

**LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN  
PERULANGAN WHILE**



**Oleh:**

**Naufal Arkan Octyandi**

**2511533005**

**MATA KULIAH ALGORITMA PEMROGRAMAN**

**DOSEN PENGAMPU: DR. WAHYUDI, S.T, M.T**

**ASISTEN LABOR: AUFAN TAUFIQURRAHMAN**

**FAKULTAS TEKNOLOGI INFORMASI**

**PEPARTEMEN INFORMATIKA**

**UNIVERSITAS ANDALAS**

**PADANG, 23 SEPTEMBER 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, 2025**

**Tim Penyusun**

**Naufal Arkan Octyandi**

## DAFTAR ISI

<b>KATA PENGANTAR .....</b>	i
<b>DAFTAR ISI .....</b>	ii
<b>BAB I PENDAHULUAN .....</b>	1
1.1 Latar Belakang Praktikum .....	1
1.2 Tujuan Praktikum .....	1
<b>BAB II PENULISAN LAPORAN PRAKTIKUM.....</b>	2
2.1 Perulangan While .....	2
2.2 Do While.....	3
2.3 Sentinel Loop.....	3
<b>BAB III KESIMPULAN.....</b>	6
3.1 Ringkasan.....	6
<b>DAFTAR PUSTAKA.....</b>	7

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang Praktikum**

Dalam pemrograman, perulangan (looping) merupakan salah satu konsep dasar yang sangat penting untuk mengotomatisasi eksekusi kode secara berulang tanpa harus menulis perintah yang sama berkali-kali. Salah satu jenis perulangan yang umum digunakan adalah perulangan while. Struktur perulangan ini digunakan ketika jumlah iterasi belum diketahui secara pasti, namun terdapat suatu kondisi logis yang menjadi penentu berlanjut atau berhentinya proses perulangan. Melalui perulangan while, program dapat melakukan suatu proses secara berulang selama kondisi bernilai benar (true). Apabila kondisi bernilai salah (false), maka perulangan akan berhenti secara otomatis. Pada Pekan 6 kita sudah melakukan uji coba terhadap perulangan while itu sendiri dalam sebuah kode program java. Tujuannya agar mahasiswa memahami secara praktis mengenai cara kerja dan penerapan struktur perulangan tersebut.

#### **1.2 Tujuan Praktikum**

1. Mengetahui konsep dasar perulangan while serta cara kerjanya dalam pemrograman.
2. Mampu menuliskan dan menjalankan program sederhana yang menggunakan struktur perulangan while.
3. Memahami perbedaan antara perulangan while dengan jenis perulangan lainnya seperti for dan do while.
4. Melatih kemampuan logika dalam menentukan kondisi berhenti (terminating condition) pada suatu perulangan.
5. Mengembangkan keterampilan dalam menyusun program yang efektif menggunakan struktur kontrol while.

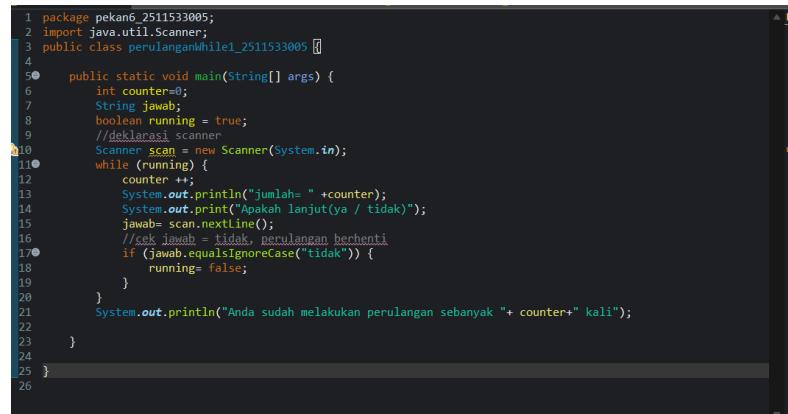
## BAB II

### PENULISAN LAPORAN PRAKTIKUM

#### 2.1 Perulangan While

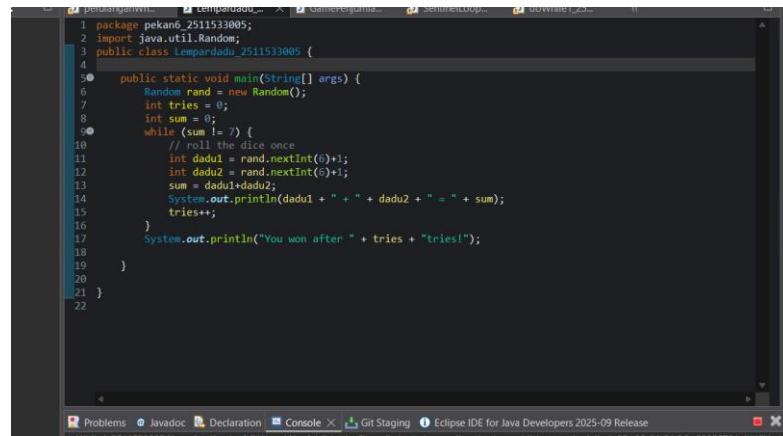
Perulangan while digunakan ketika kita tidak tahu pasti berapa kali perulangan akan dijalankan, tetapi kita tahu kondisi yang harus dipenuhi agar perulangan terus berjalan. Selama kondisi bernilai true (benar), maka blok kode di dalam while akan terus dieksekusi.

Bisa kita liat contohnya di pekan 6 kali ini



```
1 package pekan6_2511533005;
2 import java.util.Scanner;
3 public class perulanganWhile1_2511533005 {
4
5     public static void main(String[] args) {
6         int counter=0;
7         String jawab;
8         boolean running = true;
9         //deklarasi scanner
10        Scanner scan = new Scanner(System.in);
11        while (running) {
12            counter++;
13            System.out.println("jumlah= " +counter);
14            System.out.print("Apakah lanjut(ya / tidak)");
15            jawab= scan.nextLine();
16            //cek jawab = tidak, perulangan berhenti
17            if (jawab.equalsIgnoreCase("tidak")) {
18                running= false;
19            }
20        }
21        System.out.println("Anda sudah melakukan perulangan sebanyak "+ counter+" kali");
22
23    }
24
25 }
26
```

Lalu ada contoh 1 lagi



```
1 package pekan6_2511533005;
2 import java.util.Random;
3 public class LemparanDadu_2511533005 {
4
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             // roll the dice once
11             int dadu1 = rand.nextInt(6)+1;
12             int dadu2 = rand.nextInt(6)+1;
13             sum = dadu1+dadu2;
14             System.out.println(dadu1 + " " + dadu2 + " = " + sum);
15             tries++;
16         }
17         System.out.println("You won after " + tries + " tries!");
18
19     }
20
21 }
22
```

Kedua program ini sama-sama menggunakan struktur perulangan while, namun keduanya memiliki perbedaan dari segi logika dan cara kerja perulangannya.

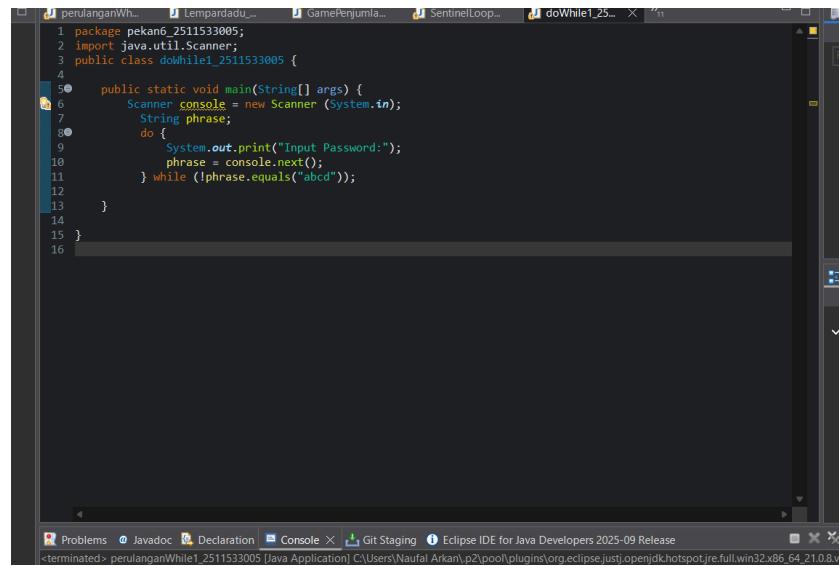
Pada program perulanganWhile1, perulangan dikendalikan oleh input pengguna. Program akan terus berjalan selama pengguna menjawab “ya”, dan akan berhenti apabila pengguna mengetik “tidak”. Sementara itu, pada program

LemparDadu, perulangan berlangsung secara otomatis berdasarkan kondisi logika internal dalam program. Perulangan akan terus dijalankan selama nilai variabel sum belum sama dengan 7, dan akan berhenti ketika hasil penjumlahan dua dadu menghasilkan angka 7. Secara keseluruhan, perbedaan utama antara kedua program terletak pada sumber kondisi berhentinya perulangan.

## 2.2 Do While

Perulangan do while mirip dengan while, tapi kondisinya diperiksa di akhir. Artinya, kode di dalam perulangan dijalankan minimal satu kali, meskipun kondisi langsung salah di awal.

Bisa kita liat contohnya di pekan 6 kali ini



The screenshot shows the Eclipse IDE interface with several tabs at the top: 'perulanganWhile1\_2511533005', 'tempdadu...', 'GamePerjudia...', 'SentinelLoop...', and 'doWhile1\_25...'. The main editor window displays the following Java code:

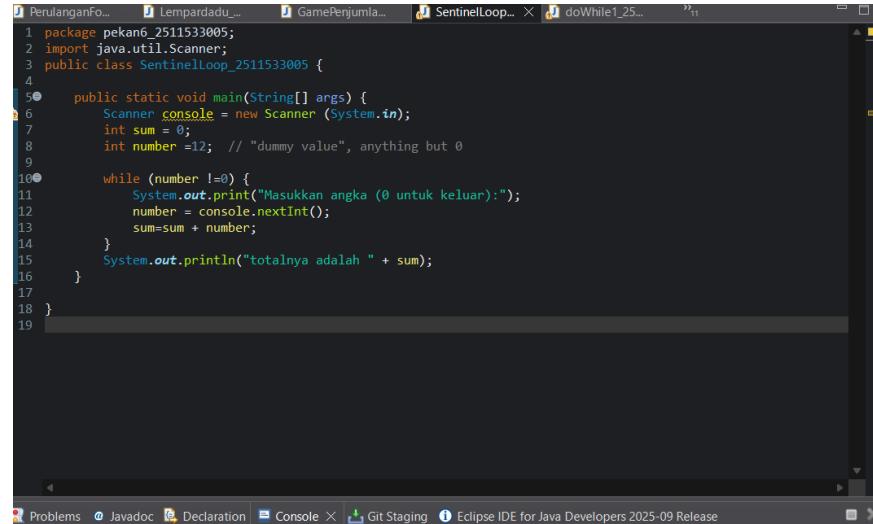
```
1 package pekan6_2511533005;
2 import java.util.Scanner;
3 public class doWhile1_2511533005 {
4
5     public static void main(String[] args) {
6         Scanner console = new Scanner (System.in);
7         String phrase;
8         do {
9             System.out.print("Input Password:");
10            phrase = console.next();
11        } while (!phrase.equals("abcd"));
12    }
13
14 }
15
16
```

The code defines a class named 'doWhile1\_2511533005' with a main method. It uses a Scanner to read input from the console. A do-while loop is used to repeatedly prompt the user for a password until it matches 'abcd'. The code is terminated by a closing brace on line 15.

## 2.3 Sentinel Loop

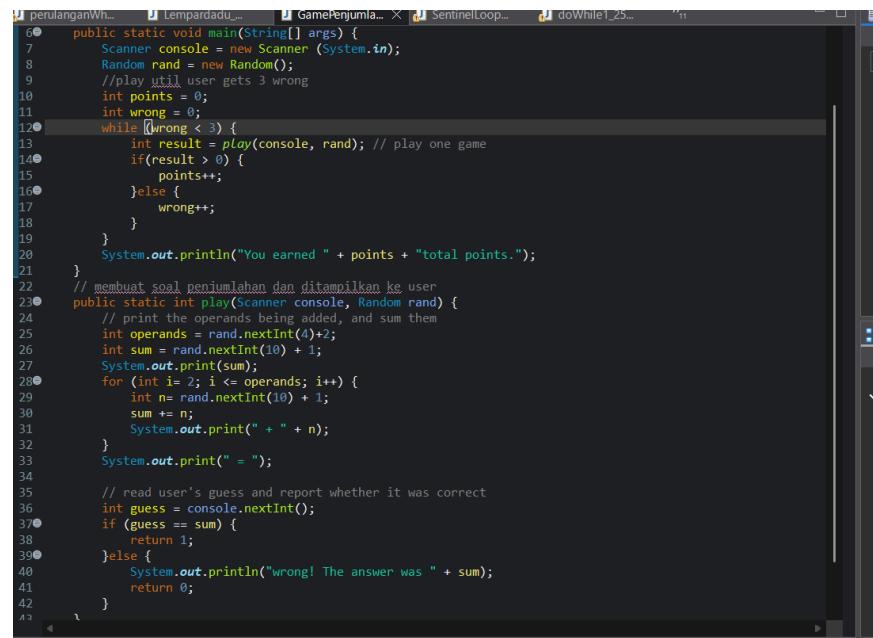
Sentinel loop adalah jenis perulangan yang berhenti saat menemukan nilai tertentu (disebut *sentinel value*). Biasanya digunakan ketika jumlah data tidak diketahui, dan pengguna akan memberi tanda khusus untuk menghentikan input.

Bisa kita liat contohnya di pekan 6 kali ini



```
1 package pekan6_2511533005;
2 import java.util.Scanner;
3 public class SentinelLoop_2511533005 {
4
5    public static void main(String[] args) {
6        Scanner console = new Scanner (System.in);
7        int sum = 0;
8        int number =12; // "dummy value", anything but 0
9
10       while (number !=0) {
11           System.out.print("Masukkan angka (0 untuk keluar):");
12           number = console.nextInt();
13           sum=sum + number;
14       }
15       System.out.println("totalnya adalah " + sum);
16   }
17
18 }
```

Lalu ada contoh 1 lagi



```
1 public static void main(String[] args) {
2     Scanner console = new Scanner (System.in);
3     Random rand = new Random();
4     //play until user gets 3 wrong
5     int points = 0;
6     int wrong = 0;
7     while (wrong < 3) {
8         int result = play(console, rand); // play one game
9         if(result > 0) {
10             points++;
11         }else {
12             wrong++;
13         }
14     }
15     System.out.println("You earned " + points + "total points.");
16 }
17 // membuat soal penjumlahan dan ditampilkan ke user
18 public static int play(Scanner console, Random rand) {
19     // print the operands being added, and sum them
20     int operands = rand.nextInt(4)+2;
21     int sum = rand.nextInt(10) + 1;
22     System.out.print(sum);
23     for (int i= 2; i <= operands; i++) {
24         int n= rand.nextInt(10) + 1;
25         sum += n;
26         System.out.print(" + " + n);
27     }
28     System.out.print(" = ");
29     // read user's guess and report whether it was correct
30     int guess = console.nextInt();
31     if (guess == sum) {
32         return 1;
33     }else {
34         System.out.println("wrong! The answer was " + sum);
35         return 0;
36     }
37 }
```

Kedua program sama-sama menggunakan konsep perulangan, tetapi memiliki tujuan dan cara kerja yang berbeda dalam penerapan logikanya. Pada program GamePenjumlahan, perulangan digunakan untuk membuat permainan interaktif di mana pengguna harus menjawab hasil penjumlahan dua angka acak yang

diberikan oleh komputer. Program akan terus memberikan soal baru selama pengguna menjawab dengan benar, kemudian berhenti saat pengguna menjawab salah atau memilih untuk keluar. Jenis perulangan seperti ini berfokus pada validasi jawaban dan kondisi permainan, di mana keputusan berhenti ditentukan oleh kebenaran input dari pengguna terhadap hasil operasi matematika. Sedangkan pada program SentinelLoop, perulangan digunakan untuk membaca atau memproses data berulang hingga ditemukan nilai penanda (sentinel) yang menandakan akhir input. Pada program ini, perulangan akan terus berjalan selama nilai yang dimasukkan pengguna tidak sama dengan nilai sentinel yang sudah ditentukan sebelumnya. Secara keseluruhan, perbedaan paling mendasar antara keduanya terletak pada fungsi dan kondisi berhentinya perulangan.

## **BAB III**

### **KESIMPULAN**

#### **3.1 Ringkasan**

Berdasarkan hasil praktikum yang dilakukan pada berbagai program berbasis perulangan while, dapat disimpulkan bahwa struktur perulangan while memiliki peran penting dalam mengontrol jalannya program berdasarkan kondisi logika yang ditentukan. Setiap program menerapkan konsep while dengan karakteristik dan tujuan yang berbeda, namun prinsip dasarnya tetap sama, yaitu menjalankan blok perintah selama kondisi bernilai benar (true). Secara keseluruhan, dapat disimpulkan bahwa perulangan while sangat fleksibel dan dapat digunakan untuk berbagai kebutuhan, baik yang dikontrol oleh pengguna (sentinel loop) maupun oleh logika program (conditional loop). Praktikum ini menunjukkan bahwa pemahaman terhadap kondisi logika dan pengendalian perulangan sangat penting untuk membangun program yang efisien, interaktif, dan sesuai dengan tujuan yang diinginkan.

## **DAFTAR PUSTAKA**

DuniaIlkom. (2020). *Tutorial Belajar Java Part 35: Perulangan DO WHILE Bahasa Java.*

<https://www.duniaIlkom.com/tutorial-belajar-java-perulangan-do-while-bahasa-java/> DuniaIlkom

Daisma Bali. (2019). *Java – Algoritma Perulangan.*

[https://daismabali.com/artikel\\_detail/8/1/Java---Algoritma-Perulangan.html](https://daismabali.com/artikel_detail/8/1/Java---Algoritma-Perulangan.html)  
[daismabali.com](http://daismabali.com)

Petani Kode. (2016). *Belajar Java: Memahami 2 Jenis Perulangan dalam Java.*

<https://www.petanikode.com/java-perulangan/> Petani Kode

KodeDasar. (2023). *3+ Contoh Program Looping pada Java dengan For, While, Do-While.*

<https://kodedasar.com/blog/perulangan-java/>