

**LAPORAN PRAKTIKUM**  
**ALGORITMA & PEMROGRAMAN**  
**“Laporan Praktikum Pekan 6”**

Disusun Oleh:  
Rizky Fadhlurrahman Almatria  
2511532006  
Dosen Pengampu : Dr. Wahyudi, S.T, M.T.  
Asisten Praktikum : Muhammad Zaki



DAPARTEMEN INFORMATIKA  
FAKULTAS TEKNOLOGI  
INFORMASI UNIVERSITAS  
ANDALAS

2025

## **KATA PENGANTAR**

Puji syukur saya panjatkan ke hadirat Tuhan Yang Maha Esa atas rahmat dan karunia-Nya, sehingga dapat menyelesaikan laporan praktikum ini dengan baik. Tidak lupa juga saya ucapkan terima kasih kepada bapak Wahyudi. Dr. S.T.M.T sebagai dosen pembimbing dan Muhammad Zaki Al Hafiz yang telah membantu dalam pelaksanaan praktikum ini. Laporan ini saya susun untuk memenuhi tugas praktikum Algoritma dan Pemrograman dari praktikum pertemuan pertama mengenai pemahaman dasar pemrograman bahasa Java menggunakan IDE Eclipse. Praktikum ini bertema pembuatan kode dasar, yaitu pengulangan menggunakan while dan do while, sentinel loop, dan randomize. Saya berharap laporan ini dapat memberikan manfaat bagi pembaca dalam memahami konsep dasar pemrograman Java.

Laporan ini dibuat dengan harapan dapat memberikan pemahaman dasar mengenai penggunaan Bahasa pemrograman Java, khususnya mengenai hal struktur dasar program dan fungsi.

Saya menyadari bahwa laporan ini masih jauh dari sempurna, oleh sebab itu, kritik dan saran sangat diharapkan demi penyempurnaan laporan di masa mendatang.

Padang 05 Oktober, 2025

Penulis

## **Daftar Isi**

### **Daftar Isi**

BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Tujuan .....	1
1.3 Manfaat.....	1
BAB II.....	3
PEMBAHASAN.....	3
2.1 Perulangan While .....	3
2.1.1 Langkah Kerja Program .....	3
2.1.3Hasil Output .....	4
2.1.4 Analisis Hasil .....	4
2.2 If-Else.....	5
2.2.1 Langkah Kerja Program .....	5
2.2.2 Contoh Pemrograman .....	5
2.2.3 Hasil Output .....	6
2.2.4 Analisis Hasil .....	6
2.3 Multi If .....	6
2.3.1 Langkah Kerja Program .....	6
2.3.2 Contoh Pemrograman .....	7
2.3.3 Hasil Output .....	7
2.3.4 Analisis Hasil .....	7
2.4 Nested If .....	8
2.4.1 Langkah Kerja Program .....	8
2.4.2 Contoh Pemrograman .....	9
2.4.3 Hasil Output .....	10
2.4.4 Analisis Hasil .....	10
2.5 Switch Case.....	10
2.5.1 Langkah Kerja Program .....	10
2.5.2 Contoh Pemrograman .....	11
2.5.3 Hasil Output .....	11
2.5.4 Analisis Hasil .....	11
BAB III.....	12
KESIMPULAN .....	12

DAFTAR PUSTAKA .....	13
----------------------	----

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Praktikum Pekan 6, Disusun dan dirancang untuk memperkenalkan struktur pengulangan menggunakan prosedur while dan do while, dan beberapa contoh penggunaannya, lalu juga menjelaskan mengenai penggunaan Syntax random dalam kode program, yang akan 84w di implementasikan dan dicontohkan menggunakan bahasa Java dan *Eclipse* Sebagai IDE nya.

#### **1.2 Tujuan**

1. Memahami bagaimana cara penggunaan perulangan while.
2. Melihat pengimplementasian perulangan while.
3. Memahami bagaimana menggunakan fungsi randomize.
4. Memahami bagaimana cara penggunaan perulangan do while.

#### **1.3 Manfaat**

1. Meningkatkan kemampuan mahasiswa dalam menulis kode program menggunakan bahasa pemrograman Java dengan logika kondisional yang benar.
2. Memberikan pemahaman simple tentang perulangan.
3. Memperkuat keterampilan debugging dan analisis kondisi menggunakan Eclipse.
4. Memahami fondasi untuk program kompleks seperti sistem preulangan.
5. Mendorong pemikiran logis dalam merancang solusi pemrograman.

## **BAB II**

### **PEMBAHASAN**

#### **2.1 Perulangan While**

Program ini mendemonstrasikan penggunaan struktur perulangan while dengan. Contohnya adalah menggunakan program yang menanyakan apakah program ingin diulang dan menghitung berapa kali perulangan dilakukan dan menampilkan total perulangan yang telah dilakukan.

##### **2.1.1 Langkah Kerja Program**

- Program dimulai dan kemudian mendeklarasikan 3 buah variable yaitu “Counter” dengan tipe int sebagai penghitung jumlah pengulangan, “running” yang bertipe Boolean yang diberi nilai ‘true’ sebagai penentu jalan atau tidaknya program, dan “jawab” dengan tipe String sebagai penampung input.
- Perulangan dimulai jika “running” bernilai true dalam perulangan terdapat proses
  - a. “Counter” ditambah 1.
  - b. Cetak “Jumlah = (*‘Counter’*) “.
  - c. Cetak “Apakah ingin lanjut? (ya / tidak)”.
  - d. Input “jawab”.
  - e. Jika Jawab sama dengan “tidak”.
    - running = false
  - f. pengecekan kondisi perulangan.
- Ketika perulangan berakhir maka cetak “Anda melakukan perulangan sebanyak = (*‘Counter’*)”.
- Program berakhir.

## 2.1.2 Contoh Pemrograman

```
1 package pekan6_2511532006;
2 import java.util.Scanner;
3
4 public class PerulanganWhile1_2511532006 {
5
6     public static void main(String[] args) {
7         // TODO Auto-generated method stub
8         int counter =0;
9         String jawab ="";
10        boolean running = true;
11
12        Scanner scan = new Scanner(System.in);
13
14
15    while(running) {
16        counter++;
17        System.out.println("Jumlah =" + counter);
18        System.out.println("Apakah Lanjut? (ya/tidak)");
19        jawab = scan.nextLine();
20        if(jawab.equalsIgnoreCase("tidak")) {
21            running = false;
22        }
23    }
24    System.out.println("Anda sudah melakukan pengulangan sebanyak" + counter);
25
26}
27
28 }
```

## 2.1.3 Hasil Output

```
<terminated> PerulanganWhile1_2511532006 [Java Application]
Jumlah =1
Apakah Lanjut? (ya/tidak)
ya
Jumlah =2
Apakah Lanjut? (ya/tidak)
tidak
Anda sudah melakukan pengulangan sebanyak2
```

## 2.1.4 Analisis Hasil

Dari program yang sudah kita buat, kita lihat program ini melakukan perulangan selama beberapa kali dan menghitungnya di setiap perulangan. Bisa kita lihat bagaimana perbedaan perulangan while dengan perulangan for. Perulangan for memiliki nilai yang jelas untuk menentukan berapa kali total pengulangan yang di lakukan, sedangkan di perulangan while kita memasukkan statement pengecekan didalam nya, seperti tadi, apakah Boolean ini bernilai true, atau misal string ini bernilai "A". dan banyak contoh lainnya. Sehingga penggunaan perulangan while bisa membantu kita membuat program yang lebih kompleks dan interaktif berkat kemampuan pengulangannya yang menggunakan statement.

## 2.2 Sentinel Loop

Sentinel loop Adalah pengulangan yang akan terus-terusan berjalan dan akan berhenti Ketika satu angka khusus atau yang disini disebut **Angka Sentinel** diinputkan. Pada program ini kita akan mempraktekan bagaimana cara kerja sentinel loop tersebut dengan variasi menjumlahkan semua nilai yang sudah kita inputkan.

### 2.2.1 Langkah Kerja Program

- Program dimulai dan kemudian mendeklarasikan 2 buah variable, “sum” & “number” dengan tipe integer yang Dimana sum bernilai 0 dan number bernilai 1
  - Perulangan dimulai jika “number” bernilai tidak sama dengan 0.  
dalam perulangan terdapat proses :
    - a. Cetak “Masukkan angaka (0 untuk keluar): “.
    - b. Input nilai “Number”
    - c. Nilai “sum” diisi menjadi sum + number
    - d. pengecekan kondisi perulangan.
  - Ketika perulangan berakhir maka cetak “Totalmnya adalah(‘sum’”).
- Program berakhir.

### 2.2.2 Contoh Pemrograman

```
1 package pekan6_2511532006;
2 import java.util.Scanner;
3
4 public class SentinelLoop_2511532006 {
5     public static void main(String[] args) {
6         Scanner console = new Scanner(System.in);
7         int sum = 0;
8         int number = 1;
9
10        while (number != 0) {
11            System.out.println("Masukkan angaka (0 untuk keluar):");
12            number = console.nextInt();
13            sum = sum + number;
14        }
15        System.out.println("totalnya adalah "+ sum);
16    }
17}
```

### **2.2.3 Hasil Output**

```
<terminated> SentinelLoop_2511532006 [Java Application]
Masukkan angaka (0 untuk keluar):
10
Masukkan angaka (0 untuk keluar):
20
Masukkan angaka (0 untuk keluar):
30
Masukkan angaka (0 untuk keluar):
40
Masukkan angaka (0 untuk keluar):
0
totalnya adalah 100
```

### **2.2.4 Analisis Hasil**

Dari program yang sudah kita buat, kita bisa lihat bagaimana sentinel loop bekerja. Dia mengulang proses yang sama yaitu pencetakan, penginputan angka, penjumlahan sampai input angkanya Adalah 0. Itulah contoh dari sentinel loop yang bisa dilakukan menggunakan perulangan while.

## **2.3 Randomize system (Lempar Dadu)**

Program ini menunjukkan bagaimana menggunakan fitur randomize yang digabungkan dengan while. Dengan penggabungan itu program ini bisa membuat kita melakukan pengulangan lempar dadu terus menerus dan berakhir jika nilainya sesuai dengan yang kita inginkan. Lalu menunjukkan berapa total perulangan yang sudah dilakukan.

### **2.3.1 Langkah Kerja Program**

- Program dimulai dan kemudian mendeklarasikan 2 buah variable yaitu “tries” & “sum” yang bertipe integer dengan nilai 0.
- perulangan dilakukan dan dilakukan jika sum tidak sama dengan 7.  
dalam perulangan terdapat:
  - a. inisialisasi variable “dadu1” yang bernilai random antara 1-6.
  - b. inisialisasi variable “dadu2” yang bernilai random antara 1-6.
  - c. “Sum” berubah nilai menjadi “dadu1” + “dadu2”.
  - d. Cetak “(‘dadu1’ + ‘dadu2’ = ‘sum’)
  - e. “tries” + 1
- jika perulangan berakhir maka Cetak “You won after (‘tries’) tries!”
- Program berakhir

### 2.3.2 Contoh Pemrograman

```
1 package pekan6_2511532006;
2 import java.util.*;
3
4 public class LemparDadu_2511532006 {
5     public static void main(String[] Args) {
6         Random rand = new Random();
7         int tries = 0;
8         int sum = 0;
9         while (sum != 7) {
10             int dadu1 = rand.nextInt(6) + 1;
11             int dadu2 = rand.nextInt(6) + 1;
12             sum = dadu1+dadu2;
13             System.out.println(dadu1 + " + " + dadu2 + " = " + sum);
14             tries++;
15         }
16         System.out.println("You won after " + tries + " tries!");
17     }
18 }
```

### 2.3.3 Hasil Output

```
<terminated> LemparDadu_2511532006
6 + 3 = 9
5 + 1 = 6
2 + 3 = 5
4 + 6 = 10
3 + 4 = 7
You won after 5 tries!
```

### 2.3.4 Simpulan

Dari program yang sudah kita buat, kita lihat bahwa dengan menggunakan pengulangan while dan system randomize, kita bisa membuat program dadu dan menghitungnya dengan cepat. Menggunakan randomize, kita bisa memasukkan variable dengan nilai acak didalam program kita.

## 2.4 Randomize system (Game penjumlahan)

Program ini mendemonstrasikan penggunaan fungsi random dengan menggunakan beberapa variable sehingga menciptakan permainan penjumlahan dengan berbagai angka random, dan sistem nyawa. Disini kita juga sedikit mempelajari mengenai penggunaan ‘method’.

### 2.4.1 Langkah Kerja Program

- Buat method baru dengan nama “play” dan tipe output int yang memiliki 2 increment yaitu scanner dengan nama “console” dan random dengan nama “rand”
  - dalam method tersebut terdapat proses yaitu:
    - a. Deklarasi “Operands” dengan nilai = random antara 2 - 5
    - b. Deklarasi “Sum” dengan nilai = random antara 1-10
    - c. Cetak “Sum”
    - d. lakukan pengulangan menggunakan for dengan konfigurasi int “I” = 2, “I” lebih kecil sama dengan “Operands” maka:
      - a. int n = random antara 1-10
      - b. sum ditambah n
      - c. Cetak “+ (‘sum’)“
      - e. Cetak “= ”
    - f. Deklarasi variable “Guess” dengan tipe data integer dan diisi dengan console;
    - g. Jika nilai “Guess” sama dengan “Sum”
      - maka method mengeluarkan nilai 1
      - jika tidak maka Cetak “Wrong! The answer is (‘sum’) dan method Maka method mengeluarkan nilai 0 atau stop method
- 
- Program dimulai dan kemudian mendeklarasikan 2 buah variable yaitu Nilai dengan tipe data int bernama “Points” dan “Wrong”.
  - perulangan berjalan selama nilai “wrong” kecil dari 3 dalam perulangan terjadi:
    - a. deklarasi variable “Result” Adalah nilai dari method ‘play’
    - b. jika nilai “Result” besar dari 0 maka point++, jika tidak maka “Wrong” + 1
  - Cetak “You earned (‘Points’) total points”
  - Program berakhir

## 2.4.2 Contoh Pemrograman

```
1 package pekan6_2511532006;
2 import java.util.*;
3
4
5 public class Gamepenjumlahan_2511532006 {
6
7     public static int play(Scanner console, Random rand) {
8         int operands = rand.nextInt(4)+2;
9         int sum = rand.nextInt(10)+ 1;
10        System.out.print(sum);
11        for (int i = 2; i<operands; i++) {
12            int n = rand.nextInt(10)+1;
13            sum += n;
14            System.out.print(" + " + n);
15        }
16        System.out.println(" = ");
17
18        int guess = console.nextInt();
19        if (guess == sum) {
20            return 1;
21        }else {
22            System.out.println("Wrong! The answer was "+sum);
23            return 0;
24        }
25    }
26
27    public static void main(String[] args) {
28        Scanner console = new Scanner(System.in);
29        Random rand = new Random();
30
31        int points = 0;
32        int wrong = 0;
33        while (wrong < 3) {
34            int result = play(console, rand);
35            if (result > 0) {
36                points++;
37            }else wrong++;
38        }
39        System.out.println("You earned " + points + " total points");
40    }
41}
```

### 2.4.3 Hasil Output

```
<terminated> Gamepenjumlahan_2511532006 [a]
10 + 4 =
14
8 + 9 + 1 =
18
4 + 3 + 5 + 6 + 2 =
0
Wrong! The answer was 20
10 + 2 + 9 + 1 + 4 =
0
Wrong! The answer was 26
9 + 1 + 3 + 6 =
0
Wrong! The answer was 19
You earned 2 total points
```

### 2.4.4 Simpulan

Dari program yang sudah kita buat, kita bisa melihat bagaimana penggunaan method, pengulangan for dan while di gabungkan dalam satu program. Dengan method kita bisa menghemat pengetikan dan tidak mengganggu optimalisasi penggunaan syntax, kita tinggal panggil aja lalu method tersebut dapat di eksekusi berkali2-kali. Dengan while kita bisa membuat sistem nyawa dan membuat game ini menjadi menantang, lalu dengan for kita bisa menggenerate soal yang sepenuhnya random

## 2.5 Do While

Program ini mendemonstrasikan penggunaan do while di dalam program dengan demonstrasi disini Adalah program input password

### 2.5.1 Langkah Kerja Program

- Program dimulai dan kemudian mendeklarasikan 1 buah variable yaitu “Phrase” dengan tipe data String
- pengulangan di awali dengan melakukan proses yaitu:
  - a. Cetak “Input password”
  - b. Input nilai “Phrase”
  - c. lakukan pengecekan apakah phrase tidak sama dengan ‘abcd’ jika iya maka ulangi proses.

- Program berakhir

## 2.5.2 Contoh Pemrograman

```
1 package pekan6_2511532006;
2 import java.util.Scanner ;
3
4 public class DoWhile1_2511532006 {
5
6    public static void main(String[] args) {
7        Scanner console = new Scanner(System.in);
8        String phrase;
9    do {
10            System.out.println("Input password");
11            phrase = console.next();
12        } while (!phrase.equals("abcd"));
13    }
14 }
15
```

## 2.5.3 Hasil Output

```
<terminated> Gamepenjumlahan_2511532006
10 + 4 =
14
8 + 9 + 1 =
18
4 + 3 + 5 + 6 + 2 =
0
Wrong! The answer was 20
10 + 2 + 9 + 1 + 4 =
0
Wrong! The answer was 26
9 + 1 + 3 + 6 =
0
Wrong! The answer was 19
You earned 2 total points
```

## 2.5.4 Simpulan

Dari program yang sudah kita buat, kita bisa lihat bahwa perbedaan while dengan do while, do while melakukan proses dahulu, jadi setidaknya perulangan dilakukan 1 kali. Sedangkan while melakukan pengecekan di awal, sehingga bisa jadi program tidak dijalankan sama sekali.

## **BAB III**

### **KESIMPULAN**

Dari praktikum ini, kita bisa melihat bagaimana perulangan while dan do while berjalan. Apa perbedaannya, bagaimana penggunaanya di Bahasa java

Secara keseluruhan, tujuan dari praktikum pertemuan keenam berhasil mencapai tujuannya yaitu memperkenalkan mengenai perulangan while dan do while. Penggunaan Scanner dan Eclipse memfasilitasi interaksi, sementara konsep ini menjadi pondasi untuk aplikasi lebih kompleks.

Akhirnya, pemahaman ini membuka jalan untuk topik lanjutan di pemograman java. Dengan praktikum ini berhasil melatih para mahasiswa agar berpikir dengan logika dan menciptakan program yang interaktif dan dapat berjalan dengan otomatis tanpa melakukan banyak syntax. Yang mana akan sangat berguna di real-world nantinya.

## **DAFTAR PUSTAKA**

1. Schildt, Herbert. (2019). *Java: The Complete Reference*. McGraw-Hill Education.
2. Oracle. (2023). *The Java™*.
3. Deitel, Paul & Deitel, Harvey. (2020). *Java How to Program*. Pearson Education.
4. Horstmann, Cay S. (2018). *Core Java Volume I—Fundamentals*. Addison Wesley.S

