

Jobsheet 7

Nama: Rizqi Bagus Andrean

Absen: 25

Kelas: TI-1D

Percobaan 1

1. Terdapat tiga komponen perulangan pada sintaks FOR. Berdasarkan Percobaan 1 tersebut, sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program yang telah dibuat!

int i = 1 adalah inisialisasi

i <= 50 adalah kondisi

i++ update

2. Jelaskan alur kerja dari potongan kode program berikut!

```
for (int i = 1; i <= 50; i++) {  
    if (i % kelipatan == 0) {  
        jumlah += i;  
        counter++;  
    }  
}
```

Jadi program tersebut merupakan perulangan dimana perulangan di lakukan dari 1 sampai 50.

Didalam kode tersebut ada if dimana jika i di modulus dengan kelipatan dan sisanya adalah 0, maka jumlah akan di tambah dengan jumlah + i

Dan counternya akan ditambah 1

3. Modifikasi kode program yang telah dibuat dengan menambahkan variabel baru untuk menghitung rata-rata dari seluruh bilangan kelipatan yang ditentukan! Push dan commit kode program ke github.

```

import java.util.Scanner;

class ForKelipatan25{

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);

        int kelipatan, jumlah = 0, counter = 0;

        System.out.print("Masukkan bilangan kelipatan (1-9): ");
        kelipatan = scan.nextInt();

        for (int i = 1; i <= 50; i++) {
            if (i % kelipatan == 0) {
                jumlah += i;
                counter++;
            }
        }

        double avg = (double) jumlah / counter;

        System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter );
        System.out.printf("Jumlah bilangan kelipatan %d adalah %d\n", kelipatan, jumlah );
        System.out.print("Rata-rata bilangan kelipatan " + kelipatan + " adalah " + avg);

    }
}

```

4. Buatlah file baru dengan nama WhileKelipatanNoAbsen.java. Buatlah kode program dengan tujuan serupa tetapi menggunakan WHILE. Push dan commit kode program ke github.

```

import java.util.Scanner;

class WhileKelipatan25{

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);

        int kelipatan, jumlah = 0, counter = 0;

        System.out.print("Masukkan bilangan kelipatan (1-9): ");
        kelipatan = scan.nextInt();

        int i = 1;
        while (i <= 50) {
            if (i % kelipatan == 0) {
                jumlah += i;
                counter++;
            }
            i++;
        }

        double avg = (double) jumlah / counter;

        System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter );
        System.out.printf("Jumlah bilangan kelipatan %d adalah %d\n", kelipatan, jumlah );
        System.out.print("Rata-rata bilangan kelipatan " + kelipatan + " adalah " + avg);

    }
}

```

Percobaan 2

1. Tunjukkan bagian kode program yang digunakan sebagai syarat untuk menghentikan perulangan WHILE! Berapa kali perulangan dilakukan?

```

while (i < jumlahKaryawan) {

```

2. Pada potongan kode berikut, Apa yang sebenarnya terjadi jika variabel jabatan berisi nilai "DIREKTUR"? Apa peran CONTINUE yang dituliskan di dalam sintaks perulangan?

```
if (jabatan.equalsIgnoreCase("direktur")) {  
    continue;  
}
```

Jika berisii direktur maka dia akan menjalankan fungsi continue, Fungsi continue berfungsi untuk melanjutkan perulangan atau men-skip perulangan, sehingga dilanjutkan ke index berikutnya. Jadi setelah di skip dia tidak mengeksekusi kode dibawahnya didalam blok while.

3. Mengapa komponen update i++ diletakkan di posisi tengah, tidak di bagian akhir statement? Pindahkan i++ di bagian akhir, lalu jalankan kembali program dengan memasukkan "direktur" sebagai jabatan karyawan pertama. Apa yang terjadi? Jelaskan!

Jika diletakkan dibawah, apabila jabatan = direktur maka perulangan akan lanjut ke index berikutnya tanpa mengeksekusi i++; alhasil jumlah perulangan akan semakin banyak.

4. Modifikasi kode program untuk handle jabatan yang invalid seperti contoh berikut:

```
Masukkan jumlah karyawan: 3  
Pilihan jabatan - Direktur, Manajer, Karyawan  
Masukkan jabatan karyawan ke-1: manajer  
Masukkan jumlah jam lembur: 10  
Pilihan jabatan - Direktur, Manajer, Karyawan  
Masukkan jabatan karyawan ke-2: direktur  
Masukkan jumlah jam lembur: 5  
Pilihan jabatan - Direktur, Manajer, Karyawan  
Masukkan jabatan karyawan ke-3: pegawai  
Masukkan jumlah jam lembur: 4  
Jabatan invalid  
Pilihan jabatan - Direktur, Manajer, Karyawan  
Masukkan jabatan karyawan ke-3: karyawan  
Masukkan jumlah jam lembur: 4  
Total gaji lembur: 1300000.0
```

```

import java.util.Scanner;

public class WhileGaji25 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int jumlahKaryawan, jumlahJamLembur;
        double gajiLembur = 0, totalGajiLembur = 0;
        String jabatan;

        System.out.print("Masukkan jumlah karyawan: ");
        jumlahKaryawan = scan.nextInt();

        int i = 0;

        while (i < jumlahKaryawan) {
            System.out.print("Pilihan jabatan - Direktur, Manajer, Karyawan");
            System.out.print("Masukkan jabatan karyawan ke- " + (i + 1) + ": ");
            jabatan = scan.next();

            System.out.print("Masukkan jumlah jam lembur: ");
            jumlahJamLembur = scan.nextInt();
            i++;

            if (jabatan.equalsIgnoreCase("direktur")){
                continue;
            } else if (jabatan.equalsIgnoreCase("manajer")){
                gajiLembur = jumlahJamLembur * 100000;
            } else if (jabatan.equalsIgnoreCase("karyawan")) {
                gajiLembur = jumlahJamLembur * 75000;
            } else{
                i--;
                System.out.println("Jabatan invalid");
                continue;
            }

            totalGajiLembur += gajiLembur;
        }

        System.out.print("Total gaji lembur: " + totalGajiLembur );
    }
}

```

Percobaan 3

1. Apa kegunaan sintaks BREAK di dalam sintaks perulangan?
Break digunakan untuk menghentikan perulangan
2. Modifikasi kode program sehingga jika jumlah hari cuti yang ingin diambil lebih besar daripada jatah yang tersisa, program tidak berhenti sehingga pengguna masih memiliki kesempatan untuk mengisi jumlah hari sesuai jatah cuti.

```

import java.util.Scanner;

public class DoWhileCuti25 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int jatahCuti, jumlahHari;
        String konfirmasi;

        System.out.print("Masukkan jatah cuti: ");
        jatahCuti = sc.nextInt();

        do{
            System.out.print("Apakah anda ingin mengambil cuti (y/t) ? ");
            konfirmasi = sc.next();

            if (konfirmasi.equalsIgnoreCase("y")){
                System.out.print("Jumlah hari: ");
                jumlahHari = sc.nextInt();

                if (jumlahHari <= jatahCuti) {
                    jatahCuti -= jumlahHari;
                    System.out.println("Sisa jatah cuti anda adalah " + jatahCuti);

                    if (jatahCuti == 0) {
                        System.out.println("Jatah cuti anda sudah habis");
                        break;
                    }
                }
            }else{
                System.out.println("Sisa jatah cuti anda tidak mencukupi, tinggal sisa: " +
                    jatahCuti + " hari");
            }
        }while(jatahCuti > 0);
    }
}

```

3. Push dan commit kode program ke github
4. Pada saat input konfirmasi, ketikkan "t", apa yang terjadi? Mengapa demikian?
Program akan terus berjalan, karena tidak ada fungsi break
5. Modifikasi kode program sehingga saat pengguna mengetikkan "t" sebagai input konfirmasi, maka program akan berhenti

```

import java.util.Scanner;

public class DoWhileCuti25 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int jatahCuti, jumlahHari;
        String konfirmasi;

        System.out.print("Masukkan jatah cuti: ");
        jatahCuti = sc.nextInt();

        do{
            System.out.print("Apakah anda ingin mengambil cuti (y/t) ? ");
            konfirmasi = sc.next();

            if (konfirmasi.equalsIgnoreCase("y")){
                System.out.print("Jumlah hari: ");
                jumlahHari = sc.nextInt();

                if (jumlahHari <= jatahCuti) {
                    jatahCuti -= jumlahHari;
                    System.out.println("Sisa jatah cuti anda adalah " + jatahCuti);

                    if (jatahCuti == 0) {
                        System.out.println("Jatah cuti anda sudah habis");
                        break;
                    }
                }
            }else{
                System.out.println("Sisa jatah cuti anda tidak mencukupi, tinggal sisa: " +
                    jatahCuti + " hari");
                break;
            }
        }while(jatahCuti > 0);
    }
}

```

6. Push dan commit kode program ke github

Tugas

```
public class DeretBil25 {  
    public static void main(String[] args) {  
        int i = 25;  
        int total = 0;  
  
        while (i ≥ 1) {  
            total += i;  
            i--;  
        }  
  
        System.out.print("Total deret bilangan 25: " + total);  
    }  
}
```

```
public class Deret1sampai50 {  
    public static void main(String[] args)  
    {  
        for (int i = 1; i ≤ 50; i++) {  
            if (i % 3 ≠ 0) {  
                System.out.print(i + " ");  
            }  
        }  
    }  
}
```



```
public class Deret1sampai50 {  
    public static void main(String[] args)  
    {  
        for (int i = 1; i ≤ 50; i++) {  
            if (i % 3 ≠ 0) {  
                System.out.print(i + " ");  
            }  
        }  
    }  
}
```