

**LAPORAN PRAKTIKUM ALGORITMA PEMROGRAMAN**  
**THE FOR LOOP**



disusun Oleh:

Tiara Amalia Insani

NIM 2511532019

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

Asisten Praktikum : Jovantri Immanuel Gulo

**DEPARTEMEN INFORMATIKA**  
**FAKULTAS TEKNOLOGI INFORMASI**  
**UNIVERSITAS ANDALAS**  
**PADANG**

**2025**

## **KATA PENGANTAR**

Pedoman ini disusun sebagai rujukan resmi bagi mahasiswa Departemen Informatika dalam penyusunan laporan praktikum pada mata kuliah Pemrograman Dasar dengan Java. Dokumen ini tidak hanya memberikan gambaran umum mengenai format penulisan, tetapi juga menguraikan secara rinci sistematika laporan, tata cara penyajian isi, serta contoh penulisan kode program yang dilengkapi dengan referensi ilmiah. Melalui panduan ini, mahasiswa diharapkan mampu menyusun laporan yang tidak sekadar memenuhi aspek administratif, tetapi juga mencerminkan ketelitian, keteraturan, dan penerapan kaidah penulisan akademik pada tingkat dasar. Dengan demikian, laporan praktikum yang dihasilkan dapat berfungsi sebagai media pembelajaran, dokumentasi kegiatan, sekaligus sarana untuk melatih keterampilan menulis ilmiah yang akan bermanfaat dalam jenjang studi selanjutnya.

Padang, 29 Oktober 2025

Penyusun

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>i</b>
<b>DAFTAR ISI.....</b>	<b>ii</b>
<b>DAFTAR GAMBAR.....</b>	<b>iii</b>
<b>BAB I.....</b>	<b>1</b>
<b>PENDAHULUAN .....</b>	<b>1</b>
<b>1. 1 Latar Belakang.....</b>	<b>1</b>
<b>1.2 Tujuan Praktikum.....</b>	<b>1</b>
<b>1.3 Manfaat.....</b>	<b>1</b>
<b>BAB II .....</b>	<b>2</b>
<b>PEMBAHASAN .....</b>	<b>2</b>
<b>2.1 THE FOR LOOP .....</b>	<b>2</b>
2.1.1 Perulangan For .....	2
2.1.2 Increment/decrement.....	2
2.1.3 Operator Relasional dan Logika .....	2
2.1.4 Nested For .....	3
<b>2.2 Praktikum .....</b>	<b>3</b>
2.3.1 Alat dan Bahan.....	3
2.3.2 Program.....	3
<b>BAB III.....</b>	<b>10</b>
<b>KESIMPULAN DAN SARAN .....</b>	<b>10</b>
<b>3.1 Kesimpulan.....</b>	<b>10</b>
<b>3.2 Saran .....</b>	<b>10</b>
<b>DAFTAR PUSTAKA .....</b>	<b>11</b>

## DAFTAR GAMBAR

Gambar 2.1 .....	3
Gambar 2.2 .....	4
Gambar 2.3 .....	5
Gambar 2.4 .....	6
Gambar 2.5 .....	7
Gambar 2.6 .....	8
Gambar 2.7 .....	9

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Dalam pemrograman, perulangan digunakan untuk mengeksekusi pernyataan atau blok kode secara berulang sesuai dengan jumlah yang ditentukan tanpa harus menulis perintah yang sama berulang ulang kali. Salah satu jenis perulangan yang sering digunakan adalah perulangan for, perulangan ini berguna Ketika jumlah pengulangan sudah diketahui.

### **1.2 Tujuan Praktikum**

Tujuan dari pelaksanaan praktikum antara lain sebagai berikut:

1. Membantu mahasiswa memahami konsep dasar pemrograman Java melalui penerapan langsung.
2. Melatih kemampuan menulis, mengompilasi, dan mengeksekusi program dengan mengikuti aturan *sintaksis* Java
3. Membiasakan mahasiswa bekerja sistematis dalam menyusun laporan yang memuat analisis hasil praktikum.
4. Menanamkan sikap teliti, disiplin, serta tanggung jawab dalam melaksanakan kegiatan laboratorium.
5. Belajar bagaimana membuat program yang bisa mengambil keputusan otomatis sesuai kondisi
6. Mempelajari konsep perulangan menggunakan *for loop* atau *nested for*.
7. Mengetahui cara kerja increment, decrement, serta operator relasional dan logika dalam java
8. Melatih kemampuan logika berpikir dalam menulis program melibatkan kondisi dan perulangan.

### **1.3 Manfaat**

Mahasiswa diharapkan semakin terbiasa menggunakan perulangan dalam pemrograman, dan bisa memperoleh pemahaman mengenai penggunaan *for loop* dan *nested for* untuk menjalankan proses berulang secara efisien.

## BAB II

### PEMBAHASAN

#### 2.1 THE FOR LOOP

##### 2.1.1 Perulangan For

```
for (inisialisasi; kondisi; increment/decrement{  
    // blok kode yang diulang  
}
```

- Inisialisasi : dieksekusi sekali diawal sebelum perulangan dimulai.
- Kondisi : menentukan apakah perulangan akan terus dijalankan, jika bernilai true, maka blok kode dijalankan, jika false, perulangan berhenti.
- Increment/decrement : bagian yang menambah atau mengurangi nilai variabel penghitung setiap kali pengulangan dilakukan.

##### 2.1.2 Increment/decrement

Operator increment (++) digunakan untuk menambah nilai variabel sebesar 1, sedangkan decrement (--) untuk mengurangnya sebesar 1.

```
Int x = 2;  
x++;    //sekarang x bernilai 3  
x--;    //sekarang x bernilai 2
```

untuk nilai desimal juga berlaku

```
double gpa = 1.5;  
gpa--;    //sekarang gpa bernilai 0.5
```

selain itu, java juga menyediakan bentuk singkat (shortcut assignment)

```
x += 3;    // sama dengan x = x + 3;  
y -= 2;    // sama dengan y = y - 2;  
z *= 4;    // sama dengan z = z * 4;  
a /= 5;    // sama dengan a = a / 5;  
b %= 3;    // sama dengan b = b % 3;
```

##### 2.1.3 Operator Relasional dan Logika

Operator relasional untuk membandingkan 2 nilai.

- > lebih besar
- >= lebih besar sama dengan
- < lebih kecil
- <= lebih kecil sama dengan
- == sama dengan
- != tidak sama dengan

Operator logika digunakan untuk menggabungkan kondisi

- && dan (AND)
- || atau (OR)
- ! tidak (NOT)

#### 2.1.4 Nested For

Adalah perulangan di dalam perulangan. Biasanya digunakan untuk menampilkan pola atau table data. Pada nested for terdapat:

- Outer loop → perulangan luar (utama)
- Inner loop → perulangan dalam (berulang di setiap putaran outer loop)

## 2.2 Praktikum

### 2.3.1 Alat dan Bahan

1. Laptop/computer dengan eclipse terinstal.
2. Materi praktikum mengenai The For Loop.

### 2.3.2 Program

#### 1. PerulanganFor1

```

1 package pekan5;
2
3 public class PerulanganFor1 {
4
5     public static void main(String[] args) {
6         for (int i=1; i<= 10; i++) {
7             System.out.println(i);
8         }
9     }
10 }
11
12 }
```

Gambar 2.1

Langkah-langkah :

- a. *Public class* PerulanganFor1 { → mendeklarasikan class.
- b. *Public static void main (String[] args){* tempat program java dieksekusi pertama kali. Agar program bisa dijalankan.

- c. *For (int i = 1; i <= 10; i++)*{ membuat perulangan for, berfungsi mengulang kode di dalamnya sebanyak 10 kali
  - *Int i = 1* ➔ nilai awal variabel i adalah 1
  - *i <= 10* ➔ kondisi agar perulangan terus berjalan selama i masih <= 10
  - *i++* ➔ setiap putaran nilai i bertambah 1.
- d. *System.out.println(i)*; menampilkan nilai i.
- e. *}* ➔ menutup for, main, dan class.

## 2. PerulanganFor2

```

1 package pekan5;
2
3 public class PerulanganFor2 {
4
5     public static void main(String[] args) {
6         for (int i=1; i<= 10; i++) {
7             System.out.print(i+" ");
8         }
9
10    }
11
12 }
```

Gambar 2.2

Langkah-langkah :

- a. *Public class PerulanganFor2*{ ➔ mendeklarasikan class.
- b. *Public static void main (String[] args)*{ tempat program java dieksekusi pertama kali. Agar program bisa dijalankan.
- c. *For (int i = 1; i <= 10; i++)*{ membuat perulangan for, berfungsi mengulang kode di dalamnya sebanyak 10 kali
  - *Int i = 1* ➔ nilai awal variabel i adalah 1
  - *i <= 10* ➔ kondisi agar perulangan terus berjalan selama i masih <= 10
  - *i++* ➔ setiap putaran nilai i bertambah 1.
- d. *System.out.print(i+" ");* menampilkan nilai i tanpa pindah baris dan tambahan" " membedri spasi antar angka
- e. *}* ➔ menutup for, main, dan class.



### 3. PerulanganFor3

```
1 package pekan5;
2
3 public class PerulanganFor3 {
4
5     public static void main(String[] args) {
6         int jumlah=0;
7         for(int i=1; i<=10; i++) {
8             System.out.print(i);
9             jumlah=jumlah+i;
10            if(i<10) {
11                System.out.print(" + ");
12            }
13        }
14        System.out.println();
15        System.out.println("Jumlah = "+jumlah);
16    }
17
18 }
```

Gambar 2.3

Langkah-langkah:

- Public class PerulanganFor3 {* → mendeklarasikan class.
- Public static void main (String[] args){* tempat program java dieksekusi pertama kali. Agar program bisa dijalankan.
- Int jumlah=0;* → mendeklarasikan variabel jumlah sebagai tempat menyimpan hasil penjumlahan.
- For (int i = 1; i <= 10; i++){* melakukan perulangan dari nilai i=1 sampai i=10
- System.out.print(i);* menampilkan nilai i tanpa pindah baris
- Jumlah=jumlah+i;* menambahkan nilai i ke variabel jumlah
- If (i<10){*  
*System.out.print("+");* → tanda + muncul diantara angka
- }* → menutup if dan for
- System.out.println();* pindah ke baris baru setelah perulangan selesai
- System.out.println("jumlah="+jumlah);* menampilkan total hasil penjumlahan semua angka.
- }* → menutup main, dan class.

### 4. PerulanganFor4

```

1 package pekan5;
2
3 import java.util.Scanner;
4
5 public class PerulanganFor4 {
6
7     public static void main(String[] args) {
8         int jumlah=0;
9         int batas;
10        Scanner input = new Scanner(System.in);
11        System.out.print("Masukkan nilai batas=");
12        batas=input.nextInt();
13        input.close();
14        for (int i=1; i<=batas; i++) {
15            System.out.print(i);
16            jumlah=jumlah+i;
17            if (i<batas) {
18                System.out.print(" + ");
19            } else {
20                System.out.print(" = ");
21            }
22        }
23        System.out.println(jumlah);
24    }
25 }

```

Gambar 2.4

Langkah-langkah :

- a. *Import java.util.Scanner;* ➔ mengimpor class scanner untuk membaca input dari keyboard
- b. *Public class PerulanganFor4{* ➔ mendeklarasikan class.
- c. *Public static void main (String[] args){* tempat program java dieksekusi pertama kali. Agar program bisa dijalankan.
- d. *Int jumlah=0;* ➔ mendeklarasikan variabel jumlah sebagai tempat menyimpan hasil penjumlahan.
- e. *Int batas;* ➔ variabel untuk menyimpan batas angka yang dimasukkan pengguna
- f. *Scanner input = new Scanner (System.in);* ➔ agar bisa membaca data dari keyboard
- g. *System.out.print("Masukkan nilai batas=");* ➔ menyimpan angka yang dimasukkan oleh pengguna ke dalam variabel batas.
- h. *Input.close();* ➔ menutup scanner setelah digunakan
- i. *For (int i = 1; i <= batas; i++){* melakukan perulangan dari nilai i=1 sampai i=batas
- j. *System.out.print(i);* menampilkan nilai i tanpa pindah baris
- k. *Jumlah=jumlah+i;* menambahkan nilai i ke variabel jumlah

- l. *If (i<batas){* ➔ jika i belum mencapai batas terakhir, maka jalankan isi dari if
- m. *System.out.print(" + ")* ➔ menampilkan tanda tambah diantara angka;
- n. *}else{* ➔ jika I sudah sama dengan batas maka jalankan else
- o. *System.out.print(" = ");* ➔ menampilkan sama dengan setelah angka terakhir
- p. *}* ➔ menutup else dan for
- q. *System.out.println(jumlah);* menampilkan total hasil penjumlahan dan pindah baris.
- r. *}* ➔ menutup main, dan class.

## 5. nestedFor0

```

1 package pekan5;
2
3 public class nestedFor0 {
4
5     public static void main(String[] args) {
6         for (int line = 1; line <=5; line++) {
7             for (int j =1; j<= (-1 * line + 5); j ++){
8                 System.out.print(".");
9             }
10            System.out.print(line);
11            System.out.println();
12        }
13    }
14 }
15 }

```

Gambar 2.5

Langkah-langkah:

- a. *Public class nestedFor0{* ➔ mendeklarasikan class.
- b. *Public static void main (String[] args){* tempat program java dieksekusi pertama kali. Agar program bisa dijalankan..
- c. *For (int line = 1; line <= 5; line++){* mengatur baris dari 1 sampai 5
- d. *For (int j = 1; j <= (-1\*line+ 5); j++){* ➔ perulangan dalam (nested loop) untuk mencetak titik "." di setiap baris. Rumus  $(-1 \cdot \text{line} + 5)$  untuk menentukan berapa banyak titik yang dicetak
- e. *System.out.print(".");* mencetak titik tanpa pindah baris
- f. *}* ➔ menutup for dalam

- g. *System.out.print*("line"); ➔ mencetak angka baris diakhir titik
- h. *System.out.println*(); ➔ pindah baris baru
- i. } ➔ menutup for luar, main, class

#### 6. nestedFor1

```

1 package pekan5;
2
3 public class nestedFor1 {
4
5     public static void main(String[] args) {
6         for (int i = 1; i<=5; i++) {
7             for (int j=1; j<=5; j++) {
8                 System.out.print("*");
9             }
10            System.out.println();
11            //to end the line
12        }
13    }
14 }
15
16 }
```

Gambar 2.6

#### Langkah-langkah

- a. *Public class nestedFor1*{ ➔ mendeklarasikan class.
- b. *Public static void main (String[] args){* tempat program java dieksekusi pertama kali. Agar program bisa dijalankan..
- c. *For (int i = 1; i <= 5; i++){* ➔ membuat umlah baris, variabel i menunjukkan baris seberapa
- d. *For (int j = 1; j <=5); j++){* ➔ perulangan dalam (nested loop) untuk mencetak bintang di setiap kolom dalam satu baris, variabel j menunjukkan kolom seberapa.
- e. *System.out.print*("\*"); mencetak bintang tanpa pindah baris
- f. } ➔ menutup for dalam
- g. *System.out.println*(); ➔ pindah baris baru
- h. } ➔ menutup for luar, main, class

#### 7. nestedFor2

```

1 package pekan5;
2
3 public class nestedFor2 {
4
5     public static void main(String[] args) {
6         for (int i = 0; i<=5; i++) {
7             for (int j=0; j<=5; j++) {
8                 System.out.print(i+j+" ");
9             }
10            System.out.println();
11            //to end the line
12        }
13    }
14 }
15
16 }

```

Gambar 2.7

Langkah-langkah:

- a. *Public class nestedFor2{* ➔ mendeklarasikan class.
- b. *Public static void main (String[] args){* tempat program java dieksekusi pertama kali. Agar program bisa dijalankan..
- c. *For (int i = 0; i <= 5; i++){* *perulangan luar*, menentukan baris dari tabel
- d. *For (int j = 0; j <= 5; j++){* ➔ perulangan dalam (nested loop) untuk menentukan kolom dari tabel (j=0 sampai 5)
- e. *System.out.print(i+j+" ");* mencetak hasil penjumlahan i+j di baris yang sama lalu diberi spasi
- f. *}* ➔ menutup for dalam
- g. *System.out.println();* ➔ pindah baris baru
- h. *}* ➔ menutup for luar, main, class

## **BAB III**

### **KESIMPULAN DAN SARAN**

#### **3.1 Kesimpulan**

Dari praktikum ini dapat disimpulkan bahwa `for` digunakan untuk menjalankan perintah secara berulang ulang dengan jumlah tertentu, sedangkan `nested for` digunakan untuk membuat pola atau tampilan dua dimensi melalui perulangan dalam perulangan. Dengan praktikum ini, dipahami bahwa struktur perulangan sangat penting dalam pemrograman karna mempermudah, dan membuat kode lebih efisien.

#### **3.2 Saran**

1. Praktikum berikutnya sebaiknya dilengkapi dengan lebih banyak contoh program nyata (kasus sehari-hari) supaya lebih mudah memahami penerapan tipe data dalam menyelesaikan masalah.
2. Materi pembelajaran sebaiknya diberikan sebelum praktikum dimulai, agar dapat dipahami terlebih dahulu, sehingga pada saat praktikum lebih mudah mengerti.

## DAFTAR PUSTAKA

- Dokumen Institusi

[1] Departemen Informatika, *Pedoman Penulisan Laporan Praktikum Java*. Padang: Universitas Andalas, 2025.

- Buku

[2] H. M. Deitel and P. J. Deitel, *Java: How to Program (Early Objects)*, 11th ed. Pearson, 2018.

- Sumber daring

[3] Oracle, “The Java Tutorials,” 2023. [Daring]. Tersedia pada: <https://docs.oracle.com/javase/tutorial/>. [Diakses: 1-Okt-2025].