

Nama : Abdul Muid  
Kelas : TI 1D  
NIM : 244107020006  
No. Absen : 01

## Percobaan 1

### Langkah Langkah

1. Buat repository baru beri nama daspro-jobsheet7 dan lakukan cloning
2. Buka folder repository di visual studio code kemudian buat file baru dengan nama SiakadForNoAbsen.java
3. Buat struktur dasar program java dan tambahkan library scanner dibagian atas, selanjutnya Buat deklarasi variable seperti berikut

```
1 import java.util.Scanner;  
2 public class SiakadFor01 {  
3  
4     public static void main(String[] args) {  
5  
6         Scanner input = new Scanner(System.in);  
7  
8         double nilai, tertinggi = 0, terendah = 100 ;
```

4. Buat struktur perulangan for dan tambahkan perintah untuk memasukkan nilai mahasiswa di dalam perulangan. Lalu buat dua kondisi terpisah untuk mengecek nilai tertinggi dan terendah lakukan seperti berikut

```
1 for (int i=1; i<= 10; i++){  
2     System.out.print("Masukkan nilai mahasiswa ke-" + i + " : ");  
3     nilai = input.nextDouble();  
4     if (nilai > tertinggi){  
5         tertinggi = nilai;  
6     }if (nilai < terendah){  
7         terendah = nilai;  
8     }  
9 }
```

5. Diluar for tampilkan nilai tertinggi dan terendah

```
1 System.out.println("Nilai tertinggi: " + tertinggi);  
2 System.out.println("Nilai terendah: " + terendah);
```

6. Compile dan run program
7. Commit dan push ke github

### Hasil kode program (output)

```
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
```

## Jawaban Pertanyaan

- Inisialisasi `int i = 1` (menentukan nilai awal dari variabel pengulangan)
  - Kondisi `i <= 10` (perulangan akan terus berjalan selama `i` kurang dari sama dengan 10)
  - Increment `i++` (nilai `i` akan terus bertambah 1 setiap setelah iterasi)
- Variabel tertinggi diinisialisasi 0 karena asumsi awal bahwa nilai pertama yang ditemui adalah yang terbesar. Setiap nilai baru yang lebih besar akan menggantikan nilai variabel tertinggi ini. Sedangkan variabel terendah diinisialisasi 100 karena asumsi awalnya adalah nilai pertama yang ditemui adalah yang terkecil. Setiap nilai baru yang lebih kecil akan menggantikan nilai variabel terendah ini. Yang terjadi jika inisialisasi dibalik maka pada nilai tertinggi akan tetap 100 dan nilai terendah akan tetap 0
- Fungsi pada potongan kode tersebut yaitu nilai akan dibandingkan dengan tertinggi dan terendah jika itu benar maka variabel tertinggi dan terendah akan diganti nilai
- Modifikasi kode program untuk mahasiswa lulus dan tidak lulus

```
1  int jmlLulus=0, jmlTidakLulus=0;
2      double nilai, tertinggi = 0, terendah = 100 ;
3
4      for (int i=1; i<= 10; i++){
5          System.out.print("Masukkan nilai mahasiswa ke-" + i + ": ");
6          nilai = input.nextDouble();
7          if (nilai > tertinggi){
8              tertinggi = nilai;
9          }if (nilai < terendah){
10             terendah = nilai;
11         }if (nilai >= 60){
12             jmlLulus++;
13         }else {
14             jmlTidakLulus++;
15         }
16     }
17
18     System.out.println("Nilai tertinggi: " + tertinggi);
19     System.out.println("Nilai terendah: " + terendah);
20     System.out.println("Jumlah mahasiswa yang lulus: " + jmlLulus);
21     System.out.println("Jumlah mahasiswa yang tidak lulus: " + jmlTidakLulus);
```

### Output

```
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
Jumlah mahasiswa yang lulus: 9
Jumlah mahasiswa yang tidak lulus: 1
```

- Commit dan push kode program ke Github

## Percobaan 2

### Langkah Langkah

1. Buat file baru beri nama SiakadWhileNoAbsen.java
2. Buatlah struktur dasar java dan tambahkan library scanner dibagian atas, selanjutnya buat deklarasi seperti berikut

```
1 import java.util.Scanner;
2 public class SiakadWhile01 {
3     public static void main(String[] args) {
4
5         Scanner input = new Scanner(System.in);
6
7         int nilai, jml, i=0;
```

3. Tuliskan kode program untuk menerima input banyaknya mahasiswa

```
1 System.out.print("Masukkan jumlah mahasiswa: ");
2 jml = input.nextInt();
```

4. Buat struktur perulangan while, didalamnya tambahkan perintah memasukkan nilai mahasiswa. Kemudian buat kondisi if untuk mengecek valid atau tidak nilai yang dimasukkan, Selanjutnya tambahkan pemilihan if else if else untuk menampilkan kategori nilai huruf. Lakukan seperti berikut

```
1 while (i < jml){
2     System.out.print("Masukkan nilai mahasiswa ke-" + (i+1) + ": ");
3     nilai = input.nextInt();
4
5     if(nilai < 0 || nilai > 100){
6         System.out.println("Nilai tidak valid. Masukkan lagi nilai yang valid!");
7         continue;
8     }
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 }
```

5. Compile dan run program
6. Commit dan push ke github

## Hasil kode program (Output)

```
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
```

## Jawaban Pertanyaan

1. a. Pada nilai  $< 0$  || nilai  $> 100$  kondisi ini digunakan untuk memeriksa apakah nilai yang dimasukkan berada di luar rentang yang valid, yaitu antara 0 hingga 100. Jika nilai tidak berada dalam rentang tersebut, maka dianggap sebagai nilai yang tidak valid.
2. b. Perintah continue akan langsung mengembalikan eksekusi ke awal loop (jika kode ini berada di dalam sebuah loop). Artinya, program akan langsung meminta pengguna untuk memasukkan nilai lagi tanpa menjalankan kode yang ada di bawahnya dalam iterasi saat itu.
3. `i++` diletakkan diakhir karena pada perulangan while akan dijalankan dulu sebelum `i` ditambahkan artinya output akan dimulai mahasiswa ke 1 jika diletakkan diawal maka akan nilai akan ditambah dulu outputnya dimulai dari mahasiswa ke 2
4. perulangan while akan berjalan 19 kali
5. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya"!

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

6. Commit dan push kode program ke Github

## Percobaan 3

### Langkah Langkah

1. Buat file baru beri nama KafeWhileNoAbsen.java
2. Buatlah struktur dasar java dan tambahkan library scanner dibagian atas, selanjutnya buat deklarasi seperti berikut

```
1 int kopi, teh, roti;
2 String namaPelanggan;
3 double hargakopi=12000, hargateh=7000, hargaroti=20000, totalharga;
```

3. Buat struktur perulangan do while (true), 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 lakukan seperti berikut

```
1  do {
2      System.out.print("Masukkan nama pelanggan (ketik 'batal' untuk keluar): ");
3      namaPelanggan = input.nextLine();
4
5      if (namaPelanggan.equalsIgnoreCase("batal")){
6          System.out.println("Transaksi dibatalkan");
7          break;
8      }
9      System.out.print("Masukkan jumlah kopi: ");
10     kopi = input.nextInt();
11     System.out.print("Masukkan jumlah teh: ");
12     teh = input.nextInt();
13     System.out.print("Masukkan jumlah roti: ");
14     roti = input.nextInt();
15     totalharga = (kopi*hargakopi) + (teh*hargateh) + (roti*hargaroti);
16     System.out.println("Total yang harus dibayar: Rp " + totalharga);
17     input.nextLine();
18 } while(true);
19
20 System.out.println("Semua transaksi selesai.");
```

4. Compile dan run program
5. Commit dan push ke github

### Hasil Kode Program (Output)

```
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Rena
Masukkan jumlah kopi: 3
Masukkan jumlah teh: 0
Masukkan jumlah roti: 1
Total yang harus dibayar: Rp 56000.0
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Yuni
Masukkan jumlah kopi: 1
Masukkan jumlah teh: 4
Masukkan jumlah roti: 2
Total yang harus dibayar: Rp 80000.0
Masukkan nama pelanggan (ketik 'batal' untuk keluar): batal
Transaksi dibatalkan
Semua transaksi selesai.
```

### Jawaban Pertanyaan

1. Apabila nama pelanggan yang dimasukkan pertama kali adalah “batal”, maka perulangan tetap dilakukan 1 kali
2. Kondisi berhenti yang digunakan adalah break
3. Nilai true membuat perulangan dijalankan tanpa batas waktu sampai dihentikan untuk menghentikannya yaitu dengan kondisi tertentu seperti break
4. Karena pada perulangan do-while perulangan akan terus dijalankan. Perulangan dijalankan setidaknya satu kali

# Tugas 1

## Kode Program

```
1 import java.util.Scanner;
2 public class Tugasno1 {
3
4     public static void main(String[] args) {
5
6         Scanner sc = new Scanner(System.in);
7
8         int jmlTiket, totalTiket=0;
9         double hargaTiket = 50000, totalHarga, diskon = 0, hargaAwal, totalHargaPenjualan=0 ;
10
11     do {
12         System.out.print("Masukkan jumlah tiket (ketik 0 untuk menghentikan transaksi): ");
13         jmlTiket = sc.nextInt();
14
15         if (jmlTiket == 0){
16             System.out.println("Transaksi dihentikan");
17             break;
18         }
19         if (jmlTiket < 0){
20             System.out.println("Jumlah tiket tidak valid. Masukkan ulang jumlah tiket");
21             continue;
22         }
23         if (jmlTiket > 4 && jmlTiket <= 10 ){
24             diskon = 0.1;
25         } else if (jmlTiket > 10){
26             diskon = 0.15;
27         }
28
29         hargaAwal = jmlTiket*hargaTiket;
30         totalHarga = hargaAwal-(hargaAwal*diskon);
31         totalTiket += jmlTiket;
32         totalHargaPenjualan += totalHarga;
33
34         System.out.println("Total harga untuk " + jmlTiket + " tiket: Rp " + totalHarga);
35     } while(true);
36
37     System.out.println("Total tiket yang terjual " + totalTiket);
38     System.out.println("Total harga penjualan tiket: Rp " + totalHargaPenjualan);
39
40     sc.close();
41 }
42 }
43
```

## Hasil (Output)

```
Masukkan jumlah tiket (ketik 0 untuk menghentikan transaksi): 2
Total harga untuk 2 tiket: Rp 100000.0
Masukkan jumlah tiket (ketik 0 untuk menghentikan transaksi): 4
Total harga untuk 4 tiket: Rp 200000.0
Masukkan jumlah tiket (ketik 0 untuk menghentikan transaksi): 11
Total harga untuk 11 tiket: Rp 467500.0
Masukkan jumlah tiket (ketik 0 untuk menghentikan transaksi): 0
Transaksi dihentikan
Total tiket yang terjual 17
Total harga penjualan tiket: Rp 767500.0
```

## Tugas 2

### Kode program

```
1  import java.util.Scanner;
2  public class Tugasno2 {
3
4      public static void main(String[] args) {
5
6          Scanner sc = new Scanner(System.in);
7
8          int jenis, durasi, total = 0;
9
10         do {
11             System.out.print("Masukkan jenis kendaraan (1 mobil, 2 motor, 0 keluar): ");
12             jenis = sc.nextInt();
13
14             if(jenis == 1 || jenis == 2 ){
15                 System.out.print("Masukkan durasi parkir (jam): ");
16                 durasi = sc.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                 }
24             }
25
26             System.out.println("Total pembayaran parkir " + total);
27         } while(jenis !=0);
28     }
29 }
30
```

### Hasil (Output)

```
Masukkan jenis kendaraan (1 mobil, 2 motor, 0 keluar): 2
Masukkan durasi parkir (jam): 3
Total pembayaran parkir 6000
Masukkan jenis kendaraan (1 mobil, 2 motor, 0 keluar): 1
Masukkan durasi parkir (jam): 2
Total pembayaran parkir 12000
Masukkan jenis kendaraan (1 mobil, 2 motor, 0 keluar): 0
Total pembayaran parkir 12000
```