

LAPORAN PRAKTIKUM ALGORITMA PEMROGRAMAN

PERULANGAN FOR



DOSEN PENGAMPU:

DR. WAHYUDI, S.T, M.T

Asisten Lab:

AUFAN TAUFIQURRAHMAN

Disusun Oleh :

M.YAZEM AGVA ROIZ

2511533003

FAKULTAS TEKNOLOGI INFORMASI

DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

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, 2025

Tim Penyusun

DAFTAR ISI

BAB 1.....	4
1.1 Latar Belakang.....	4
1.2 Tujuan.....	5
1.3 Manfaat.....	5
BAB 2.....	5
2.1 Teori.....	5
2.2 Code Pemrograman.....	6
a. Code 1.....	6
b. Code 2.....	8
c. Code 3.....	9
d. Code 4.....	10
e. Code 5.....	11
f. Code 6.....	12
g. Code 7.....	13
BAB 3.....	14
3.1 Kesimpulan.....	14
3.2 Saran.....	14

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Dalam pemrograman sendiri sangatlah banyak kasus yang memerlukan perulangan dari suatu blok kode demi tujuan tertentu seperti keefisienan atau fleksibilitas, oleh karena itu bahasa pemrograman umumnya menyediakan fitur berupa keyword yang dapat digunakan untuk mengulangi suatu blok kode tanpa harus membuat blok kode yang sama berkali-kali.

Java merupakan bahasa pemrograman yang telah dikenal luas dan populer, menjadi suatu hal yang lumrah jikalau java juga memiliki fitur untuk dapat mengulangi sebuah blok kode pada pemrogramannya, kerana hal itulah penulis akan menjelaskan perulangan FOR pada java pada tulisan kali ini.

1.2 Tujuan

Tujuan dari penulisan laporan ini antara lain:

- a. Memahami Perulangan FOR
- b. Memahami kapan dan dimana saat yang tepat dalam menggunakan perulangan FOR

1.3 Manfaat

Manfaat dari penulisan makalah ini adalah pemahaman dari fitur perulangan FOR pada bahasa java serta menggunakan perulangan FOR pada kasus kasus nyata pada code untuk memahami manfaat dan urgensi dari perulangan FOR.

BAB 2

PEMBAHASAN

2.1 Teori

Perulangan FOR merupakan fitur yang memungkinkan melakukan perulangan pada sebuah kode, perulangan di sebuah kode umumnya disebut LOOP. fitur ini sendiri memungkinkan untuk mengulangi suatu kode yang sama sembari merubah variabel variabel pada setiap iterasi perulangan untuk mendapatkan perilaku yang diinginkan tanpa harus menulis kode yang sama secara manual.

Pada bahasa pemrograman java, perulangan FOR atau bisa disebut FOR LOOP dapat digunakan menggunakan keyword *for* dengan cara menginisialisasikannya memerlukan 3 area seperti berikut:

```
for(inisialisasi; kondisi; update){  
    // code  
}
```

Contoh penulisan sintaks sebagai berikut:

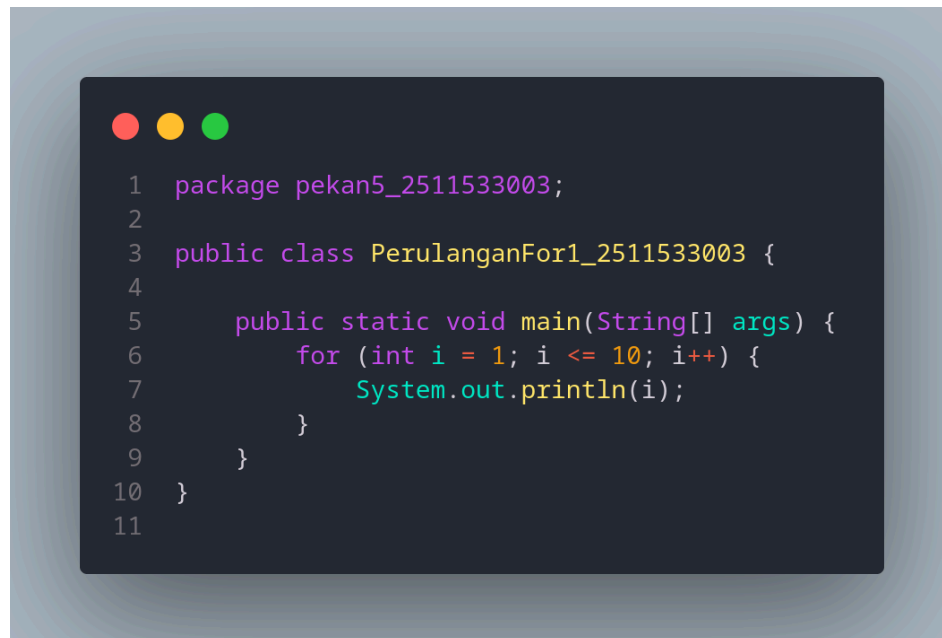
```
for (int i = 0; i < 5; i++){  
    system.out.println(i);  
}
```

Di Dalam FOR LOOP terdapat suatu kode, kode inilah yang akan diulangi sebanyak iterasi dan jumlah iterasi tergantung pada sintaks yang telah dituliskan terlebih dahulu, pada kasus diatas, terlebih dahulu menginisialisasikan sebuah variabel *i* dengan value 0, lalu akan divalidasi apakah *i* < 5 jika ia maka *i* akan diupdate dengan *i++* serta

menjalankan code yang terkandung dan jikalau $i < 5$ bernilai false maka perulangan tersebut akan berhenti.

2.2 Code Pemrograman

a. Code 1

A screenshot of a code editor with a dark background and light-colored text. The code is a Java program that uses a for loop to print numbers from 1 to 10. The code is as follows:

```
1 package pekan5_2511533003;
2
3 public class PerulanganFor1_2511533003 {
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
```

Penjelasan:

Code diatas merupakan code sederhana dalam penggunaan perulangan for, dari penulisan sintaks dapat dilihat bahwa for akan berjalan sebanyak 10 kali serta menjalankan code yang terkandung di dalamnya juga sebanyak 10 kali.

Output:

1
2
3
4

5
6
7
8
9
10

b. Code 2

A screenshot of a code editor window with a dark background and light-colored text. The code is written in Java and uses a for loop to print numbers from 1 to 10. The code is as follows:

```
1 package pekan5_2511533003;  
2  
3 public class PerulanganFor2_2511533003 {  
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
```

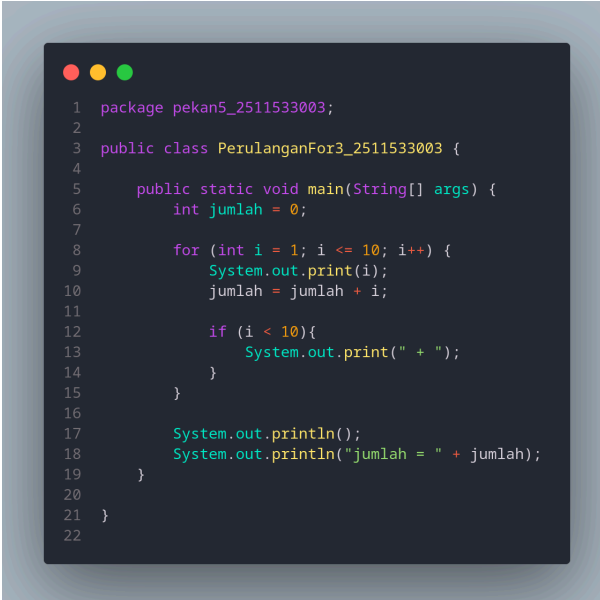
Penjelasan:

Code diatas merupakan code sederhana dalam penggunaan perulangan for yang sama seperti kose sebelumnya, namun perbedaannya pada penggunaan method print() dari pada println().

Output:

1 2 3 4 5 6 7 8 9 10

c. Code 3



```
1 package pekan5_2511533003;
2
3 public class PerulanganFor3_2511533003 {
4
5     public static void main(String[] args) {
6         int jumlah = 0;
7
8         for (int i = 1; i <= 10; i++) {
9             System.out.print(i);
10            jumlah = jumlah + i;
11
12            if (i < 10){
13                System.out.print(" ");
14            }
15        }
16
17        System.out.println();
18        System.out.println("jumlah = " + jumlah);
19    }
20
21 }
22
```

Penjelasan:

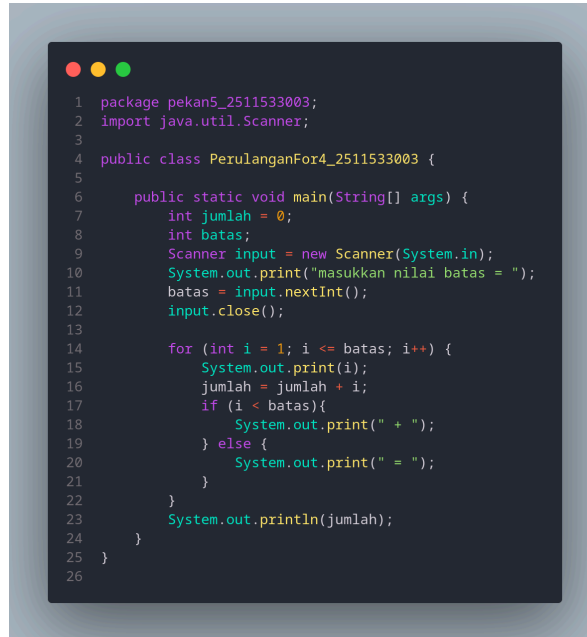
Kode diatas memiliki tujuan untuk menghitung jumlah angka 1 sampai 10.

Output:

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

Jumlah = 55

d. Code 4



```
1 package pekan5_2511533003;
2 import java.util.Scanner;
3
4 public class PerulanganFor4_2511533003 {
5
6     public static void main(String[] args) {
7         int jumlah = 0;
8         int batas;
9         Scanner input = new Scanner(System.in);
10        System.out.print("masukkan nilai batas = ");
11        batas = input.nextInt();
12        input.close();
13
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 }
26
```

Penjelasan:

Kode diatas memiliki tujuan untuk menghitung jumlah angka 1 sampai angka batas yang dimasukkan oleh user.

Output:

masukkan nilai batas = 20

1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 + 11 + 12 + 13 + 14 + 15 + 16 + 17 + 18 +
19 + 20 = 210

e. Code 5

```
1 package pekan5_2511533003;
2
3 public class nestedFor0_2511533003 {
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
16 }
17
```

Penjelasan:

Kode diatas merupakan kode nested loop atau dengan kata lain kode yang menggunakan loop didalam loop

Output:

....1


...2

..3

.4

5

f. Code 6

A screenshot of a Java code editor with a dark background and light-colored text. The code is a Java program that uses nested for loops to print a 5x5 grid of asterisks. The code is as follows:

```
1 package pekan5_2511533003;
2
3 public class nestedFor1_2511533003 {
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 }
17
```

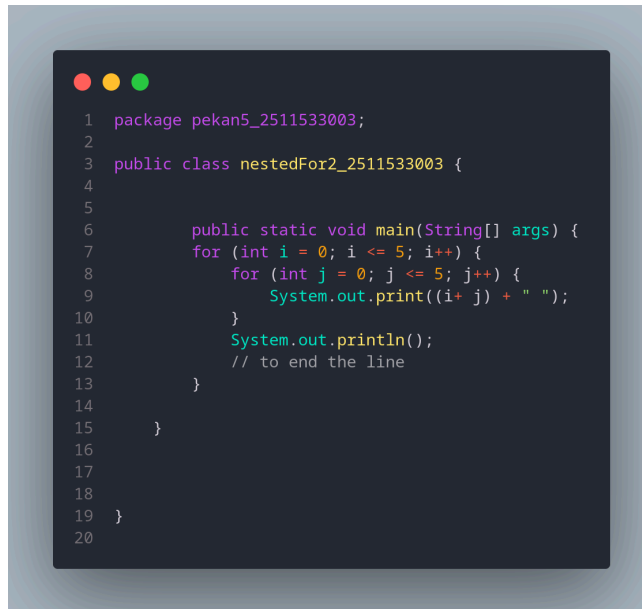
Penjelasan:

Kode diatas mirip dengan kode sebelumnya yaitu menggunakan for di dalam for yang akan memberikan output berupa persegi

Output:

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

g. Code 7

A screenshot of a Java code editor with a dark background and light-colored text. The code is a Java program with a package declaration, a class declaration, and a main method containing two nested for loops. The code is as follows:

```
1 package pekan5_2511533003;
2
3 public class nestedFor2_2511533003 {
4
5
6     public static void main(String[] args) {
7         for (int i = 0; i <= 5; i++) {
8             for (int j = 0; j <= 5; j++) {
9                 System.out.print((i+ j) + " ");
10            }
11            System.out.println();
12            // to end the line
13        }
14    }
15 }
16
17
18
19 }
20
```

Penjelasan:

Kode diatas mirip dengan kode sebelumnya dengan output berbentuk persegi, namun perbedaan pada hal yang dicetaknya saja.

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 3

KESIMPULAN

3.1 Kesimpulan

Perulangan For sangatlah membantu terkhususnya saat kasus yang memerlukan perulangan pada suatu kode berkali kali, dari contoh program sendiri dapat dilihat banyak operasi yang makin efisien dan fleksibel menggunakan perulangan for beserta perulangan for sangatlah membantu dalam membuat sebuah pola

3.2 Saran

Laporan praktikum yang telah penulis tuliskan, masih memiliki kekurangan. Penulis sendiri pun juga membuka saran dan kritikan untuk meningkatkan kualitas laporan pratikum ini.

DAFTAR PUSTAKA

Oracle. The For Statement. diakses pada 31 oktober 2025.

<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html>