

LAPORAN PRAKTIKUM
PEMROGRAMAN ALGORITMA DAN PEMROGRAMAN
LOGIKA, KELAS OBJEK, DAN MANIPULASI STRING

Disusun Oleh:

Muhammad Fharel

2511531010

Dosen Pengampu:

Dr. Wahyudi S.T.M.T

Asisten Pratikum:

Muhammad Zaki Al Hafiz



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS

2025

KATA PENGANTAR

Segala puji penulis panjatkan ke hadirat Allah SWT atas rahmat dan karunia-Nya sehingga laporan praktikum Algoritma dan Pemograman pada tanggal 12 November 2025 yang membahas tentang bahasa pemograman java spesifiknya pada bagian Logika, Kelas Objek, Manipulasi String dan bagaimana penerapannya untuk membuat *system* yang sesuai dengan kebutuhan. Materi ini penting karena menjadi fondasi dalam memahami pemograman.

Ucapan terima kasih ditujukan kepada dosen pengampu, asisten praktikum, serta rekan-rekan yang telah membantu dalam proses pelaksanaan praktikum. Penulis menyadari bahwa penulisan laporan masih jauh dari kata sempurna, oleh karena itu kritik dan saran sangat diharapkan untuk penyempurnaan di kemudian hari. Semoga laporan ini memberikan manfaat dan menambah wawasan pembaca.

Padang, 15 November 2025

Penulis

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI.....	ii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Tujuan Praktikum	1
1.3 Manfaat Praktikum	1
BAB II PEMBAHASAN	2
2.1 Pengertian Logika, Kelas Objek dan Manipulasi String	2
2.2 Pembuatan Package dan Class Pekan 7	2
2.3 Program Pertama (Bilangan Prima).....	4
2.3.1 Output.....	4
2.3.2 Penjelasan Singkat	5
2.4 Program Kedua (Mahasiswa)	5
2.4.1 Output.....	6
2.4.2 Penjelasan Singkat	6
2.5 Program Ketiga (Panggil Mahasiswa).....	6
2.5.1 Output.....	6
2.5.2 Penjelasan singkat	7
2.6 Program Keempat (Panggil mahasiswa 2)	7
2.6.1 Output.....	8
2.6.2 Penjelasan Singkat	8
2.7 Program Kelima (Panggil Mahasiswa 3).....	8
2.7.1 Output.....	9
2.7.2 Penjelasan Singkat	10

2.8	Program Keenam (String 1).....	10
2.8.1	Output.....	10
2.8.2	Penjelasan Singkat	11
2.9	Program Ketujuh (String 2).....	11
2.9.1	Output.....	11
2.9.2	Penjelasan Singkat	12
BAB III PENUTUP		13
3.1	Kesimpulan.....	13
DAFTAR PUSTAKA		14

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pemrograman Java merupakan salah satu fondasi penting dalam pengembangan perangkat lunak *modern*. Bahasa ini bersifat *object-oriented*, sehingga mahasiswa perlu memahami konsep logika program, kelas, objek, serta manipulasi *string* sebelum mempelajari materi yang lebih kompleks. Praktikum ini dilakukan sebagai sarana untuk menerapkan konsep dasar Java secara langsung melalui pembuatan program sederhana seperti pengecekan bilangan prima, pengelolaan data mahasiswa menggunakan kelas dan objek, serta penggunaan metode manipulasi *string* bawaan Java.

1.2 Tujuan Praktikum

Tujuan dari praktikum ini adalah:

1. Menerapkan logika pemrograman dasar pada studi kasus bilangan prima.
2. Memahami konsep kelas, objek, serta penggunaan metode *setter–getter*.
3. Menggunakan manipulasi *string* dalam berbagai bentuk operasi.
4. Menguji keterkaitan antara input pengguna dan proses dalam program Java.

1.3 Manfaat Praktikum

Manfaat dari praktikum ini adalah:

1. Mahasiswa memahami alur berpikir komputasional melalui struktur logika.
2. Mengembangkan kemampuan membuat dan memanfaatkan kelas objek.
3. Melatih penggunaan fungsi manipulasi *string* dalam bahasa Java.
4. Menjadi dasar untuk mempelajari pemrograman Java tingkat lanjut seperti struktur data dan GUI.

BAB II

PEMBAHASAN

2.1 Pengertian Logika, Kelas Objek dan Manipulasi String

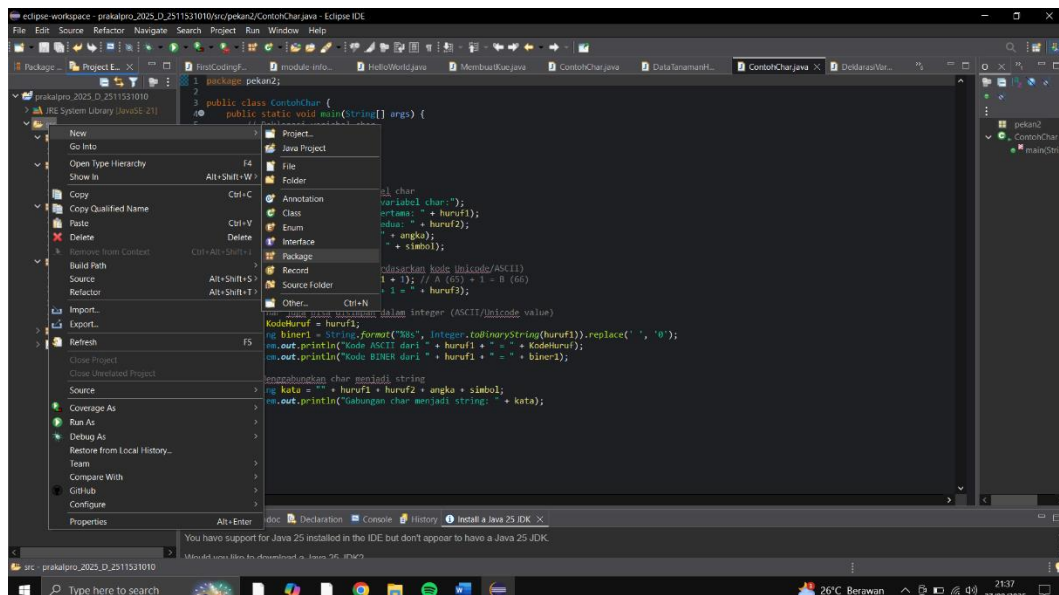
Logika program adalah aturan berpikir yang digunakan komputer untuk menentukan alur keputusan, misalnya menggunakan *if*, *looping*, atau perbandingan. Kesalahan umum mahasiswa biasanya menganggap logika hanya soal benar atau salah, padahal logika juga mencakup aliran eksekusi dan kondisi yang ditetapkan sendiri.

Kelas dan objek adalah konsep inti pemrograman berorientasi objek. Kelas berfungsi sebagai cetakan (*blueprint*), sedangkan objek adalah instans dari cetakan tersebut. Banyak mahasiswa sering mengira kelas hanya kumpulan variabel, padahal kelas juga mengatur perilaku melalui metode.

Manipulasi string merupakan proses pengolahan teks menggunakan metode bawaan Java seperti *length()*, *toUpperCase()*, *concat()*, dan *indexOf()*. *String* diubah tetapi sebenarnya membuat objek baru hal yang sering lupa diperhatikan.

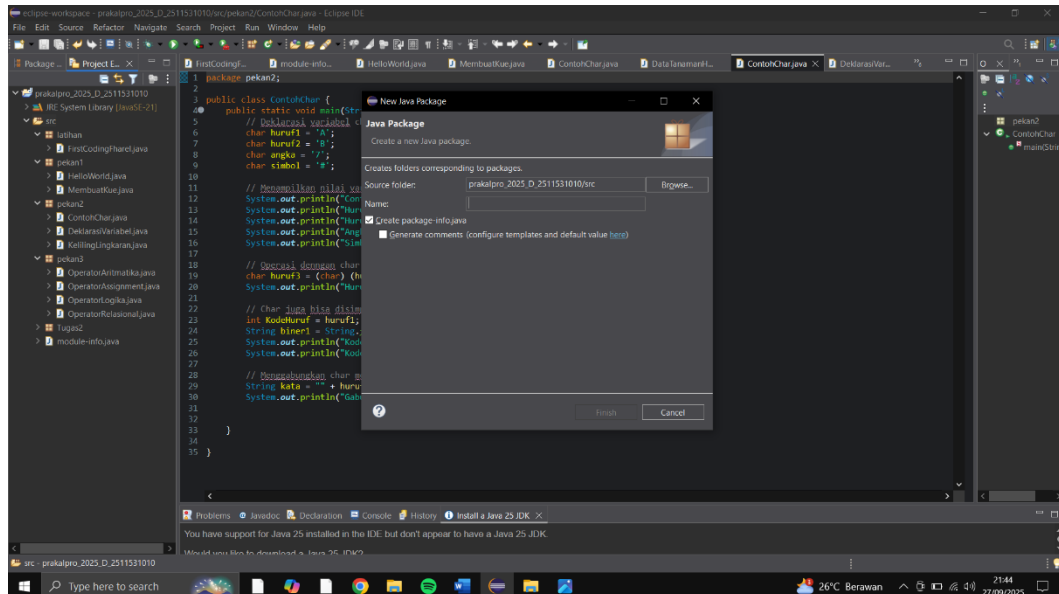
2.2 Pembuatan Package dan Class Pekan 7

1. Untuk membuat *package* baru, klik kanan pada src dan tekan “New” setelah itu pilih *package*.



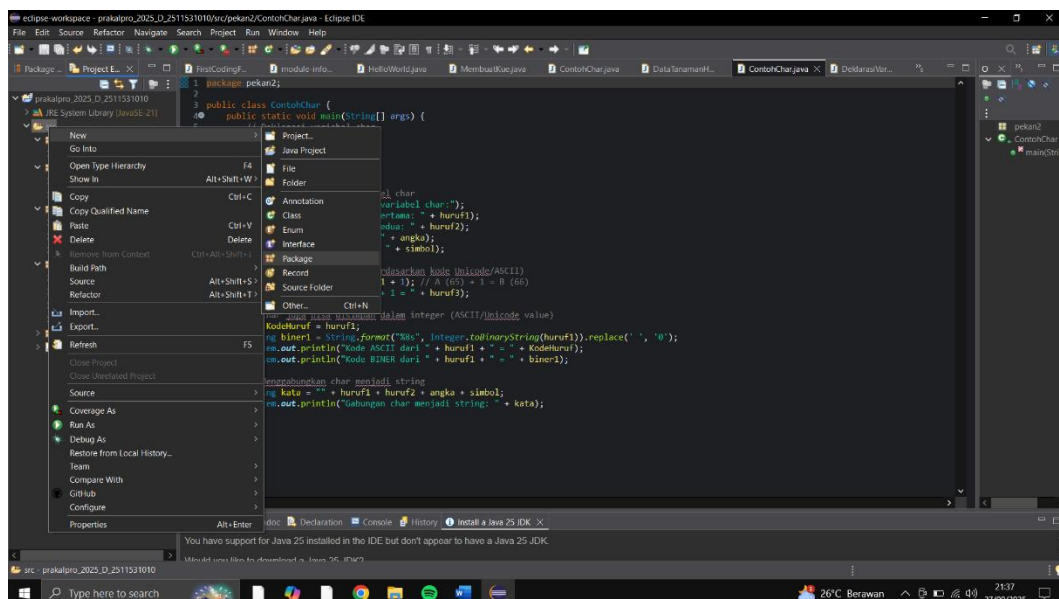
Gambar 2.1

2. Lanjut diberi nama *package* tanpa pakai spasi, huruf kapital ataupun karakter khusus. Seperti “pekan7_2511531010” dan tekan *Finish*.



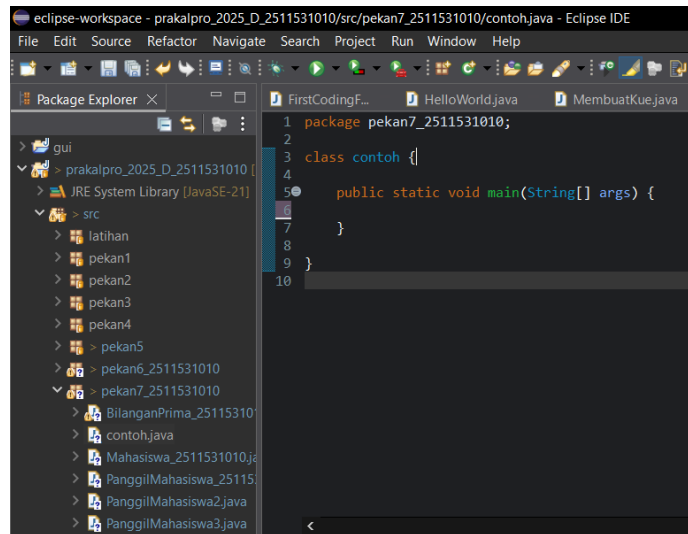
Gambar 2.2

3. Kemudian klik kanan pada *package* “pekan7_2511531010” tersebut dan klik “New” dan pilih bagian “Class” untuk memulai membuat program.



Gambar 2.3

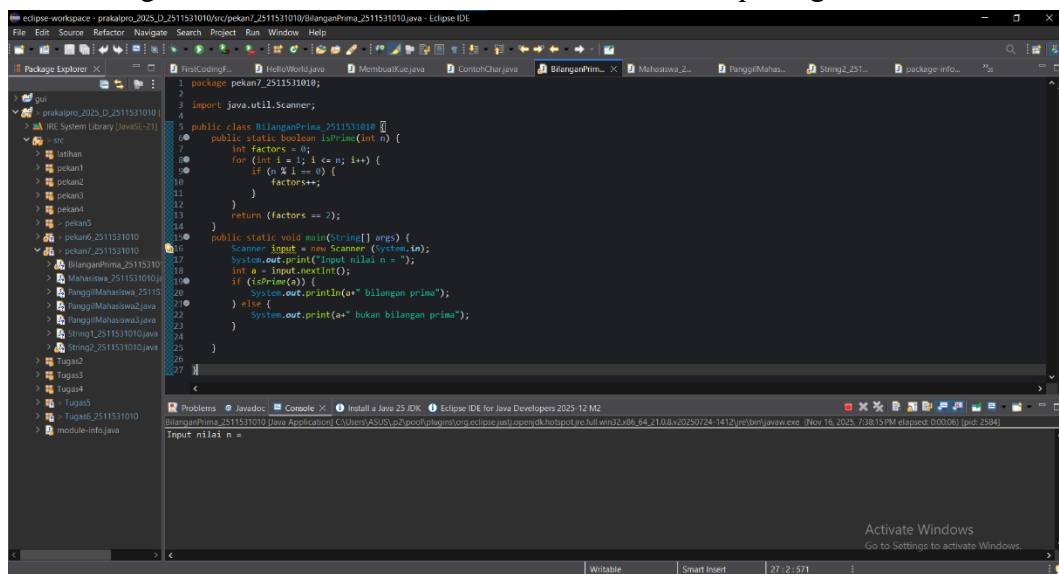
4. Buat nama *Class* yang akan dibuat pada bagian nama tanpa spasi dan menggunakan huruf kapital pada awal kata, lalu klik bagian “*public static void main(string[] args)*”. Kemudian klik *finish*.



Gambar 2.4

2.3 Program Pertama (Bilangan Prima)

Buat *Class* seperti program sebelumnya dan beri nama *Class* tersebut “BilanganPrima_2511531010”. Dan masukkan kode pemrograman berikut.



Gambar 2.5

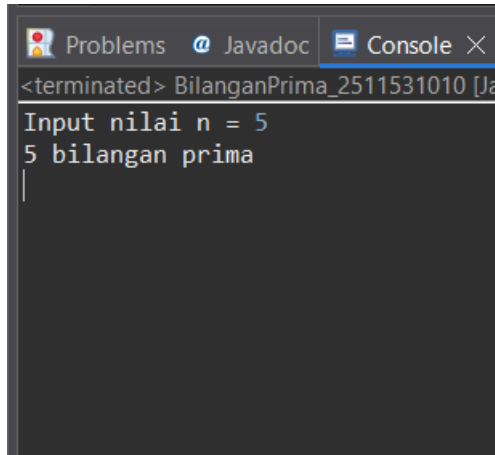
2.3.1 Output

Ketika pengguna memasukkan nilai *n*, program akan menjalankan metode *isPrime()*. Pada metode tersebut, program menghitung berapa kali *n* habis

dibagi oleh bilangan dari 1 hingga n. Jika hasil pembagian habis tepat dua kali, program menampilkan *output*:

“n bilangan prima”. Jika faktor lebih dari dua, program menampilkan:

“n bukan bilangan prima”.



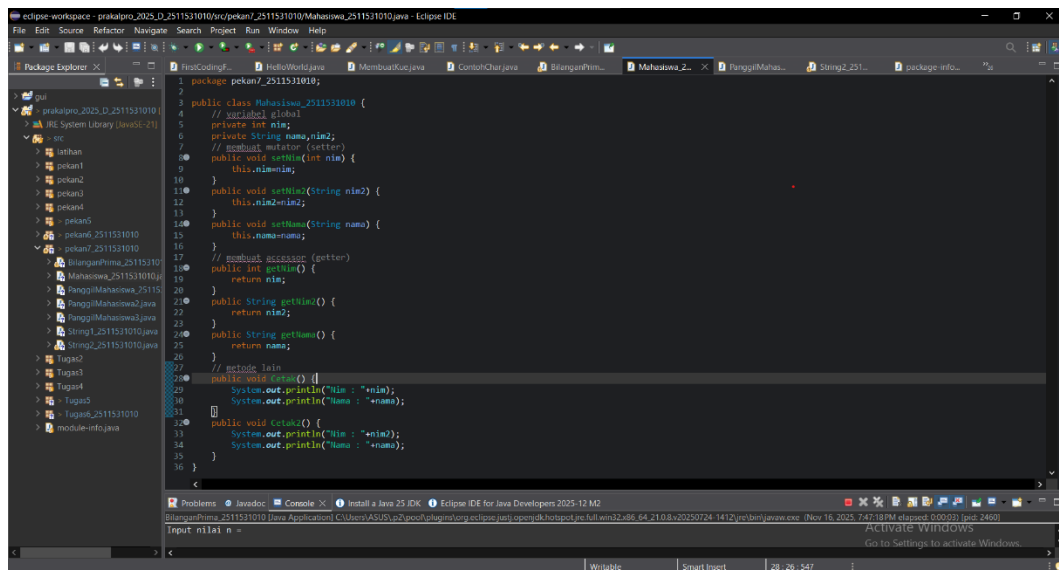
Gambar 2.6

2.3.2 Penjelasan Singkat

Program bilangan prima bekerja dengan mengecek jumlah faktor dari angka yang dimasukkan. Jika jumlah faktornya dua, angka itu prima; jika lebih banyak, maka bukan prima.

2.4 Program Kedua (Mahasiswa)

Buat *Class* seperti program sebelumnya dan beri nama *Class* tersebut “Mahasiswa_2511531010”. Dan masukkan kode pemrograman berikut.



Gambar 2.7

2.4.1 Output

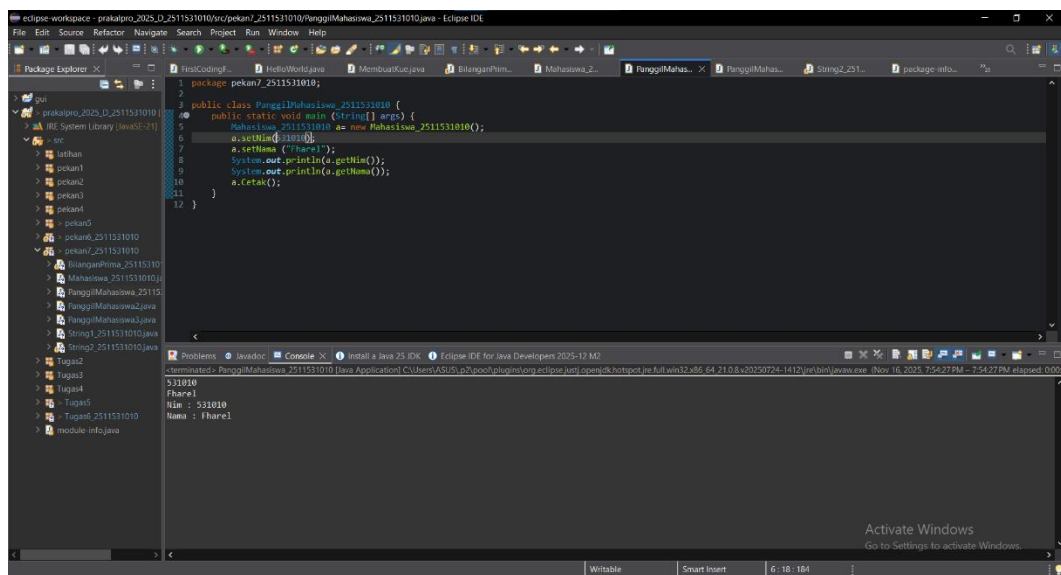
Ketika program dijalankan dengan menggunakan kelas Mahasiswa ini, *output* yang muncul berupa data yang telah dimasukkan melalui metode *setter*. Program akan menampilkan nilai *nim*, *nim2*, serta nama sesuai metode cetak yang dipanggil (*Cetak()* atau *Cetak2()*). Hasil cetak menampilkan garis informasi seperti “Nim : ...” dan “Nama : ...”, sesuai dengan data yang diberikan oleh pengguna.

2.4.2 Penjelasan Singkat

Kelas Mahasiswa ini menjadi dasar untuk menyimpan dan menampilkan data mahasiswa menggunakan konsep *setter*, *getter*, dan metode cetak. Program menunjukkan bagaimana sebuah objek dapat menyimpan nilai dan menampilkannya kembali melalui metode tertentu.

2.5 Program Ketiga (Panggil Mahasiswa)

Buat *Class* seperti program sebelumnya dan beri nama *Class* tersebut “PanggilMahasiswa_2511531010”. Dan masukkan kode pemrograman berikut.

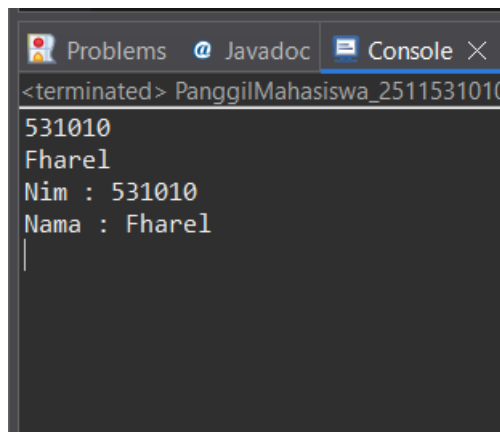


Gambar 2.8

2.5.1 Output

Program ini membuat sebuah objek Mahasiswa kemudian mengisi nilai NIM dan nama menggunakan metode *setNim()* dan *setNama()*. Setelah itu, program mencetak nilai NIM dan nama melalui *getter*, dan selanjutnya memanggil metode *Cetak()* untuk menampilkan kembali data tersebut dalam format “Nim : 531010” dan “Nama : Fharel”. Output akhirnya

menunjukkan identitas mahasiswa yang sudah ditentukan di dalam kode tanpa melibatkan input dari pengguna.



```
<terminated> PanggilMahasiswa_2511531010
531010
Fharel
Nim : 531010
Nama : Fharel
```

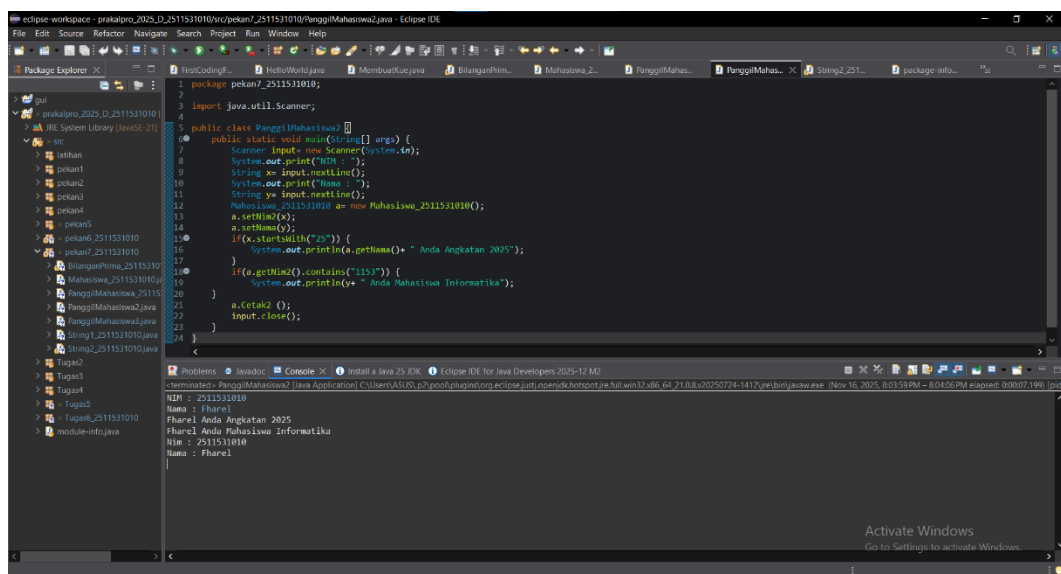
Gambar 2.9

2.5.2 Penjelasan singkat

Program ini memperlihatkan penggunaan objek mahasiswa untuk menyimpan dan mengambil data melalui *setter* dan *getter*. Outputnya berupa informasi mahasiswa yang sebelumnya sudah dimasukkan secara langsung ke dalam program.

2.6 Program Keempat (Panggil mahasiswa 2)

Buat *Class* seperti program sebelumnya dan beri nama *Class* tersebut “PanggilMahasiswa2_2511531010”. Dan masukkan kode pemrograman berikut.



```
package pekan7_2511531010;

import java.util.Scanner;

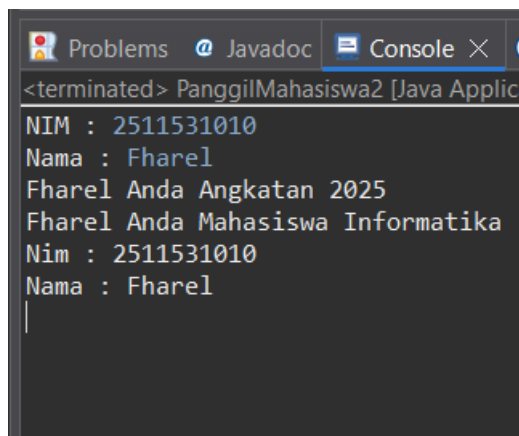
public class PanggilMahasiswa2 {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Nim : ");
        String xx = input.nextLine();
        System.out.print("Nama : ");
        String yy = input.nextLine();
        Mahasiswa_2511531010 m = new Mahasiswa_2511531010();
        m.setNim(xx);
        m.setNama(yy);
        if (m.getStartMonth("25")) {
            System.out.println(m.getNama() + " Anda Angkatan 2025");
        }
        if (m.getNim().contains("1153")) {
            System.out.println(m + " Anda Mahasiswa Informatika");
        }
        m.cetak2();
        input.close();
    }
}
```

Gambar 2.10

2.6.1 Output

Pada saat program dijalankan, pengguna diminta memasukkan NIM dan nama. Program kemudian menyimpan nilai tersebut menggunakan *setter*, lalu melakukan pengecekan tambahan menggunakan metode *string*. Jika NIM dimulai dengan “25”, program menampilkan pesan bahwa pengguna adalah angkatan 2025. Jika NIM mengandung “1153”, program menampilkan pesan bahwa pengguna merupakan mahasiswa Informatika. Setelah itu, metode *Cetak2()* dipanggil untuk menampilkan kembali data NIM dan nama yang sudah diinput. *Output* akhirnya merupakan gabungan antara identifikasi angkatan, jurusan, dan data mahasiswa itu sendiri.



```
<terminated> PanggilMahasiswa2 [Java Application]
NIM : 2511531010
Nama : Fharel
Fharel Anda Angkatan 2025
Fharel Anda Mahasiswa Informatika
Nim : 2511531010
Nama : Fharel
```

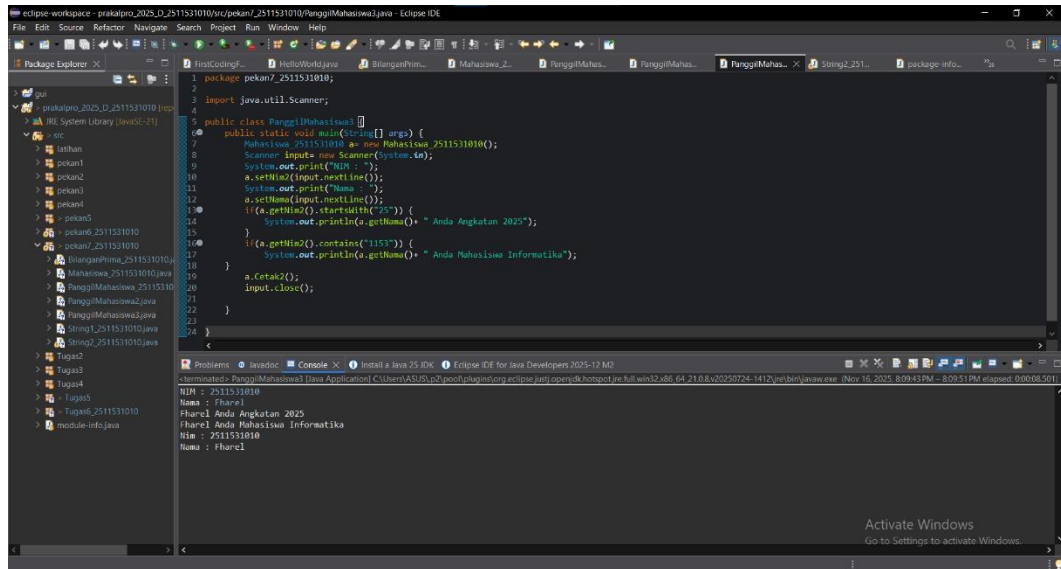
Gambar 2.11

2.6.2 Penjelasan Singkat

Program membaca NIM dan nama dari pengguna, kemudian menentukan angkatan dan jurusan berdasarkan pola pada NIM. Setelah itu, program menampilkan ulang data mahasiswa menggunakan metode cetak yang tersedia.

2.7 Program Kelima (Panggil Mahasiswa 3)

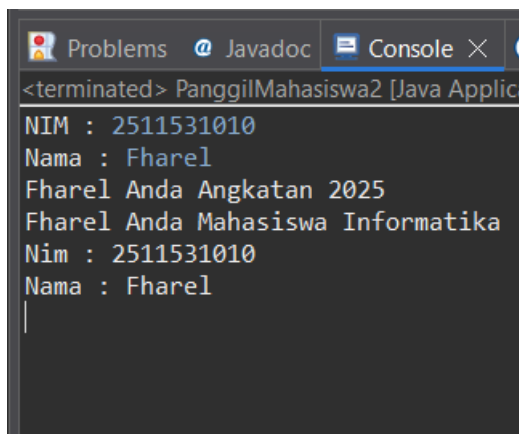
Buat *Class* seperti program sebelumnya dan beri nama *Class* tersebut “PanggilMahasiswa3_2511531010”. Dan masukkan kode pemrograman berikut.



Gambar 2.12

2.7.1 Output

Program ini bekerja hampir sama dengan Panggil Mahasiswa 2, tetapi proses pengisian atribut dilakukan langsung saat *input* diterima. Ketika pengguna memasukkan NIM dan nama, nilai tersebut langsung disimpan oleh objek Mahasiswa. Program kemudian mengecek apakah NIM termasuk angkatan 2025 atau mahasiswa Informatika dengan memeriksa awal dan isi *string*. Setelah itu, metode Cetak2() menampilkan ulang NIM dan nama yang telah diberikan. Outputnya berupa hasil analisis angkatan, jurusan, dan data mahasiswa yang telah dimasukkan.



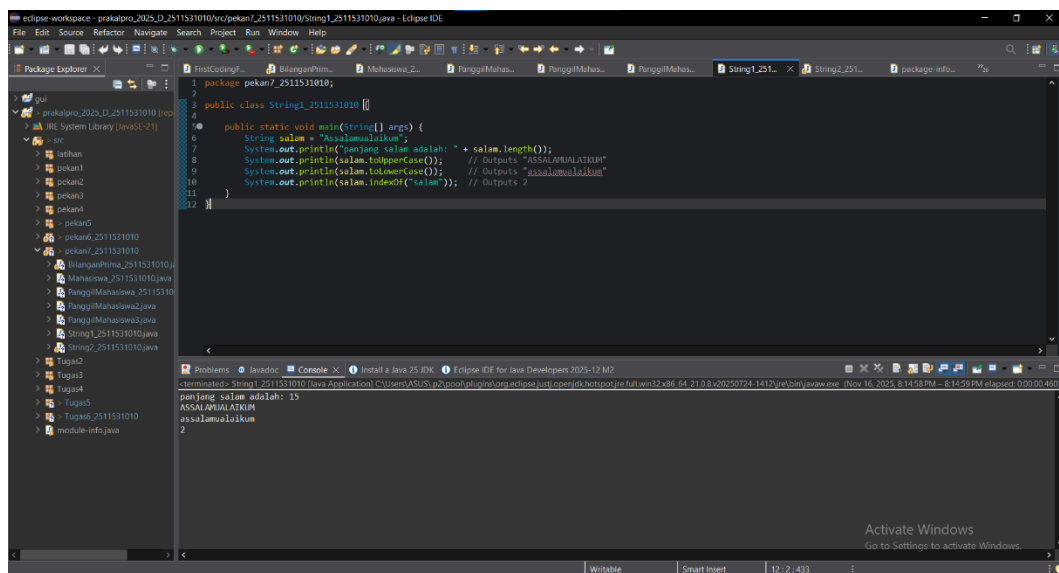
Gambar 2.13

2.7.2 Penjelasan Singkat

Program membaca NIM dan nama, memeriksa ciri-ciri tertentu pada NIM, lalu menampilkan informasi mahasiswa. Program ini menegaskan penggunaan objek, *input* pengguna, dan pemeriksaan *string*.

2.8 Program Keenam (String 1)

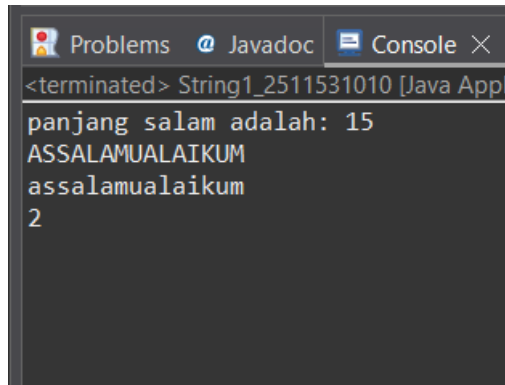
Buat *Class* seperti program sebelumnya dan beri nama *Class* tersebut “String1_2511531010”. Dan masukkan kode pemrograman berikut.



Gambar 2.14

2.8.1 Output

Saat program dijalankan, *string* “Assalamualaikum” diolah dengan beberapa metode bawaan Java. Program pertama menampilkan panjang *string* menggunakan *length()*, kemudian mengubah seluruh huruf menjadi kapital menggunakan *toUpperCase()*, serta menjadi huruf kecil menggunakan *toLowerCase()*. Program juga mencari posisi *substring* “salam” dalam *string* utama menggunakan *indexOf()*. Outputnya berupa serangkaian informasi mengenai panjang *string*, bentuk kapital, bentuk huruf kecil, dan posisi *substring* tersebut.



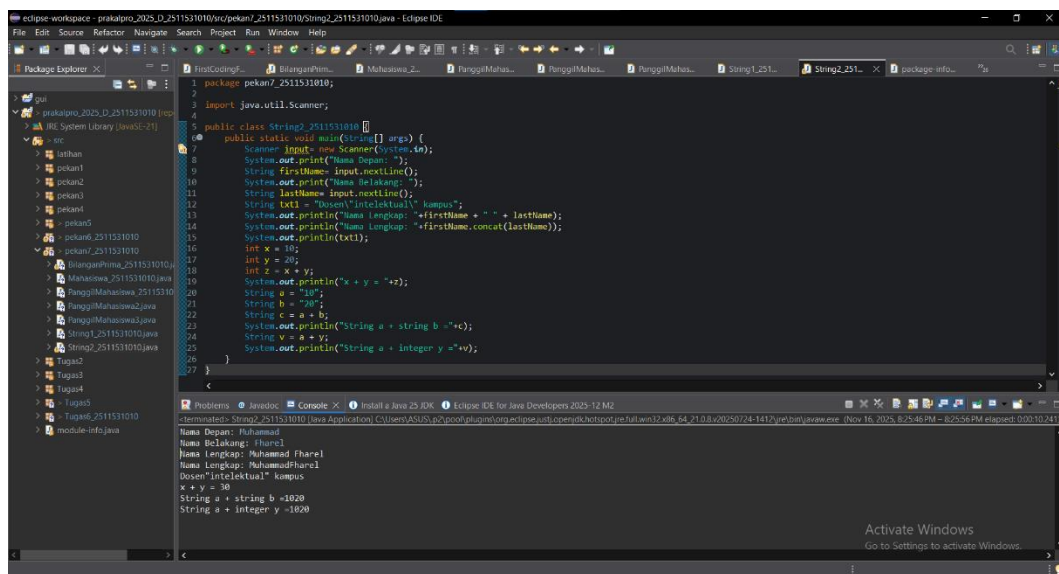
Gambar 2.15

2.8.2 Penjelasan Singkat

Program ini menampilkan contoh penggunaan metode dasar pada *string* seperti menghitung panjang, mengubah huruf, dan mencari posisi kata tertentu. Program ini menjadi latihan dasar dalam manipulasi *string* di Java.

2.9 Program Ketujuh (String 2)

Buat *Class* seperti program sebelumnya dan beri nama *Class* tersebut “String2_2511531010”. Dan masukkan kode pemrograman berikut.



Gambar 2.16

2.9.1 Output

Program meminta *input* nama depan dan nama belakang dari pengguna, kemudian menampilkan nama lengkap dengan dua cara: penggabungan menggunakan operator “+” dan metode *concat()*. Program juga mencetak isi variabel *txt1* sebagai contoh *string*. Selanjutnya, program menunjukkan perbedaan antara penjumlahan nilai integer ($x + y$) dan penggabungan

string (a + b). Terakhir, program menampilkan hasil penggabungan antara *string* dan *integer* yang menunjukkan bahwa tipe data *string* akan mengubah hasil menjadi bentuk teks.

2.9.2 Penjelasan Singkat

Program ini menggabungkan *input* pengguna dengan beberapa contoh operasi *string* dan penjumlahan angka. Program menunjukkan perbedaan cara Java memperlakukan *string* dan *integer* dalam operasi penggabungan dan penjumlahan.

BAB III

PENUTUP

3.1 Kesimpulan

Praktikum ini memberikan pemahaman dasar mengenai penerapan logika pemrograman, penggunaan kelas dan objek, serta manipulasi *string* dalam bahasa Java. Melalui serangkaian program, mulai dari pengecekan bilangan prima hingga pengelolaan data mahasiswa, mahasiswa dapat melihat bagaimana konsep teoretis diterapkan secara langsung dalam bentuk kode. Setiap program menunjukkan hubungan antara *input*, proses, dan *output*, sehingga mahasiswa dapat memahami alur kerja sebuah program secara menyeluruh. Pemahaman ini menjadi fondasi yang sangat penting sebelum mempelajari konsep OOP yang lebih kompleks, struktur data, maupun pengembangan aplikasi berbasis Java di tingkat lanjut.

DAFTAR PUSTAKA

- [1] Deitel, P. J., & Deitel, H. M.. 2017. *Java: How to Program (10th Edition)*. Pearson Education.
- [2] Oracle. 2024. *The while and do-while Statements (Java Documentation)*.
<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/while.html>.
[Diakses: 15-November-2025].
- [3] Wahana Komputer. 2021. *Belajar Pemrograman Java dari Nol*. Yogyakarta: Andi Publisher.
- [4] Raharjo, Eko. 2019. “Dasar-Dasar Pemrograman Java.” Universitas Bina Nusantara.