

**LAPORAN PRAKTIKUM**  
**AGORITMA PEMROGRAMAN**  
**“PERULANGAN WHILE dan DO WHILE”**

**disusun Oleh:**

**SOFIAN ARBA’I**

**2511533029**

**Dosen Pengampu:**

**Dr. WAHYUDI, S.T, M.T**

**Asisten Praktikum:**

**JOVANTRI IMMANUEL GULO**



**DEPARTEMEN INFORMATIKA**  
**FAKULTAS TEKNOLOGI INFORMASI**  
**UNIVERSITAS ANDALAS**

**2025**

## KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena atas rahmat dan karunia-Nya, laporan praktikum Algoritma dan Pemrograman pekan 6 dengan judul *Perulangan While dan Do While* ini dapat diselesaikan dengan baik dan tepat waktu. Laporan ini disusun sebagai bentuk dokumentasi dari kegiatan praktikum yang bertujuan untuk memahami dan mengimplementasikan konsep perulangan (*looping*).

Dalam bahasa pemrograman Java, khususnya menggunakan struktur *while* dan *do while*. Materi ini merupakan bagian penting dalam pengembangan logika program yang efisien dan dinamis, karena memungkinkan eksekusi berulang berdasarkan kondisi tertentu. Laporan ini membahas sintaks dasar, alur eksekusi, serta penerapan *while* dan *do while* dalam berbagai studi kasus sederhana.

Penyusunan laporan ini tidak terlepas dari bantuan, bimbingan, dan dukungan berbagai pihak. Oleh karena itu, saya ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Dosen pengampu mata kuliah Praktikum Algoritma dan Pemrograman yang telah memberikan ilmu, arahan, serta bimbingannya.
2. Asisten praktikum yang telah membimbing dan memberikan penjelasan selama kegiatan praktikum berlangsung.

Saya meminta maaf jika banyak terdapat kesalahan pada penulisan laporan ini. Oleh karena itu, saya menerima kritik dan saran yang membangun demi perbaikan di masa mendatang.

Padang, 28 Oktober 2025

Sofian Arba'i

## DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI .....	ii
BAB I PENDAHULUAN .....	1
1.1    Latar Belakang .....	1
1.2    Tujuan .....	1
1.3    Manfaat .....	1
BAB II PEMBAHASAN .....	3
2.1    Pengertian Perulangan While dan Do While .....	3
2.2    Perbedaan Perulangan While dan Do... While .....	3
2.3    Aplikasi Perulangan While dan Do... While .....	4
2.3.1    Kode Program Perulangan While .....	4
2.3.2    Kode Program Perulangan While .....	6
2.3.3    Kode Program Perulangan While .....	7
2.3.4    Kode Program Perulangan While .....	9
2.3.5    Kode Program Perulangan Do... While .....	10
BAB III PENUTUP .....	12
3.1    Kesimpulan .....	12
3.2    Saran .....	12
DAFTAR PUSTAKA.....	13

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Dalam pemrograman, perulangan merupakan salah satu konsep dasar yang sangat penting untuk dipahami. Perulangan memungkinkan program untuk mengeksekusi suatu blok kode secara berulang-ulang selama kondisi tertentu terpenuhi. Dua struktur perulangan yang umum digunakan dalam bahasa pemrograman Java adalah *while* dan *do while*. Kedua struktur ini memiliki karakteristik dan kegunaan masing-masing yang dapat disesuaikan dengan kebutuhan logika program.

Pemahaman yang baik terhadap perulangan *while* dan *do while* sangat diperlukan untuk membangun program yang efisien, terutama dalam kasus yang melibatkan pengulangan dengan jumlah iterasi yang tidak pasti. Oleh karena itu, praktikum ini dirancang untuk memberikan pengalaman langsung kepada mahasiswa dalam mengimplementasikan kedua jenis perulangan tersebut melalui studi kasus dan latihan pemrograman.

### **1.2 Tujuan**

Tujuan dari praktikum pekan 6 sebagai berikut:

1. Memahami konsep dasar dan perbedaan antara struktur perulangan *while* dan *do while* dalam Java.
2. Mampu mengimplementasikan perulangan *while* dan *do while* dalam program Java sesuai dengan kebutuhan logika.
3. Melatih kemampuan analisis dalam menentukan jenis perulangan yang tepat untuk menyelesaikan suatu permasalahan.
4. Meningkatkan keterampilan debugging dan pengujian program yang menggunakan struktur perulangan.

### **1.3 Manfaat**

Manfaat yang didapatkan setelah praktikum sebagai berikut:

1. Mahasiswa dapat mengembangkan logika pemrograman yang lebih sistematis dan efisien.
2. Mahasiswa mampu memilih dan menerapkan struktur perulangan yang sesuai dalam berbagai konteks pemrograman.
3. Mahasiswa memperoleh pengalaman praktis dalam menyusun program yang melibatkan proses iteratif.
4. Mahasiswa lebih siap menghadapi tantangan pemrograman lanjutan yang memerlukan pemahaman mendalam tentang kontrol alur program.

## **BAB II**

### **PEMBAHASAN**

#### **2.1 Pengertian Perulangan While dan Do While**

Perulangan while merupakan sebuah pernyataan mengulang suatu proses yang belum diketahui jumlah perulangannya. Ciri utama penggunaan pernyataan perulangan while adalah akan diuji syarat yang ditentukan terlebih dahulu. Sehingga jika syarat tidak terpenuhi maka pernyataan pada perulangan akan berhenti. Adapun bentuk perulangan while, sebagai berikut: While (condition) {statement(s);}. Perulangan while berguna untuk memproses suatu pernyataan atau beberapa pernyataan beberapa kali. Selama ungkapan bernilai benar, pernyataan akan selalu dikerjakan. Pernyataan perulangan dengan while akan selalu dikerjakan jika ungkapan selalu benar. Oleh karena itu, kita harus membuat kondisi suatu saat ungkapan bernilai salah agar perulangan berakhir.

Perulangan do...while sama dengan perulangan while yaitu pernyataan mengulang suatu proses yang belum diketahui jumlah perulangannya. Ciri utama penggunaan do...while terdapat pada pernyataan perulangan do...while yang mana akan dilihat terlebih dahulu instruksi/perintah yang ditentukan. Setelah itu, baru dilakukannya uji syarat yang telah ditetapkan. Adapun bentuk umum perulangan do...while, sebagai berikut: Do {statement(s)} While (condition). Perulangan dengan do...while ini juga digunakan untuk mengerjakan sebuah atau sekelompok pernyataan berulang-ulang. Bedanya dengan while adalah pernyataan do...while akan mengecek kondisi di belakang, sementara while cek kondisi ada di depan.

#### **2.2 Perbedaan Perulangan While dan Do...While**

Berikut adalah beberapa perbedaan dari perulangan While dan Do...While:

1. Perulangan While memeriksa kondisi terlebih dahulu lalu mengeksekusi pernyataan, sedangkan perulangan do while akan mengeksekusi pernyataan minimal satu kali, kemudian kondisi diperiksa.
2. Perulangan While adalah perulangan dengan kendali masuk, sedangkan perulangan do while adalah perulangan dengan kendali keluar.
3. Pada perulangan while, kita tidak perlu menambahkan titik koma di akhir kondisi while, namun kita perlu menambahkan titik koma di akhir kondisi while pada perulangan do- while.
4. Pernyataan while loop dieksekusi nol kali jika kondisinya salah, sedangkan pernyataan do- while dieksekusi setidaknya sekali.
5. Perulangan While memungkinkan inisialisasi variabel penghitung sebelum memulai badan perulangan, sedangkan perulangan do while memungkinkan inisialisasi variabel penghitung sebelum dan sesudah memulai badan perulangan.

## **2.3 Aplikasi Perulangan While dan Do...While**

Pada pertemuan praktikum pekan 6 mahasiswa diinstruksikan untuk menyalin kode program tentang perulangan while dan do...while yang telah dibuat oleh dosen. Kode program tersebut diharapkan sebagai latihan dan contoh untuk kode program perulangan while dan do...while. Berikut adalah kode program yang diberikan oleh dosen untuk praktikum pekan 6.

### **2.3.1 Kode Program Perulangan While**

Kode program perulangan while pertama adalah sebuah kode program yang menanyakan kepada pengguna apakah ingin melanjutkan perulangan dengan mengetik “ya” atau “tidak”. Jika pengguna mengetik “tidak”, maka kode program akan berhenti. Berikut adalah kode program nya.

```

1 package pekan6_2511533029;
2 import java.util.Scanner;
3 public class perulanganWhile1_2511533029 {
4
5     public static void main(String[] args) {
6         int counter=0;
7         String jawab;
8         boolean running= true;
9         Scanner scan= new Scanner(System.in);
10        while (running) {
11            counter++;
12            System.out.println("jumlah = "+ counter);
13            System.out.print("Apakah lanjut (ya/tidak?)");
14            jawab = scan.nextLine();
15            if (jawab.equalsIgnoreCase("tidak")) {
16                running= false;
17                scan.close();
18            }
19        }
20        System.out.println("anda sudah melakukan perulangan sebanyak "+counter+" kali");
21    }
22 }
23
24
25

```

Kode Program 2.1

#### Langkah pembuatan kode program

1. Membuat project dan class baru
2. Mendeklarasikan variabel yang diperlukan
3. Membuat Scanner supaya data diinputkan sesuai keinginan pengguna
4. Menulis Struktur Perulangan While
5. Menampilkan Hasil dan Meminta Input
6. Membaca Jawaban dan Mengecek Kondisi

Program Java tersebut berfungsi untuk melakukan perulangan dengan menggunakan struktur while loop, di mana pengguna dapat memilih apakah ingin melanjutkan atau menghentikan perulangan. Program diawali dengan mendeklarasikan variabel counter untuk menghitung jumlah perulangan, jawab untuk menyimpan input pengguna, dan running sebagai kondisi pengontrol perulangan yang bernilai awal true. Objek Scanner dibuat untuk membaca input dari keyboard.

Selama variabel running bernilai true, program akan terus menjalankan blok perulangan. Di dalamnya, nilai counter akan bertambah satu setiap kali loop berjalan, kemudian ditampilkan ke layar



dengan pesan “jumlah = ...”. Setelah itu, program menanyakan kepada pengguna apakah ingin melanjutkan perulangan dengan mengetik “ya” atau “tidak”. Jika pengguna mengetik “tidak”, maka kondisi running diubah menjadi false, sehingga perulangan berhenti dan objek Scanner ditutup.

### 2.3.2 Kode Program Perulangan While

Kode program perulangan while selanjutnya adalah sebuah kode program yang mensimulasikan pelemparan dua buah dadu hingga jumlah keduanya bernilai 7, kemudian menampilkan berapa kali percobaan yang dilakukan sampai kondisi tersebut tercapai. Berikut adalah kode programnya.

```

1 package pekan6_2511533029;
2 import java.util.Random;
3 public class Lempardadu_2511533029 {
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             int dadu1 = rand.nextInt(6);
11             int dadu2 = rand.nextInt(6);
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 }
19
20 }

```

Kode Program 2.2

Langkah pembuatan kode program

1. Membuat project dan class baru
2. Menambahkan import.java.util.random
3. Membuat objek random dan variable
4. Menulis struktur perulangan while, while (sum != 7)
5. Menghasilkan nilai acak dadu lalu menjumlahkannya, sum = dadu1 + dadu2;
6. Menghitung Jumlah Percobaan, dan tampilkan hasil akhir.

Program Java tersebut mensimulasikan pelemparan dua buah dadu hingga jumlah keduanya bernilai 7. Program menggunakan kelas Random untuk menghasilkan angka acak pada kedua dadu, lalu menjumlahkannya dan menampilkan hasil setiap lemparan. Variabel tries digunakan untuk menghitung berapa kali percobaan dilakukan. Selama total nilai kedua dadu belum sama dengan 7, perulangan while terus berjalan. Setelah jumlahnya menjadi 7, loop berhenti dan program menampilkan pesan kemenangan beserta jumlah percobaan yang dibutuhkan.

### 2.3.3 Kode Program Perulangan While

Kode program perulangan while selanjutnya merupakan sebuah kode program yaitu penjumlahan sederhana yang menguji kemampuan pengguna dalam menjawab soal matematika acak. Berikut adalah kode programnya.

```

3 import java.util.Random;
4 public class GamePenjumlahan_2511533029 {
5
6     public static void main(String[] args) {
7         Scanner console= new Scanner(System.in);
8         Random rand= new Random ();
9         // play until user get 3 wrong
10        int points = 0;
11        int wrong = 0;
12        while (wrong <3) {
13            int result = play (console, rand);
14            if (result > 0) {
15                points++;
16            } else {
17                wrong++;
18            }
19        }
20        System.out.println("you earned " + points + "total points.");
21    }
22
23    //membuat soal penjumlahan da ditampilkan ke user
24    public static int play (Scanner console, Random rand) {
25        int operands= rand.nextInt(4)+2;
26        int sum= rand.nextInt(10)+1;
27        System.out.println(sum);
28
29        for (int i=2; i<=operands; i++) {
30            int n= rand.nextInt(10)+1;
31            sum += n;
32            System.out.print(" + " + n);
33        }
34        System.out.print(" = ");
35        // read user guess and report wether it was correct
36        int guess= console.nextInt();
37        if (guess==sum) {
38            return 1;
39        }else {
40            System.out.println("wrong! the answer was " +sum);
41            return 0;
42        }
43    }
44 }

```

Kode Program 2.3

Langkah pembuatan kode program

1. Membuat class baru dengan nama `_NIM`
2. Buat objek Scanner dan Random dan jangan lupa import nya agar kode bisa dijalankan
3. Buat variabel points dan wrong untuk mencatat skor dan kesalahan.
4. Gunakan loop while (`wrong < 3`) agar permainan berlanjut sampai pengguna salah tiga kali.
5. Panggil metode `play()` untuk membuat dan menampilkan soal.
6. Di dalam `play()`, buat angka acak dan tampilkan operasi penjumlahan ke layar.
7. Baca jawaban pengguna dan cek kebenarannya menggunakan `if (guess == sum)`

Program Java di atas merupakan game penjumlahan sederhana yang menguji kemampuan pengguna dalam menjawab soal matematika acak. Program menggunakan kelas Scanner untuk membaca input pengguna dan kelas Random untuk menghasilkan angka acak.

Ketika dijalankan, program akan terus memberikan soal penjumlahan acak selama pengguna belum salah sebanyak tiga kali. Setiap kali pengguna menjawab dengan benar, nilai points bertambah satu; jika salah, nilai wrong bertambah. Setelah pengguna melakukan tiga kesalahan, permainan berhenti dan total poin akan ditampilkan.

Metode `play()` bertugas membuat soal penjumlahan acak. Program menentukan berapa banyak bilangan yang akan dijumlahkan (antara 2 hingga 5), lalu menghasilkan angka acak antara 1 hingga 10 untuk tiap bilangan tersebut. Program menampilkan soal ke layar, meminta jawaban pengguna, dan memeriksa apakah jawabannya benar. Jika benar, mengembalikan nilai 1; jika salah, menampilkan jawaban yang benar dan mengembalikan nilai 0.

### 2.3.4 Kode Program Perulangan While

Kode program perulangan while selanjutnya merupakan sebuah kode program untuk menjumlahkan angka yang dimasukkan oleh pengguna hingga pengguna mengetik angka 0 sebagai tanda berhenti. Berikut adalah kode programnya.

```

1 package pekan6_2511533029;
2 import java.util.Scanner;
3 public class Sentinelloop_2511533029 {
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 berhenti):");
12            number= console.nextInt();
13            sum= sum+ number;
14            console.close();
15        }
16        System.out.println("totalnya adalah " + sum);
17    }
18
19 }

```

Kode Program 2.3

Langkah pembuatan kode program

1. Membuat class baru dengan nama \_NIM
2. Import library java.util.Scanner
3. buat objek Scanner untuk menerima input pengguna.
4. Deklarasikan variabel sum untuk menampung hasil penjumlahan dan number dengan nilai awal bukan 0.
5. Gunakan loop while (number != 0) untuk meminta angka berulang kali.
6. Tambahkan setiap angka ke variabel sum.
7. Jika pengguna memasukkan angka 0, loop berhenti.

Program Java pada gambar tersebut berfungsi untuk menjumlahkan angka yang dimasukkan oleh pengguna hingga pengguna mengetik angka 0 sebagai tanda berhenti. Program ini

menggunakan konsep sentinel loop, yaitu pengulangan yang akan berhenti ketika nilai tertentu (dalam hal ini angka 0) dimasukkan.

Saat dijalankan, program akan terus meminta pengguna untuk memasukkan angka melalui Scanner. Setiap angka yang dimasukkan akan ditambahkan ke variabel sum untuk menghitung totalnya. Jika pengguna memasukkan angka 0, maka kondisi while (number != 0) akan menjadi salah, dan loop berhenti. Setelah itu, program menampilkan jumlah total dari semua angka yang telah dimasukkan.

### 2.3.5 Kode Program Perulangan Do...While

Kode program terakhir merupakan kode program untuk perulangan do...while, yaitu sebuah kode program untuk meminta pengguna memasukkan password hingga benar, menggunakan struktur perulangan do-while. Berikut adalah kode programnya.

```

1 package pekan6_2511533029;
2 import java.util.Scanner;
3 public class doWhile1_2511533029 {
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.nextLine();
11        } while (!phrase.equals("abcd"));
12
13    }
14
15 }
```

Kode Program 2.5

Langkah pembuatan kode program

1. Membuat class baru dengan nama\_NIM
2. Membuat Import.java.util.Scanner.
3. Membuat objek Scanner untuk input dari pengguna.
4. Deklarasikan variabel phrase bertipe String.
5. Gunakan perulangan do-while untuk meminta input password.
6. Di dalam blok do, tampilkan pesan "input password:" lalu baca input dengan nextLine().

7. Cek kondisi pada bagian while (!phrase.equals("abcd")).
8. Jika input salah, ulangi; jika benar, hentikan loop.

Program Java pada gambar di atas berfungsi untuk meminta pengguna memasukkan password sampai benar, menggunakan perulangan do-while. Program akan terus meminta input password sampai pengguna mengetikkan kata sandi yang benar, yaitu “abcd”.

Penjelasannya, pertama program membuat objek Scanner untuk membaca input dari keyboard, dan mendeklarasikan variabel phrase bertipe String untuk menampung password yang dimasukkan. Pada blok do, program meminta pengguna untuk menginput password, kemudian menyimpannya ke dalam variabel phrase. Setelah itu, kondisi pada while (!phrase.equals("abcd")) akan diperiksa — jika password tidak sama dengan “abcd”, maka pengulangan akan dijalankan kembali. Sebaliknya, jika password yang dimasukkan benar, maka loop berhenti.

Struktur do-while di sini memastikan bahwa blok perintah dijalankan minimal satu kali, meskipun kondisi salah sejak awal.

## **BAB III**

### **PENUTUP**

#### **3.1 Kesimpulan**

Praktikum pekan ke-6 mengenai perulangan *while* dan *do while* dalam bahasa pemrograman Java memberikan pemahaman yang mendalam tentang struktur kontrol alur program yang bersifat iteratif. Melalui berbagai studi kasus dan implementasi kode, mahasiswa dapat memahami bahwa perulangan *while* melakukan pengecekan kondisi sebelum mengeksekusi pernyataan, sedangkan *do while* mengeksekusi pernyataan terlebih dahulu sebelum memeriksa kondisi. Perbedaan ini menjadikan masing-masing struktur memiliki keunggulan tersendiri tergantung pada kebutuhan logika program. Praktikum ini juga memperlihatkan bagaimana perulangan dapat digunakan dalam berbagai skenario, seperti validasi input, simulasi acak, permainan interaktif, dan pengolahan data berulang. Dengan latihan yang diberikan, mahasiswa mampu meningkatkan kemampuan analisis, debugging, dan pemilihan struktur perulangan yang tepat dalam menyelesaikan masalah pemrograman.

#### **3.2 Saran**

disarankan kepada mahasiswa untuk terus berlatih membuat program dengan variasi logika dan kondisi yang berbeda. Latihan mandiri akan membantu memperkuat intuisi dalam memilih struktur perulangan yang sesuai. Selain itu, dosen dan asisten praktikum diharapkan dapat memberikan tantangan tambahan berupa studi kasus yang lebih kompleks agar mahasiswa terbiasa berpikir algoritmik dan menyusun solusi yang efisien. Penulisan dokumentasi dan penjelasan kode juga perlu dilakukan secara sistematis agar memudahkan proses evaluasi dan pembelajaran ulang.

## DAFTAR PUSTAKA

- [1] Dadan Rahmat, Sylpha A'zizah, Sri Mulyani "PERKALIA MENGGUNAKAN BAHASA PEMROGRAMAN PERULANGAN (LOOPING) BERBANTU SUBLIME TEXT DAN XAMPP," Journal of Information Technology Research, Vol. 3, no. 2, e-Issn: 2745-375, p-Issn: 2776-8546, 2022.
- [2] Evi Pratiwi, Konsep Dasar Algoritma dan Pemrograman dengan Bahasa Java, , 2020.
- [3] GURU99, "Perbedaan antara perulangan while dan do- while di C, ". [Daring]. Tersedia pada: [Perbedaan antara perulangan while dan do- while di C](#) [Diakses: 5-Nov-2025].