

# **LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN**

## **“Perulangan While dan Do-While”**



**disusun Oleh:**

**Muhammad Faiz An-Anri**

**NIM 2511532021**

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

**Asisten Pratikum: Jovuel Imanuntri Gulo**

**FAKULTAS TEKNOLOGI INFORMASI**

**DEPARTEMEN INFORMATIKA**

**UNIVERSITAS ANDALAS**

**2025**

## DAFTAR ISI

<b>DAFTAR ISI.....</b>	<b>i</b>
<b>KATA PENGANTAR.....</b>	<b>ii</b>
<b>BAB I 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 Praktikum.....</b>	<b>1</b>
<b>BAB II PEMBAHASAN .....</b>	<b>2</b>
<b>2.1 Program perulanganWhile1 .....</b>	<b>2</b>
<b>2.2 Program Lempardadu.....</b>	<b>3</b>
<b>2.3 Program GamePenjumlahan.....</b>	<b>4</b>
<b>2.3 Program doWhile1 .....</b>	<b>5</b>
<b>2.4 Program SentinelLoop .....</b>	<b>6</b>
<b>BAB III KESIMPULAN .....</b>	<b>7</b>
<b>3.1 Ringkasan Hasil Praktikum .....</b>	<b>7</b>

## **KATA PENGANTAR**

Puji syukur kehadiran Allah SWT atas limpahan rahmat dan karunia-Nya sehingga laporan praktikum dengan judul “Perulangan While dan Do-While” ini dapat disusun dan diselesaikan dengan baik. Laporan ini disusun sebagai salah satu bentuk pelaksanaan kegiatan praktikum pada mata kuliah Algoritma dan Pemrograman.

Ucapan terima kasih disampaikan kepada Bapak Dr. Wahyudi, S.T., M.T. selaku dosen pengampu, serta kepada asisten laboratorium yang telah memberikan bimbingan selama kegiatan praktikum berlangsung.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna, baik dari segi penulisan maupun isi. Oleh karena itu, saran dan kritik yang bersifat membangun sangat diharapkan demi perbaikan di masa yang akan datang. Harapannya, laporan ini dapat memberikan manfaat dan menjadi referensi bagi pembelajaran konsep perulangan dalam bahasa pemrograman Java.

## **BAB 1**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Dalam proses pemrograman, sering kali sebuah program perlu menjalankan satu atau beberapa perintah secara berulang sampai suatu kondisi tertentu terpenuhi. Jika dilakukan tanpa struktur perulangan, penulisan kode akan menjadi panjang.

Perulangan while digunakan ketika program perlu mengeksekusi blok kode selama suatu kondisi bernilai true, sedangkan do-while akan menjalankan blok kode terlebih dahulu sebelum memeriksa kondisi, sehingga minimal satu kali iterasi selalu dilakukan.

#### **1.2 Tujuan Praktikum**

Tujuan dari praktikum ini adalah:

1. Memahami konsep dasar dan perbedaan antara struktur perulangan while dan do-while dalam bahasa pemrograman Java.
2. Mengimplementasikan perulangan while dan do-while untuk menyelesaikan berbagai kasus, seperti pengulangan input, simulasi permainan, dan pengolahan data.

#### **1.3 Manfaat Praktikum**

Manfaat yang diperoleh dari praktikum ini antara lain:

1. Menambah pemahaman mengenai perulangan yang tidak memiliki batas iterasi tetap.
2. Meningkatkan keterampilan menulis kode dengan menggunakan kondisi logika sebagai pengendali perulangan.
3. Memberikan pengalaman dalam membuat program interaktif yang melibatkan input pengguna secara berulang.

## BAB 2

### PEMBAHASAN

#### 2.1 Program perulanganWhile1

```

1 package pekan6_2511532001;
2
3 import java.util.Scanner;
4
5 public class perulanganWhile1_2511532001 {
6
7     public static void main(String[] args) {
8
9         int counter=0;
10        String jawab;
11        boolean running = true;
12
13        Scanner scan = new Scanner(System.in);
14        while (running) {
15            counter++;
16            System.out.println("Jumlah = "+counter);
17            System.out.println("Apakah lanjut (ya / tidak?)");
18            jawab= scan.nextLine();
19
20            if (jawab.equalsIgnoreCase("tidak")) {
21                running = false;
22            }
23        }
24        System.out.println("Anda sudah melakukan perulangan sebanyak "+counter+" kali");
25    }
26 }
27
28
29 }

```

Program ini menggunakan perulangan while untuk menjalankan pengulangan berdasarkan jawaban pengguna. Selama variabel running bernilai true, program akan terus menghitung jumlah pengulangan yang telah dilakukan dan menanyakan apakah pengguna ingin melanjutkan atau tidak. Kondisi `equalsIgnoreCase("tidak")` digunakan agar program berhenti tanpa mempermasalahkan huruf besar atau kecil.

## 2.2 Program Lempardadu

```
1 package pekan6_2511532001;
2
3 import java.util.Random;
4
5 public class Lempardadu_2511532001 {
6
7     public static void main(String[] args) {
8         Random rand = new Random();
9         int tries = 0;
10        int sum = 0;
11        while (sum != 7) {
12
13            int dadu1 = rand.nextInt (6) + 1;
14            int dadu2 = rand.nextInt (6) + 1;
15            sum = dadu1 + dadu2;
16            System.out.println(dadu1 + " + " + dadu2 + " = " + sum);
17            tries++;
18        }
19        System.out.println("You won after " + tries + " tries!");
20    }
21 }
22
23 }
```

Program ini mensimulasikan permainan lempar dadu random. Dua buah angka acak dari 1–6 dihasilkan dan dijumlahkan setiap kali perulangan berjalan. Perulangan akan terus dijalankan selama jumlah kedua dadu belum sama dengan 7. Setelah kondisi terpenuhi ( $\text{sum} == 7$ ), program berhenti dan menampilkan jumlah percobaan (tries) yang diperlukan.

## 2.3 Program GamePenjumlahan

```

1 package pekan6_2511532001;
2
3 import java.util.Random;
4
5 import java.util.Scanner;
6
7
8 public class GamePenjumlahan_2511532001 {
9
10     public static void main(String[] args) {
11         Scanner console = new Scanner(System.in);
12         Random rand = new Random();
13
14         int points = 0;
15         int wrong = 0;
16         while (wrong < 3) {
17             int result = play(console, rand);
18             if (result > 0) {
19                 points++;
20             } else {
21                 wrong++;
22             }
23         }
24         System.out.println("You earned " + points + " total points.");
25     }
26     public static int play(Scanner console, Random rand) {
27         int operands = rand.nextInt(4) + 2;
28         int sum = rand.nextInt(10) + 1;
29         System.out.print(sum);
30
31         for (int i = 2; i <= operands; i++) {
32             int n = rand.nextInt(10) + 1;
33             sum += n;
34             System.out.print(" + " + n);
35
36         }
37         System.out.print(" = ");
38
39         int guess = console.nextInt();
40         if (guess == sum) {
41             return 1;
42
43             return 1;
44         } else {
45             System.out.println("Wrong! The answer was "+sum);
46             return 0;
47         }
48     }
49 }
50
51

```

Program ini adalah permainan penjumlahan sederhana. Pemain diberi soal penjumlahan acak yang dihasilkan oleh fungsi play().

Setiap jawaban benar menambah 1 poin, sedangkan jawaban salah menambah jumlah kesalahan (wrong). Perulangan while akan terus berjalan hingga jumlah kesalahan mencapai 3 kali, lalu program berhenti dan menampilkan total skor yang diperoleh.

#### 2.4 Program doWhile1

```
1 package pekan6_2511532001;
2
3 import java.util.Scanner;
4
5 public class doWhile1_2511532001 {
6
7     public static void main(String[] args) {
8         Scanner console = new Scanner(System.in);
9         String phrase;
10        do {
11            System.out.print("Input Password: ");
12            phrase = console.next();
13            } while (!phrase.equals("abcd"));
14        }
15
16    }
```

Program ini menggunakan struktur do-while untuk meminta pengguna memasukkan kata sandi (password).

Blok perulangan dijalankan minimal satu kali untuk membaca input pengguna, lalu memeriksa apakah nilai yang dimasukkan sama dengan kata sandi yang benar, yaitu "abcd". Jika tidak sesuai, program akan meminta input ulang hingga kondisi terpenuhi.



## 2.5 Program SentinelLoop

```
1 package pekan6_2511532001;
2
3 import java.util.Scanner;
4
5 public class SentinelLoop_2511532001 {
6
7     public static void main(String[] args) {
8         Scanner console = new Scanner(System.in);
9         int sum = 0;
10        int number = 12; //dummy value
11
12        while (number != 0) {
13            System.out.print("Masukkan angka (0 untuk keluar): ");
14            number = console.nextInt();
15            sum = sum + number;
16        }
17        System.out.println("totalnya adalah "+sum);
18    }
19
20 }
```

Program ini merupakan penerapan sentinel-controlled loop, yaitu perulangan yang berhenti ketika pengguna memasukkan nilai tertentu (nilai sentinel). Dalam hal ini, angka 0 digunakan sebagai tanda berhenti.

Selama input bukan 0, nilai akan terus dijumlahkan ke variabel sum. Setelah pengguna mengetik 0, kondisi `number != 0` menjadi salah dan program keluar dari perulangan, menampilkan hasil total seluruh angka yang dimasukkan.

## **BAB 3**

### **KESIMPULAN**

#### **3.1 Ringkasan Hasil Praktikum**

Dari hasil praktikum yang telah dilakukan pada materi Perulangan While dan Do-While, dapat disimpulkan bahwa kedua struktur perulangan ini sangat berguna dalam situasi di mana jumlah pengulangan belum diketahui di awal program. Struktur while digunakan untuk mengeksekusi blok kode selama kondisi tertentu bernilai true, sedangkan do-while menjamin bahwa blok kode akan dijalankan minimal satu kali sebelum kondisi diperiksa.

Kedua jenis perulangan ini memungkinkan program menjadi lebih fleksibel, efisien, dan interaktif karena dapat beradaptasi terhadap masukan atau keputusan dari pengguna. Dalam implementasinya, konsep ini sering digunakan untuk melakukan proses berulang seperti validasi data, pengulangan input, simulasi permainan, dan pemrosesan data yang tidak memiliki batas tetap.

Agar pemahaman terhadap materi perulangan menjadi lebih mendalam, sebaiknya latihan tidak hanya terbatas pada contoh dasar, tetapi juga mencakup penerapan dalam konteks yang lebih kompleks seperti permainan interaktif, simulasi proses, dan pengolahan data numerik.

## **DAFTAR PUSTAKA**

- [1] Oracle, “The while and do-while Statements,” The Java Tutorials: Control Flow Statements, 2023. [Daring]. Tersedia pada:  
<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/while.html> [Diakses: 5 Nov 2025].
- [2] W3Schools, “Java While Loop,” Java Tutorial, 2025. [Daring]. Tersedia pada:  
[https://www.w3schools.com/java/java\\_while\\_loop.asp](https://www.w3schools.com/java/java_while_loop.asp). [Diakses: 5 Nov 2025].