

**LAPORAN PRAKTIKUM**  
**ALGORITMA DAN PEMOGRAMAN**  
**METHOD DAN STRING PADA JAVA**  
**PEKAN 7**

Disusun Oleh :

KINAYA NOVRYA MANDA  
(2511531016)

Dosen Pengampu :

DR. WAHYUDI, S.T, M.T

Asisten Praktikum :

MUHAMMAD ZAKI AL HAFIZ



DEPARTEMEN INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ANDALAS

2025

## KATA PENGANTAR

Puji syukur penulis sampaikan kehadirat Allah SWT. Salawat dan salam disampaikan kepada Nabi Muhammad SAW. Karena thaufik dan hidayah-Nya, laporan praktikum Java ini dapat diselesaikan dengan baik dan tepat waktu. Laporan ini disusun sebagai salah satu bentuk pertanggungjawaban dari kegiatan praktikum yang telah dilaksanakan, sekaligus sebagai sarana untuk memperdalam pemahaman mengenai konsep dasar serta penerapan bahasa pemrograman Java.

Melalui praktikum ini, penulis memperoleh pengalaman langsung dalam memahami *method* dan *string* yang memungkinkan program membuat blok kode berulang kali dan mengolah data berdasarkan kondisi tertentu menggunakan Java. Diharapkan laporan ini dapat memberikan gambaran yang jelas mengenai materi yang telah dipelajari serta hasil dari percobaan yang dilakukan selama praktikum berlangsung.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna, baik dari segi isi maupun penyajiannya. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa yang akan datang.

Akhir kata, penulis mengucapkan terima kasih kepada dosen pengampu, asisten laboratorium, serta semua pihak yang telah memberikan bimbingan, arahan, dan dukungan sehingga laporan praktikum ini dapat tersusun. Semoga laporan ini dapat bermanfaat bagi pembaca, khususnya dalam memahami dasar-dasar pemrograman Java.

Padang, November 2025

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>i</b>
<b>DAFTAR ISI .....</b>	<b>ii</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>iii</b>
<b>BAB 1 PENDAHULUAN</b>	
1.1 Latar Belakang.....	1
1.2 Tujuan Praktikum.....	2
1.3 Manfaat Praktikum.....	2
<b>BAB 2 PEMBAHASAN</b>	
2.1 Kode Program.....	3
2.2 Langkah Kerja .....	11
2.3 Analisis Hasil.....	20
<b>BAB 3 KESIMPULAN</b>	
3.1 Hasil Praktikum .....	21
3.2 Saran Pengembangan .....	25
<b>DAFTAR KEPUSTAKAAN .....</b>	<b>26</b>

## DAFTAR LAMPIRAN

Kode Program 2.1 (Kode Program BilanganPrima).....	3
Kode Program 2.2 (Kode Program Mahasiswa) .....	4
Kode Program 2.3 (Kode Program PanggilMahasiswa).....	5
Kode Program 2.4 (Kode Program PanggilMahasiswa2).....	7
Kode Program 2.5 (Kode Program String1) .....	8
Kode Program 2.6 (Kode Program String2) .....	9
Gambar 2.1 (Langkah 1 BilanganPrima) .....	11
Gambar 2.2 (Langkah 2 BilanganPrima) .....	11
Gambar 2.3 (Langkah 3 BilanganPrima) .....	12
Gambar 2.4 (Langkah 1 Mahasiswa).....	12
Gambar 2.5 (Langkah 2 Mahasiswa).....	12
Gambar 2.6 (Langkah 3 Mahasiswa).....	13
Gambar 2.7 (Langkah 4 Mahasiswa).....	13
Gambar 2.8 (Langkah 5 Mahasiswa).....	13
Gambar 2.9 (Langkah 1 PanggilMahasiswa) .....	14
Gambar 2.10 (Langkah 2 PanggilMahasiswa) .....	14
Gambar 2.11 (Langkah 3 PanggilMahasiswa) .....	14
Gambar 2.12 (Langkah 4 PanggilMahasiswa) .....	15
Gambar 2.13 (Langkah 1 PanggilMahasiswa2) .....	15
Gambar 2.14 (Langkah 2 PanggilMahasiswa2) .....	15
Gambar 2.15 (Langkah 3 PanggilMahasiswa2) .....	16
Gambar 2.16 (Langkah 4 PanggilMahasiswa2) .....	16

Gambar 2.17 (Langkah 5 PanggilMahasiswa2) .....	16
Gambar 2.18 (Langkah 1 String1) .....	17
Gambar 2.19 (Langkah 2 String1) .....	17
Gambar 2.20 (Langkah 3 String1) .....	17
Gambar 2.21 (Langkah 4 String1) .....	18
Gambar 2.22 (Langkah 1 String2) .....	18
Gambar 2.23 (Langkah 2 String2) .....	18
Gambar 2.24 (Langkah 3 String2) .....	19
Gambar 2.25 (Langkah 4 String2) .....	19
Gambar 2.26 (Langkah 5 String2) .....	19
Gambar 3.1 (Hasil Praktikum BilanganPrima) .....	21
Gambar 3.2 (Hasil Praktikum PanggilMahasiswa) .....	21
Gambar 3.3 (Hasil Praktikum PanggilMahasiswa2) .....	22
Gambar 3.4 (Hasil Praktikum String1) .....	23
Gambar 3.5 (Hasil Praktikum String2) .....	23

## BAB 1

### PENDAHULUAN

#### 1.1 Latar Belakang

Pemrograman Java merupakan salah satu bahasa yang paling banyak digunakan dalam dunia pengembangan perangkat lunak, terutama karena sifatnya yang berorientasi objek dan kemampuannya dalam menangani data secara efisien. Dalam proses pembelajaran dasar pemrograman Java, pemahaman mengenai *method* dan *string* menjadi salah satu fondasi penting yang harus dikuasai oleh mahasiswa. *Method* berfungsi sebagai blok kode yang dapat dipanggil berulang kali sehingga mempermudah pengelolaan program yang kompleks, sedangkan *string* digunakan untuk mengolah data teks yang sangat sering muncul dalam aplikasi nyata.

Pada praktikum ini, mahasiswa diperkenalkan lebih dalam mengenai cara membuat, memanggil, dan memanfaatkan *method* dalam Java. Penggunaan *method* bertujuan agar program lebih terstruktur, modular, dan mudah diperbaiki ketika terjadi kesalahan. Di sisi lain, materi mengenai *string* membantu mahasiswa memahami berbagai operasi dasar seperti mencari karakter, mengubah huruf menjadi besar atau kecil, serta mendapatkan panjang teks. Penguasaan kedua konsep ini akan membantu mahasiswa meningkatkan kemampuan logika sekaligus keterampilan teknis dalam menulis kode.

Selain itu, perkembangan teknologi saat ini menuntut pengembang untuk membuat program yang tidak hanya berjalan dengan baik, tetapi juga mudah dikembangkan dan dipelihara. *Method* dan *string* merupakan dua konsep fundamental yang mendukung tercapainya tujuan tersebut. Dengan melakukan praktikum mengenai kedua materi ini, mahasiswa diharapkan dapat membangun fondasi pemrograman yang kuat sebelum mempelajari konsep lanjutan seperti class, objek, dan struktur data yang lebih kompleks.

## 1.2 Tujuan Praktikum

Tujuan dilakukannya praktikum ini adalah sebagai berikut :

1. Memahami konsep dasar *method* dalam Java serta cara membuat dan memanggilnya.
2. Menguasai berbagai operasi dasar pada *string* seperti *length()*, *toUpperCase()*, *toLowerCase()*, dan *indexOf()*.
3. Melatih kemampuan mahasiswa dalam menulis kode yang lebih rapi, modular, dan mudah dipahami.
4. Meningkatkan pemahaman mahasiswa terhadap peran *method* dan *string* dalam penyelesaian masalah pemrograman.

## 1.3 Manfaat Praktikum

Manfaat dilakukannya praktikum ini adalah sebagai berikut :

1. Mahasiswa mampu membuat program yang lebih terstruktur dan efisien menggunakan *method*.
2. Mahasiswa dapat mengolah data teks dengan benar menggunakan berbagai fungsi *string*.
3. Memperkuat dasar logika pemrograman yang dibutuhkan untuk materi lanjutan dalam Java.
4. Membantu mahasiswa terbiasa membangun program yang mudah dikembangkan, diperbaiki, dan dikelola.

## BAB II

### PEMBAHASAN

#### 2.1 Kode Program

##### 2.1.1 BilanganPrima

```

1 package Pekan7_2511531016;
2 import java.util.Scanner;
3 public class BilanganPrima_2511531016 {
4
5     public static boolean isPrime(int n) {
6         int factors = 0;
7         for (int i = 1; i <= n; i++) {
8             if (n % i == 0) {
9                 factors++;
10            }
11        }
12        return (factors == 2);
13    }
14
15    public static void main(String[] args) {
16        Scanner input = new Scanner(System.in);
17        System.out.print("Input nilai n = ");
18        int a = input.nextInt();
19        if (isPrime(a)) {
20            System.out.println(a + " bilangan prima");
21        } else {
22            System.out.println(a + " bukan bilangan prima");
23        }
24    }
25 }
26 }

```

#### Kode Program 2.1 (Kode Program BilanganPrima)

##### Uraian Kode Program :

1. *package Pekan7\_2511531016;*  
→ Menunjukkan bahwa program berada dalam paket bernama *Pekan7\_2511531016*.
2. *import java.util.Scanner;*  
→ Mengimpor kelas *Scanner* untuk membaca input dari *keyboard*.
3. *public class BilanganPrima\_2511531016 {*  
→ Mendefinisikan kelas utama bernama *BilanganPrima\_2511531016*.
4. *public static boolean isPrime(int n) { ... }*  
→ Membuat metode *isPrime* untuk menentukan apakah suatu angka merupakan bilangan prima. Di dalamnya terdapat variabel *factors* sebagai penghitung pembagi dan perulangan *for (int i = 1; i <= n; i++)*



yang mengecek setiap angka dari 1 hingga n. Jika  $n \% i == 0$ , maka jumlah faktor ditambah. Metode mengembalikan *true* jika jumlah faktor tepat 2.

5. *public static void main(String[] args) { ... }*

→ Metode utama yang dijalankan pertama kali saat program dieksekusi.

6. *Scanner input = new Scanner(System.in);*

→ Membuat objek *Scanner* untuk membaca input pengguna.

7. *System.out.print("Input nilai n = "); int a = input.nextInt();*

→ Menampilkan permintaan input lalu menyimpan nilai yang dimasukkan pengguna ke variabel a.

8. *if (isPrime(a)) ... else ...*

→ Mengecek apakah nilai a adalah bilangan prima dengan memanggil metode *isPrime*. Jika benar, program menampilkan “a bilangan prima”, jika tidak program menampilkan “a bukan bilangan prima”.

### 2.1.2 Mahasiswa

```

1 package Pekan7_2511531016;
2
3 public class Mahasiswa2511531016 {
4     //variabel global
5     private int nim;
6     private String nama, nim2;
7     //membuat mutator (setter)
8     public void setNim (int nim) {
9         this.nim=nim;
10    }
11    public void setNim2 (String nim2) {
12        this.nim2=nim2;
13    }
14    public void setName (String nama) {
15        this.nama=nama;
16    }
17    //membuat accessor (getter)
18    public int getNim() {
19        return nim;
20    }
21    public String getNim2() {
22        return nim2;
23    }
24    public String getName() {
25        return nama;
26    }
27    //metode lain
28    public void Cetak() {
29        System.out.println("Nim : "+nim);
30        System.out.println("Nama : "+nama);
31    }
32    public void Cetak2() {
33        System.out.println("Nim : "+nim2);
34        System.out.println("Nama : "+nama);
35    }
36 }

```

**Kode Program 2.2 (Kode Program Mahasiswa)**

### Uraian Kode Program :

1. *package Pekan7\_2511531016;*  
→ Menunjukkan bahwa program berada dalam paket bernama *Pekan7\_2511531016*.
2. *public class Mahasiswa\_2511531016 {*  
→ Mendefinisikan kelas utama bernama *Mahasiswa\_2511531016*.
3. *private int nim; private String nama, nim2;*  
→ Mendeklarasikan variabel global yang bersifat *private*, yaitu *nim*, *nama*, dan *nim2*, sehingga hanya bisa diakses lewat metode *setter* dan *getter*.
4. *public void setNim(), setNim2(), setNama()*  
→ Merupakan metode *mutator (setter)* untuk mengubah nilai atribut *nim*, *nim2*, dan *nama*. Masing-masing menggunakan kata kunci *this* agar jelas bahwa yang diubah adalah variabel milik objek.
5. *public int getNim(), public String getNim2(), getNama()*  
→ Merupakan metode *accessor (getter)* yang berfungsi untuk mengambil/mengembalikan nilai dari atribut *nim*, *nim2*, dan *nama*.
6. *public void Cetak() dan public void Cetak2()*  
→ Metode tambahan untuk menampilkan data mahasiswa ke layar. *Cetak()* menampilkan *nim* dan *nama*, sedangkan *Cetak2()* menampilkan *nim2* dan *nama*.

### 2.1.3 PanggilMahasiswa

```

1 package Pekan7_2511531016;
2
3 public class PanggilMahasiswa_2511531016 {
4
5     public static void main(String[] args) {
6         Mahasiswa_2511531016 a= new Mahasiswa_2511531016();
7         a.setNim(1016);
8         a.setNama("Kinaya");
9         System.out.println(a.getNim());
10        System.out.println(a.getNama());
11        a.Cetak();
12    }
13
14 }
```

### Kode Program 2.3 (Kode Program PanggilMahasiswa)

### Uraian Kode Program :

1. *package Pekan7\_2511531016;*  
→ Menunjukkan bahwa program berada dalam paket bernama *Pekan7\_2511531016*.
2. *public class PanggilMahasiswa\_2511531016 {*  
→ Mendefinisikan kelas utama bernama *PanggilMahasiswa\_2511531016*.
3. *public static void main(String[] args)*  
→ Metode utama yang pertama kali dijalankan ketika program dieksekusi.
4. *Mahasiswa\_2511531016 a = new Mahasiswa\_2511531016();*  
→ Membuat objek baru bernama *a* dari kelas *Mahasiswa\_2511531016* sehingga dapat digunakan untuk memanggil *setter*, *getter*, dan metode lain.
5. *a.setNim(1016);* dan *a.setNama("Kinaya");*  
→ Mengisi atau menetapkan nilai *nim* dan *nama* untuk objek *a* menggunakan metode *setter*.
6. *System.out.println(a.getNim());* dan *System.out.println(a.getNama());*  
→ Menampilkan nilai *nim* dan *nama* yang telah disimpan melalui metode *getter*.
7. *a.Cetak();*  
→ Memanggil metode *Cetak()* dari kelas *Mahasiswa* untuk menampilkan kembali data mahasiswa ke layar.

### 2.1.4 PanggilMahasiswa2

```

1 package Pekan7_2511531016;
2 import java.util.Scanner;
3 public class PanggilMahasiswa2_2511531016 {
4
5     public static void main(String[] args) {
6         Scanner input= new Scanner(System.in);
7         System.out.print("NIM: ");
8         String x= input.nextLine();
9         System.out.print("Nama: ");
10        String y= input.nextLine();
11        Mahasiswa_2511531016 a= new Mahasiswa_2511531016();
12        a.setNim2(x);
13        a.setNama(y);
14        if(x.startsWith("25")) {
15            System.out.println(y+ " anda angkatan 2025");
16        }
17        if(x.contains("1153")) {
18            System.out.println("Anda Mahasiswa Informatika");
19        }
20        a.Cetak2();
21        input.close();
22    }
23 }
24 }

```

**Kode Program 2.4 (Kode Program PanggilMahasiswa2)**

#### Uraian Kode Program :

1. *package Pekan7\_2511531016;*  
→ Menunjukkan bahwa program berada dalam paket *Pekan7\_2511531016*.
2. *import java.util.Scanner;* dan *public class PanggilMahasiswa2\_2511531016*  
→ Mengimpor *Scanner* untuk membaca input dari pengguna dan mendefinisikan kelas utama program.
3. *public static void main(String[] args)*  
→ Metode utama yang dijalankan pertama kali.
4. *Scanner input = new Scanner(System.in);*  
*x = input.nextLine(); y = input.nextLine();*  
→ Membaca input NIM dan Nama dari pengguna.
5. *Mahasiswa\_2511531016 a = new Mahasiswa\_2511531016();*  
*a.setNim2(x); a.setNama(y);*  
→ Membuat objek mahasiswa dan mengisi data NIM serta Nama menggunakan *setter*.

6. *if (x.startsWith("25")) ... dan if (x.contains("1153")) ...*

→ Mengecek pola tertentu pada NIM: “25” yaitu angkatan 2025 dan “1153” yaitu mahasiswa Informatika.

7. *a.Cetak2(); input.close();*

→ Menampilkan data mahasiswa dan menutup input.

### 2.1.5 String1

```

1 package Pekan7_2511531016;
2 public class String1_2511531016 {
3
4     public static void main(String[] args) {
5         String salam = "Assalamualaikum";
6         System.out.println("panjang salam adalah: " + salam.length());
7         System.out.println(salam.toUpperCase()); // Outputs "ASSALAMUALAIKUM"
8         System.out.println(salam.toLowerCase()); // Outputs "assalamualaikum"
9         System.out.println(salam.indexOf("salam")); // Outputs "ASSALAMUALAIKUM"
10    }
11
12 }
```

#### Kode Program 2.5 (Kode Program String1)

#### Uraian Kode Program :

1. *package Pekan7\_2511531016;*

→ Menunjukkan bahwa program berada dalam paket bernama *Pekan7\_2511531016*.

2. *public class String1\_2511531016* dan *public static void main(String[] args)*

→ Mendefinisikan kelas utama program dan metode utama yang dijalankan pertama kali.

3. *String salam = "Assalamualaikum";*

→ Membuat variabel string bernama *salam* dengan nilai “Assalamualaikum”.

4. *System.out.println(...)*

→ Menampilkan berbagai hasil operasi string, yaitu:

- *salam.length()* → menampilkan panjang teks

- *salam.toUpperCase()* → mengubah teks menjadi huruf besar
- *salam.toLowerCase()* → mengubah teks menjadi huruf kecil
- *salam.indexOf("salam")* → mencari posisi kata “salam” dalam *string*

### 2.1.6 String2

```

1 package Pekan7_2511531016;
2 public class String2_2511531016 {
3
4     public static void main(String[] args) {
5         String firstName = "Kinaya";
6         String lastName = "Novrya";
7         String txt1 = "Dosen\intelektual\" kampus";
8         System.out.println("Nama Lengkap: "+firstName + " " + lastName);
9         System.out.println("Nama Lengkap: "+firstName.concat(lastName));
10        System.out.println(txt1);
11        int x = 10;
12        int y = 20;
13        int z = x + y;
14        System.out.println("x+ y= "+z);
15        String a = "10";
16        String b = "20";
17        String c = a + b;
18        System.out.println("String a + string b =" +c);
19        String v = a + y;
20        System.out.println("String a + integer y =" +v);
21    }
22 }
23

```

**Kode Program 2.6 (Kode Program String2)**

#### Uraian Kode Program :

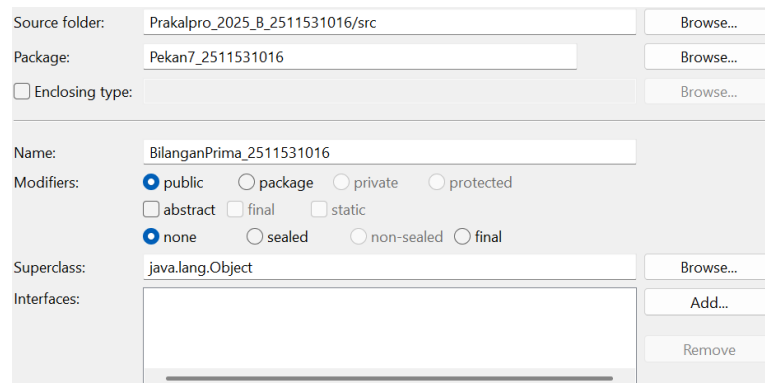
1. *package Pekan7\_2511531016;*  
→ Menunjukkan bahwa program berada dalam paket bernama *Pekan7\_2511531016*.
2. *public class String2\_2511531016*  
→ Mendefinisikan kelas utama bernama *String2\_2511531016*.
3. *public static void main(String[] args)*  
→ Merupakan metode utama yang dijalankan pertama kali saat program dieksekusi.
4. *String firstName = "Kinaya"; String lastName = "Novrya";*  
→ Membuat dua variabel string untuk menyimpan nama depan dan nama belakang.

5. `String txt1 = "Dosen\"intelektual\" kampus";`  
 → Membuat string yang berisi tanda kutip menggunakan karakter *escape* `\`.
6. `System.out.println("Nama Lengkap: " + firstName + " " + lastName);`  
 → Menampilkan nama lengkap dengan menggabungkan *string* menggunakan operator `+`.
7. `System.out.println("Nama Lengkap: " + firstName.concat(lastName));`  
 → Menggabungkan nama depan dan nama belakang menggunakan fungsi `.concat()`.
8. `System.out.println(txt1);`  
 → Menampilkan teks yang berisi tanda kutip.
9. `int x = 10; int y = 20; int z = x + y;`  
 → Mendeklarasikan variabel *integer* dan melakukan operasi penjumlahan angka secara matematis.
10. `System.out.println("x + y = " + z);`  
 → Menampilkan hasil penjumlahan dua bilangan *integer*.
11. `String a = "10"; String b = "20"; String c = a + b;`  
 → Menggabungkan dua *string* berisi angka menggunakan operator `+` hasilnya menjadi "1020" (bukan operasi matematika).
12. `System.out.println("String a + string b = " + c);`  
 → Menampilkan hasil penggabungan dua *string* angka.
13. `String v = a + y;` → Menggabungkan string "10" dengan integer 20, menghasilkan "1020" setelah integer otomatis dikonversi menjadi string.
14. `System.out.println("String a + integer y = " + v);`  
 → Menampilkan hasil penggabungan *string* dan *integer*.

## 2.2 Langkah Kerja

### 2.2.1 BilanganPrima

1. Membuat kelas baru bernama *BilanganPrima* dalam *package* *Pekan7\_2511531016*.



Source folder: Prakalpro\_2025\_B\_2511531016/src Browse...

Package: Pekan7\_2511531016 Browse...

☐ Enclosing type: Browse...

Name: BilanganPrima\_2511531016

Modifiers: ☒ public ☐ package ☐ private ☐ protected  
☐ abstract ☐ final ☐ static  
☒ none ☐ sealed ☐ non-sealed ☐ final

Superclass: java.lang.Object Browse...

Interfaces: Add... Remove

**Gambar 2.1 (Langkah 1 BilanganPrima)**

2. Buat variabel untuk menghitung jumlah pembagi dari suatu angka. Lakukan mengecek pembagi mulai dari 1 sampai n, jika n habis dibagi i maka faktor bertambah. Jika jumlah faktor hanya 2 (1 dan dirinya sendiri), maka angka dianggap bilangan prima.

```

1 package Pekan7_2511531016;
2 import java.util.Scanner;
3 public class BilanganPrima_2511531016 {
4
5     public static boolean isPrime(int n) {
6         int factors = 0;
7         for (int i = 1; i <= n; i++) {
8             if (n % i == 0) {
9                 factors++;
10            }
11        }
12        return (factors == 2);
13    }
14 }

```

**Gambar 2.2 (Langkah 2 BilanganPrima)**



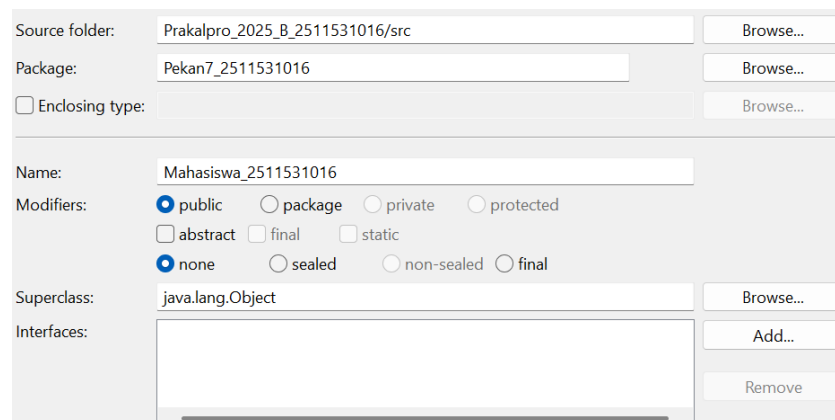
3. Menerima input angka dari pengguna dan mengecek apakah angka tersebut prima menggunakan fungsi *isPrime*.

```
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    System.out.print("Input nilai n = ");
    int a = input.nextInt();
    if (isPrime(a)) {
        System.out.println(a + " bilangan prima");
    } else
        System.out.println(a + " bukan bilangan prima");
}
```

**Gambar 2.3 (Langkah 3 BilanganPrima)**

### 2.2.2 Mahasiswa

1. Membuat kelas baru bernama *Mahasiswa* dalam package *Pekan7\_2511531016*.



Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected  
☐ abstract ☐ final ☐ static  
☒ none ☐ sealed ☐ non-sealed ☐ final

Superclass:

Interfaces:

**Gambar 2.4 (Langkah 1 Mahasiswa)**

2. Mendefinisikan variabel yang hanya bisa diakses lewat *setter* dan *getter*.

```
//variabel global
private int nim;
private String nama, nim2;
```

**Gambar 2.5 (Langkah 2 PerulanganFor2)**

3. Buat *mutator* (*setter*) lalu isi atau ubah nilai nim, nim2, nama mahasiswa.

```
//membuat mutator (setter)
public void setNim (int nim) {
    this.nim=nim;
}
public void setNim2 (String nim2) {
    this.nim2=nim2;
}
public void setNama (String nama) {
    this.nama=nama;
}
```

**Gambar 2.6 (Langkah 3 Mahasiswa)**

4. Buat *accessor* (*getter*) lalu ambil atau baca nilai nim, nim2, nama mahasiswa.

```
//membuat accessor (getter)
public int getNim() {
    return nim;
}
public String getNim2() {
    return nim2;
}
public String getNama() {
    return nama;
}
```

**Gambar 2.7 (Langkah 4 Mahasiswa)**

5. Metode lain yaitu tampilkan data mahasiswa ke layar.

```
//metode lain
public void Cetak() {
    System.out.println("Nim : "+nim);
    System.out.println("Nama : "+nama);
}
public void Cetak2() {
    System.out.println("Nim : "+nim2);
    System.out.println("Nama : "+nama);
}
```

**Gambar 2.8 (Langkah 5 Mahasiswa)**

### 2.2.3 PanggilMahasiswa

1. Membuat kelas baru bernama *PanggilMahasiswa* dalam *package* *Pekan\_2511531016*.

Source folder:

Package:

☐ Enclosing type:

---

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected  
☐ abstract ☐ final ☐ static  
☒ none ☐ sealed ☐ non-sealed ☐ final

Superclass:

Interfaces:

**Gambar 2.9 (Langkah 1 PanggilMahasiswa)**

2. Program dijalankan dan masuk ke *main*.

```
public static void main(String[] args) {
```

**Gambar 2.10 (Langkah 2 PanggilMahasiswa)**

3. Buat objek mahasiswa agar bisa menggunakan setter, getter dan metode lainnya. Lalu, isi nim dan nama objek mahasiswa.

```
Mahasiswa_2511531016 a= new Mahasiswa_2511531016();
a.setNim(1016);
a.setNama("Kinaya");
```

**Gambar 2.11 (Langkah 3 PanggilMahasiswa)**

4. Tampilkan nim dan nama yang disimpan dalam objek, lalu cetak untuk menampilkan data mahasiswa lengkap.

```
System.out.println(a.getNim());
System.out.println(a.getNama());
a.Cetak();
```

**Gambar 2.12 (Langkah 4 PanggilMahasiswa)**

### 2.2.4 PanggilMahasiswa2

1. Membuat kelas baru bernama *PanggilMahasiswa2* dalam *package* *Pekan7\_2511531016*.

Source folder:

Package:

☐ Enclosing type:

---

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected  
☐ abstract ☐ final ☐ static  
☒ none ☐ sealed ☐ non-sealed ☐ final

Superclass:

Interfaces:

**Gambar 2.13 (Langkah 1 PanggilMahasiswa2)**

2. Program dijalankan dan masuk ke *main*.

```
public static void main(String[] args) {
```

**Gambar 2.14 (Langkah 2 PanggilMahasiswa2)**

3. Menerima input NIM dan Nama dari pengguna menggunakan *Scanner* input.

```
Scanner input= new Scanner(System.in);
System.out.print("NIM: ");
String x= input.nextLine();
System.out.print("Nama: ");
String y= input.nextLine();
```

**Gambar 2.15 (Langkah 3 PanggilMahasiswa2)**

4. Simpan NIM dan Nama kedalam objek mahasiswa.

```
Mahasiswa_2511531016 a= new Mahasiswa_2511531016();
a.setNim2(x);
a.setNama(y);
```

**Gambar 2.16 (Langkah 4 PanggilMahasiswa2)**

5. Cek apakah NIM dimulai dengan “25”, yang berarti angkatan 2025 dan cek apakah NIM berisi kode prodi Informatika. Lalu tampilkan data mahasiswa melalui metode *Cetak2* dan tutup *keyboard*.

```
if(x.startsWith("25")) {
    System.out.println(y+ " anda angkatan 2025");
}
if(x.contains("1153")) {
    System.out.println("Anda Mahasiswa Informatika");
}
a.Cetak2();
input.close();
```

**Gambar 2.17 (Langkah 5 PanggilMahasiswa2)**

### 2.2.5 String1

1. Membuat kelas baru bernama *String1* dalam *package* *Pekan7\_2511531016*.

The screenshot shows the 'New Class' dialog box. The 'Source folder' field contains 'Prakalpro\_2025\_B\_2511531016/src'. The 'Package' field contains 'Pekan7\_2511531016'. The 'Name' field contains 'String1\_2511531016'. Under 'Modifiers', the 'public' radio button is selected. Other options include 'package', 'private', 'protected', 'abstract', 'final', 'static', 'none', 'sealed', 'non-sealed', and 'final'. The 'Superclass' field contains 'java.lang.Object'. There are 'Browse...', 'Add...', and 'Remove' buttons on the right side.

**Gambar 2.18 (Langkah 1 String1)**

2. Program dijalankan dan masuk ke *main*.

```
public static void main(String[] args) {
```

**Gambar 2.19 (Langkah 2 String1)**

3. Buat variabel *string* untuk melakukan berbagai operasi teks. Lalu, hitung dan tampilkan panjang teks.

```
String salam = "Assalamualaikum";
System.out.println("panjang salam adalah: " + salam.length());
```

**Gambar 2.20 (Langkah 3 String1)**

4. Mengubah seluruh huruf menjadi huruf kapital, huruf kecil, dan mencari kata “salam” di dalam teks.

```
System.out.println(salam.toUpperCase()); // Outputs "ASSALAMUALAIKUM"
System.out.println(salam.toLowerCase()); // Outputs "assalamualaikum"
System.out.println(salam.indexOf("salam")); // Outputs "ASSALAMUALAIKUM"
```

**Gambar 2.21 (Langkah 4 String1)**

## 2.2.6 String2

1. Membuat kelas baru bernama *String2* dalam *package* *Pekan7\_2511531016*.

**Gambar 2.22 (Langkah 1 String2)**

2. Program dijalankan dan masuk ke *main*.

```
public static void main(String[] args) {
```

**Gambar 2.23 (Langkah 2 String2)**

3. Simpan nama depan dan nama belakang dalam bentuk *String*. Gunakan *escape character* \” untuk menampilkan tanda kutip didalam teks. Lalu, gabungkan dua *string* menggunakan *operator* + dan tampilkan. Gabungkan nama depan dan belakang menggunakan *method* .concat(). Tampilkan teks yang sudah menggunakan *escape character*.

```
String firstName = "Kinaya";
String lastName = "Novrya";
String txt1 = "Dosen\"intelektual\" kampus";
System.out.println("Nama Lengkap: "+firstName + " " + lastName);
System.out.println("Nama Lengkap: "+firstName.concat(lastName));
System.out.println(txt1);
```

**Gambar 2.24 (Langkah 3 String2)**

4. Jumlahkan dua bilangan bulan (10 + 20) dan tampilkan.

```
int x = 10;
int y = 20;
int z = x + y;
System.out.println("x+ y= "+z);
```

**Gambar 2.25 (Langkah 4 String2)**

5. Buktikan bahwa *string* tidak dijumlahkan seperti angka, melainkan digabungkan, sehingga hasilnya menjadi “1020”. Ketika *string* digabung dengan *integer*, Java otomatis mengubah *integer* menjadi *string*, sehingga “10” + 20 menghasilkan “1020”.

```
String a = "10";
String b = "20";
String c = a + b;
System.out.println("String a + string b =" +c);
String v = a + y;
System.out.println("String a + integer y =" +v);
```

**Gambar 2.26 (Langkah 5 String2)**



### 2.3 Analisis Hasil

Berdasarkan hasil percobaan dari keenam kode program yang telah dijalankan, dapat dilihat bahwa semuanya masih berkaitan dengan materi dasar pemrograman Java, seperti perulangan, *operator*, *method*, dan pengolahan *string*. Secara umum, tiap program berjalan sesuai teori yang ada di materi praktikum maupun referensi Java. Hampir semua program dieksekusi dari metode *main()*, sehingga urutan prosesnya mengikuti apa yang dituliskan di dalam blok kode tersebut.

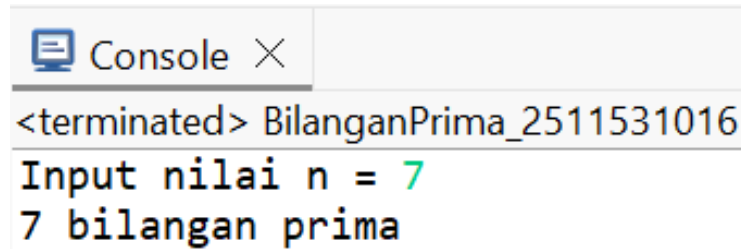
Pada beberapa program pertama, bagian perulangan (*for*, *while*, dan *do-while*) menunjukkan bagaimana Java akan mengulang suatu perintah selama kondisi yang diberikan masih terpenuhi. Hal ini sesuai dengan teori kontrol perulangan di Java, di mana inisialisasi, pengecekan kondisi, dan perubahan nilai variabel dilakukan secara berurutan. Hasil output dari program juga memperlihatkan bahwa setiap iterasi berjalan sesuai logika yang ditulis. Misalnya, ketika kondisi sudah tidak memenuhi syarat, perulangan otomatis berhenti. Ini membuktikan bahwa cara kerja perulangan di Java sesuai dengan dasar teori yang ada di modul dan dokumentasi Java.

Pada program terakhir yang membahas tentang *string* dan operasi dasar, terlihat bahwa Java memperlakukan *string* sebagai objek yang tidak bisa diubah begitu saja (*immutable*). Penggabungan *string* menggunakan *+* maupun *.concat()* menghasilkan objek *string* baru, bukan mengubah *string* lama. Selain itu, program juga menunjukkan perbedaan antara penjumlahan angka dan penggabungan *string*. Ketika angka dijumlahkan, hasilnya mengikuti operasi matematika. Tapi ketika angka dan *string* digabungkan, maka angka tersebut otomatis berubah menjadi *string*. Hasil output seperti “1020” membuktikan bahwa operator *+* pada *string* bukan menambah angka, tetapi menyatukan teks.

## BAB III KESIMPULAN

### 3.1 Hasil Praktikum

#### 3.1.1 BilanganPrima



```

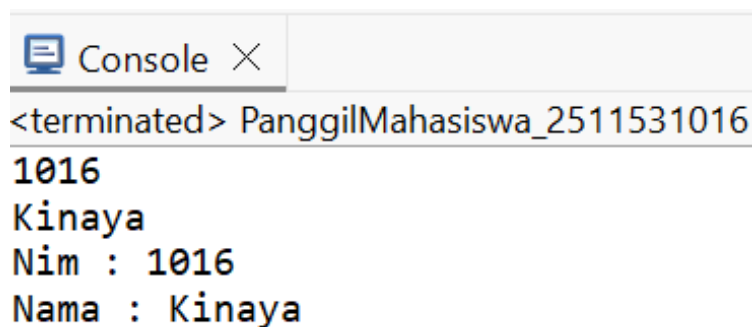
Console X
<terminated> BilanganPrima_2511531016
Input nilai n = 7
7 bilangan prima

```

**Gambar 3.1 (Hasil Praktikum BilanganPrima)**

Output diatas menampilkan Input nilai n, dimana program meminta pengguna memasukan nilai n, disini pengguna memasukan angka 7 lalu program menampilkan “7 bilangan prima”. Ini berarti program dibuat untuk memeriksa apakah angka yang dimasukkan pengguna merupakan bilangan prima atau bukan.

#### 3.1.2 PanggilMahasiswa



```

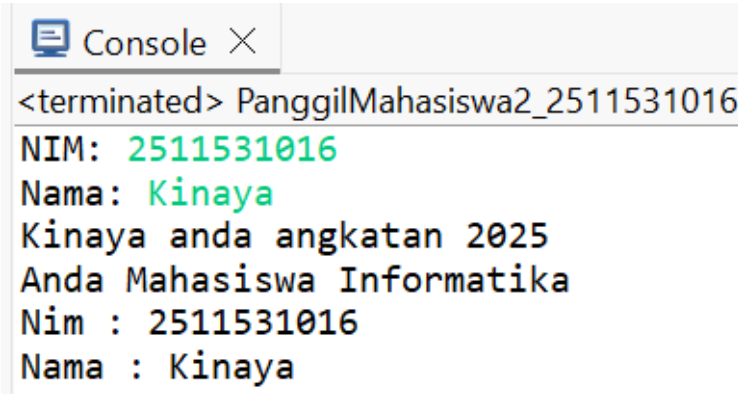
Console X
<terminated> PanggilMahasiswa_2511531016
1016
Kinaya
Nim : 1016
Nama : Kinaya

```

**Gambar 3.2 (Hasil Praktikum PanggilMahasiswa)**

Output di atas menampilkan data mahasiswa berupa NIM dan Nama, yaitu “1016” dan “Kinaya”, baik secara langsung maupun dalam format yang sudah diberi label di konsol. Program ini kemungkinan dibuat agar data identitas mahasiswa dapat muncul secara terstruktur, di mana NIM dan Nama dicetak satu per satu, lalu ditampilkan kembali dengan format “Nim : xxx” dan “Nama : xxx”, sehingga hasil output mudah dipahami sesuai dengan informasi dasar yang ingin disampaikan.

### 3.1.3 PanggilMahasiswa2



```

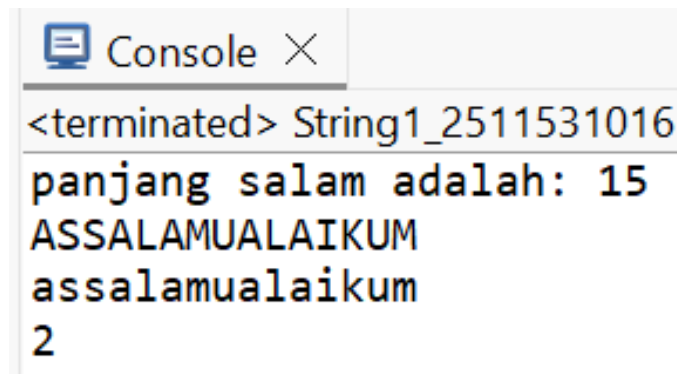
Console X
<terminated> PanggilMahasiswa2_2511531016
NIM: 2511531016
Nama: Kinaya
Kinaya anda angkatan 2025
Anda Mahasiswa Informatika
Nim : 2511531016
Nama : Kinaya

```

**Gambar 3.3 (Hasil Praktikum PanggilMahasiswa2)**

Output di atas menampilkan informasi lengkap mengenai seorang mahasiswa dengan NIM “2511531016” dan nama “Kinaya”, dimana program meminta pengguna memasukan nim dan nama pada konsol, maka akan menampilkan tambahan keterangan seperti angkatan tahun 2025 serta status sebagai mahasiswa Informatika, sehingga program ini dibuat untuk memperlihatkan data identitas beserta atribut tambahan secara berurutan dan terstruktur agar pengguna dapat menerima informasi mahasiswa secara jelas sesuai dengan logika dan urutan yang telah ditentukan di dalam program.

### 3.1.4 String1

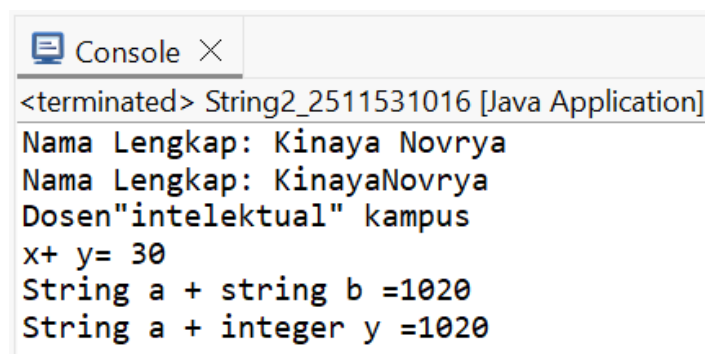


```
<terminated> String1_2511531016
panjang salam adalah: 15
ASSALAMUALAIKUM
assalamualaikum
2
```

**Gambar 3.4 (Hasil Praktikum String1)**

Output di atas mendemonstrasikan beberapa fungsi *string*. Pertama, menampilkan panjang (jumlah karakter) dari *string* "salam", yaitu 15; kedua, menampilkan *string* yang sama dalam format huruf kapital semua (ASSALAMUALAIKUM) melalui fungsi *toUpperCase()*; ketiga, menampilkan *string* yang sama dalam format huruf kecil semua (assalamualaikum) melalui fungsi *toLowerCase()*; dan terakhir, menampilkan hasil angka 2, yang merupakan hasil dari operasi lain, yaitu mencari posisi kata "salam" (menggunakan *indexOf()*) karakter dalam *string* tersebut.

### 3.1.5 String2



```
<terminated> String2_2511531016 [Java Application]
Nama Lengkap: Kinaya Novrya
Nama Lengkap: KinayaNovrya
Dosen"intelektual" kampus
x+ y= 30
String a + string b =1020
String a + integer y =1020
```

**Gambar 3.5 (Hasil Praktikum String2)**

Output ini secara berurutan menunjukkan penggabungan *string* yang mencantumkan nama lengkap dengan dan tanpa spasi (Kinaya Novrya, KinayaNovrya), sebuah *string literal* yang berisi tanda kutip (Dosen"intelektual" kampus), hasil dari penjumlahan dua variabel numerik ( $x + y = 30$ ), dan dua contoh hasil dari penggabungan *string* dengan angka di mana penggabungan *string* a dan *string* b menghasilkan 1020, serta penggabungan *string* a dengan *integer* y juga menghasilkan 1020, yang menunjukkan bahwa operasi penjumlahan dalam konteks ini diutamakan sebagai operasi penggabungan string.

### 3.2 Saran Pengembangan

Berdasarkan hasil analisis dari keenam kode program *method* dan *string* yang telah dijalankan, terdapat beberapa saran pengembangan agar program menjadi lebih interaktif, efisien, dan bermanfaat dalam penerapan nyata.

Pertama, bisa menambahkan input dari pengguna menggunakan *Scanner*, sehingga data yang ditampilkan tidak hanya bersifat statis tapi lebih interaktif. Selain itu, pada program perulangan, dapat menambahkan validasi atau kondisi tambahan agar hasil *looping* lebih variatif, misalnya menampilkan angka ganjil atau genap saja. Pada program yang menggunakan *string*, dapat mencoba berbagai *method string* lainnya seperti *replace()*, *substring()*, atau *equalsIgnoreCase()* untuk memperluas pemahaman mengenai manipulasi teks di Java.

Dari sisi struktur kode, program juga bisa dikembangkan dengan membuat *method* khusus agar tidak semua perintah dituliskan langsung di dalam *main()*. Hal ini akan membuat program lebih rapi dan mudah dibaca. Terakhir, juga bisa menambahkan komentar pada kode untuk menjelaskan fungsi bagian-bagian tertentu agar lebih mudah dipahami saat dipelajari kembali. Dengan pengembangan-pengembangan sederhana seperti ini, program akan terlihat lebih baik dan mencerminkan pemahaman dasar pemrograman yang semakin meningkat.

## DAFTAR PUSTAKA

- [1] Oracle, “*Java Platform, Standard Edition Documentation*,” Oracle.com.  
<https://docs.oracle.com/javase/> (diakses Nov. 2024).
- [2] P. Deitel dan H. Deitel, *Java: How to Program*, 11th ed. New Jersey: Pearson, 2018.
- [3] W3Schools, “*Java Tutorial*,” W3schools.com.  
<https://www.w3schools.com/java/> (diakses Nov. 2024).
- [4] J. Bloch, *Effective Java*, 3rd ed. Addison-Wesley, 2018.
- [5] S. Malik, *Core Java: Concepts and Applications*. New York: McGraw-Hill, 2019.