

Laporan Hasil Praktikum



Aryo Adi Putro
1H Teknik Informatika
2341720084

Jl. Soekarno Hatta No. 9, Jaitmulyo, Kec. Lowokwaru, Kota Malang
Telp. : 0341-404424, 404425
E-Mail : www.polinema.ac.id

Jobsheet 7

Perulangan 1

1. Tujuan

- Mahasiswa dapat menjelaskan format penulisan program perulangan bagian 1
- Mahasiswa dapat mengimplementasikan flowchart perulangan bagian 1 menggunakan bahasa pemrograman Java

2. Praltikum

2.1 Percobaan 1: Menghitung Bilangan Kelipatan Menggunakan FOR

- Kode program

```
package jobSheet7.src;

import java.util.*;
public class ForPerulangan06 {
    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++;
            }
        }
        System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter);
        System.out.printf("Total bilangan kelipatan %d dari 1 sampai 50 adalah %d\n", kelipatan, jumlah);

        scan.close();
    }
}
```

- Output

```
PS C:\Kuliah\Coding Tugas> & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:50746' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Aorys\AppData\Roaming\Code\User\workspaceStorage\9370c1a1b36c24efe96c8fe037320c87\redhat.java\jdt_ws\Coding Tugas_c8fbfc4b\bin' 'jobSheet7.src.ForPerulangan06'
Masukkan bilangan kelipatan (1-9) : 3
Banyaknya bilangan 3 dari 1 sampai 50 adalah 16
Total bilangan kelipatan 3 dari 1 sampai 50 adalah 408
PS C:\Kuliah\Coding Tugas>
```

- Commit

- Pertanyaan
- 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!

- 2.

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

Inisialisasi : deklarasi dan inisialisasi variabel counter (variabel pengontrol perulangan)

Kondisi : batas atau syarat agar perulangan tetap dieksekusi

Update : perubahan nilai variabel counter pada setiap putaran perulangan (increment atau decrement)

- 3. Jelaskan alur kerja dari potongan kode program berikut!

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

- Menginisialisasi variable i terlebih dahulu, lalu memeriksa variabel apakah sesuai dengan kondisi yang ada. Jika kondisi bernilai true maka akan dilanjutkan ke pemeriksaan kondisi kedua. Dan jika bernilai true, mengeksekusi pernyataan yang ada. Kemudian dilakukan increment terhadap variable i. Terakhir dilakukan perulangan ke kondisi awal hingga kondisi tidak terpenuhi.
- 4. 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.

```

package jobSheet7.src.modif;

import java.util.*;
public class ForPerulangan061 {
    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 rata = (double) jumlah / counter;
        System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter);
        System.out.printf("Total bilangan kelipatan %d dari 1 sampai 50 adalah %d\n", kelipatan, jumlah);
        System.out.printf("Rata-rata bilangan kelipatan %d dari 1 sampai 50 adalah %.2f\n", kelipatan, rata);

        scan.close();
    }
}

```

```

PS C:\Kuliah\Coding Tugas> & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:53352' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Aorys\AppData\Roaming\Code\User\workspaceStorage\9370c1a1b36c24efe96c8fe037320c87\redhat.java\jdt_ws\Coding Tugas_c8fbfc4b\bin' 'jobSheet7.src.modif.ForPerulangan061'
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 27.50
PS C:\Kuliah\Coding Tugas>

```

ForPerulangan061.java

modif perulangan for

now

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

```

package jobSheet7.src.modif;

import java.util.*;
public class WhilePerulangan06 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int kelipatan, counter=0, jumlah=0;
        double rata;

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

        int i=1;
        while (i <= 50) {
            i++;
            if (i % kelipatan == 0) {
                jumlah += i;
                counter++;
            }
        }
        rata = (double) jumlah / counter;
        System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter);
        System.out.printf("Total bilangan kelipatan %d dari 1 sampai 50 adalah %d\n", kelipatan, jumlah);
        System.out.printf("Rata-rata bilangan kelipatan %d dari 1 sampai 50 adalah %.2f\n", kelipatan, rata);

        sc.close();
    }
}

```

```

PS C:\Kuliah\Coding Tugas> c:: cd 'c:\Kuliah\Coding Tugas'; & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:53394' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Aorys\AppData\Roaming\Code\User\workspaceStorage\9370c1a1b36c24efe96c8fe037320c87\redhat.java\jdt_ws\Coding Tugas_c8fbfc4b\bin' 'jobSheet7.src.modif.WhilePerulangan06'
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 27.50
PS C:\Kuliah\Coding Tugas>

```

WhilePerulangan06.java

modif perulangan for

now

2.2 Percobaan 2 : Menghitung Gaji Lembur Karyawan Menggunakan WHILE dan CONTINUE

- Kode program

```
package jobSheet7.src;

import java.util.*;
public class WhileGaji06 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);

        int jumlahKaryawan, jumlahJamLembur;
        double gajiLembur=0, totalGajiLembur=0;
        String jabatan;

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

        int i = 0;
        while (i < jumlahKaryawan) {
            System.out.println("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;
            }

            totalGajiLembur += gajiLembur;
        }

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

        scan.close();
    }
}
```

- Output

```
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-1: direktur
Masukkan jumlah jam lembur : 12
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-2: manajer
Masukkan jumlah jam lembur : 6
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-3: karyawan
Masukkan jumlah jam lembur : 3
Total gaji lembur : 825000.0
PS C:\Kuliah\Coding Tugas>
```

- Commit

WhileGaji06.java

pertemuan ke 7

now

- Pertanyaan
1. Tunjukkan bagian kode program yang digunakan sebagai syarat untuk menghentikan perulangan WHILE! Berapa kali perulangan dilakukan?
- while (i < jumlahKaryawan). Pengulangan dilakukan tergantung dari input jumlah karyawan yang dimasukkan oleh pengguna.
2. Pada potongan kode berikut, Apa yang sebenarnya terjadi jika variabel jabatan berisi nilai “DIREKTUR”? Apa peran CONTINUE yang dituliskan di dalam sintaks perulangan?

```
Masukkan jumlah karyawan: 3
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-1: manajer
Masukkan jumlah jam lembur: 1
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-2: direktur
Masukkan jumlah jam lembur: 10
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-3: karyawan
Masukkan jumlah jam lembur: 5
Total gaji lembur: 475000.0
```

- Jika diisi nilai direktur program akan tetap berjalan dan tidak error karena menggunakan equalsIgnoreCase. Peran Continue adalah untuk melewati statement yang ada di bagian Direktur dan melanjutkannya ke pernyataan selanjutnya.
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!
- Input dari jabatan dan jam lembur karyawan akan muncul terus menerus tanpa henti. Hal ini terjadi karena nilai dari variabel i yang tidak di increment, tidak berada di posisi update. Yang mengakibatkan nilai dari variabel i akan terus 0 dan tidak akan menjadi lebih dari nilai variabel jumlahKaryawan.
4. Modifikasi kode program untuk menghandle jabatan yang invalid seperti contoh berikut:

```

package jobSheet7.src;

import java.util.*;
public class WhileGaji06 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);

        int jumlahKaryawan, jumlahJamLembur;
        double gajiLembur=0, totalGajiLembur=0;
        String jabatan;

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

        int i = 0;

        while (i < jumlahKaryawan) {
            System.out.println("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 {
                System.out.println("Jabatan invalid");
                i--;
                continue;
            }

            totalGajiLembur += gajiLembur;
        }

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

        scan.close();
    }
}

```

5. Push dan commit kode program ke github

WhileGaji06.java

modif while

now

2.3 Percobaan 3 : Menghitung Jatah Cuti Menggunakan DO-WHILE

- Kode program

```
package jobSheet7.src;

import java.util.*;
public class DoWhileCuti06 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int jatahCuti, jumlahHari;
        String konfirmasi;

        System.out.println("Jatah cuti : ");
        jatahCuti = sc.nextInt();

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

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

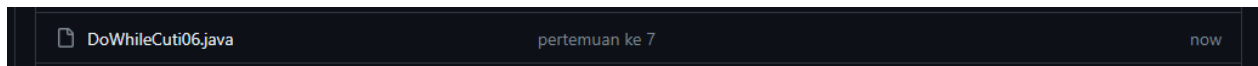
                if (jumlahHari <= jatahCuti) {
                    jatahCuti -= jumlahHari;
                    System.out.println("Sisa jatah cuti : " + jatahCuti);
                } else {
                    System.out.println("Sisa jatah cuti Anda tidak mencukupi");
                    break;
                }
            }
        } while (jatahCuti > 0);

        sc.close();
    }
}
```

- Output

```
PS C:\Kuliah\Coding Tugas> & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:50823' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Aorys\AppData\Roaming\Code\User\workspaceStorage\9370c1a1b36c24efe96c8fe037320c87\redhat.java\jdt_ws\Coding Tugas_c8fbfc4b\bin' 'jobSheet7.src.DoWhileCuti06'
Jatah cuti :
12
Apakah Anda ingin mengambil cuti (y/t)?
y
Jumlah hari : 3
Sisa jatah cuti : 9
Apakah Anda ingin mengambil cuti (y/t)?
y
Jumlah hari : 5
Sisa jatah cuti : 4
Apakah Anda ingin mengambil cuti (y/t)?
y
Jumlah hari : 4
Sisa jatah cuti : 0
```

- Commit



- Pertanyaan
 1. Apa kegunaan sintaks BREAK di dalam sintaks perulangan?
- Untuk menghentikan perulangan secara paksa.
- 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.

```
package jobSheet7.src.modif;

import java.util.*;
public class DoWhileCuti061 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int jatahCuti, jumlahHari;
        String konfirmasi;

        System.out.println("Jatah cuti : ");
        jatahCuti = sc.nextInt();

        do {
            System.out.println("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 : " + jatahCuti);
                } else {
                    System.out.println("Sisa jatah cuti Anda tidak
mencukupi");
                }
            }
        } while (jatahCuti > 0);

        sc.close();
    }
}
```

3. Push dan commit kode program ke github



4. Pada saat input konfirmasi, ketikkan "t", apa yang terjadi? Mengapa demikian?

- Akan terjadi perulangan karena tidak diberikan kondisi khusus jika memilih “t”.
5. Modifikasi kode program sehingga saat pengguna mengetikkan “t” sebagai input konfirmasi, maka program akan berhenti

```
package jobSheet7.src.modif;

import java.util.*;
public class DoWhileCuti061 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int jatahCuti, jumlahHari;
        String konfirmasi;

        System.out.println("Jatah cuti : ");
        jatahCuti = sc.nextInt();

        do {
            System.out.println("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 : " + jatahCuti);
                } else {
                    System.out.println("Sisa jatah cuti Anda tidak mencukupi");
                }
            } else if (konfirmasi.equalsIgnoreCase("t")) {
                break;
            }
        } while (jatahCuti > 0);

        sc.close();
    }
}
```

6. Push dan commit kode program ke github

DoWhileCuti061.java

modif cuti

13 minutes ago

3. 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