

**LAPORAN PRATIKUM
PEMROGRAMAN ALGORITMA DAN PEMROGRAMAN**

LAPORAN PRAKTIKUM PEKAN 5

**Disusun Oleh:
Muhammad Aufa Rafiki**

2511531012

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

Asisten Praktikum: Muhammad Zaki Al Hafiz



**DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS**

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.

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR PUSTAKA	ii
BAB I PENDAHULUAN.....	1
1.1 Pengertian Pratikum.....	1
1.2 Tujuan Pratikum.....	1
1.3 Persyaratan Pratikum.....	1
1.4 Waktu dan Tempat Pratikum.....	2
BAB II PENULISAN LAPORAN PRATIKUM.....	3
2.1 For Loops	3
2.2 Perulangan For 1	3
2.3 Perulangan For 2	4
2.4 Perulangan For 3	4
2.5 Perulangan For 4	5
2.6 Nested For.....	6
2.7 Nested For 1	7

2.8 Nested For 2	8
BAB III KESIMPULAN	10
DAFTAR PUSTAKA	11

BAB I

PENDAHULUAN

1.1 Pengertian Pratikum

Praktikum Java adalah kegiatan pembelajaran yang dilakukan di laboratorium komputer untuk mengasah keterampilan mahasiswa dalam memahami serta menerapkan konsep pemrograman Java. Kegiatan ini tidak hanya menekankan pada penguasaan teori, tetapi juga pada latihan penyusunan kode program, pengujian, hingga analisis hasil eksekusi. Praktikum dipandang sebagai wahana latihan yang menjembatani pemahaman konseptual dengan kemampuan teknis pemrograman.

1.2 Tujuan Pratikum

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.** Meningkatkan keterampilan dalam memecahkan masalah (*problem solving*) dengan pendekatan algoritmik.
- 4.** Membiasakan mahasiswa bekerja sistematis dalam menyusun laporan yang memuat analisis hasil praktikum.
- 5.** Menanamkan sikap teliti, disiplin, serta tanggung jawab dalam melaksanakan kegiatan laboratorium.

1.3 Persyaratan Pratikum

Agar praktikum berjalan lancar, mahasiswa perlu memenuhi beberapa persyaratan berikut:

- 1.** Telah mengikuti perkuliahan teori Pemrograman Java sebagai dasar pemahaman.
- 2.** Membawa perlengkapan yang diperlukan, antara lain laptop atau komputer yang sudah terpasang Java Development Kit (JDK) dan Integrated Development Environment (IDE) yang direkomendasikan.

3. Mengikuti setiap sesi praktikum sesuai jadwal yang ditetapkan dan hadir minimal sesuai ketentuan program studi.
4. Mematuhi tata tertib laboratorium, termasuk menjaga keamanan data, perangkat, serta lingkungan kerja.
5. Menyusun laporan praktikum dengan format dan aturan yang telah ditetapkan dalam pedoman ini.

1.4 Waktu dan Tempat Pratikum

Pelaksanaan praktikum Java mengikuti kalender akademik yang berlaku pada program studi. Setiap sesi praktikum dilaksanakan sesuai jadwal yang ditentukan oleh dosen pengampu. Tempat kegiatan umumnya berlangsung di laboratorium komputer, namun pada kondisi tertentu dapat dilaksanakan secara mandiri dengan perangkat masing-masing, selama memenuhi syarat teknis yang ditetapkan.

BAB II

PEMBAHASAN

2.1 FOR LOOPS

For loops (atau perulangan for) adalah struktur kontrol dalam pemrograman yang digunakan untuk mengulang blok kode sebanyak jumlah tertentu.

Biasanya, for loop digunakan ketika jumlah perulangan sudah diketahui sebelumnya.

2.2 PERULANGAN FOR 1

1. Code Program

```
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
```

Gambar 2.1

2. Output

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

Gambar 2.2

3. Kerja Program

- Sistem akan memproses data $i=1$
- Lalu akan mengecek kondisi apakah $i \leq 10$
- Jika benar maka akan mencetak i
- Lalu untuk baris selanjutnya $i++$ (artinya $i+1$)
- Proses akan berulang sampai $i \leq 10$ bernilai salah

2.3 PERULANGAN FOR 2

1. Code Program

```
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 }
13
```

Gambar 2.3

2. Output

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

Gambar 2.4

3. Kerja Prograam

- Sistem akan memproses data i=1
- Lalu akan mengecek kondisi apakah i<=10
- Jika benar maka akan mencetak i
- Lalu untuk kolom selanjutnya i++(artinya i+1)
- Proses akan berulang sampai i<=10 bernilai salah

2.4 PERULANGAN FOR 3

1. Kode Program

```
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 2.5

2. Output

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

Gambar 2.6

3. Kerja Program

- Menginisialisasi jumlah=0
- Sistem akan memproses data i=1
- Lalu akan mengecek kondisi apakah i<=10
- Jika benar maka akan mencetak i
- Dan jumlah = jumlah+i
- Jika i<10, maka print (“+”)
- Lalu untuk kolom selanjutnya i++(artinya i+1)
- Proses akan berulang sampai i<=10 bernilai salah
- Jika salah maka perulangan akan berhenti
- Lalu print jumlah

2.5 PERULANGAN FOR 4

1. Kode Program

```
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 }
26
27
```

Gambar 2.7

2. Output

```
Masukkan nilai batas = 12
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 + 11 + 12 = 78
```

Gambar 2.8

3. Kerja Program

- Menginisialisasi jumlah=0 dan batas
- Meminta input nilai batas dari pengguna
- Sistem akan memproses data i=1
- Lalu akan mengecek kondisi apakah i<=batas
- Jika benar maka akan mencetak i
- Dan jumlah = jumlah+i
- Jika i<batas bernilai benar, maka print (" + ")
 - jika bernilai salah, maka print (" = ")
- Lalu untuk kolom selanjutnya i++(artinya i+1)
- Proses akan berulang sampai i<=batas bernilai salah
- Jika salah maka perulangan akan berhenti
- Lalu print jumlah

2.6 NESTED FOR

1. Kode Program

```
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
16 }
17 }
```

Gambar 2.9

2. Output

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

Gambar 2.10

3. Kerja Program

- inialisasi line=1
- Lalu akan mengecek kondisi apakah i<=5
- Jika benar, maka Inialisasi j=1
 - Jika salah, maka perulangan akan berhenti
- Lalu cek kondisi apakah j <= (-1*line+5)
- Jika benar print (“.”)”
 - Lalu untuk kolom selanjutnya j++(artinya j+1)
 - Proses akan berulang sampai j <= (-1*line+5) bernilai salah
- jika salah maka print (line)
- println () untuk membuat baris baru
- lalu line++ (line+1)
- berulang sampai line bernilai salah

2.7 Nested for 1

1. Kode Program

```
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
```

Gambar 2.11

2. Output

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

Gambar 2.12

3. Kerja Program

- inialisasi i=1
- Lalu akan mengecek kondisi apakah i<=5
- Jika benar, maka Inialisasi j=1

- Jika salah, maka perulangan akan berhenti
- Lalu cek kondisi apakah $j \leq 5$
- Jika benar print (“*”)
 - Lalu untuk kolom selanjutnya $j++$ (artinya $j+1$)
 - Proses akan berulang sampai $j \leq 5$ bernilai salah
- jika salah, print ()
- lalu $i++$ ($i+1$)
- berulang sampai sampai i bernilai salah

2.8 NESTED FOR 2

1. Kode Program

```

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 }
17

```

Gambar 2.13

2. 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

```

Gambar 2.14

3. Kerja Program

- inialisasi $i=0$
- Lalu akan mengecek kondisi apakah $i \leq 5$
- Jika benar, maka Inisialisasi $j=0$
 - Jika salah, maka perulangan akan berhenti

- Lalu cek kondisi apakah $j \leq 5$
- Jika benar print (i+j+ " ")
 - Lalu untuk kolom selanjutnya $j++$ (artinya $j+1$)
 - Proses akan berulang sampai $j \leq 5$ bernilai salah
- jika salah, print ()
- lalu $i++$ ($i+1$)
- berulang sampai sampai i bernilai salah

BAB III

KESIMPULAN

Dari pembahasan yang telah dilakukan, dapat disimpulkan bahwa for loops merupakan salah satu struktur perulangan yang sangat penting dalam pemrograman karena memungkinkan eksekusi kode secara berulang dengan jumlah iterasi yang telah ditentukan. Dengan menggunakan for loops, proses yang bersifat repetitif dapat dilakukan dengan lebih efisien, terstruktur, dan mudah dipahami.

Ada bentuk pengembangan dari for loop yang memungkinkan penggunaan satu perulangan di dalam perulangan lainnya yang disebut nested for loop atau perulangan bersarang.

Secara keseluruhan, pemahaman yang baik tentang for loop dan nested for loop sangat penting bagi seorang programmer, karena kedua konsep ini menjadi dasar dalam pembuatan algoritma yang efektif dan efisien di berbagai bahasa pemrograman.

DAFTAR PUSTAKA

- [1] Oracle, *The for Statement (Java™ Tutorials)*, 2024. [Online]. Available: <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html>
- [2] J. Gosling, B. Joy, G. Steele, and G. Bracha, *The Java Language Specification*, 5th ed., Boston: Addison-Wesley, 2022.