

LAPORAN PRAKTIKUM DASAR PEMROGRAMAN
JOB SHEET 7: PERULANGAN 1



Nama: Afifah Khoirunnisa

NIM: 2341720250

Kelas: 1B

Prodi : D-IV Teknik Informatika

2.1 Percobaan 1: Menghitung Bilangan Kelipatan Menggunakan FOR

Source Code:

```
import java.util.Scanner; //Menambah library scanner untuk user dapat menginput data

public class ForKelipatan03 {
    Run | Debug
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int kelipatan, counter, total; //Inisialisasi variabel menggunakan tipe data int
        total = 0;
        counter = 0;


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




        //Menggunakan perulangan for dengan dimulai inisialisasi i = 1, kondisi jika i <= 50, dan menggunakan i increment
        //Menggunakan if jika nilai i dibagi kelipatan sama dengan 0, maka total akan bertambah 1, dan mulai menghitung
        for (int i = 1; i <= 50; i++) {
            if (i % kelipatan == 0) {
                total += i;
                counter++;
            }
        }
        System.out.printf(format:"Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter);
        System.out.printf(format:"Total bilangan kelipatan %d dari 1 sampai 50 adalah %d\n", kelipatan, total);
    }
}
```

Output:


```
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM PEMROGRAMAN DASAR\PRAKTIKUM\JOBSHEET 7> javac ForKelipatan03.java
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM PEMROGRAMAN DASAR\PRAKTIKUM\JOBSHEET 7> java ForKelipatan03
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
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM PEMROGRAMAN DASAR\PRAKTIKUM\JOBSHEET 7> |
```


Push ke- github:

 daspro-jobsheet7 Public Pin Unwatch 1

 main  1 branch  0 tags

Go to file Add file <> Code

 afifahnis17 For Kelipatan a07f778 now 2 commits

 ForKelipatan03.java

For Kelipatan

now

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!

Jawaban:

Tiga komponen dalam for yaitu:

- a. Inisialisasi : Untuk mengatur nilai awal yang akan digunakan untuk menginisialisasi variabel.
- b. Kondisi : Untuk menentukan kapan suatu perulangan terus berjalan atau berhenti.
- c. Increment : Untuk mengubah nilai variabel control perulangan setiap kali perulangan di eksekusi.

2. Jelaskan alur kerja dari potongan kode program berikut!

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

Jawaban:

Alur kerja dari potongan kode adalah sebagai berikut.

1. Sebuah variabel `i` diinisialisasi dengan nilai 1.
 2. Kemudian, perulangan `for` dimulai. Perulangan ini akan terus berjalan selama nilai `i` kurang dari atau sama dengan 50 (`i <= 50`).
 3. Pada setiap iterasi perulangan, kondisi `if` diperiksa. Kondisi ini memeriksa apakah nilai `i` adalah kelipatan dari suatu bilangan yang disimpan dalam variabel `kelipatan`. Jika nilai `i` memenuhi kondisi tersebut (yaitu, jika `i` adalah kelipatan dari `kelipatan`), maka langkah-langkah berikut dilakukan:
 - i. Nilai `i` ditambahkan ke variabel `total`.
 - ii. Variabel `counter` (yang mungkin telah diinisialisasi sebelumnya) ditingkatkan sebesar 1.
 4. Setelah langkah-langkah di atas selesai dieksekusi untuk iterasi tertentu, nilai `i` ditingkatkan sebesar 1 (`i++`) dan perulangan berlanjut ke iterasi berikutnya.
 5. Proses ini akan terus berlanjut hingga nilai `i` mencapai 50 atau lebih. Setelah itu, perulangan akan berhenti.
 6. Setelah perulangan selesai, variabel `total` akan berisi jumlah dari semua bilangan yang merupakan kelipatan dari `kelipatan` dalam rentang 1 hingga 50, dan variabel `counter` akan berisi jumlah bilangan tersebut.
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.

Jawaban;

Source code:

```
double rata2 =(double) total/jumlahKelipatan; // menambah variabel baru untuk menghitung rata-rata
System.out.printf(format:"Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter);
System.out.printf(format:"Total bilangan kelipatan %d dari 1 sampai 50 adalah %d\n", kelipatan, total);
System.out.printf("Rata-rata bilangan kelipatan "+ kelipatan + " dari 1 sampai 50 adalah " + rata2 );
```

Output:

```
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
```

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

Jawaban:

Source Code:

```
1 import java.util.Scanner; //Menambah library scanner untuk user dapat menginput data
2
3 public class WhileKelipatan03 {
4     public static void main(String[] args) {
5         Scanner scan = new Scanner(System.in);
6         int kelipatan, counter, total; //Inisialisasi variabel menggunakan tipe data int
7         int jumlahKelipatan = 0;
8         total = 0;
9         counter = 0;
10
11         System.out.println("Masukkan bilangan kelipatan (1-9) : ");
12         kelipatan = scan.nextInt();
13
14         int i = 1;
15         while (i <= 50) {
16             if (i % kelipatan == 0) {
17                 total += i;
18                 counter++;
19             }
20             i++;
21         }
22
23         i = 1;
24         while (i <= 50) {
25             if (i % kelipatan == 0) {
26                 jumlahKelipatan++;
27             }
28             i++;
29         }
30
31         double rata2 =(double) total/jumlahKelipatan; // menambah variabel baru untuk menghitung rata-rata
32         System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter);
33         System.out.printf("Total bilangan kelipatan %d dari 1 sampai 50 adalah %d\n", kelipatan, total);
34         System.out.printf("Rata-rata bilangan kelipatan "+ kelipatan + " dari 1 sampai 50 adalah " + rata2 );
35     }
36 }
37 }
38
```

Output:

```
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
```

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

Source Code :

```
1  import java.util.Scanner;
2
3  public class WhileGaji03 {
4      public static void main(String[] args) {
5          Scanner scan = new Scanner(System.in);
6          int jumlahKaryawan, jumlahJamLembur;
7          double gajiLembur, totalGajiLembur;
8          gajiLembur = 0;
9          totalGajiLembur = 0;
10         String jabatan;
11
12         System.out.println("Masukkan jumlah karyawan: ");
13         jumlahKaryawan = scan.nextInt();
14         scan.nextLine(); // Membersihkan newline character dari buffer
15
16         int i = 0;
17
18         while (i < jumlahKaryawan) {
19             System.out.println("Pilihan jabatan - Direktur, Manajer, Karyawan");
20             System.out.println("Masukkan jabatan karyawan ke-" + (i + 1) + ": ");
21             jabatan = scan.nextLine();
22             System.out.println("Masukkan jumlah jam lembur: ");
23             jumlahJamLembur = scan.nextInt();
24             scan.nextLine(); // Membersihkan newline character dari buffer
25             i++;
26
27             if (jabatan.equalsIgnoreCase("direktur")) {
28                 continue;
29             } else if (jabatan.equalsIgnoreCase("manager")) {
30                 gajiLembur = jumlahJamLembur * 100000;
31             } else if (jabatan.equalsIgnoreCase("karyawan")) {
32                 gajiLembur = jumlahJamLembur * 75000;
33             }
34
35             totalGajiLembur += gajiLembur;
36             System.out.println("Total gaji lembur: " + totalGajiLembur);
37
38         }
39     }
40
41 }
42
```

Output:

```
Masukkan jumlah karyawan:
3
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-1:
manajer
Masukkan jumlah jam lembur:
1
Total gaji lembur: 0.0
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: 375000.0
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM PEMROGRAMAN DASAR\PRAKTIKUM\JOBSHEET 7> |
```

Pertanyaan

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

Jawaban:

Jika jumlah karyawan yang dimasukkan lebih dari 1, maka perulangan akan dilakukan. Perulangan bergantung pada jumlah karyawan yang di input.

```
while (i < jumlahKaryawan) {
    System.out.println(x:"Pilihan jabatan - Direktur, Manajer, Karyawan");
    System.out.println("Masukkan jabatan karyawan ke-" + (i + 1) + ": ");
    jabatan = scan.nextLine();
    System.out.println(x:"Masukkan jumlah jam lembur: ");
    jumlahJamLembur = scan.nextInt();
    scan.nextLine(); // Membersihkan newline character dari buffer
    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:

Jika pengguna mengisi kolom jabatan dengan nilai “Direktur”, maka tidak akan ada tambahan gaji lembur, sehingga program akan dilanjutkan perulangannya. Peran CONTINUE yaitu untuk melanjutkan program.

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 update i++ dipindahkan ke akhir program, maka perulangan jumlah karyawan tidak dijalankan oleh program. Dan jika memasukkan direktur, maka tidak akan terjadi apa-apa.

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
```

Jawaban:

```
1  import java.util.Scanner;
2
3  public class WhileGaji03 {
4      public static void main(String[] args) {
5          Scanner scan = new Scanner(System.in);
6          int jumlahKaryawan, jumlahJamLembur;
7          double gajiLembur, totalGajiLembur;
8          gajiLembur = 0;
9          totalGajiLembur = 0;
10         String jabatan;
11
12         System.out.println("Masukkan jumlah karyawan: ");
13         jumlahKaryawan = scan.nextInt();
14         scan.nextLine(); // Membersihkan newline character dari buffer
15
16         int i = 0;
17
18         while (i < jumlahKaryawan) {
19             System.out.println("Pilihan jabatan - Direktur, Manajer, Karyawan");
20             System.out.println("Masukkan jabatan karyawan ke-" + (i + 1) + ": ");
21             jabatan = scan.nextLine();
22             System.out.println("Masukkan jumlah jam lembur: ");
23             jumlahJamLembur = scan.nextInt();
24             scan.nextLine(); // Membersihkan newline character dari buffer
25             i++;
26
27             if (jabatan.equalsIgnoreCase("direktur")) {
28                 continue;
29             } else if (jabatan.equalsIgnoreCase("manajer")) {
30                 gajiLembur = jumlahJamLembur * 100000;
31             } else if (jabatan.equalsIgnoreCase("karyawan")) {
32                 gajiLembur = jumlahJamLembur * 75000;
33             } else {
34                 System.out.println("Jabatan invalid");
35                 continue; // Kembali ke awal loop saat jabatan tidak valid
36             }
37
38             totalGajiLembur += gajiLembur;
39             System.out.println("Total gaji lembur: " + totalGajiLembur);
40
41         }
42     }
43 }
44
45
46
47
```

5. Push dan commit kode program ke github

Commit

Menambahkan continue untuk mengulang kesalahan input

main

afifahnisa17 committed 1 minute ago

Showing 1 changed file with 5 additions and 2 deletions.

7 WhileGaji03.java

...	@@ -31,8 +31,11 @@ public static void main(String[] args) {		
31	31		gajilembur = jumlahJamLembur * 100000;
32	32		} else if (jabatan.equalsIgnoreCase("karyawan")) {
33	33		gajilembur = jumlahJamLembur * 75000;
34	-		}
35	-		
	34	+	} else {
	35	+	System.out.println("Jabatan invalid");
	36	+	continue; // Kembali ke awal loop saat jabatan tidak valid
	37	+	}

2.3 Percobaan 3: Menghitung Jatah Cuti Menggunakan DO-WHILE

Source code:

```
1  import java.util.Scanner;
2
3  public class DoWhileCuti03 {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6          int jatahCuti, jumlahHari;
7          String konfirmasi;
8
9          System.out.print("Jatah cuti: ");
10         jatahCuti = sc.nextInt();
11
12         do {
13             System.out.println("Apakah Anda ingin mengambil cuti (y/t)");
14             konfirmasi = sc.next();
15
16             if (konfirmasi.equalsIgnoreCase("y")) {
17                 System.out.println("Jumlah hari: ");
18                 jumlahHari = sc.nextInt();
19
20                 if (jumlahHari <= jatahCuti) {
21                     jatahCuti -= jumlahHari;
22                     System.out.println("Sisa jatah cuti: " + jatahCuti);
23                 } else {
24                     System.out.println("Sisa jatah cuti Anda tidak mencukupi");
25                     break;
26                 }
27             }
28         } while (jatahCuti > 0);
29     }
30
31 }
32
```

Output:

```
Jatah cuti: 12
Apakah Anda ingin mengambil cuti (y/t)
y
Jumlah hari:
4
Sisa jatah cuti: 8
Apakah Anda ingin mengambil cuti (y/t)
y
Jumlah hari:
5
Sisa jatah cuti: 3
Apakah Anda ingin mengambil cuti (y/t)
y
Jumlah hari:
4
Sisa jatah cuti Anda tidak mencukupi
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM PEMROGRAMAN DASAR\PRAKTIKUM\JOBSHEET 7>
```

Pertanyaan

1. Apa kegunaan sintaks BREAK di dalam sintaks perulangan?

Jawaban:

Break digunakan untuk menghentikan perulangan lebih awal sebelum mencapai titik akhir kondisi 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.

Jawaban:

Source code:

```
import java.util.Scanner;

public class DOWhileCuti03 {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int jatahCuti, jumlahHari;
        String konfirmasi;

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

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

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

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

Output:

```
Jatah cuti: 12
Apakah Anda ingin mengambil cuti (y/t)
y
Jumlah hari:
10
Sisa jatah cuti: 2
Apakah Anda ingin mengambil cuti (y/t)
y
Jumlah hari:
9
Sisa jatah cuti Anda tidak mencukupi
Apakah Anda ingin mengambil cuti (y/t)
```

3. Push dan commit kode program ke github
Jawaban:

 main

 afifahnisa17 committed 1 minute ago

Showing 1 changed file with 1 addition and 1 deletion.

2

DoWhileCuti03.java

22	22	System.out.println("Sisa jatah cuti: " + jatahCuti);
23	23	} else {
24	24	System.out.println("Sisa jatah cuti Anda tidak mencukupi");
25	-	break;
25	+	continue;
26	26	}
27	27	}
28	28	} while (jatahCuti > 0);
29	29	}

4. Pada saat input konfirmasi, ketikkan “t”, apa yang terjadi? Mengapa demikian?
Jawaban:

```
Jatah cuti: 12
Apakah Anda ingin mengambil cuti (y/t)
t
Apakah Anda ingin mengambil cuti (y/t)
t
Apakah Anda ingin mengambil cuti (y/t)
```

Saat mengetikkan t, akan terjadi pengulangan pertanyaan. Hal tersebut karena program akan berjalan jika user mengetikkan y.

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

Jawaban:

Source Code:

```
1  import java.util.Scanner;
2
3  public class DoWhileCuti03 {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6          int jatahCuti, jumlahHari;
7          String konfirmasi;
8
9          System.out.print("Jatah cuti: ");
10         jatahCuti = sc.nextInt();
11
12         do {
13             System.out.println("Apakah Anda ingin mengambil cuti (y/t)");
14             konfirmasi = sc.next();
15
16             if (konfirmasi.equalsIgnoreCase("y")) {
17                 System.out.println("Jumlah hari: ");
18                 jumlahHari = sc.nextInt();
19
20                 if (jumlahHari <= jatahCuti) {
21                     jatahCuti -= jumlahHari;
22                     System.out.println("Sisa jatah cuti: " + jatahCuti);
23                 } else {
24                     System.out.println("Sisa jatah cuti Anda tidak mencukupi");
25                     continue;
26                 }
27             } else {
28                 System.out.println("Program berhenti");
29                 break;
30             }
31         } while (jatahCuti > 0);
32     }
33
34 }
35
```

Output:

```
Jatah cuti: 12
Apakah Anda ingin mengambil cuti (y/t)
t
Program berhenti
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM PEMROGRAMAN DASAR\PRAKTIKUM\JOBSHEET 7> |
```

6. Push dan commit kode program ke github.
Jawaban:

Commit

Menambahkan value dari else, jika kita user memasukkan selain y

main

afifahnisa17 committed 1 minute ago

Showing 1 changed file with 4 additions and 1 deletion.

▼ 5 ■■■■■ DoWhileCuti03.java

↑	@@ -24,7 +24,10 @@	public static void main(String[] args) {
24	24	System.out.println("Sisa jatah cuti Anda tidak mencukupi");
25	25	continue;
26	26	}
27	-	}
	27 +	} else
	28 +	System.out.println("Program berhenti");
	29 ++	break;
	30 +	
28	31	} while (jatahCuti > 0);
29	32	}
30	33	