

Pertanyaan 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!

Jawaban: inisiasi : `int i = 1`; kondisi : `i <= 50`; update : `i++`; statement : `if (i % kelipatan == 0)`

2. Jelaskan alur kerja dari potongan kode program berikut!

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

Jawaban: nilai awal i adalah 1, Program akan menghitung berulang sampai kurang atau sama dengan 50 jika modul dari nilai i dengan nilai kelipatan jika modulnya sama dengan 0 maka program akan menambah nilai total dan counter

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.

4. Jawaban:

```
rata_rata = (double) total/counter;
```

```
• Masukkan bilangan kelipatan (1-9): 5  
Banyaknya bilangan 5 dari 1 sampai 50 adalah 10  
Total bilangan kelipatan 5 dari 1 sampai 50 adalah 275  
Rata Rata bilangan kelipatan 5 dari 1 sampai 50 adalah 27.5
```

5. 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;  
  
public class WhileKelipatanNoAbsen {  
    public static void main (String args[]) {  
        Scanner input02 = new Scanner(System.in);  
  
        int kelipatan, total = 0, jumlah = 0, counter = 0;  
        double rata_rata;  
  
        System.out.print("Masukkan bilangan kelipatan (1-9): ");  
        kelipatan = input02.nextInt();  
  
        int i = 1;  
  
        while (i <= 50) {  
            i++;  
            if (i % kelipatan == 0) {  
                total += i;  
                counter++;  
            }  
        }  
  
        rata_rata = (double) total/counter;  
  
        System.out.println("Banyaknya bilangan " + kelipatan + " dari 1 sampai 50 adalah " + counter);  
        System.out.println("Total bilangan kelipatan " + kelipatan + " dari 1 sampai 50 adalah " + total);  
        System.out.println("Rata Rata bilangan kelipatan " + kelipatan + " dari 1 sampai 50 adalah " + rata_rata);  
    }  
}
```

Pertanyaan 2

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

```
int i = 0;
while (i < jumlah_karyawan ) {
```

Jawaban: program akan berhenti jika nilai “jumlah_karyawan” lebih kecil dari nilai “i “

2. Pada potongan kode berikut,

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

Apa yang sebenarnya terjadi jika variabel jabatan berisi nilai “DIREKTUR”? Apa peran CONTINUE yang dituliskan di dalam sintaks perulangan?

Jawaban: Karna terdapat code equalsIgnoreCase maka program akan tidak memperdulikan besar kecilnya huruf dan tetap masuk kedalam kondisi tersebut, fungsi continue adalah untuk melanjutkan program karna kondisi “direktur” tidak memiliki rumus atau perhitungan.

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!

Jawaban: Jika i++ diletakkan diakhir akan terikat dengan kondisi if else pada program dan karna pada kondisi “direktur” terdapat continue maka program akan infinite looping.

4. Modifikasi kode program untuk menghandle 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
```

Jawaban: Dengan menambah kondisi else pada kondisi akhir

```
    } else {  
        System.out.println(x:"Jabatan tidak ditemukan");  
    }  
}
```

```
PS C:\Users\noval\Documents\MatkulDASPRO\jobsheet\jobsheet7> java WhileGaji02.java  
Masukkan jumlah karyawan: 1  
Pilihan Jabatan - Direktur, Manajer, Karyawan  
Masukkan Jabatan Karyawan ke-1: pegawai  
Masukkan Jumlah Jam Lembur: 1  
Jabatan tidak ditemukan  
Total Gaji Lembur: 0.0
```

5. Push dan commit kode program ke github

Pertanyaan 3

1. Apa kegunaan sintaks BREAK di dalam sintaks perulangan?

Jawaban: Untuk mengakhiri program agar tidak berlanjut pada kondisi berikutnya

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.

Jawaban:

```
import java.util.Scanner;

public class DoWhileCuti02{
    Run | Debug
    public static void main (String args[]){

        Scanner input02 = new Scanner(System.in);

        int jatahCuti, jumlahHari;
        String konfirmasi;

        System.out.print(s:"Jatah Cuti: ");
        jatahCuti = input02.nextInt();

        do {

            System.out.print(s:"Apakah Anda ingin mengambil cuti (y/t)? ");
            konfirmasi = input02.next();

            if (konfirmasi.equalsIgnoreCase(anotherString:"y")) {
                System.out.print(s:"Jumlah hari: ");
                jumlahHari = input02.nextInt();

                if (jumlahHari <= jatahCuti) {
                    jatahCuti -= jumlahHari;
                    System.out.println("Sisa jatah cuti: " + jatahCuti);
                } else {
                    System.out.print(s:"Sisa Jatah Cuti Anda Tidak Mencukupi\nApakah Anda ingin menambah jatah cuti (y/t)? ");
                    konfirmasi = input02.next();

                    if (konfirmasi.equalsIgnoreCase(anotherString:"y")) {
                        System.out.print(s:"Jatah Cuti: ");
                        jatahCuti = input02.nextInt();
                        continue;
                    } else {
                        break;
                    }
                }
            } else {
                break;
            }
        } while (jatahCuti > 0);
    }
}
```

3. Push dan commit kode program ke github
4. Pada saat input konfirmasi, ketikkan "t", apa yang terjadi? Mengapa demikian?

Jawaban: Program akan terus berulang karna tidak memiliki kondisi saat menginput "t"

5. Modifikasi kode program sehingga saat pengguna mengetikkan "t" sebagai input konfirmasi, maka program akan berhenti

Jawaban: Dengan menambah kondisi else dan break pada akhir program

```
}else{  
    break;  
}  
while (jatahCutu > 0);
```

6. Push dan commit kode program ke github

Tugas

- Implementasikan flowchart yang telah dibuat pada tugas pertemuan 7 Matakuliah Dasar Pemrograman terkait project ke dalam kode program
- Push dan commit hasil kode program anda ke repository project Anda
- Catatan: tugas hanya boleh menerapkan materi dari pertemuan 1 hingga pertemuan 7.