

LAPORAN PRATIKUM PEKAN 4
ALGORITMA DAN PEMROGRAMAN

disusun Oleh:

ALIF MUHAMMAD IHSAN

NIM 2511531004

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

Asisten Laboratorium: Muhammad Zaki Al Hafiz



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS

PADANG

2025

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah Swt atas limpahan rahmat dan karunia-Nya sehingga praktikum pemrograman Java ini dapat diselesaikan dengan baik dan laporan ini dapat disusun sebagai bentuk pertanggungjawaban akademik.

Laporan praktikum ini disusun untuk mendokumentasikan proses pembelajaran yang telah dilakukan, khususnya dalam memahami struktur dasar pemrograman Java, penggunaan metode, serta penerapan logika prosedural dalam bentuk program sederhana. Praktikum ini diharapkan dapat memperkuat pemahaman peserta terhadap konsep-konsep dasar pemrograman serta meningkatkan kemampuan teknis dalam menulis kode yang terstruktur dan efisien.

Penulis menyadari bahwa penyusunan laporan ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada dosen pembimbing, asisten praktikum, serta rekan-rekan yang telah memberikan dukungan dan masukan selama pelaksanaan praktikum.

Akhir kata, penulis berharap laporan ini dapat memberikan manfaat dan menjadi referensi yang berguna bagi pembaca dalam memahami dasar-dasar pemrograman Java. Kritik dan saran yang membangun sangat diharapkan demi penyempurnaan laporan di masa mendatang.

Padang, September 2025

Alif Muhammad Ihsan

Daftar Isi

KATA PENGANTAR.....	i
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Tujuan.....	1
1.3 Manfaat Praktikum	2
BAB II	3
PEMBAHASAN	3
2.1 Kode Program	3
2.2 Langkah Kerja	9
2.3 Analisis Hasil	10
BAB III.....	13
PENUTUP.....	13
3.1 Kesimpulan	13
3.2 Saran	13
DAFTAR PUSTAKA.....	14

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam era perkembangan teknologi informasi yang semakin pesat, pemahaman terhadap dasar-dasar pemrograman menjadi hal yang esensial bagi setiap individu yang ingin berkontribusi dalam bidang rekayasa perangkat lunak. Salah satu bahasa pemrograman yang banyak digunakan dalam dunia akademik maupun industri adalah Java. Bahasa ini dikenal karena sifatnya yang berorientasi objek, portabel, serta memiliki struktur sintaks yang relatif mudah dipahami oleh pemula.

Praktikum ini dilaksanakan sebagai bagian dari upaya untuk memperkuat pemahaman mahasiswa terhadap konsep dasar pemrograman Java, khususnya dalam hal struktur program, penggunaan metode, serta logika eksekusi program. Melalui kegiatan praktikum, peserta diharapkan mampu menerapkan teori yang telah dipelajari dalam bentuk implementasi nyata, sehingga tercipta keterkaitan antara aspek konseptual dan praktik langsung.

1.2 Tujuan

1. Memahami struktur dasar program Java, termasuk penggunaan metode, pemanggilan fungsi, dan alur eksekusi program secara sistematis.
2. Melatih kemampuan dalam menyusun algoritma prosedural, melalui implementasi langkah-langkah logis yang merepresentasikan proses nyata, seperti pembuatan kue.
3. Meningkatkan keterampilan teknis dalam menulis kode yang terstruktur dan efisien, serta membiasakan diri dengan praktik penulisan program yang baik dan mudah dipahami.

4. Mendorong kemampuan berpikir kritis dan analitis, khususnya dalam mengidentifikasi urutan proses, menyederhanakan logika, dan menerjemahkan aktivitas sehari-hari ke dalam bentuk algoritma.

1.3 Manfaat Praktikum

1. **Memperkuat pemahaman konsep dasar pemrograman Java**, sehingga peserta mampu mengaitkan teori dengan implementasi nyata.
2. **Mengasah keterampilan penyusunan algoritma prosedural**, melalui pembiasaan merinci setiap langkah kerja sebelum menerjemahkannya ke dalam kode.
3. **Meningkatkan kemampuan menulis kode yang modular dan efisien**, dengan memanfaatkan metode sebagai unit kerja yang terpisah dan terstruktur.
4. **Memberikan pengalaman langsung dalam penerjemahan proses sehari-hari ke dalam program**, sehingga peserta dapat melihat relevansi pemrograman dalam konteks kehidupan nyata.
5. **Melatih kedisiplinan, kerapian, dan kemampuan dokumentasi**, yang menjadi landasan penting bagi profesional di bidang rekayasa perangkat lunak.

BAB II

PEMBAHASAN

2.1 Kode Program

a. Nilai

```
package pekan4;

import java.util.Scanner;

public class Nilai {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        int nilai;

        Scanner input=new Scanner(System.in);

        System.out.print("Inputkan nilai angka= ");

        nilai= input.nextInt();

        input.close();

        if(nilai >=81) {

            System.out.println("A");

            } else if (nilai >=70) {

                System.out.println("B");

            } else if (nilai >=60) {

                System.out.println("C");

            } else if (nilai >=50) {

                System.out.println("D");

            } else {
```

```

        System.out.println("E");
    }

}
}

```

Hasil Output

Inputkan nilai angka= 60

C

b. Multifl

```

package pekan4;
import java.util.Scanner;
public class multifl {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int umur;
        char sim;
        Scanner a= new Scanner(System.in);
        System.out.print("Input umur anda: ");
        umur= a.nextInt();
        System.out.print("Apakah anda sudah punya sim C: ");
        sim=a.next().charAt(0);

        if ((umur >= 17) && (sim =='y')) {
            System.out.println("Anda sudah dewasa dan boleh bawa motor");
        }
        if ((umur >= 17) && (sim !='y')) {
            System.out.println("Anda sudah dewasa tetapi tidak boleh bawa motor");
        }
    }
}

```

```

if ((umur < 17) && (sim != 'y')) {
    System.out.println("Anda belum cukup Umur bawa motor");
}
if ((umur < 17) && (sim == 'y')) {
    System.out.println("Anda belum cukup umur punya sim");
}

}

}

}

```

Hasil Output

Input umur anda: 12

Apakah anda sudah punya sim C: belu,

Anda belum cukup Umur bawa motor

c. Latifl

```

package pekan4;
import java.util.Scanner;
public class latifl {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        double IPK;
        Scanner input= new Scanner (System.in);
        System.out.println("Input IPK Anda = ");
        IPK=input.nextDouble ();
    }
}

```



```

input.close ();
if (IPK>2.75) {
System.out.println("Anda harus lulus Sangat Memuaskan dengan IPK
"+IPK);
}

}

}

```

Hasil Output

```

Input IPK Anda =
3
Anda harus lulus Sangat Memuaskan dengan IPK 3.0

```

d. Nama Bulan

```

package pekan4;
import java.util.Scanner;
public class NamaBulan {

public static void main(String[] args) {
// TODO Auto-generated method stub
Scanner scanner = new Scanner(System.in);
System.out.print("Masukan angka bulan (1-12): ");
int bulan = scanner.nextInt();
switch (bulan) {
case 1:
System.out.println("Januari");

```

```
break;
case 2:
System.out.println("Februari");
break;
case 3:
System.out.println("Maret");
break;
case 4:
System.out.println("April");
break;
case 5:
System.out.println("May");
break;
case 6:
System.out.println("Juni");
break;
case 7:
System.out.println("Juli");
break;
case 8:
System.out.println("Agustus");
break;
case 9:
System.out.println("September");
break;
case 10:
System.out.println("Oktober");
break;
case 11:
System.out.println("November");
break;
case 12:
```

```
System.out.println("Desember");  
break;  
default:  
System.out.println("Angka tidak valid");  
scanner.close();  
  
}  
  
}  
  
}
```

Hasil Output

Masukan angka bulan (1-12): 10

Oktober

2.2 Langkah Kerja

Langkah-langkah yang dilakukan dalam praktikum ini adalah sebagai berikut:

1. A. Program Penilaian Huruf (Class: Nilai)

1. Membuka aplikasi IDE (Integrated Development Environment) seperti NetBeans atau IntelliJ IDEA.
2. Membuat paket baru bernama pekan4 dan membuat kelas Java dengan nama Nilai.
3. Menuliskan kode program untuk menerima input berupa nilai angka dari pengguna.
4. Menggunakan struktur percabangan if-else untuk menentukan nilai huruf berdasarkan rentang nilai:
 - $\geq 81 \rightarrow A$
 - $\geq 70 \rightarrow B$
 - $\geq 60 \rightarrow C$
 - $\geq 50 \rightarrow D$
 - $< 50 \rightarrow E$
5. Menjalankan program dan menguji dengan beberapa input, seperti nilai 60, untuk memastikan output sesuai ekspektasi.
6. Menutup scanner setelah input selesai untuk mencegah kebocoran sumber daya.

B. Program Validasi SIM dan Umur (Class: multiftl)

1. Membuat kelas baru bernama multiftl dalam paket yang sama.
2. Menuliskan kode untuk menerima dua input dari pengguna:
 - Umur (dalam bilangan bulat)
 - Status kepemilikan SIM C (dalam bentuk karakter, misalnya 'y' untuk ya)
3. Menggunakan logika if dengan operator logika && untuk memeriksa kombinasi umur dan status SIM.
4. Menampilkan pesan yang sesuai berdasarkan kondisi:
 - Dewasa dan punya SIM \rightarrow boleh bawa motor
 - Dewasa tapi belum punya SIM \rightarrow tidak boleh bawa motor

- Belum dewasa dan belum punya SIM → belum cukup umur
 - Belum dewasa tapi sudah punya SIM → belum cukup umur punya SIM
5. Melakukan uji coba dengan input umur 12 dan status SIM 'n' untuk melihat hasilnya.

C. Program Validasi IPK (Class: latifl)

1. Membuat kelas baru bernama latifl.
2. Menuliskan kode untuk menerima input berupa IPK (dalam tipe data double).
3. Menggunakan struktur if untuk memeriksa apakah IPK lebih dari 2.75.
4. Jika kondisi terpenuhi, program akan mencetak pesan bahwa pengguna lulus dengan predikat sangat memuaskan.
5. Melakukan uji coba dengan input IPK 3.0 untuk memastikan hasil sesuai.

D. Program Nama Bulan Berdasarkan Angka (Class: NamaBulan)

1. Membuat kelas baru bernama NamaBulan.
2. Menuliskan kode untuk menerima input berupa angka bulan (1–12).
3. Menggunakan struktur switch-case untuk mencetak nama bulan sesuai angka yang dimasukkan.
4. Menambahkan default untuk menangani input yang tidak valid (di luar rentang 1–12).
5. Melakukan uji coba dengan input angka 10 dan memastikan output adalah “Oktober”.

2.3 Analisis Hasil

A. Program Penilaian Huruf (Nilai)

Program ini menerima masukan nilai numerik dan mengonversinya menjadi huruf berdasarkan rentang nilai. Saat nilai 60 dimasukkan, keluaran “C” sesuai dengan rentang 60–69. Logika if-else bekerja dengan benar, mengeksekusi cabang yang sesuai tanpa konflik.

Struktur kontrol if-else merupakan mekanisme dasar untuk pengambilan keputusan dalam bahasa Java. Setiap kondisi dievaluasi secara berurutan hingga kondisi yang

terpenuhi. Pendekatan ini mudah dipahami dan diimplementasikan dalam sistem evaluasi akademik yang umum digunakan di perguruan tinggi.

Referensi:

- Deitel, P. J., & Deitel, H. M. (2017). *Java: How to Program*. Pearson Education.
- Nugroho, A. (2020). *Logika dan Algoritma dalam Pemrograman*. Penerbit Informatika.

B. Program Validasi Umur dan SIM (multifl)

Program memeriksa dua kondisi: umur dan status kepemilikan SIM C. Saat umur 12 dan status SIM 'n' dimasukkan, keluaran “Anda belum cukup umur bawa motor” muncul sesuai dengan logika gabungan &&.

Operator logika && memastikan kedua syarat terpenuhi sebelum blok kode dijalankan. Pemanfaatan operator ini lazim dalam sistem validasi data untuk menjamin aturan tertentu dipatuhi sebelum melanjutkan proses.

Referensi:

- Sebesta, R. W. (2016). *Concepts of Programming Languages*. Pearson.
- Sujarwo, A. (2019). *Dasar-Dasar Pemrograman Java*. Graha Ilmu.

C. Program Validasi IPK (latifl)

Program menerima input IPK bertipe double dan memeriksa apakah nilainya melebihi 2.75. Saat IPK 3.0 dimasukkan, program mencetak “Anda harus lulus Sangat Memuaskan dengan IPK 3.0”, sesuai ekspektasi.

Tipe data double memungkinkan representasi nilai desimal yang presisi, ideal untuk data akademik seperti IPK. Validasi numerik semacam ini penting untuk menentukan kelayakan atau status kelulusan mahasiswa.

Referensi:

- Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach*. McGraw-Hill.
- Wahana Komputer. (2020). *Pemrograman Java untuk Pemula*. Andi Publisher.
- Potensi penyempurnaan:

D. Program Nama Bulan Berdasarkan Angka (NamaBulan)

Program ini menggunakan switch-case untuk mencetak nama bulan dari angka 1–12. Input 10 menghasilkan keluaran “Oktober”, sesuai dengan implementasi.

Struktur switch-case efisien untuk menangani banyak kondisi diskrit. Dibandingkan rangkaian if-else, switch meningkatkan keterbacaan dan memudahkan penambahan kasus baru.

Referensi:

- E. Savitch, W. (2018). *Absolute Java*. Pearson.
- F. Siregar, R. (2021). *Pemrograman Java Tingkat Dasar*. Deepublish.

BAB III

PENUTUP

3.1 Kesimpulan

Program-program praktikum pekan 4 telah berhasil menerapkan struktur kontrol dasar pada bahasa pemrograman Java. Setiap program mampu memproses input sesuai logika yang dirancang dan menghasilkan keluaran yang tepat, mulai dari konversi nilai huruf, validasi umur dan kepemilikan SIM, pengecekan IPK, hingga pemetaan angka ke nama bulan. Penggunaan tipe data int, double, dan char sudah sesuai kebutuhan, begitu pula operator logika dan struktur switch-case. Meski demikian, masih ada ruang untuk meningkatkan keandalan dan keterbacaan kode agar siap menghadapi berbagai kondisi input.

3.2 Saran

1. . Menambahkan validasi input agar program menolak nilai di luar rentang yang valid dan mencegah kesalahan saat runtime.
2. Menggunakan struktur if-else if-else atau mengkombinasikan switch-case dengan grouping kasus untuk menyederhanakan alur eksekusi.
3. Memecah logika utama ke dalam beberapa metode kecil untuk memudahkan pemeliharaan, pengujian, dan dokumentasi.
4. Menutup objek Scanner atau sumber daya lain di blok finally atau setelah seluruh proses selesai agar tidak terjadi kebocoran memori.
5. Menyertakan penanganan kesalahan (error handling) berupa pesan yang jelas ketika input tidak sesuai format atau jenis data.
6. Menerapkan String.format atau printf untuk mengatur tampilan angka desimal dan teks, sehingga output menjadi lebih konsisten dan rapi.

DAFTAR PUSTAKA

Deitel, P. J., & Deitel, H. M. (2017). *Java: How to Program*. Pearson Education.

Nugroho, A. (2020). *Logika dan Algoritma dalam Pemrograman*. Penerbit Informatika.

Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach*. McGraw-Hill.

Savitch, W. (2018). *Absolute Java*. Pearson.

Sebesta, R. W. (2016). *Concepts of Programming Languages*. Pearson.

Siregar, R. (2021). *Pemrograman Java Tingkat Dasar*. Deepublish.

Sujarwo, A. (2019). *Dasar-Dasar Pemrograman Java*. Graha Ilmu.

Wahana Komputer. (2020). *Pemrograman Java untuk Pemula*. Andi Publisher.