

JobSheet 7

Praktikum Dasar Pemrograman

Perulangan 1

Marvelino Husca

254107020184

Pertanyaan 1:

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

```
for (int i = 1; i <= 10; i++){
```

Int i = 1 adalah deklarasi i sebagai awalan dari pengulangan ini, lalu i <= 10 itu adalah kondisi dimana perulangan ini akan berhenti, lalu i++ menandakan bahwa perulangan ini akan bertambah dari yang kondisi awalnya 1 sampai kurang dari sama dengan 10

2. Mengapa variabel tertinggi diinisialisasi 0 dan terendah diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

Hal ini dimaksudkan untuk mencari nilai tertinggi dan dibandingkan dengan nilai kebalikkannya. Contoh untuk mencari nilai tertinggi menggunakan nilai 0 sebagai patokan, ketika ada nilai yang lebih dari 0 maka dianggap itu yang tertinggi lalu dimasukkan ke variabel tertinggi. Begitupun juga sebaliknya, nilai terendah patokannya kurang dari 100

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

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

Ini digunakan untuk mengecek apakah nilai itu lebih dari 0, jika iya maka akan dimasukkan ke variabel tertinggi, begitu juga dengan nilai itu kurang dari 100 atau nggak, itu akan masuk ke variabel terendah

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!

```

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    double nilai, tertinggi = 0, terendah = 100;
    int lulus = 0, tidaklulus = 0;

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

        if (nilai >= 0 && nilai <= 100) {
            if (nilai > tertinggi) {
                tertinggi = nilai;
            }
            if (nilai < terendah) {
                terendah = nilai;
            }

            if (nilai >= 60) {
                lulus += 1;
            } else {
                tidaklulus += 1;
            }
        } else {
            System.out.println("Nilai tidak valid");
            i--;
        }
    }

    System.out.println("Nilai tertinggi: " + tertinggi);
    System.out.println("Nilai terendah: " + terendah);
    System.out.println("lulus: " + lulus);
    System.out.println("tidaklulus: " + tidaklulus);
}

```

Pertanyaan 2:

1. Pada potongan kode berikut, tentukan maksud dan kegunaan dari sintaks berikut:

```

if (nilai < 0 || nilai > 100) {
    System.out.println("Nilai tidak valid. Masukkan lagi nilai yang valid!");
    continue;
}

```

- a. `nilai < 0 || nilai > 100`: ini digunakan untuk pengecekan apakah nilai diluar range 0 – 100, jika hasilnya minus atau diatas 100 maka code dibawahnya akan dijalankan
 - b. `continue`: ini berarti ketika nilai tidak valid, sistem akan mengulang-ulang terus sampai nilainya menjadi valid
2. Mengapa sintaks `i++` dituliskan di akhir perulangan WHILE? Apa yang terjadi jika posisinya dituliskan di awal perulangan WHILE?
Sintaks `i++` diperlukan agar nilai `i` bertambah sampai dengan kondisi yang telah ditentukan, di modul sampai nilai `i` nya kurang dari nilai mahasiswanya. Jika sintaks ini dituliskan di awal maka iterasi pertama akan di skip dan menuju ke iterasi ke dua
 3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan WHILE akan berjalan?
19 kali karena `i` dimulai dari 0, jika `i` dimulai dari 1 maka perulangan sampai 18
 4. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya!"

```

public class SiakadWhile13 {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int nilai, jml;
        int i = 0;

        System.out.print(s:"Masukkan jumlah mahasiswa: ");
        jml = sc.nextInt();

        while (i<jml){
            System.out.print("Masukkan nilai mahasiswa ke-" + (i+1) + ": ");
            nilai = sc.nextInt();

            if (nilai < 0 || nilai > 100){
                System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid");
                continue;
            }

            if (nilai > 80 && nilai <= 100){
                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah A\nBagus, pertahankan nilainya!");
            } else if (nilai > 73 && nilai <= 80){
                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B+");
            } else if (nilai > 65 && nilai <= 73){
                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah B");
            } else if (nilai > 60 && nilai <= 65){
                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C+");
            } else if (nilai > 50 && nilai <= 60){
                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah C");
            } else if (nilai > 39 && nilai <= 50){
                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah D");
            } else {
                System.out.println("Nilai mahasiswa ke-" + (i + 1) + " adalah E");
            }
            i++;
        }
    }
}

```

Pertanyaan 3

1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah “batal”, maka berapa kali perulangan dilakukan?
Perulangan tepat satu kali dilakukan
2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!
Break digunakan untuk menghentikan iterasi itu dan selanjutnya sehingga program berhenti
3. Apa fungsi dari penggunaan nilai true pada kondisi DO-WHILE?
Fungsi nya untuk terus mengulang perintah yang ada di do sampai ada break untuk mengakhiri loopnya
4. Mengapa perulangan DO-WHILE tersebut tetap berjalan meskipun tidak ada komponen inisialisasi dan update?
Karena perulangan DO-WHILE tidak bergantung pada jumlah perulangan nya dan bergantung pada user kapan program atau perulangan tersebut diakhiri