

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN
LAPORAN PRAKTIKUM PEKAN 3

Disusun Oleh:

Syasya Halwa Gazwani

(2511531018)

Dosen Pengampu:

Dr. Wahyudi, S.T, M.T

Asisten Praktikum:

Muhammad Zaki Al Hafiz



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
PADANG
2025

KATA PENGANTAR

Assalamu'alaikum warahmatullahi wabarakatuh,

Segala puji bagi Allah SWT yang telah memberikan kesempatan serta kemudahan kepada penulis untuk dapat menyelesaikan Laporan Praktikum pada mata kuliah Algoritma Pemrograman, sehingga Laporan Praktikum ini dapat dikumpulkan dengan tepat waktu. Atas rahmat dan karunianya Laporan Praktikum dapat terselesaikan dengan baik. Sholawat serta salam juga penulis sampaikan kepada baginda tercinta yaitu Nabi Muhammad SAW. yang kita nantikan syafa'atnya di akhirat nanti. Laporan Praktikum ini bertujuan untuk menambah wawasan para pembaca untuk lebih memperdalam ilmu yang ada pada makalah ini.

Dalam penyusunan Laporan Praktikum ini, penulis mengalami banyak kesulitan dan penulis menyadari bahwa Laporan Praktikum ini masih jauh dari kesempurnaan. Untuk itu, penulis sangat mengharapkan kritik dan saran yang membangun demi kesempurnaan pada Laporan Praktikum ini. Penulis mengucapkan terima kasih kepada bapak Dr. Wahyudi, S.T, M.T. selaku dosen mata kuliah Algoritma Pemrograman yang telah memberikan tugas ini sehingga dapat menambah pengetahuan dan wawasan sesuai dengan mata kuliah yang penulis tekuni. Penulis juga mengucapkan terima kasih kepada semua pihak yang telah membagi sebagian pengetahuannya sehingga penulis dapat menyelesaikan Laporan Praktikum ini.

Padang, September 2025

Penulis

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI	ii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan	1
BAB II PEMBAHASAN	3
2.1 LatIf1	3
2.1.1 Kode Program	3
2.1.2 Langkah Kerja	3
2.1.3 Analisis Hasil	4
2.2 Ifelse1	5
2.2.1 Kode Program	5
2.2.2 Langkah Kerja	5
2.2.3 Analisis Hasil	6
2.3 MultiIf	7
2.3.1 Kode Program	7
2.3.2 Langkah Kerja	7
2.3.3 Analisis Hasil	8
2.4 Nilai	9
2.4.1 Kode Program	9
2.4.2 Langkah Kerja	9
2.4.3 Analisis Hasil	10
2.5 Nama Bulan	11
2.5.1 Kode Program	11

2.5.2	Langkah Kerja	12
2.5.3	Analisis Hasil	13
KESIMPULAN.....		14
DAFTAR PUSTAKA.....		15

BAB I PENDAHULUAN

1.1 Latar Belakang

Pemrograman atau *programming* merupakan sebuah proses menulis, menguji dan memperbaiki (*debug*), serta memelihara kode yang dapat membangun suatu program komputer. Kode ini ditulis dalam berbagai bahasa pemrograman atau sering disebut juga bahasa komputer. Tujuan dari pemrograman adalah untuk memuat suatu program yang dapat melakukan suatu perhitungan atau “pekerjaan” sesuai dengan keinginan pemrograman. Untuk melakukan pemrograman, diperlukan keterampilan dalam algoritma, logika, bahasa pemrograman, dan pada banyak kasus, pengetahuan-pengetahuan lain seperti matematika. Bahasa Pemrograman, atau biasanya disebut bahasa komputer atau *computer language programming*, yaitu sebuah instruksi standar untuk mengendalikan sebuah komputer. Seperangkat aturan *syntax* dan *semantic* yang digunakan untuk memberikan sebuah definisi pada program komputer dikenal sebagai bahasa pemrograman. Dengan menggunakan bahasa ini, seorang *programmer* dapat dengan tepat menentukan data yang akan diproses oleh komputer, bagaimana penyimpanan dan transfernya, dan tindakan apa yang harus diambil dalam berbagai keadaan.

1.2 Tujuan

- 1.2.1 Memahami konsep dasar tipe data dalam bahasa pemrograman Java serta penerapannya dalam pembuatan program sederhana.
- 1.2.2 Mengembangkan keterampilan analisis terhadap hasil keluaran program sesuai dengan logika yang diterapkan.

1.3 Manfaat Praktikum

- 1.3.1 Menambah wawasan dan keterampilan mahasiswa dalam mengelola data menggunakan bahasa pemrograman Java sebagai dasar untuk pemrograman yang lebih lanjut.

- 1.3.2 Memberikan pemahaman dasar kepada mahasiswa mengenai penggunaan tipe data dalam bahasa pemrograman Java, sehingga dapat menjadi bekal dalam mengembangkan program yang lebih kompleks di tahap berikutnya.

BAB II PEMBAHASAN

2.1 LatIf1

2.1.1 Kode Program

```

1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class latIf1 {
6     public static void main(String[] args) {
7         double IPK;
8         Scanner input=new Scanner(System.in);
9         System.out.print("Input IPK Anda = ");
10        IPK=input.nextDouble();
11        input.close();
12        if (IPK>2.75) {
13            System.out.println("Anda Lulus Sangat Memuaskan dengan IPK "+IPK);
14        }
15    }
16 }
17 }

```

Gambar 2.1.1 Kode Program latIf1.

2.1.2 Langkah Kerja

2.1.2.1 Menyatakan bahwa *class* ini berada pada package pekan4.

```
1 package pekan4;
```

Gambar 2.1.2.1 Menyimpan *class* dalam package pekan4.

2.1.2.2 Mengimpor *class Scanner* untuk membaca input.

```
3 import java.util.Scanner;
```

Gambar 2.1.2.2 Mengimpor *class Scanner*.

2.1.2.3 Deklarasi *class publik* yang dimana nama kelas harus sama dengan nama file. Mulai memasuki titik program.

```
5 public class latIf1 {
6     public static void main(String[] args) {
```

Gambar 2.1.2.3 Deklarasi *class publik*.

2.1.2.4 Mendeklarasikan variabel IPK bertipe *double* untuk menyimpan nilai IPK yang di input.

```
7         double IPK;
```

Gambar 2.1.2.4 Deklarasi variabel IPK.

2.1.2.5 Membuat objek *Scanner* bernama input yang membaca dari keyboard.

```
8 Scanner input=new Scanner(System.in);
```

Gambar 2.1.2.5 Mmembuat objek *Scanner*.

2.1.2.6 Menampilkan *prompt* ke layar.

```
9 System.out.print("Input IPK Anda = ");
```

Gambar 2.1.2.6 Menampilkan *prompt*.

2.1.2.7 Membaca nilai pecahan (*double*) yang diketik *user* dan menyimpannya ke IPK. Lalu menutup *Scanner*.

```
10 IPK=input.nextDouble();
11 input.close();
```

Gambar 2.1.2.7 Membaca nilai *double* dan menutup *Scanner*.

2.1.2.8 Memastikan nilai IPK lebih besar samadengan 2,75. Jika nilai benar, nilai dicetak ke layar.

```
12 if (IPK>2.75) {
    System.out.println("Anda Lulus Sangat Memuaskan dengan IPK "+IPK);
}
```

Gambar 2.1.2.8 Memastikan nilai dan mencetak ke layar.

2.1.3 Analisis Hasil

```
Input IPK Anda = 3,25
Anda Lulus Sangat Mmuaskan dengan IPK 3.25
```

Gambar 2.1.3 Hasil Kode Program latIf1.

Gambar diatas merupakan hasil Kode Program latIf1 yang telah ditampilkan. Program ini menghasilkan output yang berbeda tergantung pada nilai IPK yang dimasukkan. Jika pengguna memasukkan IPK lebih besar atau samadengan 2,75, maka kondisi pada pernyataan if terpenuhi dan program akan menampilkan pesan “Ada Lulus Sangat Memuaskan dengan IPK...” sesuai dengan angka yang dimasukkan.

Namun, jika nilai IPK yang dimasukkan kurang dari 2,75, kondisi if tidak terpenuhi sehingga program langsung berhenti tanpa memberikan pesan apapun, yang membuat pengguna tidak mendapatkan informasi lebih lanjut.

2.2 Ifelse1

2.2.1 Kode Program

```

1 package pekan4;
2
3 import java.util.Scanner;
4 public class Ifelse1 {
5     public static void main(String[] args) {
6         double IPK;
7         Scanner input=new Scanner(System.in);
8         System.out.print("Input IPK Anda = ");
9         IPK=input.nextDouble();
10        input.close();
11        if (IPK>2.75) {
12            System.out.println("Anda Lulus Sangat Memuaskan dengan IPK "+IPK);
13        } else {
14            System.out.println("Anda Tidak Lulus");
15        }
16    }
17 }
18 }

```

Gambar 2.1.1 Kode Program ifelse1.

2.2.2 Langkah Kerja

2.2.2.1 Variabel IPK bertipe *double* dibuat untuk menyimpan nilai IPK yang dimasukkan.

```

6         double IPK;

```

Gambar 2.2.2.1 Membuat variabel bertipe *double*.

2.2.2.2 Objek *Scanner* bernama input dibuat untuk membaca data dari System.in. Lalu program menampilkan teks “Input IPK Anda =” di layar untuk meminta pengguna memasukkan nilai IPK.

```

7         Scanner input=new Scanner(System.in);
8         System.out.print("Input IPK Anda = ");

```

Gambar 2.2.2.2 Membuat objek *Scanner* dan menampilkan teks.

2.2.2.3 Pengguna mengetikkan angka, lalu nilainya disimpan ke variabel `IPK` melalui `input.nextDouble()`. Objek *Scanner* ditutup.

```
9          IPK=input.nextDouble();
10         input.close();
```

Gambar 2.2.2.3 Input angka dan menutup objek *Scanner*.

2.2.2.4 Program melakukan percabangan menggunakan *if else*.

```
11         if (IPK>2.75) {
12             System.out.println("Anda Lulus Sangat Memuaskan dengan IPK "+IPK);
13         } else {
```

Gambar 2.2.2.4 Menggunakan *if else*.

2.2.2.5 Menampilkan “Anda Tidak Lulus” jika nilai `IPK` kurang dari 2,75.

```
14         System.out.println("Anda Tidak Lulus");
```

Gambar 2.2.2.5 Menampilkan “Anda Tidak Lulus”.

2.2.3 Analisis Hasil

```
Input IPK Anda = 2,30
Anda Tidak Lulus
```

```
Input IPK Anda = 3,40
Anda Lulus Sangat Memuaskan dengan IPK 3.4
```

Gambar 2.2.3 Hasil Kode Program `ifelse1`.

Gambar diatas merupakan hasil output dari Kode Program `ifelse1` yang telah ditampilkan. Program ini akan menghasilkan output yang berbeda tergantung pada nilai `IPK` yang dimasukkan pengguna. Jika nilai `IPK` yang dimasukkan lebih besar atau sama dengan 2.75, maka kondisi pada percabangan `if` akan bernilai benar sehingga program mencetak pesan “Anda Lulus Sangat Memuaskan dengan `IPK` ...” dengan menampilkan nilai `IPK` yang dimasukkan. Sebaliknya, jika nilai `IPK` kurang dari 2.75, kondisi `if` tidak terpenuhi dan program akan menjalankan blok `else` yang menampilkan pesan “Anda Tidak Lulus”. Dengan begitu, program ini selalu memberikan hasil, baik ketika `IPK` memenuhi syarat kelulusan maupun ketika tidak.

2.3 MultiIf

2.3.1 Kode Program

```

1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class multiIf {
6     public static void main(String[] args) {
7         int umur;
8         char sim;
9         Scanner a= new Scanner(System.in);
10        System.out.print("Input umur anda: ");
11        umur= a.nextInt();
12        System.out.print("Apakah Anda Sudah Punya Sim C: ");
13        sim=a.next().charAt(0);
14        a.close();
15        if((umur >= 17)&&(sim=='y')) {
16            System.out.println("Anda Sudah dewasa dan boleh bawa motor");
17        }
18        if((umur >= 17)&&(sim!='y')) {
19            System.out.println("Anda Sudah dewasa tetapi tidak boleh bawa motor");
20        }
21        if((umur < 17)&&(sim=='y')) {
22            System.out.println("Anda Belum Cukup Umur bawa motor");
23        }
24        if((umur < 17)&&(sim=='y')) {
25            System.out.println("Anda Belum Cukup Umur punya SIM");
26        }
27    }
28 }

```

Gambar 2.3.1 Kode Program multiIf.

2.3.2 Langkah Kerja

2.3.2.1 Mendeklarasi dua variabel, umur bertipe int dan sim bertipe char.

```

7         int umur;
8         char sim;

```

Gambar 2.3.2.1 Mendeklarasi dua variabel.

2.3.2.2 Membuat objek *Scanner* untuk membaca input dari pengguna, pengguna mengetik angka yang disimpan di variabel umur, program meminta input lagi berupa jawaban, dan *Scanner* ditutup.

```

9         Scanner a= new Scanner(System.in);
10        System.out.print("Input umur anda: ");
11        umur= a.nextInt();
12        System.out.print("Apakah Anda Sudah Punya Sim C: ");
13        sim=a.next().charAt(0);
14        a.close();

```

Gambar 2.3.2.2 Membuat objek *Scanner* dan menutup *Scanner*.

2.3.2.3 Program masuk ke percabangan if untuk memeriksa kondisi. Setelah salah satu kondisi terpenuhi dan pesan ditampilkan, program selesai dijalankan.

```

15●      if((umur >= 17)&&(sim=='y')) {
16          System.out.println("Anda Sudah dewasa dan boleh bawa motor");
17      }
18●      if((umur >= 17)&&(sim!='y')) {
19          System.out.println("Anda Sudah dewasa tetapi tidak boleh bawa motor");
20      }
21●      if((umur < 17)&&(sim=='y')) {
22          System.out.println("Anda Belum Cukup Umur bawa motor");
23      }
24●      if((umur < 17)&&(sim!='y')) {
25          System.out.println("Anda Belum Cukup Umur punya SIM");

```

Gambar 2.3.2.3 Program memeriksa kondisi.

2.3.3 Analisis Hasil

```

Input umur anda: 17
Apakah Anda Sudah Punya Sim C: belum
Anda Sudah dewasa tetapi tidak boleh bawa motor

```

Gambar 2.3.3 Hasil Kode Program multiIf.

Gambar diatas merupakan hasil Kode Program dari multiIf yang telah dihasilkan. Program ini menghasilkan output yang berbeda sesuai dengan kombinasi nilai umur dan status kepemilikan SIM yang dimasukkan pengguna. Jika umur 17 tahun atau lebih dan jawaban SIM adalah y, program menampilkan pesan bahwa pengguna sudah dewasa dan boleh membawa motor. Jika umur 17 tahun atau lebih tetapi jawaban SIM bukan y, artinya pengguna belum punya SIM, maka pesan yang muncul adalah sudah dewasa tetapi tidak boleh membawa motor. Apabila umur kurang dari 17 tahun namun jawaban SIM adalah y, program menampilkan pesan bahwa pengguna belum cukup umur untuk membawa motor meskipun sudah memiliki SIM. Sedangkan jika umur kurang dari 17 tahun dan jawaban SIM bukan y, program memberikan pesan bahwa pengguna belum cukup umur untuk memiliki SIM.

2.4 Nilai

2.4.1 Kode Program

```

1 package pekan4;
2 import java.util.Scanner;
3 public class Nilai {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int nilai;
8         Scanner input=new Scanner(System.in);
9         System.out.print("Inputkan nilai angka= ");
10        nilai=input.nextInt();
11        input.close();
12
13        if(nilai >=81) {
14            System.out.println("A");
15        } else if (nilai >=70) {
16            System.out.println("B");
17        } else if (nilai >=60) {
18            System.out.println("C");
19        } else if (nilai >=50) {
20            System.out.println("D");
21        } else {
22            System.out.println("E");
23        }
24    }
25
26 }

```

Gambar 2.4.1 Kode Program Nilai.

2.4.2 Langkah Kerja

2.4.2.1 Deklarasi variabel nilai bertipe int untuk menampung angka yang dimasukkan pengguna.

```

7         int nilai;

```

Gambar 2.4.2.1 Deklarasi variabel.

2.4.2.2 Objek *Scanner* dibuat untuk membaca data dari *System.in*, program menampilkan pesan “Inputkan nilai angka=”, nilai yang diketik pengguna disimpan kedalam variabel nilai menggunakan *input.nextInt()*, dan *Scanner* ditutup.

```

8         Scanner input=new Scanner(System.in);
9         System.out.print("Inputkan nilai angka= ");
10        nilai=input.nextInt();
11        input.close();

```

Gambar 2.4.2.2 Membuat objek *Scanner* dan menutup *Scanner*.

2.4.2.3 Program kemudian memeriksa nilai dengan menggunakan struktur percabangan *if else*. Setelah salah satu kondisi yang terpenuhi, program mencetak huruf.

```
13-      if(nilai >=81) {  
14-          System.out.println("A");  
15-      } else if (nilai >=70) {  
16-          System.out.println("B");  
17-      } else if (nilai >=60) {  
18-          System.out.println("C");  
19-      } else if (nilai >=50) {  
20-          System.out.println("D");  
21-      }else {  
22-          System.out.println("E");
```

Gambar 2.3.2.3 Memeriksa nilai.

2.4.3 Analisis Hasil

```
Inputkan nilai angka= 64  
C
```

Gambar 2.4.3 Hasil Kode Program Nilai.

Gambar diatas merupakan hasil Kode Program Nilai yang telah ditampilkan. Program ini akan menghasilkan huruf tertentu sesuai dengan nilai angka yang dimasukkan pengguna. Jika pengguna menginput angka 81 atau lebih, maka program menampilkan huruf A sebagai hasil. Jika angka berada di antara 70 sampai 80, maka yang ditampilkan adalah huruf B. Jika angka berada di antara 60 sampai 69, hasilnya adalah huruf C. Jika angka di antara 50 sampai 59, program akan menampilkan huruf D. Sedangkan jika angka kurang dari 50, program menampilkan huruf E. Dengan demikian, hasil keluaran program sepenuhnya ditentukan oleh rentang nilai yang dimasukkan pengguna, dan setiap kemungkinan nilai sudah memiliki kategori yang jelas sehingga program selalu memberikan hasil berupa nilai huruf tanpa kondisi kosong.

2.5 Nama Bulan

2.5.1 Kode Program

```
1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class NamaBulan {
6     public static void main(String[] args) {
7         Scanner input=new Scanner(System.in);
8         System.out.print("Masukkan angka bulan (1 12): ");
9         int bulan = input.nextInt();
10        switch (bulan) {
11            case 1:
12                System.out.println("Januari");
13                break;
14            case 2:
15                System.out.println("Februari");
16                break;
17            case 3:
18                System.out.println("Maret");
19                break;
20            case 4:
21                System.out.println("April");
22                break;
23            case 5:
24                System.out.println("Mei");
25                break;
26            case 6:
27                System.out.println("Juni");
28                break;
29            case 7:
30                System.out.println("Juli");
31                break;
32            case 8:
33                System.out.println("Agustus");
34                break;
35            case 9:
36                System.out.println("September");
37                break;
38            case 10:
39                System.out.println("Oktober");
40                break;
41            case 11:
42                System.out.println("November");
43                break;
44            case 12:
45                System.out.println("Desember");
46                break;
47            default:
48                System.out.println("Angka tidak valid");
49        }
50        input.close();
51    }
52 }
```

Gambar 2.5.1 Kode Program Nama Bulan.

2.5.2 Langkah Kerja

- 2.5.2.1 Membuat objek *Scanner* untuk menerima input dari *keyboard*, menampilkan *prompt*, dan menyimpan input ke variabel bulan.

```

7      Scanner input=new Scanner(System.in);
8      System.out.print("Masukkan angka bulan (1 12): ");
9      int bulan = input.nextInt();
10     switch (bulan) {

```

Gambar 2.5.2.1 Membuat objek *Scanner*.

- 2.5.2.2 Mengevaluasi nilai dari variabel bulan.

```

11     case 1:
12         System.out.println("Januari");
13         break;
14     case 2:
15         System.out.println("Februari");
16         break;
17     case 3:
18         System.out.println("Maret");
19         break;
20     case 4:
21         System.out.println("April");
22         break;
23     case 5:
24         System.out.println("Mei");
25         break;
26     case 6:
27         System.out.println("Juni");
28         break;
29     case 7:
30         System.out.println("Juli");
31         break;
32     case 8:
33         System.out.println("Agustus");
34         break;
35     case 9:
36         System.out.println("September");
37         break;
38     case 10:
39         System.out.println("Oktober");
40         break;
41     case 11:
42         System.out.println("November");
43         break;
44     case 12:
45         System.out.println("Desember");
46         break;

```

Gambar 2.5.2.2 Struktur *Switch-Case*.

- 2.5.2.3 Mencetak hasil nilai bulan.

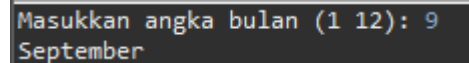
```

47     default:
48         System.out.println("Angka tidak valid");
49     }
50     input.close();

```

Gambar 2.5.2.3 Mencetak hasil nilai bulan.

2.5.3 Analisis Hasil



```
Masukkan angka bulan (1 12): 9
September
```

Gambar 2.5.3 Hasil Kode Program Nama Bulan.

Program tersebut merupakan aplikasi Java sederhana yang berfungsi untuk menampilkan nama bulan berdasarkan input angka dari pengguna. Program dimulai dengan mendefinisikan *package* dan mengimpor *library Scanner* yang digunakan untuk membaca input dari *keyboard*. Di dalam *method main*, program meminta pengguna memasukkan angka antara 1 hingga 12. Input tersebut kemudian disimpan dalam variabel bulan. Setelah mendapatkan input, program menggunakan struktur kontrol *switch* untuk mencocokkan nilai input dengan *case-case* yang tersedia. Setiap *case* mewakili satu bulan dalam setahun, mulai dari Januari untuk case 1 hingga Desember untuk case 12.

Jika pengguna memasukkan angka yang berada di luar rentang 1 sampai 12, maka program akan mengeksekusi blok *default* yang mencetak pesan bahwa angka tidak valid. Program ini menampilkan hasil yang sesuai jika input valid diberikan, misalnya jika pengguna memasukkan angka 3 maka akan muncul output "Maret". Namun jika input seperti 15 dimasukkan, maka akan muncul pesan "Angka tidak valid". Program ini juga menutup objek *Scanner* di akhir eksekusi untuk mencegah kebocoran *resource*. Secara keseluruhan, program berjalan dengan baik dan memiliki logika yang jelas untuk memetakan angka ke nama bulan. Namun, struktur *switch* yang panjang bisa disederhanakan menggunakan *array* atau struktur data lain jika ingin efisiensi dan kejelasan yang lebih baik dalam penulisan kode.

KESIMPULAN

Program tersebut dirancang untuk menerima input dari pengguna, kemudian mengolahnya melalui struktur percabangan, dan akhirnya memberikan keluaran yang sesuai dengan kondisi yang ditentukan. Pada program pertama, input berupa IPK digunakan untuk menentukan apakah seseorang lulus atau tidak dengan kriteria minimal 2.75. Pada program kedua, input berupa umur dan status kepemilikan SIM digunakan untuk menentukan kelayakan seseorang dalam membawa motor sesuai aturan yang berlaku. Sedangkan pada program terakhir, input berupa nilai angka dikonversikan menjadi nilai huruf sesuai dengan rentang tertentu. Secara umum, semua program ini memperlihatkan bagaimana *penggunaan class, method main*, input dengan *Scanner*, variabel, serta percabangan *if-else* dapat bekerja sama untuk menghasilkan keputusan yang berbeda berdasarkan data yang dimasukkan pengguna. Hal ini menunjukkan bahwa konsep dasar pemrograman seperti input, proses, dan output dapat diterapkan secara praktis untuk menyelesaikan kasus nyata.

DAFTAR PUSTAKA

- [1] M. Musfika, I. Akbar, S. F. Dewi, dan A. A. Aziz, "Bahasa Pemrograman Java Berbasis Exe-Learning," *Jurnal Ilmiah Sistem Informasi, Teknologi Informasi dan Sistem Komputer*, vol. 18, no. 1, p. 2, 2023.
- [2] W. Gata dan g. Gata, "Penerapan Bahasa Pemrograman Java dalam Sistem Informasi Penjualan Versi Desktop," *Jurnal* (tidak disebutkan nama jurnal), vol. 10, no. 1, p. 81, 2013.