

LAPORAN PRATIKUM PEKAN 5
PEMROGRAMAN ALGORITMA DAN PEMROGRAMAN
“PERULANGAN FOR”



Dosen Pengampu:
DR. Wahyudi. .S.T.M.T

Asisten Lab:
Jovantri Immanuel Gulo

Disusun Oleh:
Marcello Bayu Denar Widjaksono
2511532011

Fakultas Teknologi Informasi
Departemen Informatika
Universitas Andalas
2025

KATA PENGANTAR

Puji dan syukur senantiasa penulis panjatkan kepada Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan laporan ini guna memenuhi laporan praktikum mata kuliah Algoritma Pemrograman, dengan judul: “Perulangan For”.

Pada kesempatan ini, Penulis mengucapkan terima kasih Bapak DR. Wahyudi. .S.T.M.T yang telah memberikan tugas laporan praktikum. Penulis juga ingin mengucapkan terima kasih kepada pihak-pihak yang telah mendukung serta membantu penulis dalam penyelesaian laporan praktikum Algoritma Pemrograman.

Penulis menyadari sepenuhnya bahwa laporan ini masih jauh dari sempurna dikarenakan terbatasnya pengalaman dan pengetahuan yang penulis miliki. Oleh karena itu, penulis mengharapkan segala bentuk saran dan masukan serta kritik yang membangun dari berbagai pihak.

Akhir kata, penulis berharap Laporan ini dapat memberikan manfaat bagi perkembangan dunia pendidikan.

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
BAB I PENDAHULUAN.....	3
1.1 Latar Belakang	3
1.2 Tujuan.....	3
1.3 Manfaat Pratikum	3
BAB II PEMBAHASAN	3
2.1 Teori.....	3
2.2 Kode Pemrograman	3
BAB III KESIMPULAN	8
3.1 Kesimpulan	8
3.2 Saran	8
DAFTAR PUSTAKA	9

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat membuat sebuah program kita pasti pernah bertemu suatu kasus yang mengharuskan kita untuk membuat sebuah kode program yang dapat dieksekusi berkali-kali, dan jika dilakukan secara manual tentu ini akan menghabiskan banyak waktu dan tenaga. Untuk mengatasi hal itu kita dapat menggunakan konsep perulangan

Konsep perulangan selalu ada dalam setiap bahasa pemrograman walaupun terkadang di tiap-tiap bahasa pemrograman tipe perulangannya berbeda-beda. Pada bahasa pemrograman Java konsep perulangan ini ada 2 yaitu: for-loop dan while-loop.

1.2 Tujuan

1. Menjelaskan konsep perulangan dalam bahasa pemrograman Java.
2. Menjelaskan penerapan konsep perulangan dalam bahasa pemrograman Java.

1.3 Manfaat Pratikum

1. Memahami konsep perulangan dalam bahasa pemrograman Java.
2. Mampu memahami dan menerapkan konsep perulangan dalam bahasa Pemrograman Java.

BAB II PEMBAHASAN

2.1 Teori

loop adalah putaran yang tidak berulang, dalam bahasa pemrograman Java loop berfungsi untuk mengeksekusi blok kode berdasarkan suatu kondisi yang dinilai benar (memenuhi suatu kondisi yang telah ditetapkan) . Loop sangat berguna karena dapat menghemat waktu dan membuat pembuatan program menjadi lebih mudah. Dalam bahasa pemrograman Java terdapat 3 tipe loop yaitu for, do while, dan while. Masing-masing tipe loop ini akan saya jelaskan di bawah ini.

a. For-loop

for loop digunakan saat anda mengetahui berapa kali baris kode tersebut di jalankan dalam menggunakan for loop kita memerlukan 3 statement, yang mana tiap statement nya memiliki peran masing masing:

```
for (statement 1; statement 2; statement 3) {  
    // kode yang akan dieksekusi  
}
```

statement 1 = menetapkan variabel sebelum loop dimulai

statement 2 = menetapkan kondisi untuk loop

statement 3 = menambahkan 1 terhadap nilai pada variabel di statement 1

Dalam menggunakan for-loop, jika value/nilai pada statement 1 telah lebih dari statement 2 maka loop ini tidak akan pernah berjalan. Selain itu kita juga bisa menggunakan konsep for-loop untuk melakukan loop array.

2.2 Kode Pemrograman

Disini saya akan menjelaskan beberapa program yang telah saya buat dibawah ini:

a. Program perulanganFor1

```
public class PerulanganFor1 {
    public static void main(String [] args) {
        for (int i=1; i <= 10; i++) {
            System.out.println(i);
        }
    }
}
```

Penjelasan:

1. Pertama kita melakukan inisialisasi i = 1.
2. Lalu menetapkan kondisi i <= 10
3. Jika bernilai true maka print(i)
4. Lalu nilai i++

b. Program perulanganFor2

```
public class PerulanganFor2 {
    public static void main(String [] args) {
        for (int i=1; i <= 10; i++) {
            System.out.print(i+ " ");
        }
    }
}
```

Penjelasan:

1. Pertama kita melakukan inisialisasi i = 1.
2. Lalu menetapkan kondisi i <= 10
3. Jika bernilai true maka print(i+ " ")
4. Lalu nilai i++

Output:

1 2 3 4 5 6 7 8 9 10

c. Program perulanganFor3

```
public class PerulanganFor3 {
    public static void main (String [] args) {
        int jumlah = 0;
        for (int i=1; i <= 10; i++) {
            System.out.print(i);
            jumlah= jumlah+i;
            if ( i < 10) {
                System.out.print(" + ");
            }
        }
        System.out.println();
        System.out.println("Jumlah = " + jumlah);
    }
}
```

Penjelasan:

1. Inisialisasi var jumlah = 0
2. Inisali kondisi awal loop dengan i=1 dan I <= 10, i++.
3. Print(i)
4. Jumlah += i.
5. Jika i < 10 print(" + ")
6. Println()
7. Println("jumlah =" +)

Output:

1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10
Jumlah = 55

d. Program perulanganFor4

```
public class PerulanganFor4 {
    public static void main(String [] arg) {
        int jumlah = 0;
        int batas;
        Scanner input = new Scanner(System.in);

        System.out.print("Masukkan nilai batas =");
        batas = input.nextInt();
        input.close();

        for (int i =1; i <= batas; i++) {
            System.out.print(i);
        }
    }
}
```

```

        jumlah = jumlah + i;
        if ( i < batas) {
            System.out.print(" + ");
        } else {
            System.out.print(" = ");
        }
    }

    System.out.println(jumlah);
}
}

```

Penjelasan:

1. Inisialisasi batas dan jumlah = 0
2. Inputkan nilai batas
3. For i = 1, I <= batas, i++
4. Print(i)
5. Jumlah += i
6. Jika i < batas print(" + ")
7. Selain itu print(" = ")
8. Println(jumlah)

Output:

Masukkan nilai batas = 5
 1 + 2 + 3 + 4 + 5 = 15

e. Program nestedFor0

```

public class nestedFor0 {
    public static void main(String [] arg) {
        for ( int line = 1; line <= 5; line++) {
            for(int j = 1; j <= (-1 * line + 5); j++) {
                System.out.print(".");
            }
            System.out.print(line);
            System.out.println();
        }
    }
}

```

Penjelasan:

1. Loop dengan kondisi line = 1, line <= 5, line++
 Loop dengan kondisi j =1, j <= (-1 * line + 5), j++
 Print(".")
 Println(line)

Println()

Output:

```
....1
...2
..3
.4
5
```

f. Program nestedFor1

```
public class nestedFor1 {
    public static void main(String[] args) {
        for( int i= 1 ; i <= 5; i++) {
            for ( int j =1; j <= 5; j++) {
                System.out.print("*");
            }
            System.out.println();
            //mengakhiri line
        }
    }
}
```

Penjelasan:

1. Loop 1 (sebagai nilai kolom) dengan kondisi i = 1, i <= 5, i++
2. Loop 2 (sebagai nilai baris) yang berada di dalam loop 1 dengan kondisi j = 1, j <= 5, j++ dan print("*")
3. Println()

Output:

```
*****
*****
*****
*****
*****
```

g. Program nestedFor2

```
public class nestedFor2 {
    public static void main(String [] arg) {
        for( int i= 0 ; i <= 5; i++) {
            for ( int j =0 ; j <= 5; j++) {
                System.out.print(i + j + " ");
            }
            System.out.println();
            //mengakhiri line
        }
    }
}
```

```

    }
  }
}

```

Penjelasan:

1. Loop 1 (sebagai nilai kolom) dengan kondisi $i = 0$,
 $i \leq 5$, $i++$
2. Loop 2 (sebagai nilai baris) yang berada di dalam
loop 1 dengan kondisi $j = 0$, $j \leq 5$, $j++$ dan print
($i + j$ + " ")
3. `Println()`

Output:

```

0 1 2 3 4 5
1 2 3 4 5 6
2 3 4 5 6 7
3 4 5 6 7 8
4 5 6 7 8 9
5 6 7 8 9 10

```

BAB III

KESIMPULAN

3.1 Kesimpulan

Konsep loop memiliki peran yang sangat penting dalam membuat suatu kode program baik dalam menggunakan bahasa pemrograman Java ataupun dalam bahasa pemrograman lainnya, karena konsep loop dapat menghemat waktu kita dalam mengeksekusi blok kode yang harus dilakukan berulang-ulang .

3.2 Saran

Berdasarkan pratikum yang telah dilakukan mengenai Konsep loop dalam bahasa pemrograman Java. Penulis harapkan kita sebagai mahasiswa informatika harus dapat memahami dan mampu menerapkan konsep loop dalam membuat sebuah kode program.

DAFTAR PUSTAKA

[1] Oracle, “The Java” Tutorials: Learning The Java Language” Oracle Documentation, 2023 [Daring]. Tersedia pada: <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html>. [Diakses : 20-oktober-2025]

[2] W3schools, “While loop” 2025 [Daring]. Tersedia pada: https://www.w3schools.com/java/java_while_loop.asp. [Diakses: 22-oktober-2025]

[3] W3schools, “For loop” 2025 [Daring]. Tersedia pada: https://www.w3schools.com/java/java_for_loop.asp. [Diakses: 22-oktober-2025]

