

Latihan Dasar Pemrograman Pertemuan 6

Nama :Rasyiq Satrio Musthafa

Kelas :TI 1G

No Absen:25

2.1.3 Pertanyaan:

1. Sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program Percobaan 1!
2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?
3. Jelaskan fungsi dan alur kerja dari potongan kode berikut!

```
if (nilai > tertinggi) {  
    tertinggi = nilai;  
}  
if (nilai < terendah) {  
    terendah = 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!
5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 1”

Jawab:

1. *(int i = 1)* Inisialisasi yang berfungsi untuk mendefinisikan variabel penghitung (counter)
(i <= 10) Kondisi dengan fungsi syarat agar perulangan tetap berjalan dengan jumlah yang ditentukan.
(i++) Perubahan increment untuk menambahkan bilangan *i* sebanyak 1 kali
2. Tertinggi tidak akan pernah berubah, karena semua nilai kamu (70, 80, 90) tidak ada yang lebih besar dari 100 → hasil akhirnya: tertinggi = 100 (SALAH).
Terendah juga tidak akan berubah, karena semua nilai lebih besar dari 0 → hasil akhirnya: terendah = 0 (SALAH).
Akibatnya, program tidak akan benar-benar mencatat nilai tertinggi dan terendah dari input pengguna.

- Potongan kode itu digunakan untuk membandingkan setiap nilai mahasiswa yang diinput dengan nilai tertinggi dan terendah yang sudah tersimpan sebelumnya.

```
jobsheet7 > .\SiakadFor25.java > SiakadFor25 > main(String[])
1 package jobsheet7;
2
3 import java.util.Scanner;
4
5 public class SiakadFor25 {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9         double nilai, tertinggi = 0, terendah = 100;
10        int lulus = 0, tdklulus = 0;
11
12        for (int i = 1; i <= 10; i++) {
13            System.out.print("masukan 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
22            if (nilai >= 60) {
23                lulus++;
24            } else {
25                tdklulus++;
26            }
27        }
28        System.out.println("Nilai tertinggi: " + tertinggi);
29        System.out.println("Nilai terendah: " + terendah);
30        System.out.println("Jumlah siswa lulus: " + lulus);
31        System.out.println("Jumlah siswa tidak lulus: " + tdklulus);
32    }
33 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
ItewMessages - -cp "C:\Users\IrsanPC\AppData\Local\Microsoft\Windows\WorkspacesStorage\workspaceStorage\005F3d7da4095b2
masukan nilai mahasiswa ke-1: 76,5
masukan nilai mahasiswa ke-2: 82,3
masukan nilai mahasiswa ke-3: 62,1
masukan nilai mahasiswa ke-4: 88,4
masukan nilai mahasiswa ke-5: 65,9
masukan nilai mahasiswa ke-6: 67,9
masukan nilai mahasiswa ke-7: 90,1
masukan nilai mahasiswa ke-8: 55,3
masukan nilai mahasiswa ke-9: 73,7
masukan nilai mahasiswa ke-10: 78,6
Nilai tertinggi: 90.1
Nilai terendah: 55.3
Jumlah siswa lulus: 9
Jumlah siswa tidak lulus: 1
PS E:\Github Utilities\PraktikumDaspro\PraktikumDaspro>
```

-
-
-
-
- Sudah.

2.2.3 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`
2. Mengapa sintaks `++` dituliskan di akhir perulangan `WHILE`? Apa yang terjadi jika posisinya dituliskan di awal perulangan `WHILE`?
 3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan `WHILE` akan berjalan?
 4. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya"!
 5. Commit dan push hasil modifikasi Anda ke Github dengan pesan "Modifikasi Percobaan 2"

Jawab:

1. a. Untuk mengatur batas agar nilai yang di input tidak melebihi 100 atau kurang dari 0
b. `continue` berfungsi untuk langsung loncat ke *next iteration* tanpa mengeksekusi kode yang ada setelah `continue` di dalam loop itu. `Continue` bukan keluar dari loop, tapi hanya melewati sisa perintah di iterasi yang sedang berjalan.
2. Karena jika nilai `i` ditempatkan di awal program `while`, perulangan tidak sesuai dengan yang diinginkan. contohnya jika menginput nilai mahasiswa pertama, yang ditunjukkan Adalah siswa ke-2 dan seterusnya.
3. Di program `while` pertama, Loop ini akan berjalan sampai `i` menjadi 19. Jadi berulang 19 kali. Saat masuk loop kedua, kondisi masih benar ($19 \leq 19$). Loop berjalan sekali, lalu `i` menjadi 20. Setelah itu kondisi salah ($20 \leq 19 = \text{false}$), sehingga berhenti. Maka, Jika jumlah mahasiswa = 19, maka total perulangan `while` yang dijalankan adalah 20 kali.

```
jobsheet7 > J siakadWhile25.java > $ siakadWhile25 > main(String[])
1 package jobsheet7;
2 import java.util.Scanner;
3
4 public class siakadWhile25 {
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7
8         int nilai, jml, i = 0;
9
10        System.out.print("masukan jumlah mahasiswa: ");
11        jml = sc.nextInt();
12
13        while (i < jml) {
14            i++;
15
16
17            while (i <= jml) {
18                System.out.print("masukan nilai mahasiswa ke-" + i + " :");
19                nilai = sc.nextInt();
20
21                if (nilai < 0 || nilai > 100) {
22                    System.out.println(x:"Nilai tidak valid. Masukan lagi nilai yang valid!");
23                    continue;
24                }
25                if (nilai > 80 && nilai <=100) {
26                    System.out.println("nilai mahasiswa ke-" + i + " adalah A \n Bagus, Pertahankan nilainya!");
27                } else if (nilai > 73 && nilai <=80) {
28                    System.out.println("Nilai mahasiswa ke-" + i + " adalah B+");
29                } else if (nilai > 65 && nilai <=73) {
30                    System.out.println("Nilai mahasiswa ke-" + i + " adalah B");
31                } else if (nilai > 60 && nilai <=65) {
32                    System.out.println("Nilai mahasiswa ke-" + i + " adalah C+");
33                } else if (nilai > 50 && nilai <=60) {
34                    System.out.println("Nilai mahasiswa ke-" + i + " adalah C");
35                } else if (nilai > 39 && nilai <=50) {
36                    System.out.println("Nilai mahasiswa ke-" + i + " adalah D");
37                } else {
38                    System.out.println("Nilai mahasiswa ke-" + i + " adalah E");
39                }
40                i++;
41            }
42        }
43    }
44 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\Github Utilities\PraktikumDaspro\PraktikumDaspro> e:: cd 'e:\Github Utilities\PraktikumDaspro\PraktikumDaspro'; &
ailsInExceptionMessages' '-cp' 'C:\Users\ironPC\AppData\Roaming\Code\User\workspaceStorage\db5f3dfda40556cd6a75b7e1599bc
nilai mahasiswa ke-1 adalah A
    Bagus, Pertahankan nilainya!
masukan nilai mahasiswa ke-2 :30
Nilai mahasiswa ke-2 adalah E
PS E:\Github Utilities\PraktikumDaspro\PraktikumDaspro> |
```

- 4.
5. Sudah

2.3.3 Pertanyaan

```
package jobseet7;
```

```
import java.util.Scanner;
```

```
public class kafeMakhluk2 {
```

```
    Run | Debug
```

```
    public static void main(String[] args) {
```

```
        Scanner sc = new Scanner(System.in);
```

```
        int kopi, teh, roti, hargaKopi = 12000, hargaTeh = 7000, hargaRoti = 6000;
```

```
        String namaPelanggan;
```

```
do{
```

```
    System.out.print("Masukan nama pelanggan (ketik \" batal\" untuk keluar): ");
```

```
    namaPelanggan = sc.nextLine();
```

```
    if (namaPelanggan.equalsIgnoreCase(anotherString:"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();
```

```
int totalHarga = (kopi * hargaKopi) + (teh * hargaTeh) + (roti * hargaRoti);
```

```
System.out.println("Total yang harus di bayar: Rp " + totalHarga);
```

```
sc.nextLine();
```

```
} while (true);
```

```
system.out.println("Semua transaksi selesai.");
```

```
PROBLEMS OUTPUT DEBUG-CONSOLE TERMINAL PORTS
```

```
Pg 8 : Github Utilities>PraktikumDespen>PraktikumDespen & '%Program Files%\Java\jdk-9\bin/java.exe' %*
```

```
%cd%>%cd%User>workspace>storage>dmsfddfdw@ncc.dia70.net1986c213>cmdsh .java|dt_wjvPraktikumDespen
```

```
Jumlah kopi: 2
```

```
Jumlah teh: 3
```

```
Jumlah roti: 4
```

```
Total yang harus di bayar: Rp 125000
```

```
Masukkan nama pelanggan (ketik " batal" untuk keluar): Renu
```

```
Jumlah kopi: 3
```

```
Jumlah teh: 0
```

```
Jumlah roti: 1
```

```
Total yang harus di bayar: Rp 59000
```

```
Masukkan nama pelanggan (ketik " batal" untuk keluar): Yuni.
```

```
Jumlah kopi: 1
```

```
Jumlah teh: 4
```

```
Jumlah roti: 2
```

```
Total yang harus di bayar: Rp 80000
```

```
Masukkan nama pelanggan (ketik " batal" untuk keluar): BATAL
```

```
transaksi dibatalkan.
```

```
Semua transaksi selesai.
```

1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah “batal”, maka berapa kali perulangan dilakukan?
2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!
3. Apa fungsi dari penggunaan nilai true pada kondisi DO-WHILE?
4. Mengapa perulangan DO-WHILE tersebut tetap berjalan meskipun tidak ada komponen inisialisasi dan update?

Jawab:

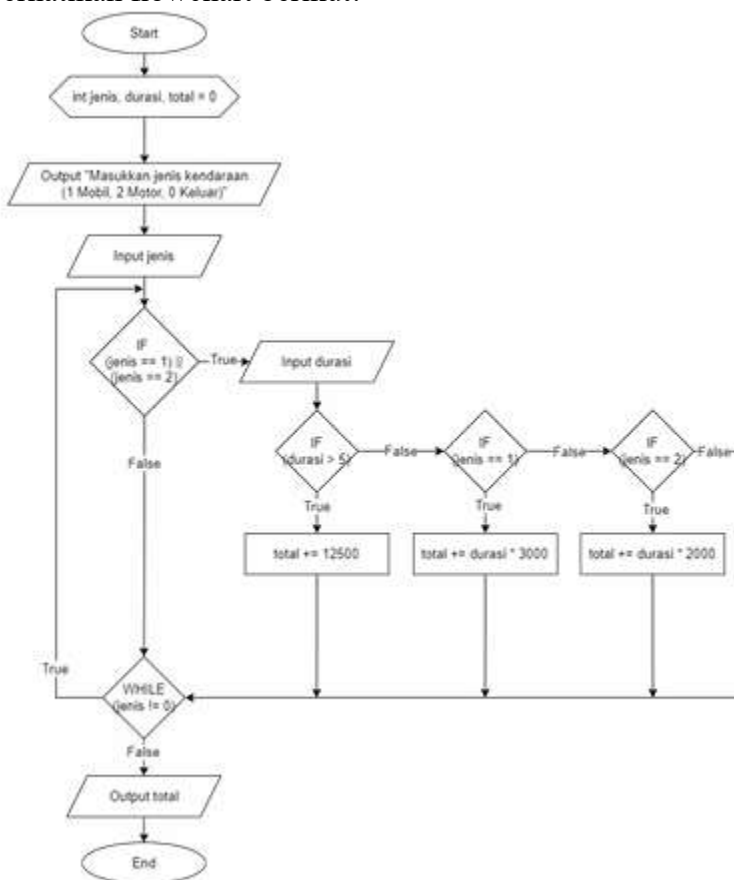
1. Tidak ada pengulangan yang dilakukan. Karena program langsung ke bagian while.
2. Saat pengguna memasukkan kata batal ataupun BATAL. Atau saat mengisi bagian integer memasukkan variable selain angka.
3. Penggunaan nilai true pada while(true) digunakan untuk memastikan perulangan berlangsung terus-menerus dan hanya berhenti berdasarkan kondisi jika variable yang diinputkan Adalah true(batal di dalam blok perulangan, bukan dari kondisi pada bagian while).
4. Perulangan do-while pada kode tersebut tetap berjalan karena kondisi pengulangannya selalu bernilai benar, yaitu while(true).

TUGAS 3.

- 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.

Commit dan push program Anda ke Github dengan pesan “Tugas 1”

- Perhatikan flowchart berikut!



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! Commit dan push program Anda ke Github dengan pesan “Tugas 2”