

Nama : Rayhan Giri Putra  
NIM : 244107060122  
Kelas : SIB - 1E

## Percobaan 1

### Source Code

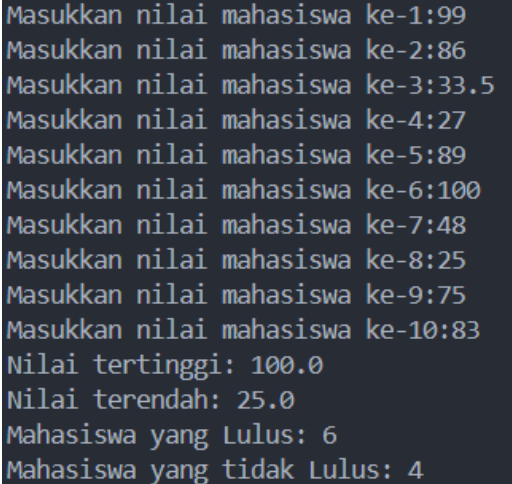
```
J SiakadFor14.java > SiakadFor14 > main(String[])
1  import java.util.Scanner;
2  public class SiakadFor14 {
3
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner sc = new Scanner(System.in);
7          double nilai, tertinggi = 0, terendah = 100;
8
9          for (int i = 1; i <= 10; i++ ) {
10             System.out.print("Masukkan nilai mahasiswa ke-" + i + ":");
11             nilai = sc.nextDouble();
12             if (nilai > tertinggi) {
13                 tertinggi = nilai;
14             }
15             if (nilai < terendah) {
16                 terendah = nilai;
17             }
18             System.out.println("Nilai tertinggi: " + tertinggi);
19             System.out.println("Nilai terendah: " + terendah);
20
21         }
22     }
```

### Result

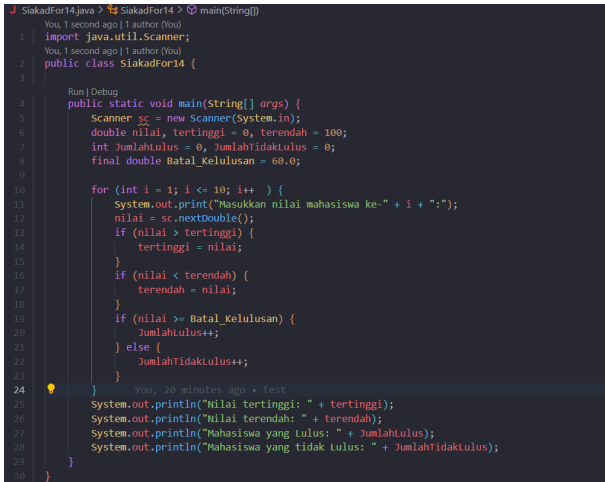
```
Masukkan nilai mahasiswa ke-1:99
Masukkan nilai mahasiswa ke-2:84
Masukkan nilai mahasiswa ke-3:24
Masukkan nilai mahasiswa ke-4:13
Masukkan nilai mahasiswa ke-5:47
Masukkan nilai mahasiswa ke-6:35
Masukkan nilai mahasiswa ke-7:100
Masukkan nilai mahasiswa ke-8:48
Masukkan nilai mahasiswa ke-9:59
Masukkan nilai mahasiswa ke-10:29
Nilai tertinggi: 100.0
Nilai terendah: 13.0
```

<https://github.com/Rayhannn18/jobsheet-2.0/blob/devel/SiakadFor14.java>

Jawaban :

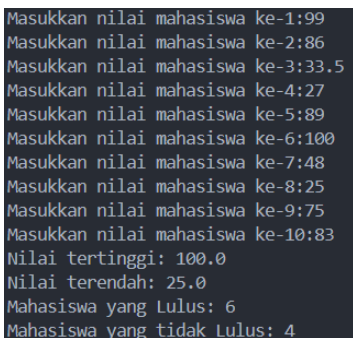
1. 

```
Masukkan nilai mahasiswa ke-1:99
Masukkan nilai mahasiswa ke-2:86
Masukkan nilai mahasiswa ke-3:33.5
Masukkan nilai mahasiswa ke-4:27
Masukkan nilai mahasiswa ke-5:89
Masukkan nilai mahasiswa ke-6:100
Masukkan nilai mahasiswa ke-7:48
Masukkan nilai mahasiswa ke-8:25
Masukkan nilai mahasiswa ke-9:75
Masukkan nilai mahasiswa ke-10:83
Nilai tertinggi: 100.0
Nilai terendah: 25.0
Mahasiswa yang Lulus: 6
Mahasiswa yang tidak Lulus: 4
```
2. Inisialisasi tertinggi dengan nilai rendah (0) dan terendah dengan nilai tinggi (100) adalah cara yang benar untuk memulai proses pencarian nilai maksimum dan minimum, karena ini memberi peluang untuk memperbarui nilai-nilai tersebut dengan data yang ada. Jika diinisialisasi sebaliknya, hasilnya bisa tidak akurat karena variabel tidak akan diperbarui sesuai dengan kondisi data yang sebenarnya.
3. Kode ini bertugas memperbarui dua variabel, yaitu tertinggi dan terendah, untuk selalu menyimpan nilai maksimum dan minimum selama iterasi berjalan. Pada akhir proses, variabel tertinggi akan menyimpan nilai terbesar dari semua nilai yang dicek, dan terendah akan menyimpan nilai terkecil.
4. source code



```
SiakadFor14.java > SiakadFor14 > main(String[])
You, 1 second ago | 1 author (You)
1  import java.util.Scanner;
2  You, 1 second ago | 1 author (You)
3  public class SiakadFor14 {
4
5      Run | Debug
6      public static void main(String[] args) {
7          Scanner sc = new Scanner(System.in);
8          double nilai, tertinggi = 0, terendah = 100;
9          int jumlahLulus = 0, jumlahTidakLulus = 0;
10         final double Batas_kelulusan = 60.0;
11
12         for (int i = 1; i <= 10; i++) {
13             System.out.print("Masukkan nilai mahasiswa ke-" + i + ":");
14             nilai = sc.nextDouble();
15             if (nilai > tertinggi) {
16                 tertinggi = nilai;
17             }
18             if (nilai < terendah) {
19                 terendah = nilai;
20             }
21             if (nilai >= Batas_kelulusan) {
22                 jumlahLulus++;
23             } else {
24                 jumlahTidakLulus++;
25             }
26         }
27         System.out.println("Nilai tertinggi: " + tertinggi);
28         System.out.println("Nilai terendah: " + terendah);
29         System.out.println("Mahasiswa yang lulus: " + jumlahLulus);
30         System.out.println("Mahasiswa yang tidak lulus: " + jumlahTidakLulus);
31     }
32 }
```

result



```
Masukkan nilai mahasiswa ke-1:99
Masukkan nilai mahasiswa ke-2:86
Masukkan nilai mahasiswa ke-3:33.5
Masukkan nilai mahasiswa ke-4:27
Masukkan nilai mahasiswa ke-5:89
Masukkan nilai mahasiswa ke-6:100
Masukkan nilai mahasiswa ke-7:48
Masukkan nilai mahasiswa ke-8:25
Masukkan nilai mahasiswa ke-9:75
Masukkan nilai mahasiswa ke-10:83
Nilai tertinggi: 100.0
Nilai terendah: 25.0
Mahasiswa yang Lulus: 6
Mahasiswa yang tidak Lulus: 4
```

5. <https://github.com/Rayhannn18/jobsheet-2.0/blob/devel/SiakadFor14.java>

percobaan kedua

source code

```
J SiakadWhile14.java > SiakadWhile14 > main(String[])
1 import java.util.Scanner;
2 public class SiakadWhile14 {
3
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         int Nilai, jml = 5, 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();
15
16            if (Nilai < 0 || Nilai > 100) {
17                System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid");
18                continue;
19            }
20
21            if (Nilai > 80 && Nilai <= 100) {
22                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A ");
23            }
24            else if (Nilai > 73 && Nilai <= 80) {
25                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B+");
26            }
27            else if (Nilai > 65 && Nilai <= 73) {
28                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B");
29            }
30            else if (Nilai > 60 && Nilai <= 65) {
31                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C+");
32            }
33            else if (Nilai > 50 && Nilai <= 60) {
34                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C");
35            }
36            else if (Nilai > 39 && Nilai <= 50) {
37                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah D");
38            }
39            else {
40                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah E");
41            }
42
43            i++;
44        }
45    }
46 }
47
```

Result

```
Masukkan jumlah mahasiswa: 5
Masukkan nilai mahasiswa ke-1: 98
Nilai mahasiswa ke-1 adalah A
Masukkan nilai mahasiswa ke-2: 56
Nilai mahasiswa ke-2 adalah C
Masukkan nilai mahasiswa ke-3: 87
Nilai mahasiswa ke-3 adalah A
Masukkan nilai mahasiswa ke-4: 75
Nilai mahasiswa ke-4 adalah B+
Masukkan nilai mahasiswa ke-5: 33
Nilai mahasiswa ke-5 adalah E
```

<https://github.com/Rayhannn18/jobsheet-2.0/blob/devel/SiakadWhile14.java>

Jawaban :

- fungsi tersebut berfungsi untuk mengidentifikasi variabel Nilai apakah variable tersebut kurang dari 0 atau lebih dari 100, jika ya maka akan mengeluarkan pesan kesalahan.
  - continue berfungsi untuk melanjutkan ke fungsi-fungsi selanjutnya.
- i++ di akhir perulangan memastikan bahwa proses input berjalan dari mahasiswa ke-1 hingga jumlah yang diinginkan sesuai dengan urutan. sedangkan, i++ di awal perulangan akan membuat perulangan dimulai dari mahasiswa berikutnya (melewatkan mahasiswa pertama), sehingga jumlah input tidak sesuai dengan yang seharusnya.
- Jika jumlah mahasiswa yang dimasukkan adalah 19, perulangan while akan berjalan sebanyak 19 kali (dari i = 0 hingga i = 18), dan berhenti ketika i = 19.
- source code

```
SiakadWhile14.java > SiakadWhile14 > main(String[])
You: 42 seconds ago | 1 author (You)
1 import java.util.Scanner;
You: 42 seconds ago | 1 author (You)
2 public class SiakadWhile14 {
3
4     Run [Debug]
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         int Nilai, jml = 5, 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();
15
16            if (Nilai < 0 || Nilai > 100) {
17                System.out.println("Nilai tidak valid. Masukkan lagi nilai yang valid");
18                continue;
19            }
20
21            if (Nilai > 80 && Nilai <= 100) {
22                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A ");
23                System.out.println("Bagus, pertahankan nilainya");
24            }
25            else if (Nilai > 73 && Nilai <= 80) {
26                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B+");
27            }
28            else if (Nilai > 65 && Nilai <= 73) {
29                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B");
30            }
31            else if (Nilai > 60 && Nilai <= 65) {
32                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C+");
33            }
34            else if (Nilai > 50 && Nilai <= 60) {
35                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C");
36            }
37            else if (Nilai > 39 && Nilai <= 50) {
38                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah D");
39            }
40            else {
41                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah E");
42            }
43            i++;
44        }
45    }
46 }
47
48
```

```
34 }
35
36
37 }
38
39
40 }
41
42
43
44 }
45
46 }
47
48
```

## Result

```
Masukkan jumlah mahasiswa: 5
Masukkan nilai mahasiswa ke-1: 97
Nilai mahasiswa ke-1 adalah A
Bagus, pertahankan nilainya
Masukkan nilai mahasiswa ke-2: 45
Nilai mahasiswa ke-2 adalah D
Masukkan nilai mahasiswa ke-3: 36
Nilai mahasiswa ke-3 adalah E
Masukkan nilai mahasiswa ke-4: 22
Nilai mahasiswa ke-4 adalah E
Masukkan nilai mahasiswa ke-5: 88
Nilai mahasiswa ke-5 adalah A
Bagus, pertahankan nilainya
```

- <https://github.com/Rayhannn18/jobsheet-2.0/blob/devel/SiakadWhile14.java>

## Percobaan ketiga

### Source code

```
J KafeDoWhile14.java > KafeDoWhile14 > main(String[])
2  public class KafeDoWhile14 {
3
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner sc = new Scanner(System.in);
7
8          int kopi, teh, roti;
9          String namaPelanggan;
10         int hargaKopi = 12000;
11         int hargaTeh = 7000;
12         int hargaRoti = 20000;
13         int totalHarga;
14
15         do {
16             System.out.print(s:"Masukkan nama pelanggan (ketik 'batal' untuk keluar): ");
17             namaPelanggan = sc.nextLine();
18             if (namaPelanggan.equalsIgnoreCase(anotherString:"batal")) {
19                 System.out.println(x:"Transaksi dibatalkan. ");
20                 break;
21             }
22             System.out.print(s:"Jumlah kopi: ");
23             kopi = sc.nextInt();
24             System.out.print(s:"Jumlah teh: ");
25             teh = sc.nextInt();
26             System.out.print(s:"Jumlah roti: ");
27             roti = sc.nextInt();
28             totalHarga = (kopi * hargaKopi) + (teh * hargaTeh) + (roti * hargaRoti);
29             System.out.println("Total yang harus dibayar: Rp " + totalHarga);
30             sc.nextLine();
31
32         }while (true);
33
34         System.out.println(x:"Semua transaksi selesai");
35         sc.close();
36
37     }
```

### Result

```
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Rena
Jumlah kopi: 4
Jumlah teh: 2
Jumlah roti: 6
Total yang harus dibayar: Rp 182000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Michael
Jumlah kopi: 7
Jumlah teh: 0
Jumlah roti: 5
Total yang harus dibayar: Rp 184000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Elsa
Jumlah kopi: 2
Jumlah teh: 5
Jumlah roti: 1
Total yang harus dibayar: Rp 79000
Masukkan nama pelanggan (ketik 'batal' untuk keluar): BATAL
Transaksi dibatalkan.
Semua transaksi selesai
```

### Jawaban :

1. Perulangan dilakukan tepat 1 kali, karena blok perulangan akan dijalankan sekali sebelum kondisi dievaluasi, meskipun nama yang dimasukkan adalah "batal" pada percobaan pertama.

2. Dalam perulangan DO-WHILE, kondisi berhenti ditentukan oleh ekspresi boolean yang diletakkan setelah kata kunci WHILE. Berikut adalah langkah-langkah untuk memahami kondisi berhenti dalam perulangan DO-WHILE:
  - **Eksekusi Awal:** Kode di dalam blok DO akan dieksekusi setidaknya sekali, tanpa memperhatikan kondisi.
  - **Kondisi Berhenti:** Setelah eksekusi blok DO, kondisi yang diletakkan di belakang WHILE akan dievaluasi. Jika kondisi ini bernilai benar (true), maka perulangan akan dilanjutkan ke iterasi berikutnya. Jika kondisi ini bernilai salah (false), maka perulangan akan berhenti.
3. Penggunaan nilai true pada kondisi dalam perulangan DO-WHILE berfungsi untuk menciptakan perulangan yang tidak akan pernah berhenti secara otomatis berdasarkan kondisi, melainkan akan terus berulang hingga ada intervensi di luar loop (seperti pernyataan break atau return).
4. perulangan DO-WHILE tetap berjalan tanpa inisialisasi dan update karena strukturnya yang memungkinkan eksekusi satu kali dan ketergantungan pada kondisi yang dievaluasi setelah eksekusi blok kode. Untuk menghindari infinite loop, penting untuk memastikan bahwa ada logika dalam blok DO yang dapat mengubah kondisi di WHILE sehingga perulangan dapat berhenti pada waktu yang tepat.

# TUGAS

## Tugas 1:

### Source code

```
J BioskopDoWhile14.java > BioskopDoWhile14
1  import java.util.Scanner;
2  public class BioskopDoWhile14 {
3
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner input = new Scanner(System.in);
7          int totalTiket = 0;
8          double totalPenjualan = 0;
9          int hargaTiket = 50000;
10         int jumlahTiket;
11
12         while (true) {
13             System.out.print(s:"Masukkan jumlah tiket yang terjual (ketik 0 untuk selesai): ");
14             jumlahTiket = input.nextInt();
15
16             if (jumlahTiket == 0) {
17                 break;
18             }
19
20             if (jumlahTiket < 0) {
21                 System.out.println(x:"Jumlah tiket tidak valid. Silakan masukkan ulang.");
22                 continue;
23             }
24
25             double totalHarga = jumlahTiket * hargaTiket;
26
27             if (jumlahTiket > 10) {
28                 totalHarga *= 0.85;
29                 System.out.println(x:"Anda mendapatkan diskon 15%");
30             } else if (jumlahTiket > 4) {
31                 System.out.println(x:"Anda mendapatkan diskon 10%");
32                 totalHarga *= 0.90;
33             }
34
35             totalTiket += jumlahTiket;
36             totalPenjualan += totalHarga;
37
38             System.out.println("Total harga tiket untuk " + jumlahTiket + " tiket: Rp " + totalHarga);
39         }
40         input.close();
41
42         System.out.println("\nTotal tiket yang terjual hari ini: " + totalTiket);
43         System.out.println("Total penjualan tiket: Rp " + totalPenjualan);
44     }
45 }
46
47
48
49
50
51
52
```

## Result :

```
Masukkan jumlah tiket yang terjual (ketik 0 untuk selesai): 4
Total harga tiket untuk 4 tiket: Rp 200000.0
Masukkan jumlah tiket yang terjual (ketik 0 untuk selesai): 5
Anda mendapatkan diskon 10%
Total harga tiket untuk 5 tiket: Rp 225000.0
Masukkan jumlah tiket yang terjual (ketik 0 untuk selesai): 56
Anda mendapatkan diskon 15%
Total harga tiket untuk 56 tiket: Rp 2380000.0
Masukkan jumlah tiket yang terjual (ketik 0 untuk selesai): 3
Total harga tiket untuk 3 tiket: Rp 150000.0
Masukkan jumlah tiket yang terjual (ketik 0 untuk selesai): 0

Total tiket yang terjual hari ini: 68
Total penjualan tiket: Rp 2955000.0
```

## Tugas 2

### Source code :

```
J Parkir14.java > Parkir14
1  import java.util.Scanner;
2  public class Parkir14 {
3
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner scanner = new Scanner(System.in);
7          int jenis, durasi, total = 0;
8
9          while (true) {
10             System.out.print(s:"Masukkan jenis kendaraan (1: Mobil, 2: Motor, 0: Keluar) : ");
11             jenis = scanner.nextInt();
12
13             if (jenis == 0) {
14                 System.out.println("Total pembayaran : Rp " + total);
15                 break;
16             }
17
18             if (jenis == 1 || jenis == 2) {
19                 System.out.print(s:"Masukkan durasi parkir (jam) : ");
20                 durasi = scanner.nextInt();
21                 total = 0;
22                 if (durasi > 5) {
23                     total += 12500;
24                 } else {
25                     if (jenis == 1) {
26                         total += durasi * 3000;
27                     } else if (jenis == 2) {
28                         total += durasi * 2000;
29                     }
30                 }
31                 System.out.println("Total pembayaran : Rp " + total);
32                 total = 0;
33
34                 } else {
35                     System.out.println(x:"Jenis kendaraan tidak valid. Silakan coba lagi.");
36                 }
37             }
38
39             scanner.close();
40         }
41     }
42
43     }
44 }
45 }
```



**Result :**

```
Masukkan jenis kendaraan (1: Mobil, 2: Motor, 0: Keluar) : 2
Masukkan durasi parkir (jam) : 24
Total pembayaran : Rp 12500
Masukkan jenis kendaraan (1: Mobil, 2: Motor, 0: Keluar) : 1
Masukkan durasi parkir (jam) : 24
Total pembayaran : Rp 12500
Masukkan jenis kendaraan (1: Mobil, 2: Motor, 0: Keluar) : 2
Masukkan durasi parkir (jam) : 18
Total pembayaran : Rp 12500
Masukkan jenis kendaraan (1: Mobil, 2: Motor, 0: Keluar) : 1
Masukkan durasi parkir (jam) : 35
Total pembayaran : Rp 12500
Masukkan jenis kendaraan (1: Mobil, 2: Motor, 0: Keluar) : 0
Total pembayaran : Rp 0
```