

LAPORAN PRAKTIKUM
ALGORITMA PEMROGRAMAN PEKAN TUJUH

String

disusun Oleh:

Zahira Nur Asyifa

2411532015

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

Asisten Praktikum: Rahmad Dwirizki Olders



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS

2025

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT karena atas rahmat dan karunia-Nya, laporan praktikum Algoritma dan Pemrograman dengan topik String, Class & Object, Method, serta penerapan logika validasi menggunakan bahasa Java ini dapat diselesaikan dengan baik. Laporan ini disusun untuk memenuhi tugas praktikum pekan keenam mata kuliah praktikum Algoritma dan Pemrograman, sekaligus sebagai sarana pembelajaran dalam memahami konsep dasar string menggunakan bahasa Java.

Melalui praktikum ini, penulis mempelajari cara memanfaatkan berbagai operasi String, penggunaan class sebagai blueprint data, penerapan method termasuk setter–getter, serta pembuatan method tambahan untuk validasi seperti pengecekan format email dan panjang password. Penulis menyadari bahwa laporan ini masih jauh dari sempurna, baik dari segi penjelasan maupun penyajian. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa mendatang. Akhir kata, penulis mengucapkan terima kasih kepada dosen pengampu, asisten praktikum, serta semua pihak yang telah memberikan bimbingan dan dukungan sehingga laporan ini dapat terselesaikan. Semoga laporan ini dapat memberikan manfaat bagi penulis sendiri maupun pembaca.

Padang, 2025

Penulis

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
DAFTAR GAMBAR	iii
BAB 1 PENDAHULUAN	4
1.1 Latar Belakang.....	4
1.2 Tujuan	4
1.3 Manfaat	4
BAB II PEMBAHASAN.....	5
2.1 Class Bilangan Prima.....	5
2.2 Class Mahasiswa	8
2.3 Class PanggilMahasiswa.....	11
2.4 Class String1	15
2.5 Class String2.....	16
BAB III KESIMPULAN.....	19
3.1 Kesimpulan	19
3.2 Saran	19
DAFTAR PUSTAKA.....	20

DAFTAR GAMBAR

Gambar 2.1 Membuat package baru	5
Gambar 2. 2 Membuat nama package.....	5
Gambar 2. 3 Membuat <i>class</i> BilanganPrima	6
Gambar 2. 4 <i>Syntax Class BilanganPrima</i>	7
<i>Gambar 2. 5 Penjelasan syntax</i>	7
<i>Gambar 2. 6 Penjelasan syntax</i>	8
<i>Gambar 2. 7 Ouput BilanganPrima</i>	8
<i>Gambar 2. 8 Class Mahasiswa</i>	8
<i>Gambar 2. 9 Syntax Mahasiswa</i>	9
<i>Gambar 2. 10 Penjelasan syntax</i>	9
<i>Gambar 2. 11 Penjelasan syntax</i>	10
<i>Gambar 2. 12 Penjelasan syntax</i>	10
<i>Gambar 2. 13 Class PanggilMahasiswa</i>	11
<i>Gambar 2. 14 Syntax PanggilMahasiswa</i>	11
<i>Gambar 2. 15 Penjelasan syntax</i>	12
<i>Gambar 2. 16 Output PanggilMahasiswa</i>	12
Gambar 2. 17 Class PanggilMahasiswa2	12
Gambar 2. 18 Syntax PanggilMahasiswa2	13
Gambar 2. 19 Penjelasan Syntax.....	13
Gambar 2. 20 Penjelasan Syntax.....	14
Gambar 2. 21 Output PanggilMahasiswa2	14
<i>Gambar 2. 22 Class String1</i>	15
<i>Gambar 2. 23 Syntax String1</i>	15
<i>Gambar 2. 24 Penjelasan syntax</i>	16
<i>Gambar 2. 25 Output String1</i>	16
<i>Gambar 2. 26 Class string2</i>	16
<i>Gambar 2. 27 Syntax String2</i>	17
<i>Gambar 2. 28 Penjelasan syntax</i>	17
<i>Gambar 2. 29 Penjelasan syntax</i>	18
Gambar 2. 30 Penjelasan Syntax.....	18
<i>Gambar 2. 31 Output String2</i>	18

BAB 1 PENDAHULUAN

1.1 Latar Belakang

String pada java adalah object yang digunakan untuk menyimpan teks. Variabel String berisi kumpulan karakter yang berada dalam kutip dua (” “). String sebenarnya merupakan sebuah object yang mengandung method yang bisa melakukan beberapa operasi. Sebagai contoh, method `length()` digunakan untuk menghitung panjang dari string, `toUpperCase()` dan `toLowerCase()` untuk mengubah string ke huruf kapital atau huruf kecil, method `indexOf()` mengembalikan posisi index karakter yang muncul pertama kali pada sebuah teks dan lainnya.

1.2 Tujuan

1. Mempelajari prinsip enkapsulasi menggunakan atribut private dan method setter/getter.
2. Mengetahui cara mengimplementasikan operasi dan manipulasi String

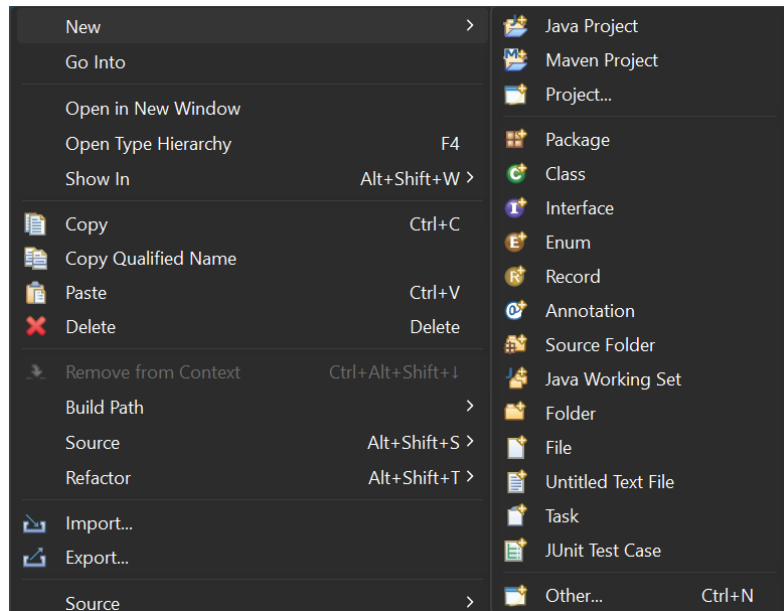
1.3 Manfaat

1. Mahasiswa dapat me tentang cara menggunakan method untuk pemrosesan data dan validasi.
2. Mahasiswa memahami perbedaan antara operasi aritmetika dan operasi string.

BAB II PEMBAHASAN

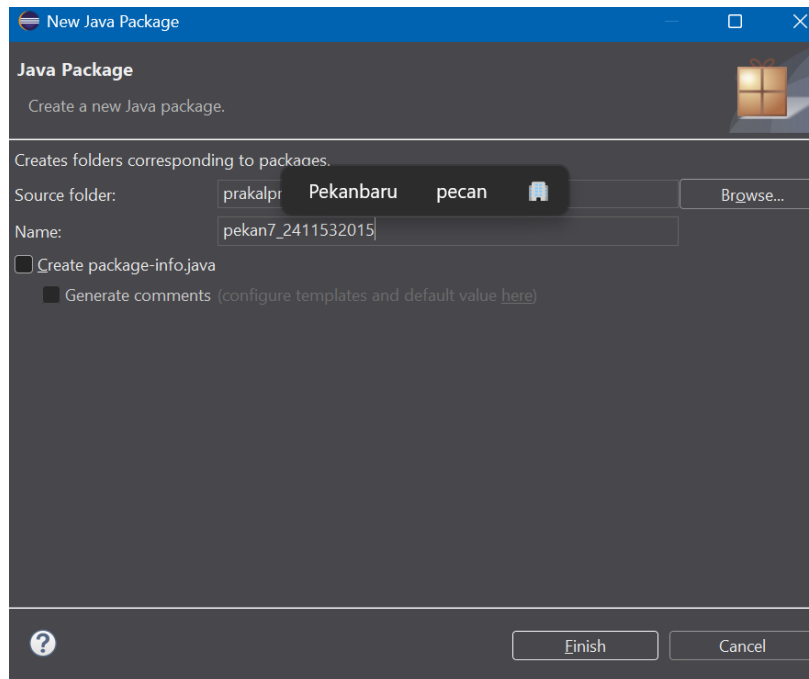
2.1 Class Bilangan Prima

1. Buat *package* baru dengan klik kanan pada src, pilih new lalu *package*



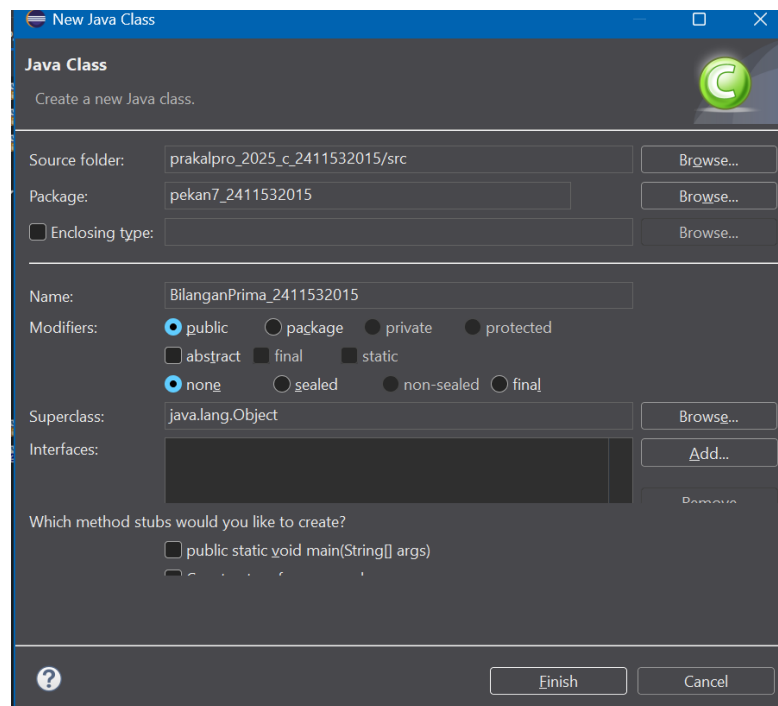
Gambar 2.1 Membuat package baru

2. Buat *package* dengan nama pekan 7



Gambar 2. 2 Membuat nama package

3. Buat *class* baru dengan nama “BilanganPrima”



Gambar 2. 3 Membuat *class* BilanganPrima

4. Buat *syntax* seperti ini

```
package pekan7_2411532015;

import java.util.Scanner;

public class BilanganPrima_2411532015 {
    public static boolean isPrime(int n) {
        int factors = 0;
        for(int i = 1; i<=n; i++) {
            if(n%i==0) {
                factors++;
            }
        }
        return (factors==2);
    }
}
```

```

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");
    }
    input.close();
}

```

Gambar 2. 4 Syntax Class BilanganPrima

5. Menghitung jumlah faktor dari bilangan n lalu melakukan perulangan dari 1 hingga n. Jika $n \% i == 0$, berarti i adalah pembagi dari n, faktor bertambah. Return factors == 2 adalah untuk menetapkan bilangan itu prima karna memiliki dua faktor

```

package pekan7_2411532015;

import java.util.Scanner;

public class BilanganPrima_2411532015 {
    public static boolean isPrime(int n) {
        int factors = 0;
        for(int i = 1; i<=n; i++) {
            if(n%i==0) {
                factors++;
            }
        }
        return (factors==2);
    }
}

```

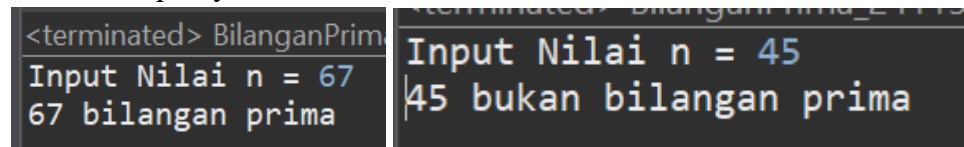
Gambar 2. 5 Penjelasan syntax

6. Pada bagian main ini program meminta input dan jika isPrime(a) bernilai true, program menampilkan bilangan tersebut adalah bilangan prima

```
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");  
    }  
    input.close();  
}
```

Gambar 2. 6 Penjelasan syntax

7. Berikut outputnya

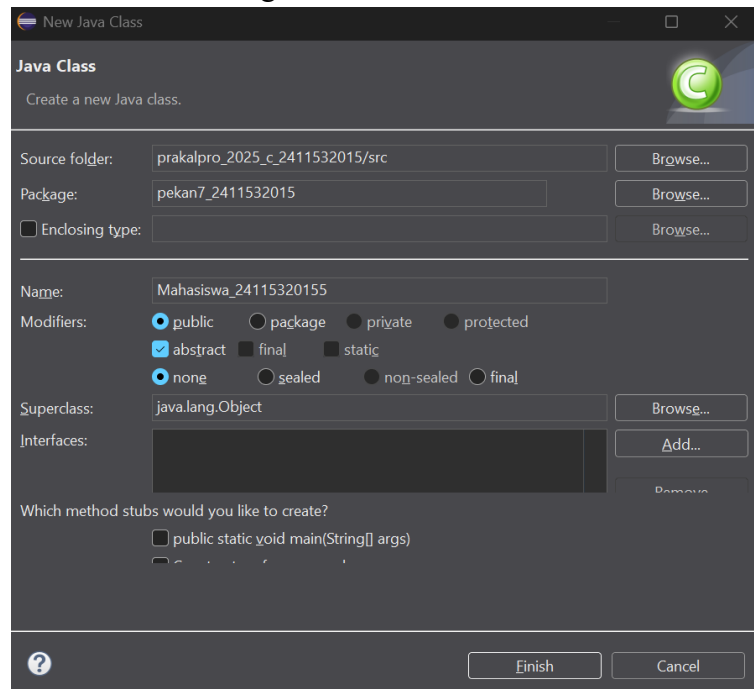


```
<terminated> BilanganPrima  
Input Nilai n = 67  
67 bilangan prima  
  
<terminated> BilanganPrima  
Input Nilai n = 45  
45 bukan bilangan prima
```

Gambar 2. 7 Ouput BilanganPrima

2.2 Class Mahasiswa

1. Buat class baru dengan nama Mahasiswa



Gambar 2. 8 Class Mahasiswa

2. Buat syntax seperti ini

```
package pekan7_2411532015;

public class Mahasiswa_2411532015 {
    private int nim;
    private String nama, nim2;

    public void setNim(int nim) {
        this.nim = nim;
    }
    public void setNama(String nama) {
        this.nama = nama;
    }
    public void setNim2(String nim2) {
        this.nim2 = nim2;
    }
    public int getNim() {
        return nim;
    }
    public String getNama() {
        return nama;
    }
    public String getNim2() {
        return nim2;
    }
    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. 9 Syntax Mahasiswa

3. Membuat variabel private

```
package pekan7_2411532015;

public class Mahasiswa_2411532015 {
    private int nim;
    private String nama, nim2;
```

Gambar 2. 10 Penjelasan syntax

4. Membuat setter dan getter method untuk mengisi nilai pada variabel serta mengambil nilai

```
public void setNim(int nim) {  
    this.nim = nim;  
}  
public void setName(String nama) {  
    this.nama = nama;  
}  
public void setNim2(String nim2) {  
    this.nim2 = nim2;  
}  
public int getNim() {  
    return nim;  
}  
public String getName() {  
    return nama;  
}  
public String getNim2() {  
    return nim2;  
}
```

Gambar 2. 11 Penjelasan syntax

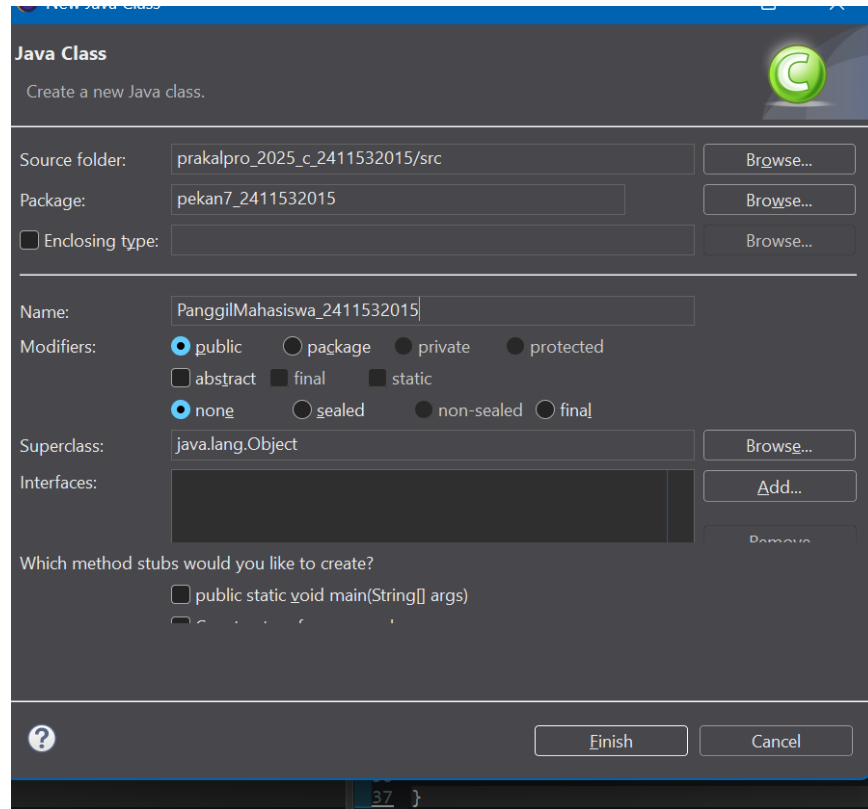
5. Membuat method untuk menampilkan nim interger dan nama lalu nim string dan nama

```
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. 12 Penjelasan syntax

2.3 Class PanggilMahasiswa

1. Buat class baru dengan PanggilMahasiswa yang merupakan class main dari Mahasiswa



Gambar 2. 13 Class PanggilMahasiswa

2. Buat syntax seperti ini

```
1 package pekan7_2411532015;
2
3 public class PanggilMahasiswa_2411532015 {
4     public static void main(String[] args) {
5         Mahasiswa_2411532015 a = new Mahasiswa_2411532015();
6         a.setNim(23532);
7         a.setNama("Rahmat");
8         System.out.println(a.getNim());
9         System.out.println(a.getNama());
10        a.Cetak();
11    }
12 }
13 }
14 }
```

Gambar 2. 14 Syntax PanggilMahasiswa

3. Program mengisi nilai nim dan nama method setter lalu menampilkan nilai menggunakan getter. Dan menampilkan data menggunakan method Cetak()

```
5      Mahasiswa_2411532015 a = new Mahasiswa_2411532015();
6      a.setNim(23532);
7      a.setNama("Rahmat");
8      System.out.println(a.getNim());
9      System.out.println(a.getNama());
10     a.Cetak();
```

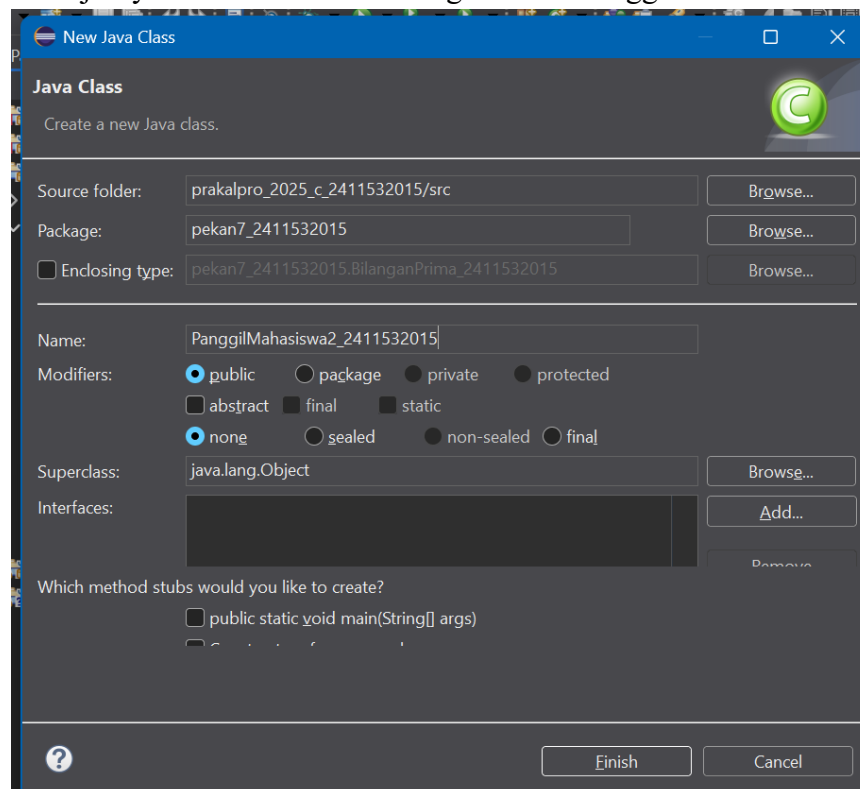
Gambar 2. 15 Penjelasan syntax

4. Output

```
23532
Rahmat
Nim : 23532
Nama : Rahmat
```

Gambar 2. 16 Output PanggilMahasiswa

5. Selanjutnya class Main kedua dengan nama PanggilMahasiswa2



Gambar 2. 17 Class PanggilMahasiswa2

6. Buat syntax seperti ini

```
package pekan7_2411532015;

import java.util.Scanner;

public class PanggilMahasiswa2_2411532015 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.println("NIM: ");
        String x = input.nextLine();
        System.out.println("Nama: ");
        String y = input.nextLine();
        Mahasiswa_2411532015 a = new Mahasiswa_2411532015();
        a.setNim2(x);
        a.setNama(y);
        if(x.startsWith("25")) {
            System.out.println(a.getNama() + " "
                               + " anda angkatan 2025");
        }
        if (a.getNim2().contains("1153")) {
            System.out.println("Anda Mahasiswa Informatika");
        }
        a.Cetak2();
        input.close();
    }
}
```

Gambar 2. 18 Syntax PanggilMahasiswa2

7. Membuat objek scanner dengan nama input

```
package pekan7_2411532015;

import java.util.Scanner;

public class PanggilMahasiswa2_2411532015 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.println("NIM: ");
        String x = input.nextLine();
        System.out.println("Nama: ");
        String y = input.nextLine();
```

Gambar 2. 19 Penjelasan Syntax

8. Menyimpan nilai melalui setter. Blok if untuk mengecek string apakah diawali 25 dan apakah mengandung 1153, lalu hasil ditampilkan

```
Mahasiswa_2411532015 a = new Mahasiswa_2411532015();
a.setNim2(x);
a.setNama(y);
if(x.startsWith("25")) {
    System.out.println(a.getNama() + ""
        + " anda angkatan 2025");
}
if (a.getNim2().contains("1153")) {
    System.out.println("Anda Mahasiswa Informatika");
}
a.Cetak2();
input.close();
}
```

Gambar 2. 20 Penjelasan Syntax

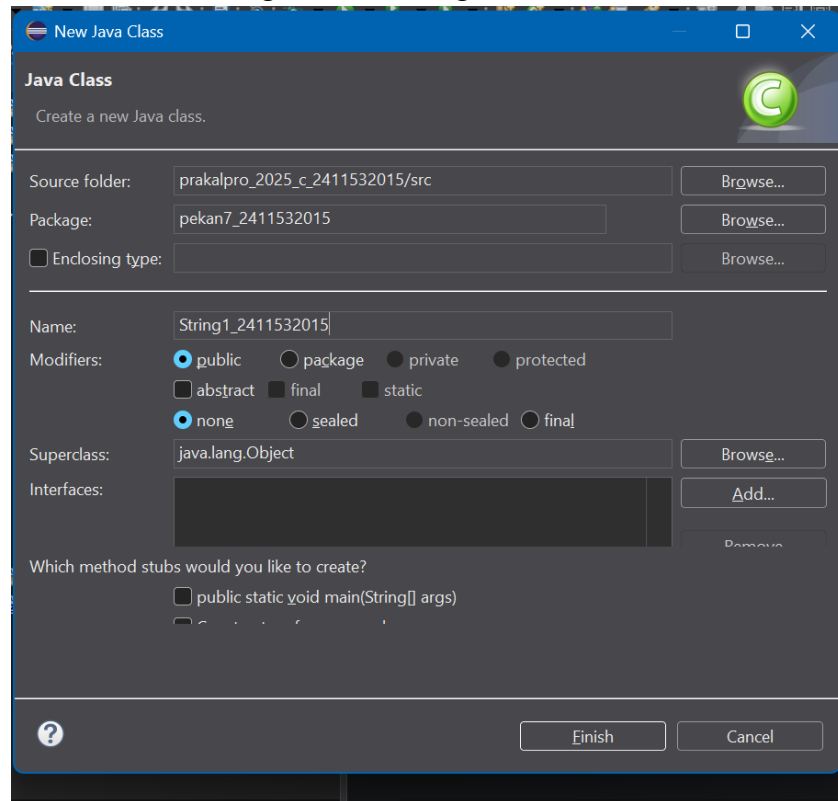
9. Output

NIM: 2411532015 Nama: han Anda Mahasiswa Informatika Nim : 2411532015 Nama : han	NIM: 2511532912 Nama: zah zah anda angkatan 2025 Anda Mahasiswa Informatika Nim : 2511532912 Nama : zah
--	--

Gambar 2. 21 Output PanggilMahasiswa2

2.4 Class String1

1. Buat class baru dengan nama String1



Gambar 2. 22 Class String1

2. Buat syntax seperti ini

```
package pekan7_2411532015;

public class String1_2411532015 {
    public static void main(String[] args) {
        String salam = "Assalamualaikum";
        System.out.println("panjang salam adalah: "
            +salam.length());
        System.out.println(salam.toUpperCase());
        System.out.println(salam.toLowerCase());
        System.out.println(salam.indexOf("salam"));
    }
}
```

Gambar 2. 23 Syntax String1

3. Program ini menampilkan operasi dasar pada String, seperti `length()` untuk menghitung jumlah karakter pada string "Assalamualaikum", `toUpperCase()` untuk mengubah seluruh huruf menjadi huruf kapital, `toLowerCase()` mengubah seluruh huruf menjadi huruf kecil, dan `indexOf()` mencari posisi awal substring "salam" pada string utama.

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

Gambar 2. 24 Penjelasan syntax

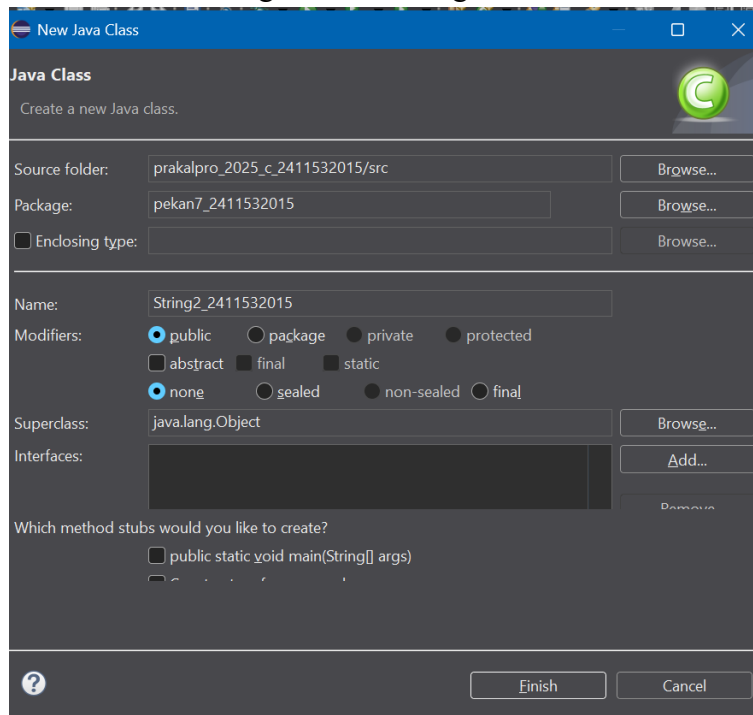
4. Output

```
panjang salam adalah: 15
ASSALAMUALAIKUM
assalamualaikum
2
```

Gambar 2. 25 Output String1

2.5 Class String2

1. Buat class baru dengan nama String2



Gambar 2. 26 Class string2

2. Buat syntax seperti ini

```
package pekan7_2411532015;

import java.util.Scanner;

public class String2_2411532015 {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Nama Depan: ");
        String firstName = input.nextLine();
        System.out.print("Nama Belakang: ");
        String lastName = input.nextLine();

        String txt1 = "Dosen\intelektual\kampus";
        System.out.println("Nama Lengkap: "+firstName + " " + lastName);
        System.out.println("Nama Lengkap: "+firstName.concat(lastName));
        System.out.println(txt1);
        int x = 10;
        int y = 20;
        int z = x+y;
        System.out.println("x + y = " +z);
        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);

        input.close();
    }

}
```

Gambar 2. 27 Syntax String2

3. Membuat objek scanner untuk input yang akan disimpan ke variabel firstName dan lastName

```
package pekan7_2411532015;

import java.util.Scanner;

public class String2_2411532015 {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Nama Depan: ");
        String firstName = input.nextLine();
        System.out.print("Nama Belakang: ");
        String lastName = input.nextLine();
    }

}
```

Gambar 2. 28 Penjelasan syntax

4. String txt1 berisi teks dengan tanda kutip dalam string. firstName + " " + lastName menggabungkan dua string menggunakan operator +. firstName.concat(lastName) menggabungkan string menggunakan method bawaan .concat()

```
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. 29 Penjelasan syntax

5. Penjumlahan integer dan penggabungan string. Variabel z merupakan hasil penjumlahan x dan y, sedangkan variabel c merupakan penggabungan antara a dan b karna menggunakan string bukan integer

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

input.close();
}
}
```

Gambar 2. 30 Penjelasan Syntax

6. Output

```
<terminated> String2_2411332013 pava Ap
Nama Depan: Zahira
Nama Belakang: Asyifa
Nama Lengkap: Zahira Asyifa
Nama Lengkap: ZahiraAsyifa
Dosen"intelektual"kampus
x + y = 30
String a + string b =1020
String a + integer y =1020
```

Gambar 2. 31 Output String2

BAB III KESIMPULAN

3.1 Kesimpulan

Berdasarkan hasil praktikum, dapat disimpulkan bahwa penggunaan class dan object memberikan manfaat besar dalam pengorganisasian data serta pembentukan program yang lebih sistematis. Setter dan getter terbukti efektif dalam menerapkan prinsip enkapsulasi untuk melindungi dan mengelola data secara terkontrol. Berbagai program yang dibuat juga menunjukkan bahwa Java menyediakan beragam method untuk manipulasi string, yang sangat membantu dalam validasi dan analisis teks.

3.2 Saran

Untuk pengembangan lebih lanjut, disarankan agar program dibuat lebih modular dengan menambahkan beberapa method tambahan agar setiap logika memiliki fungsi tersendiri dan mudah diperluas. Program juga dapat ditingkatkan dari sisi efisiensi, misalnya pada algoritma pengecekan bilangan prima yang masih bersifat brute force. Selain itu, penerapan exception handling perlu diperhatikan untuk mencegah terjadinya error ketika pengguna memasukkan input yang tidak sesuai.

DAFTAR PUSTAKA

[1] nextgen, “String pada Java”. [Daring]. Tersedia pada: <https://nextgen.co.id/string-pada-java/>