (int i = 1; i <= 10; i++) {  
9         System.out.print("Masukkan nilai mahasiswa ke-" + i + ":");  
10        nilai = sc.nextDouble();  
11        if (nilai > tertinggi) {  
12            tertinggi = nilai;  
13        }  
14        if (nilai < terendah) {  
15            terendah = nilai;  
16        }  
17    }  
18    System.out.println("Nilai tertinggi: " + tertinggi);  
19    System.out.println("Nilai terendah: " + terendah);  
20 }  
21 }
```

```
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
```

1. Sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program Percobaan 1!
 - Perulangan int i = 1
 - Kondisi i <= 10
 - Update i++
2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

- jika terbalik. Tidak ada nilai mahasiswa yang lebih besar dari 100, sehingga variable tertinggi akan tetap 100 walaupun sebenarnya tidak ada mahasiswa yang memiliki nilai tersebut.
- Tidak ada nilai mahasiswa yang lebih kecil dari 0, sehingga variable terendah akan tetap 0 walaupun tidak ada mahasiswa yang nilainya 0.

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

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

- Fungsi: untuk menentukan dan memperbarui nilai tertinggi dan terendah dari data yang dimasukkan oleh pengguna.
- Alur:
- Program membaca 1 nilai yang diinput kemudian dicek dengan baris pertama. Mengecek apakah nilai tersebut lebih besar dari nilai tertinggi sebelumnya, jika ya maka nilai tersebut akan disimpan pada variable tertinggi.
- pada baris kedua, akan dicek apakah nilai tersebut lebih kecil dari nilai terendah sebelumnya. Jika ya maka nilai itu disimpan pada variable terendah.
- Proses akan berulang hingga nilai mahasiswa ke-10 terakumulasi.

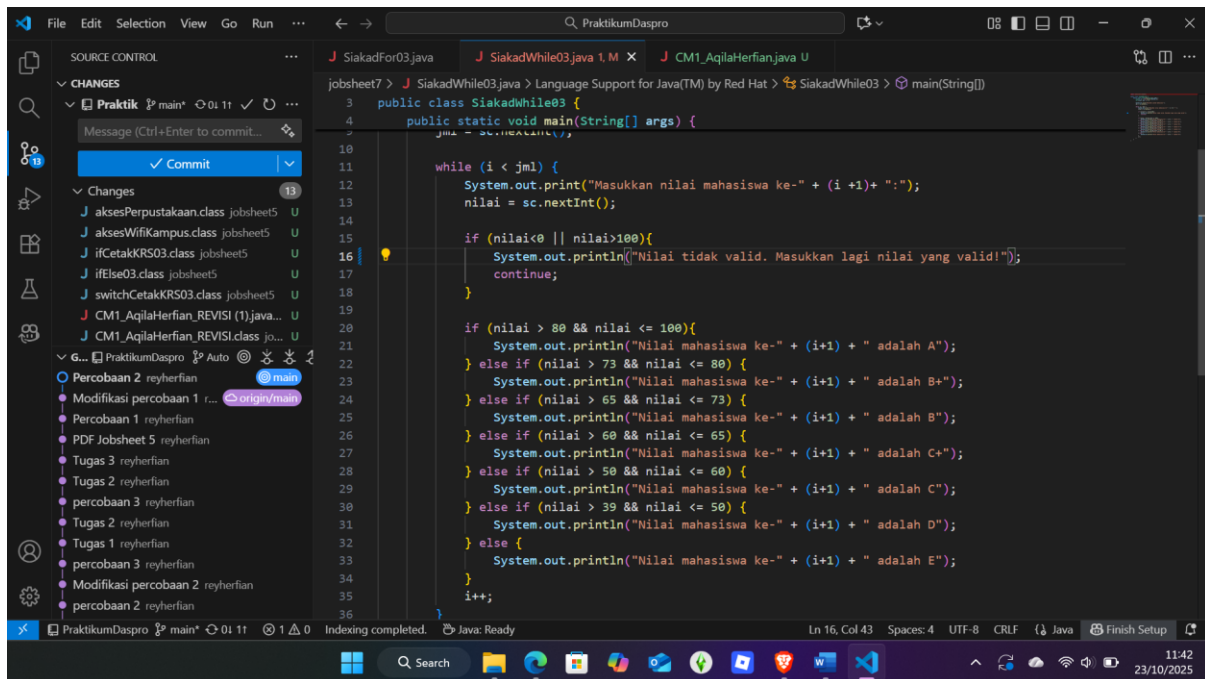
4. Modifikasi kode program sehingga terdapat perhitungan untuk menentukan berapa mahasiswa yang lulus dan yang tidak lulus berdasarkan batas kelulusan (nilai minimal 60). Tampilkan jumlah mahasiswa lulus dan tidak lulus setelah menampilkan nilai tertinggi dan terendah!

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

if (nilai >= 60) {
    lulus++;
} else {
    tLulus++;
}
}
```

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan "Modifikasi Percobaan 1"

2.2 Pertanyaan



```
Masukkan jumlah mahasiswa:5  
Masukkan nilai mahasiswa ke-1:90  
Nilai mahasiswa ke-1 adalah A  
Masukkan nilai mahasiswa ke-2:99  
Nilai mahasiswa ke-2 adalah A  
Masukkan nilai mahasiswa ke-3:78  
Nilai mahasiswa ke-3 adalah B+  
Masukkan nilai mahasiswa ke-4:60  
Nilai mahasiswa ke-4 adalah C  
Masukkan nilai mahasiswa ke-5:50  
Nilai mahasiswa ke-5 adalah D
```

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`

- untuk memastikan bahwa input berada dalam rentang valid 0-100. Jika tidak memenuhi syarat. Maka akan dianggap tidak valid dan program meminta input ulang.

- b) `Continue`

- Ketika nilai tidak valid, program tidak memproses bagian kod di bawahnya, dan langsung meminta input nilai mahasiswa lagi.

2. Mengapa sintaks `i++` dituliskan di akhir perulangan `WHILE`? Apa yang terjadi jika posisinya dituliskan di awal perulangan `WHILE`?
 - `i++` ditaruh di akhir supaya semua mahasiswa dihitung dan diproses satu per satu. Jika ditulis di awal, nilai `i` akan langsung bertambah sebelum proses input dilakukan. Akibatnya, mahasiswa pertama tidak akan diproses karena perhitungan sudah naik duluan.
3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan `WHILE` akan berjalan?
 - Maka akan berjalan 19 kali, karena `i = 0`
4. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya!"

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

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan "Modifikasi Percobaan 2"

2.3 Pertanyaan

```
do {  
    System.out.print("Masukkan nama pelanggan (ketik batal untuk keluar): ");  
    namaPelanggan = sc.nextLine();  
  
    if (namaPelanggan.equalsIgnoreCase("batal")) {  
        System.out.println("Transaksi dibatalkan.");  
        break;  
    }  
  
    System.out.print("Jumlah kopi: ");  
    kopi = sc.nextInt();  
    System.out.print("Jumlah teh: ");  
    teh = sc.nextInt();  
    System.out.print("Jumlah roti: ");  
    roti = sc.nextInt();  
  
    totalHarga = (kopi * hKopi) + (teh * hTeh) + (roti * hRoti);  
    System.out.println("Total yang harus dibayar: Rp " + totalHarga);  
    sc.nextLine();  
}  
while (true);  
  
System.out.println("Semua transaksi selesai.");
```

```

Masukkan nama pelanggan (ketik batal untuk keluar): Batal
Transaksi dibatalkan.
Semua transaksi selesai.
PS E:\PDASPRO\github\PraktikumDaspro\jobsheet7> java kafeDoWhile03
Masukkan nama pelanggan (ketik batal untuk keluar): Yahya
Jumlah kopi: 3
Jumlah teh: 0
Jumlah roti: 2
Total yang harus dibayar: Rp 76000
Masukkan nama pelanggan (ketik batal untuk keluar): █

```

1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah "batal", maka berapa kali perulangan dilakukan?
 - Jika input "batal", maka program akan berjalan 1 kali.
2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!
 - Ketika pengguna mengetik "batal" sebagai nama pelanggan.
3. Apa fungsi dari penggunaan nilai true pada kondisi DO-WHILE?
 - Digunakan agar input pelanggan terus bisa diulang sampai pengguna mengetik "batal", yang menjadi syarat berhenti.
4. Mengapa perulangan DO-WHILE tersebut tetap berjalan meskipun tidak ada komponen inisialisasi dan update?
 - Perulangan tidak ditentukan oleh nilai variable, tetapi oleh logika program yaitu input "batal" dari pengguna.

Tugas 1

```

public class Bioskop {
    public static void main(String[] args) {
        if (jmlTiket < 0) {
            System.out.println("Input tidak valid, coba lagi!");
            continue;
        }
        if (jmlTiket == 0) {
            break;
        }
        if (jmlTiket > 10) {
            diskon = 0.15;
        } else if (jmlTiket > 4) {
            diskon = 0.1;
        } else {
            diskon = 0;
        }

        tAwal = jmlTiket * hTiket;
        potongan = tAwal * diskon;
        tHarga = tAwal - potongan;

        System.out.println("Total bayar: Rp." + tHarga);

        tTiket += jmlTiket;
        tPenjualan += tHarga;
    } while (true);

    System.out.println("\nRekap total tiket terjual: " + tTiket);
}

```

```

Masukkan jumlah tiket (0 untuk selesai): 10
Total bayar: Rp.450000.0
Masukkan jumlah tiket (0 untuk selesai): 15
Total bayar: Rp.637500.0
Masukkan jumlah tiket (0 untuk selesai): 34
Total bayar: Rp.1445000.0
Masukkan jumlah tiket (0 untuk selesai): 21
Total bayar: Rp.892500.0
Masukkan jumlah tiket (0 untuk selesai): 2
Total bayar: Rp.100000.0
Masukkan jumlah tiket (0 untuk selesai): 3
Total bayar: Rp.150000.0
Masukkan jumlah tiket (0 untuk selesai): 0

```

```

Rekapan total tiket terjual: 85
Rekapan total penjualan: Rp 3675000.0
PS E:\PDASPRO\github\PraktikumDaspro\jobsheet7>

```

Tugas 2

```

sheet7 > J Taraf.java > Language Support for Java(TM) by Red Hat > Taraf > main(String[])
3 public class Taraf {
4     public static void main(String[] args) {
5
6         int jenis, durasi;
7         double total = 0;
8
9         do {
10             System.out.print("Masukkan jenis kendaraan (1 = mobil, 2 = motor, 0 = keluar: ");
11             jenis = sc.nextInt();
12
13             if (jenis == 0) {
14                 break;
15             }
16             if (jenis == 1 || jenis == 2) {
17                 System.out.print("Masukkan durasi parkir (jam): ");
18                 durasi = sc.nextInt();
19
20                 if (durasi > 5) {
21                     total += 12500;
22                 } else if (jenis == 1) {
23                     total += durasi * 3000;
24                 } else {
25                     total += durasi * 2000;
26                 }
27             } else {
28                 System.out.print("Input tidak valid, coba lagi!");
29             }
30         } while (jenis != 0);
31         System.out.print("Total pendapatan parkir hari ini: Rp. " + total);
32     }
33 }

```

```

Masukkan jenis kendaraan (1 = mobil, 2 = motor, 0 = keluar: 2
Masukkan durasi parkir (jam): 2
Masukkan jenis kendaraan (1 = mobil, 2 = motor, 0 = keluar: 1
Masukkan durasi parkir (jam): 1
Masukkan jenis kendaraan (1 = mobil, 2 = motor, 0 = keluar:
2
Masukkan durasi parkir (jam): 6
Masukkan jenis kendaraan (1 = mobil, 2 = motor, 0 = keluar: 0
Total pendapatan parkir hari ini: Rp. 19500.0

```