

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN
CONDITIONAL STATEMENT



Disusun Oleh:

Haliya Isma Husna Putri Ahmadi
2511532002

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

Asisten Praktikum:
Muhammad Zaky Al Hafiz

DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS

2025

KATA PENGANTAR

Puji dan Syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena atas Rahmat dan karunia-Nya, penulis dapat menyelesaikan lapora praktikum Algoritma dan Pemrograman pekan 4 dengan judul “*Conditional Statement*” tepat pada waktunya.

Penyusunan laporan ini bertujuan untuk memenuhi salah satu tugas dalam mata kuliah Praktikum Algoritma dan Pemrograman sekaligus sebagai bentuk pembelajaran bagi penulis mengenai penggunaan berbagai *conditionak formatting*, yang meliputi *if*, *if-else*, *if-else-if*, *multi-if* dan *switch-case*. Serta memahami bagaimana mengimplementasikannya dalam bentuk program sederhana.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, penulis sangat terbuka terhadap kritik dan saran yang membangun demi perbaikan di masa yang akan datang.

Ucapan terima kasih yang sebesar-besarnya penulis sampaikan kepada dosen pengampu, asisten praktikum, serta semua pihak yang terlibat dalam pelaksanaan praktikum dan penyusunan laporan ini. Semoga laporan ini dapat memberikan manfaat dan menambah wawasan bagi penulis maupun pembaca.

Padang, 29 September 2025

Penulis

DAFTAR ISI

KATA PENGANTAR.....	ii
DAFTAR ISI	iii
BAB I PENDAHULUAN.....	4
1.1 Latar Belakang	4
1.2 Tujuan	4
1.3 Manfaat	4
BAB II PEMBAHASAN	5
2.1 Dasar Teori.....	5
2.1.1 If.....	5
2.1.2 If-Else	5
2.1.3 If-Else-If	5
2.1.4 Multi-If.....	5
2.1.5 Switch-Case.....	5
2.2 Langkah Kerja	6
2.2.1 Pembuatan <i>Package</i> dan <i>Class</i>	6
2.2.2 Program Pertama: <i>latIf</i>	9
2.2.3 Program Kedua: <i>Ifelse1</i>	10
2.2.4 Program Ketiga: <i>Multi-If</i>	12
2.2.5 Program Keempat: <i>NamaBulan</i>	14
2.2.6 Program Kelima: <i>Nilai</i>	16
2.2.7 <i>Upload</i> ke <i>GitHub</i>	18
BAB III KESIMPULAN.....	21
3.1 Kesimpulan	21
3.2 Saran	21
DAFTAR PUSTAKA.....	22

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam pemrograman, terdapat banyak situasi di mana program perlu mengambil keputusan berdasarkan kondisi tertentu. Proses pengambilan keputusan ini disebut dengan struktur kendali percabangan (*conditional formatting*). Dengan adanya percabangan, sebuah program dapat menyelesaikan instruksi yang berbeda tergantung pada *input* atau *case* yang diberikan. Dengan ini program dapat lebih fleksibel dan sesuai dengan kebutuhan *user*.

Penerapan percabangan ini sangat penting dalam pengembangan aplikasi, untuk penyelesaian berbagai masalah dalam kehidupan sehari-hari. Seperti menentukan hasil kelulusan berdasarkan IPK, menampilkan nama bulan, mengonversi nilai angka ke dalam huruf, menguji kelayakan seorang pengemudi, dan lainnya.

1.2 Tujuan

Tujuan dari pelaksanaan praktikum ini adalah sebagai berikut:

- 1.2.1 Mempelajari dan memahami konsep struktur kendali percabangan (*conditional formatting*) dalam pemrograman *Java*.
- 1.2.2 Mampu mengimplementasikan percabangan sederhana seperti *if*, *if-else*, *if-else-if*, *multi-if* dan *switch-case* untuk suatu kondisi.
- 1.2.3 Membangun logika pemrograman untuk menyelesaikan suatu masalah dengan memanfaatkan *conditional formatting* di *Java*.

1.3 Manfaat

Manfaat yang diharapkan dari pelaksanaan praktikum ini antara lain:

- 1.3.1 Memberikan pemahaman mengenai program dapat mengambil keputusan secara otomatis berdasarkan kondisi tertentu.
- 1.3.2 Membantu mahasiswa memahami penerapan *conditional formatting* dalam kasus nyata.

BAB II

PEMBAHASAN

2.1 Dasar Teori

Dalam pemrograman, sering diperlukan pengambilan keputusan untuk menentukan jalur program. Proses pengambilan keputusan ini disebut dengan struktur kendali percabangan (*conditional statement*). Bahasa *Java* menyediakan beberapa *conditional statement* seperti *if*, *if-else*, *if-else-if*, *multi-if*, serta *switch-case*. Dengan ini, program dapat mengeksekusi instruksi yang berbeda tergantung dengan kondisi yang diberikan.

2.1.1 If

Struktur *If* digunakan untuk mengeksekusi suatu perintah apabila kondisi yang diberikan bernilai benar (*true*). Jika kondisi bernilai salah (*false*), maka perintah di dalam blok *if* akan dilewati.

2.1.2 If-Else

Syntax If-Else digunakan apabila terdapat dua kemungkinan kondisi. Jika kondisi bernilai benar (*true*), maka blok *if* dijalankan. Namun jika bernilai salah (*false*), maka blok *else* yang dijalankan.

2.1.3 If-Else-If

If-Else-If digunakan ketika terdapat lebih dari dua kondisi yang harus diperiksa. Program akan mengevaluasi kondisi dari atas ke bawah, dan akan mengeksekusi blok pertama yang bernilai benar (*true*).

2.1.4 Multi-If

Struktur *multi-If* mirip dengan *if-else-if*, hanya saja setiap kondisi dievaluasi secara terpisah. Artinya, lebih dari satu blok perintah dapat dijalankan jika kondisi yang diuji bernilai benar.

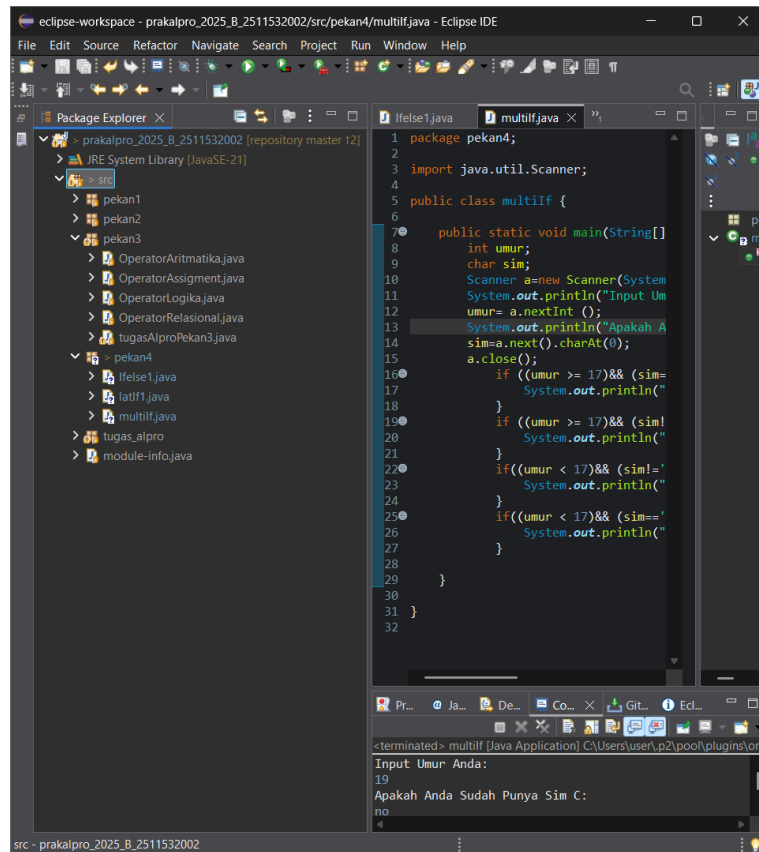
2.1.5 Switch-Case

Struktur *switch-case* digunakan ketika ada banyak kondisi yang dibandingkan dengan satu variabel. *Switch* lebih rapi digunakan dibandingkan *if-else-if* untuk pilihan yang banyak

2.2 Langkah Kerja

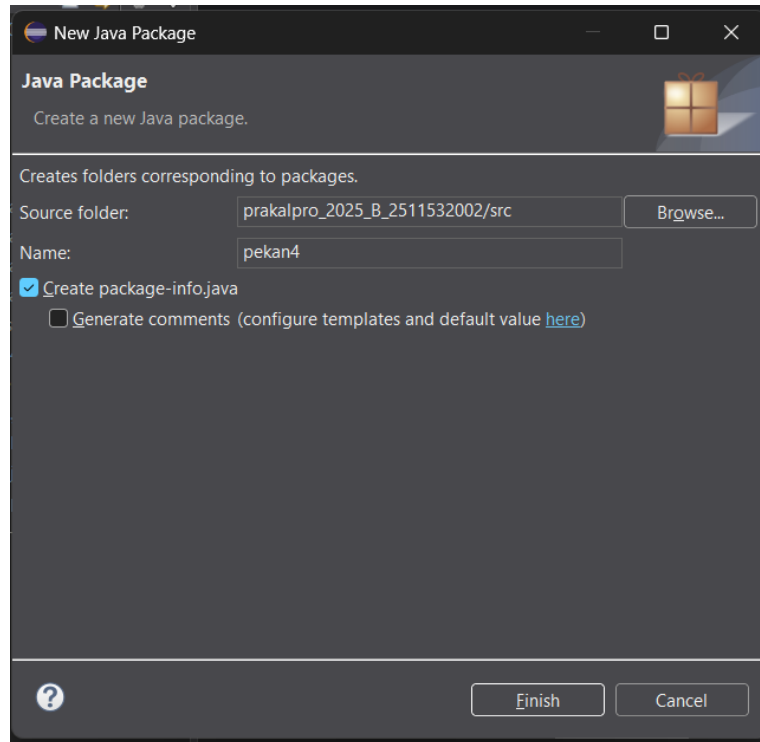
2.2.1 Pembuatan *Package* dan *Class*

1. Sebelum membuat suatu program *user* perlu untuk membuat *package* terlebih dahulu. Caranya dengan klik kanan pada *src* lalu pilih “New” dan klik “Package”.



2. 1 Pembuatan *Package* Pekan-4

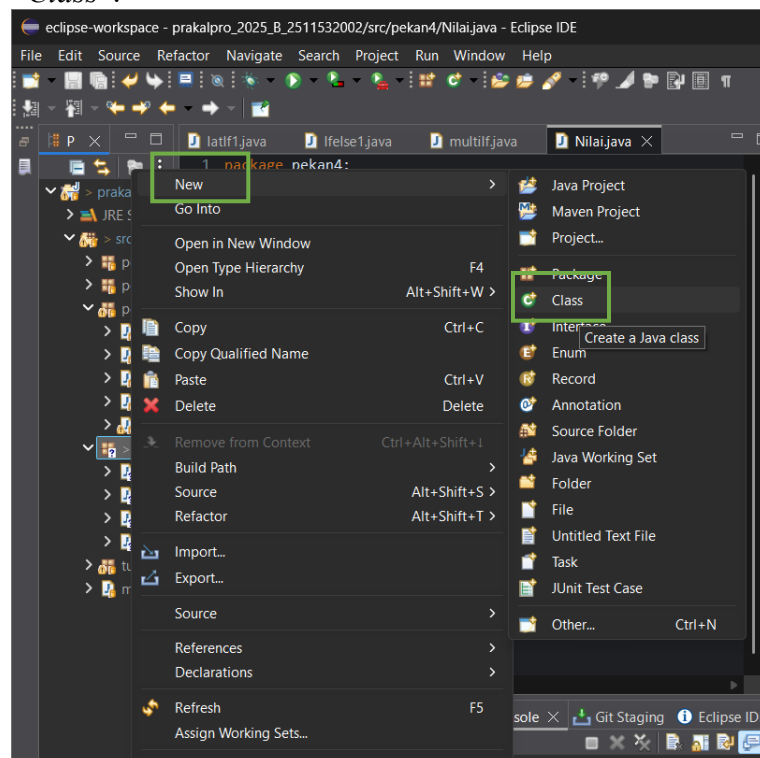
2. Setelah itu akan muncul menu “Java Package”, buat nama *package* dengan ketentuan tanpa *capslock*, *space*, dan karakter khusus lainnya. Lalu “finnish”.



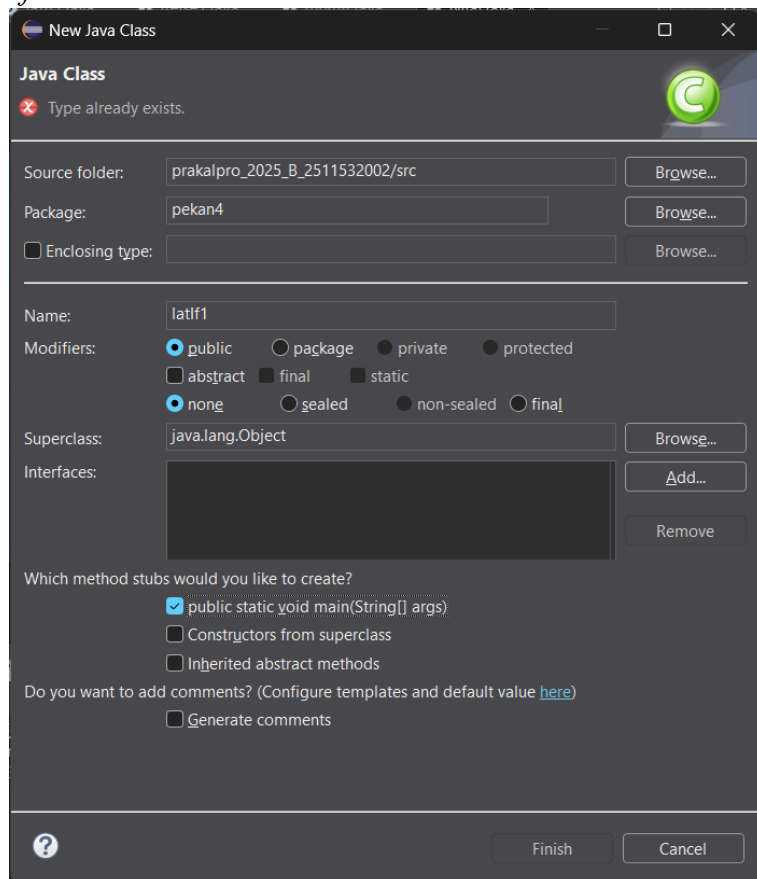
2. 2 Pembuatan Package Pekan-4

Maka *user* telah berhasil membuat *package*.

- Selanjutnya adalah membuat Membuat *class*, pada *package* yang telah dibuat tapi klik kanan lalu pilih “New” dan pilih “Class”.



4. Setelah itu pada menu “Java Class” buat nama *class*, dengan ketentuan nama harus *capslock* diawal kalimat dan tanpa spasi. Lalu pilih “Public static void main (String[] grs)” agar tidak perlu membuat *methode main()* secara manual. Selanjutnya “finish”.



2.2.2 Program Pertama: latIf

Pada *class* ini mengenai penerapan *if* sederhana. Dengan tujuan mengeksekusi sebuah blok perintah hanya jika kondisi tertentu bernilai *true*.

1. Pada *class* yang sudah dibuat, masukan *syntax* seperti berikut.

```

1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class latIf1 {
6
7     public static void main(String[] args) {
8         double IPK;
9         Scanner input=new Scanner(System.in);
10        System.out.println("Input IPK Anda = ");
11        IPK=input.nextDouble ();
12        input.close ();
13        if (IPK>2.75) {
14            System.out.println("Anda Lulus Sangat Memuaskan dengan IPK "+IPK);
15        }
16    }
17 }
18
19 }

```

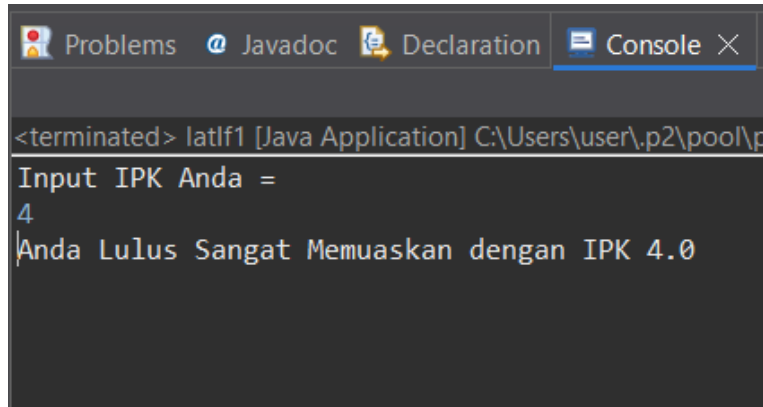
2. 3 Syntax Program LatIf

2. Setelah itu jalan kan program dengan klik tombol *Run* di pojok kiri atas pada *toolbar*.

2. 4 Running Program LatIf1

Maka pada menu *console* akan muncul hasil program seperti berikut, disini *user* dapat mencoba program dengan memasukan nilai, disini saya mencoba memasukan nilai IPK 4.

3. Maka nantinya akan menampilkan hasil program seperti berikut.

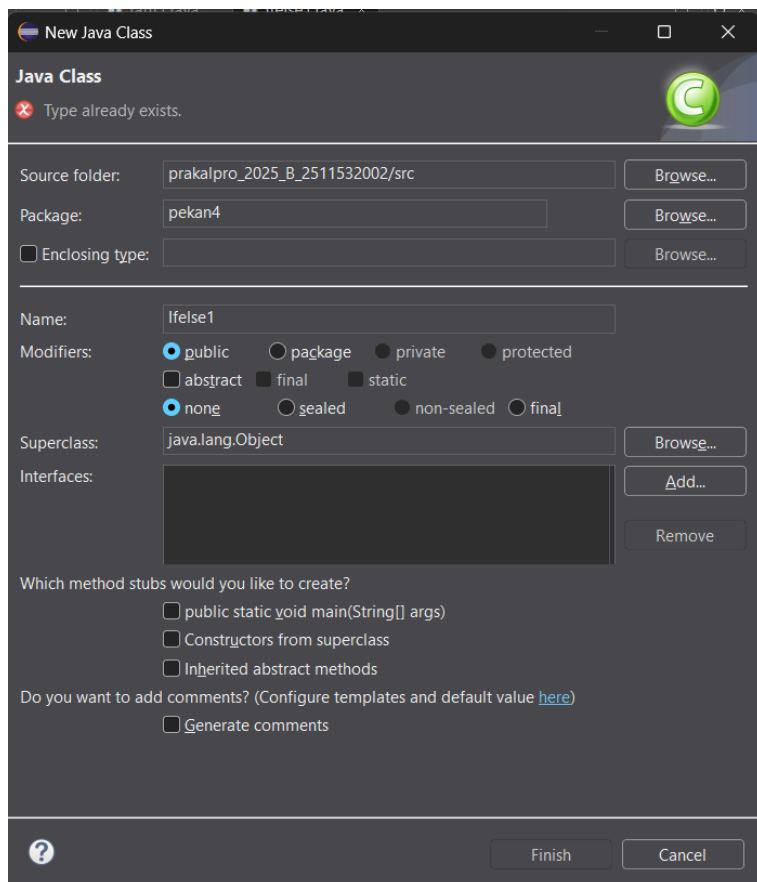


```
<terminated> latlf1 [Java Application] C:\Users\user\p2\pool\p
Input IPK Anda =
4
Anda Lulus Sangat Memuaskan dengan IPK 4.0
```

2. 5 Hasil Program Latlf1

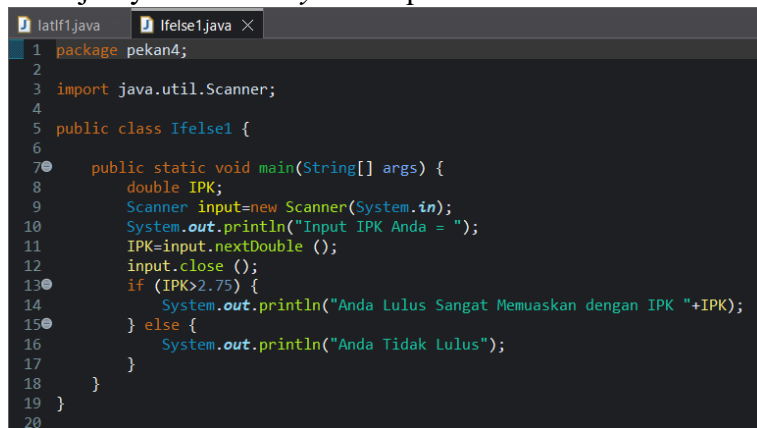
2.2.3 Program Kedua: Ifelse1

1. Buat kembali *class* baru pada *package* pekan4 dengan nama kelas “*Ifelse1*”.



2. 6 Pembuatan Class Program Ifelse1

2. Selanjutnya masukan *syntax* seperti berikut.



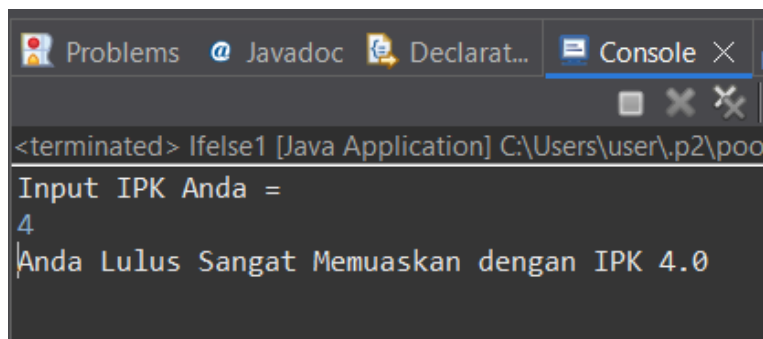
```

1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class Ifelse1 {
6
7     public static void main(String[] args) {
8         double IPK;
9         Scanner input=new Scanner(System.in);
10        System.out.println("Input IPK Anda = ");
11        IPK=input.nextDouble ();
12        input.close ();
13        if (IPK>2.75) {
14            System.out.println("Anda Lulus Sangat Memuaskan dengan IPK "+IPK);
15        } else {
16            System.out.println("Anda Tidak Lulus");
17        }
18    }
19 }
20

```

2. 7 Syintax Program ifelse1

3. Lalu jalankan program, setelah itu masukan nilai *input* yang diinginkan. Dimana jika nilai besar dari 2.75 maka lulus sangat memuaskan.



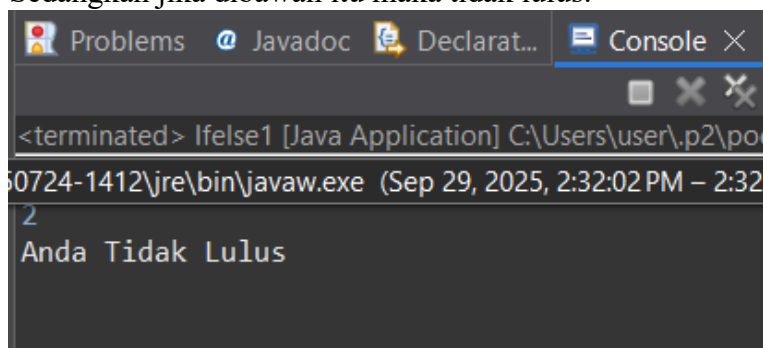
```

<terminated> Ifelse1 [Java Application] C:\Users\user\.p2\poo
Input IPK Anda =
4
Anda Lulus Sangat Memuaskan dengan IPK 4.0

```

2. 8 Hasil Program ifelse1, nilai besar dari 2.75

Sedangkan jika dibawah itu maka tidak lulus.



```

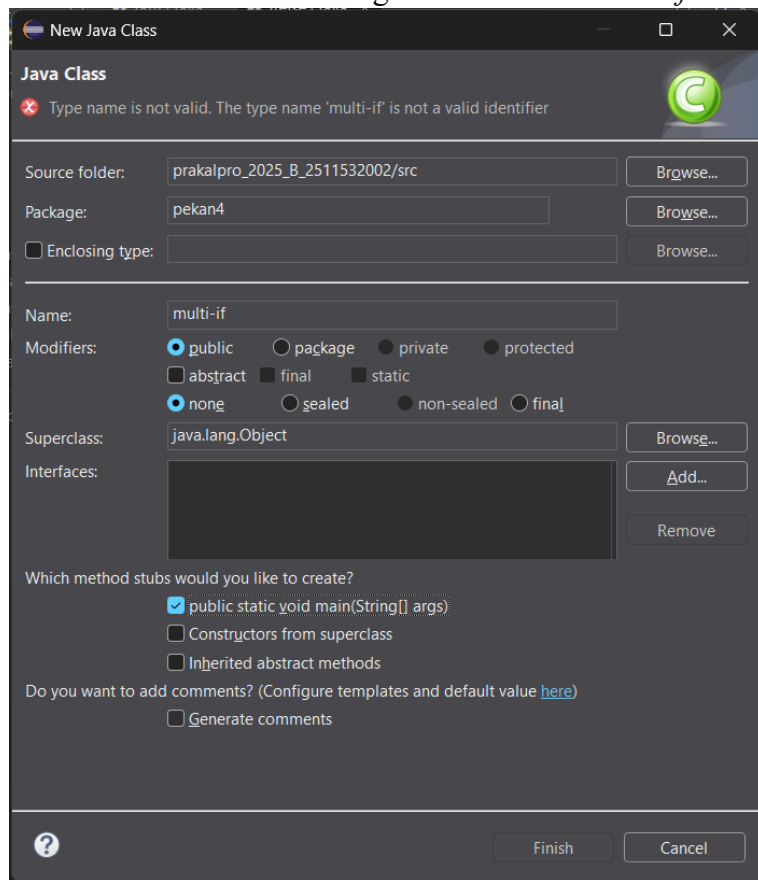
<terminated> Ifelse1 [Java Application] C:\Users\user\.p2\poo
60724-1412\jre\bin\javaw.exe (Sep 29, 2025, 2:32:02 PM – 2:32
2
Anda Tidak Lulus

```

2. 9 Hasil Program ifelse1, nilai kurang dari 2.75

2.2.4 Program Ketiga: Multi-If

1. Buat kembali *class* baru dengan nama kelas “*multi-If*”.



2. 10 Pembuatan Class multi-if

2. Selanjutnya masukan *syntax* seperti berikut.

```

1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class multiIf {
6
7     public static void main(String[] args) {
8         int umur;
9         char sim;
10        Scanner a=new Scanner(System.in);
11        System.out.println("Input Umur Anda: ");
12        umur= a.nextInt ();
13        System.out.println("Apakah Anda Sudah Punya Sim C: ");
14        sim=a.next().charAt(0);
15        a.close();
16        if ((umur >= 17)&& (sim=='y')) {
17            System.out.println("Anda Sudah Dewasa dan Boleh Bawa Motor");
18        }
19        if ((umur >= 17)&& (sim!='y')) {
20            System.out.println("Anda Sudah Dewasa, Tetapi Tidak Boleh Bawa Motor");
21        }
22        if ((umur < 17)&& (sim=='y')) {
23            System.out.println("Anda Belum Cukup Umur Bawa Motor");
24        }
25        if ((umur < 17)&& (sim!='y')) {
26            System.out.println("Anda Belum Umur Punya SIM");
27        }
28    }
29 }
30
31 }

```

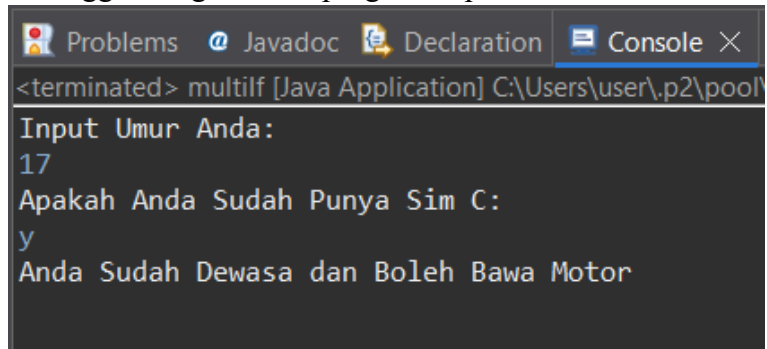
2. 11 Syntax Program multi-if

4. Lalu jalankan program, setelah itu masukan nilai *input* yang diinginkan. Disini saya memasaukan nilai berikut:

Input umur Anda:17

Apakah Anda Sudah Punya Sim C: y

Sehingga menghasilkan program seperti berikut.



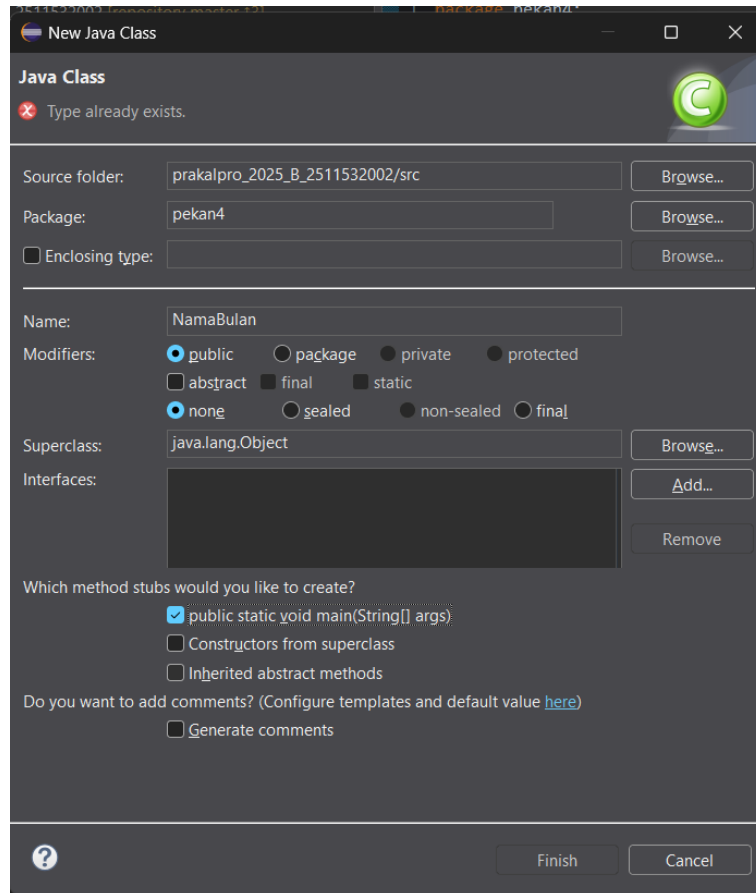
```
Problems Javadoc Declaration Console X
<terminated> multilf [Java Application] C:\Users\user\.p2\pool\
Input Umur Anda:
17
Apakah Anda Sudah Punya Sim C:
y
Anda Sudah Dewasa dan Boleh Bawa Motor
```

2. 12 Hasil Program multi-if

2.2.5 Program Keempat: NamaBulan


Program berikut menerapkan *conditional formatting switch-case*. Tujuannya adalah untuk menentukan salah satu dari banyak pilihan dari banyak pilihan berdasarkan nilai variabel tertentu.

1. Buat kembali *class* baru dengan nama kelas “NamaBulan”.



2. 13 Pembuatan Class Program NamaBulan

2. Selanjutnya masukan *syntax* seperti berikut.



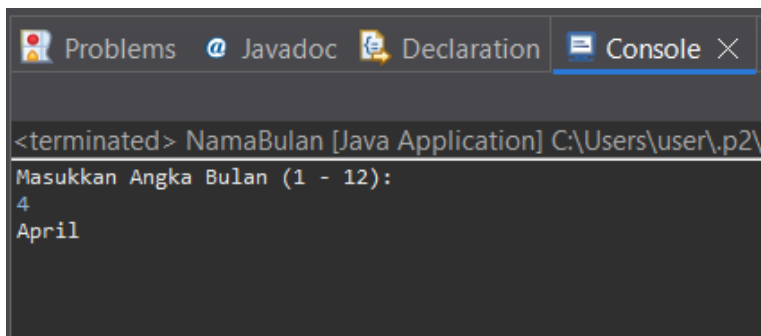
```

1  package pekan4;
2
3  import java.util.Scanner;
4
5  public class NamaBulan {
6      public static void main(String[] args) {
7          Scanner scanner=new Scanner(System.in);
8          System.out.println("Masukkan Angka Bulan (1 - 12): ");
9          int bulan = scanner.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         scanner.close();
51     }
52 }
53

```

2. 14 Syntax Program NamaBulan

5. Lalu jalankan program, setelah itu masukan nilai *input* bulan yang diinginkan. Sehingga menghasilkan program seperti berikut.



```

<terminated> NamaBulan [Java Application] C:\Users\user\.p2\
Masukkan Angka Bulan (1 - 12):
4
April

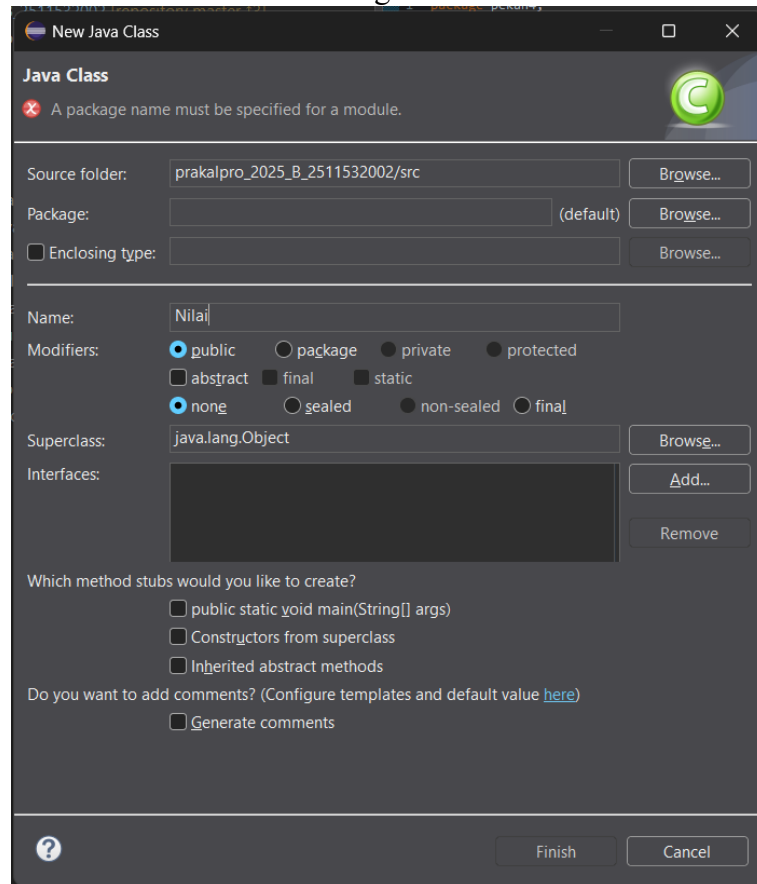
```

2. 15 Hasil Program NamaBulan

2.2.6 Program Kelima: Nilai

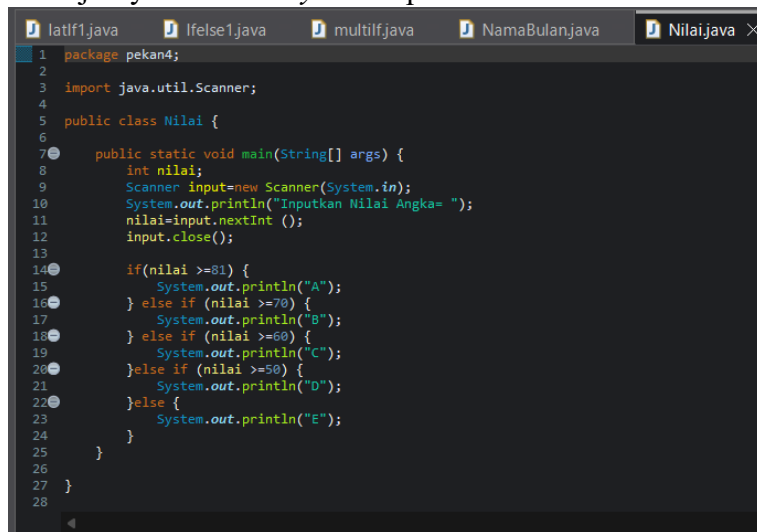
Program berikut menerapkan *conditional formatting if-else-if*. Tujuannya adalah untuk hasil dari banyak kondisi yang disusun secara berurutan.

3. Buat kembali *class* baru dengan nama kelas “Nilai”.



2. 16 Pembuatan Program Nilai

4. Selanjutnya masukan *syntax* seperti berikut.



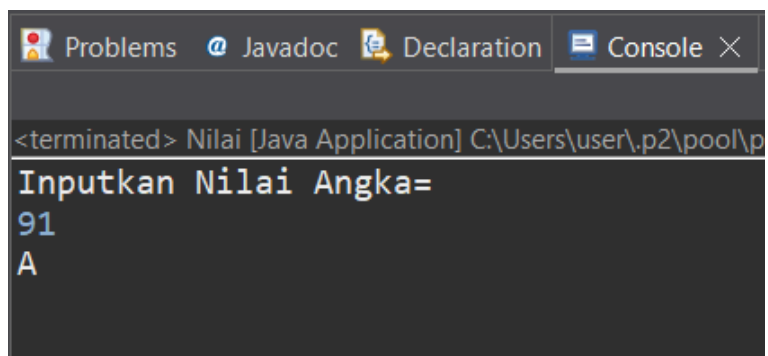
```

1 package pekan4;
2
3 import java.util.Scanner;
4
5 public class Nilai {
6
7     public static void main(String[] args) {
8         int nilai;
9         Scanner input=new Scanner(System.in);
10        System.out.println("Inputkan Nilai Angka= ");
11        nilai=input.nextInt ();
12        input.close();
13
14        if(nilai >=81) {
15            System.out.println("A");
16        } else if (nilai >=70) {
17            System.out.println("B");
18        } else if (nilai >=60) {
19            System.out.println("C");
20        }else if (nilai >=50) {
21            System.out.println("D");
22        }else {
23            System.out.println("E");
24        }
25    }
26
27 }
28

```

2. 17 Syntax Program Nilai

6. Lalu jalankan program, setelah itu masukan nilai *input* angka yang diinginkan. Sehingga menghasilkan program seperti berikut.



```

<terminated> Nilai [Java Application] C:\Users\user\.p2\pool\p
Inputkan Nilai Angka=
91
A

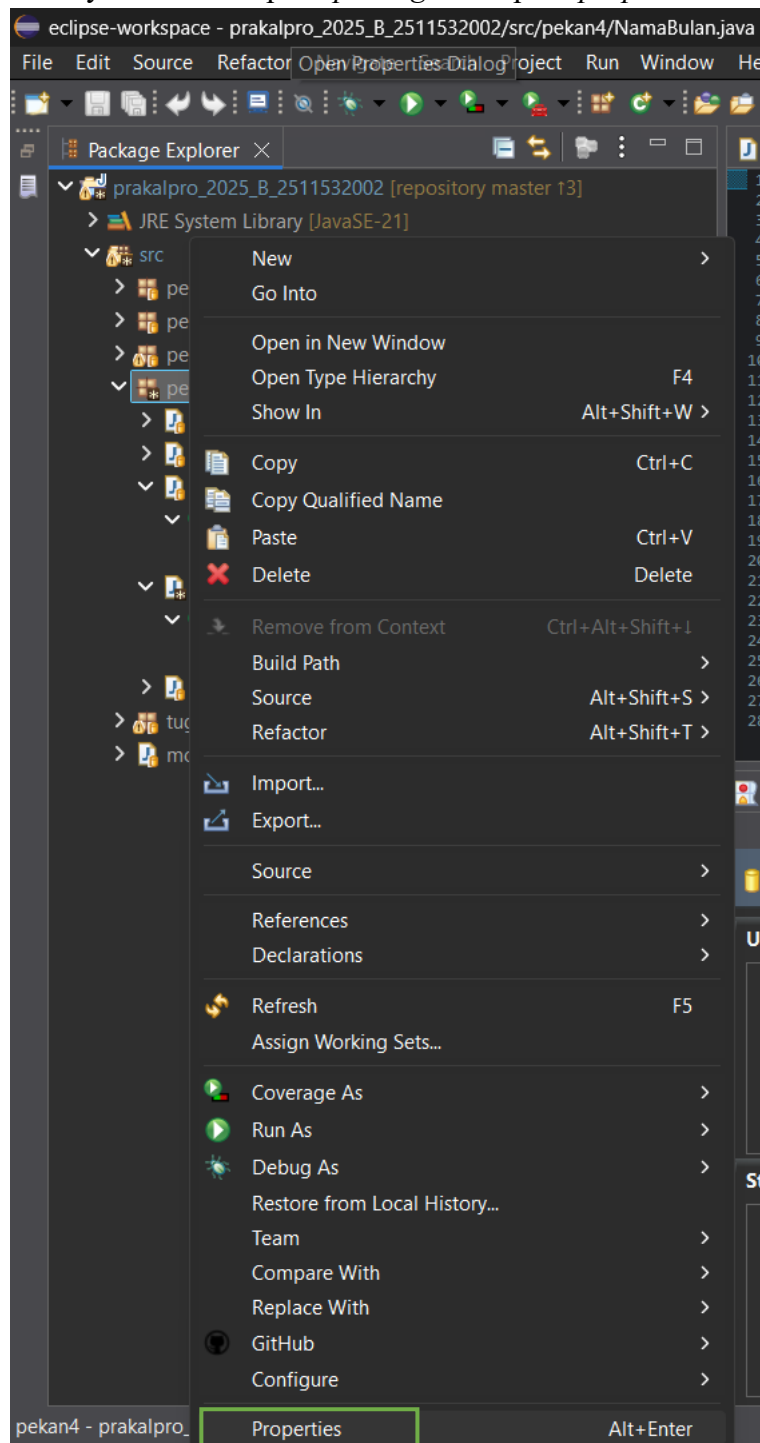
```

2. 18 Hasil Program Nilai

2.2.7 Upload ke GitHub

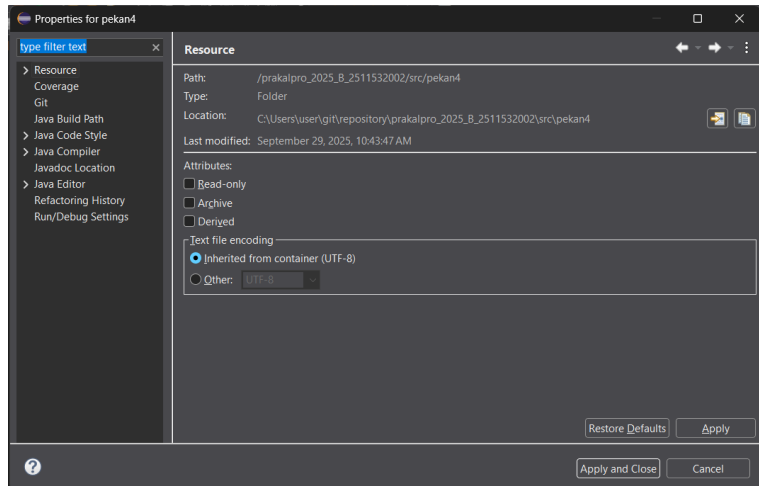
1. Setelah semua program selesai dibuat, selanjutnya adalah *user* perlu untuk memasukkan program yang dibuat di Eclipse ke *repository* GitHub. Selain dengan cara “*Team*” atau “*Commit and Push*”. Bisa juga dengan cara *upload* manual.

Caranya klik kanan pada *package*, lalu pilih *properties*.



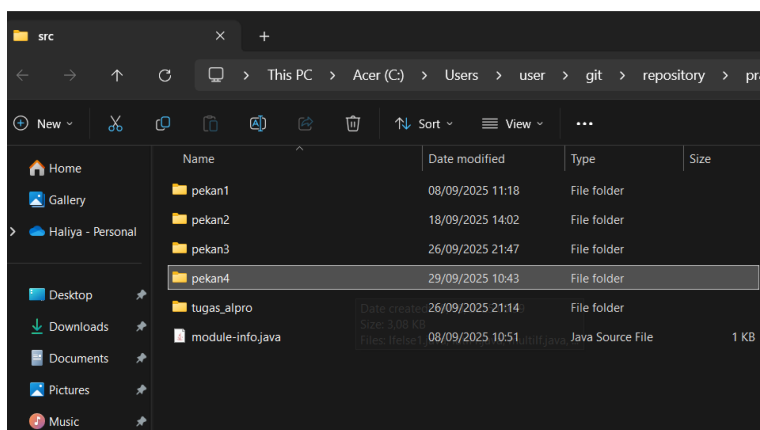
2. 19 Package Properties

2. Maka akan muncul *pop up menu* “*Properteis for Pekan4*”. Lalu klik icon “*Show in System Folder*”.



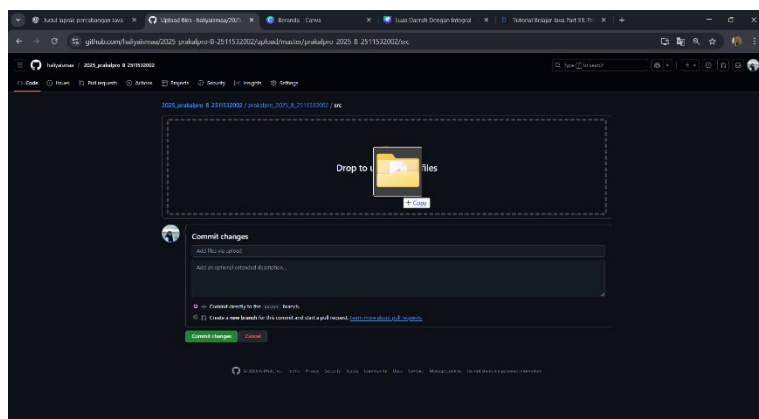
2. 20 Properties Package Pekan4

3. Maka akan menunjukan lokasi *file* “pekan4” pada *File Explorer computer*.



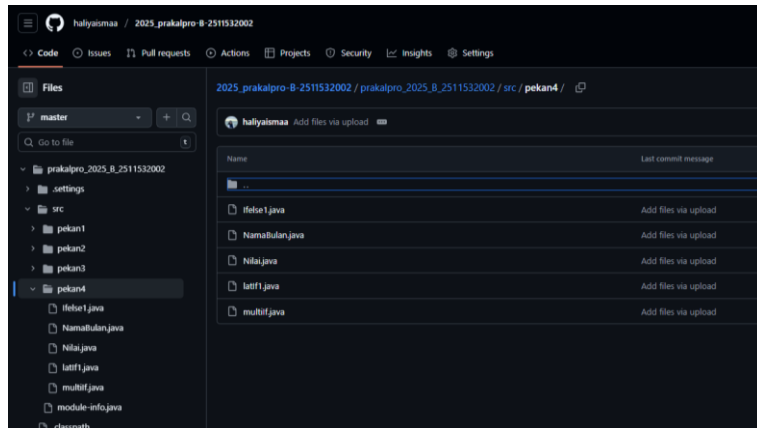
2. 21 Lokasi File Pekan4 pada File Explorer

4. Setelah itu *drag and drop* file “pekan4” ke GitHub. Lalu tambahkan keterangan jika perlu. Setelah itu klik *upload*.



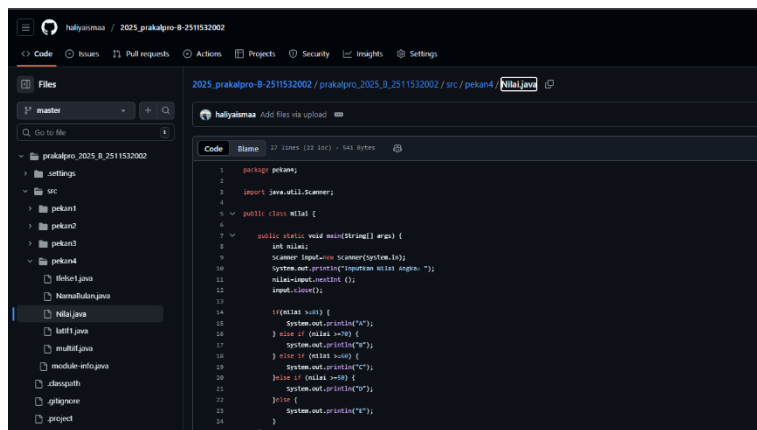
2. 22 Upload ke GitHub

5. Maka dapat dilihat pada *repository GitHub* bahwa file “pekan4” telah terunggah via *Upload*.



2. 24 Upload Pekan4 GitHub

6. Dapat kita lihat disini jika kita buka, maka akan muncul *file package* dan *class* yang telah di buat di Eclipse.



2. 23 Hasil Upoad Kode Program di GitHub.

BAB III

KESIMPULAN

3.1 Kesimpulan

Berdasarkan hasil praktikum yang telah dilakukan dengan topik “*Conditional Formating*”. Dapat disimpulkan bahwa:

1. Struktur kendali percabangan (*conditional formatting*) merupakan dasar penting dalam pemrograman, karena memungkinkan program untuk mengambil keputusan berdasarkan kondisi.
2. Mahasiswa dapat memahami implementasi setiap *conditional formatting* (*if*, *if-else*, *if-else-if*, *multi-if*, serta *switch-case*) melalui praktikum, dimana program yang dibuat berhasil dijalankan dan hasil uji coba telah sesuai.

3.2 Saran

Saran untuk pelaksanaan praktikum kedepannya adalah diharapkan agar mahasiswa lebih teliti dalam memahami *syntax* agar meminimalisir kesalahan penulisan kode. Selain itu juga diharapkan praktikum disertai dengan lebih banyak contoh kasus agar mahasiswa dapat lebih memahami konsep-konsep penerapan dari setiap operator dalam pemrograman.

DAFTAR PUSTAKA

- [1] [Oracle, “The switch Statement (Java™ Tutorials > Learning the Java Language > Language Basics),” [Daring]. Tersedia pada: <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/switch.html> [Diakses: 29-Sep-2025].
- [2] Duniaikom, “Tutorial Belajar Java: Percabangan Kondisi If-Else-If Bahasa Java,” [Daring]. Tersedia pada: <https://www.duniaikom.com/tutorial-belajar-java-percabangan-kondisi-if-else-if-bahasa-java/> [Diakses: 22-Sep-2025].
- [3] Petani Kode, “Belajar Java: Memahami 3 Bentuk Percabangan,” [Daring]. Tersedia pada: <https://www.petanikode.com/java-percabangan/> [Diakses: 22-Sep-2025].