

LAPORAN PRAKTIKUM ALGORITMA PEMROGRAMAN
SREING PADA JAVA

Disusun Oleh:

Ihsanul Zaky EL-Muhammady

NIM:2511533001

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

Asisten Laboratorium: Aufan Taufiqurrahman



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
PADANG 2025

KATA PENGANTAR

Dengan penuh rasa syukur, saya mengucapkan terima kasih kepada Tuhan Yang Maha Esa, yang telah memberikan kemudahan dan kesempatan sehingga laporan praktikum ini dapat selesai dengan baik.

Saya juga ingin menyampaikan penghargaan yang setinggi-tingginya kepada Dr. Wahyudi, S.T., M.T. selaku dosen pengampu Mata Kuliah Algoritma dan Pemrograman, yang telah memberikan bimbingan dan motivasi selama perkuliahan. Ucapan terima kasih juga saya sampaikan kepada Uda Aufan Tafiqurrahman, asisten laboratorium, yang dengan sabar dan penuh perhatian telah membimbing kami dalam praktikum ini.

Laporan ini disusun sebagai salah satu tugas dalam Mata Kuliah Pemrograman Dasar yang membahas mengenai String dalam bahasa pemrograman java.

Saya menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, segala masukan dan saran yang membangun sangat saya harapkan untuk perbaikan di masa mendatang.

Padang, 2025

IHSANUL ZAKY EL-MUHAMMADY

DAFTAR ISI

KATA PENGANTAR
DAFTAR ISI.....
BAB I PENDAHULUAN
1.1 Latar Belakang
1.2 Tujuan.....
1.3 Manfaat.....
BAB II PEMBAHASAN
2.1 Program Bilangan Prima
2.2 Program Mahasiswa
2.3 Program Panggil Mahasiswa
2.4 Program Panggil Mahasiswa 2.....
2.5 Program String1
2.6 Program String2
BAB III KESIMPULAN
3.2 Kesimpulan
3.2 Saran.....
DAFTAR PUSTAKA.....

BAB I

PENDABULUAN

1.1 Latar Belakang

Dalam pemrograman Java, string merupakan bagian penting karena bisa membantu memahami konsep class dan objek. String di Java sebenarnya Adalah sebuah class yang terdapat dalam package `java.lang`, sehingga tidak perlu diimpor secara manual. String bersifat immutable, artinya nilainya tidak bisa diubah setelah dibuat. Secara internal, string terdiri dari array bertipe char. Kita juga bisa membuat objek string menggunakan keyword `new`, seperti halnya membuat objek dari class lain.

Kedua metode di atas mengembalikan nilai bertipe boolean (true atau false) tergantung hasil perbandingan.

Dalam Java, untuk membandingkan dua objek String, terdapat beberapa metode yang bisa digunakan:

- `equals(String s)` Metode ini membandingkan isi dari dua string kecil secara langsung dengan memperhatikan perbedaan huruf besar dan kecil. Jika kedua string memiliki karakter yang sama persis (termasuk kapitalisasi), maka akan menghasilkan nilai true, jika tidak maka false
- `equalsIgnoreCase(String s)` Metode ini membandingkan isi dua string, namun tidak memperhatikan perbedaan huruf besar dan kecil. Cocok digunakan jika kita ingin membandingkan string tanpa peduli kapitalisasi

Kedua metode di atas mengembalikan nilai bertipe Boolean (true atau false) tergantung hasil perbandingan

Selain membandingkan kita juga bisa memeriksa apakah sebuah string memiliki awalan tertentu menggunakan:

- `startsWith(String s)` Metode ini mengecek apakah string diawali dengan substring tertentu. Misalnya “Hello”,`startsWith(“He”)` akan bernilai true
- `endsWith(String s)` Dengan fungsi ini, maka objek string yang bersangkutan akan diperiksa, apakah diakhiri oleh objek `s` atau `string s`, pada parameter fungsi ini.

1.2 Tujuan

1. Memahami konsep dasar kelas string dalam java
2. Membuat dan menginisialisasi objek string
3. Mempelajari berbagai method

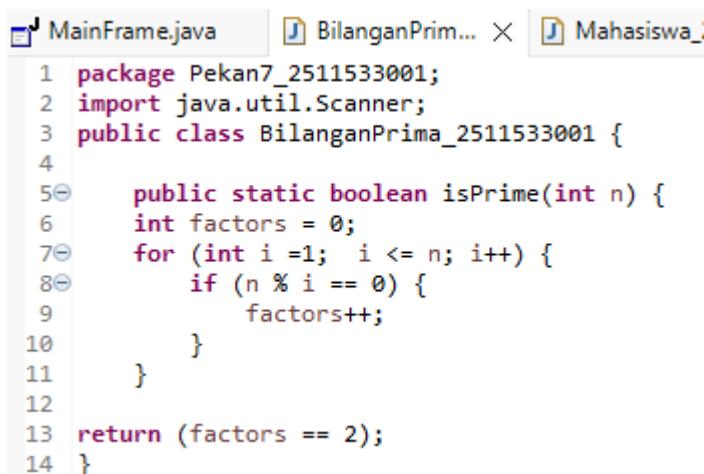
1.3 Manfaat

1. Meningkatkan pemahaman tentang cara kerja String
2. Membantu pembuatan program yang membutuhkan pengolahan dan analisis teks
3. Menambah keterampilan menggunakan method method untuk mengolah teks

BAB II

PEMBAHASAN

2.1 Program Bilangan Prima



```
1 package Pekan7_2511533001;
2 import java.util.Scanner;
3 public class BilanganPrima_2511533001 {
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    }
13    return (factors == 2);
14 }
```

Code 2.1

Nama file: BilanganPrima_2511533001

1. Variable factors untuk menghitung jumlah pembagi variable n
2. For (int i=1;i<=n;i++)
 - Untuk menentukan nilai i dari 1 hingga n
 - if (n%i ==0), saat i habis membagi n maka i adalah faktor n
 - factors akan bertambah setiap ditemukan faktor
3. Bilangan prima hanya memiliki 2 faktor
 - Jika factors == 2, adalah bilangan prima
 - Jika factors!= 2, bukan bilangan prima
4. System.out.print("Input nilai n= "); int a= input.nextInt();
 - Menampilkan perintah untuk memasukan nilai n
 - Nilai tersebut disimpan di variable a
5. isPrime(a) untuk melihat apakan nilai yang diinput adalah bil prima 4
 - Jika prima akan menampilkan pesan “bilangan prima”
 - Jika tidak akan menampilkan pesan “bukan bilangan prima”

Output:

```
Input nilai n = 5           Input nilai n = 6
5 bilangan prima            6 bukan bilangan prima
```

2.2 Program Mahasiswa

```
1 package Pekan7_2511533001;
2
3 public class Mahasiswa_2511533001 {
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;
0     }
1     public void setNim2 (String nim2) {
2         this.nim2=nim2;
3     }
4     public void setNama (String nama) {
```

Code 2.2

Nama file: Mahasiswa_2511533001

1. Privat bertujuan untuk melindungi data agar hanya dapat diakses melalui method tertentu.
2. this digunakan untuk membedakan variabel instance (this.nim) dan parameter method (nim). setNim akan menyimpan nilai ke variabel nim.
3. Method getNim akan mengembalikan nilai nim.
4. Cetak() akan menampilkan nilai nim bertipe integer dan nama.

2.3 Program Panggil Mahasiswa

```
1 package Pekan7_2511533001;
2
3 public class PanggilMahasiswa_2511533001 {
4
5     public static void main(String[] args) {
6         Mahasiswa_2511533001 a= new Mahasiswa_2511533001 ();
7         a.setNim(251153);
8         a.setNama("Zaky");
9         System.out.println(a.getNim());
0         System.out.println(a.getNama());
1         a.Cetak();
2     }
3 }
```

Code 2.3

Nama file : PanggilMahasiswa_2511533001

1. Objek dari file Mahasiswa_2511533001 akan digunakan, kita bisa mengakses semua method public yang ada di file sebelumnya.

2. setNim(251153) menyimpan nilai 251153 ke variabel nim di dalam variabel a, setnama("Zaky") menyimpan nama "Zaky" ke variabel nama
3. getNim() mengambil nilai dari variabel nim di dalam variabel a. getNama() mengambil nilai dari variabel nama.
4. a.Cetak(); memanggil method Cetak() dari file Mahasiswa_2511533001 yang menampilkan Nim dan Nama

Output:

```
251153
Zaky
Nim : 251153
Nama : Zaky
```

2.4 Program Panggil Mahasiswa 2

```
public class PanggilMahasiswa2_2511533001 {
    public static void main(String[] args) {
        Scanner input= new Scanner (System.in);
        System.out.print("NIM: ");
        String x= input.nextLine();
        System.out.print("Nama: ");
        String y= input.nextLine();
        Mahasiswa_2511533001 a= new Mahasiswa_2511533001 ();
        a.setNim2(x);
        a.setNama(y);
        if(x.startsWith("25")) {
            System.out.println(y +" anda angkatan 2025");
        }
    }
}
```

Code 2.4

Nama file: PanggilMahasiswa2_2511533001

1. Program meminta pengguna untuk memasukkan:
 - NIM (disimpan dalam variabel x)
 - Nama (disimpan dalam variabel y)
2. Karena menggunakan nextLine(), program bisa membaca seluruh teks, termasuk spasi.
3. Membuat objek a dari class Mahasiswa_2511533001. Melalui objek ini, kita bisa memanggil semua method.
4. x.startsWith("25") mengecek apakah NIM dimulai dengan angka "25".
 - Jika ya, maka dianggap angkatan tahun 2025.

- Program mencetak "anda angkatan 2025".

5. x.contains("1153") mengecek apakah NIM mengandung angka "1153" id dalamnya, program menampilkan "Anda Mahasiswa Informatika".

6. a.Cetak(); memanggil method Cetak() dari file Mahasiswa_2511533001 yang menampilkan Nim dan Nama

Output:

```
NIM: 2511533001
Nama: Zaky
Zaky anda angkatan 2025
Anda Mahasiswa Informatika
Nim2 : 2511533001
Nama : Zaky
```

2.5 Program String1

```
1 package Pekan7_2511533001;
2
3 public class String1_2511533001 {
4
5     public static void main(String[] args) {
6         String salam= "Assalamualaikum";
7         System.out.println("Panjang salam adalah: " + salam.length());
8         System.out.println(salam.toUpperCase()); //Outputs "ASSALAMUALAIKUM"
9         System.out.println(salam.toLowerCase()); //Outputs "assalamualaikum"
10        System.out.println(salam.indexOf("salam")); //Outputs 2
11    }
12 }
13 }
```

Code 2.5

Nama file: String_2511533001

1. Mendeklarasikan variable salam bertipe string dan mengisinya dengan teks "Assalamualaikum".
2. Method length() mengembalikan jumlah karakter dalam string.
3. Method toUpperCase() mengubah seluruh huruf menjadi huruf besar/kapital.
4. Method toLowerCase() mengubah seluruh huruf menjadi huruf kecil.
5. Method index of ("salam") mencari posisi awal substring "salam" di dalam "Assalamualaikum".

Output:

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

2.6 Program String2

```
1 package Pekan7_2511533001;
2
3 import java.util.Scanner;
4
5 public class String2_2511533001 {
6
7     public static void main(String[] args) {
8         Scanner input= new Scanner (System.in);
9         System.out.print("Nama Depan");
10        String firstName= input.nextLine();
11        System.out.print("Nama Belakang");
12        String lastName= input.nextLine();
13        String txt1= "Dosen\"intelektual\" kampus";
14        System.out.println("Nama Lengkap: "+firstName + " " +lastName);
```

Code 2.6

Nama file: String2_2511533001

1. firstName dan lastName menyimpan nama depan dan belakang.
2. txt1 berisi teks dengan tanda kutip ganda di dalam string. Karakter (\") digunakan untuk menampilkan tanda kutip ("")
3. "Nama Lengkap: " digabung dengan firstName, lalu ditambah spasi (" "), lalu lastName.
4. concat() adalah method bawaan dari class String untuk menggabungkan dua string. Hasilnya tidak menambahkan spasi otomatis.
5. System.out.println(txt1); Menampilkan isi txt1
6. x dan y adalah variabel bertipe int. z = x + y menjumlahkan secara aritmetika (10 + 20 = 30).
7. a dan b adalah string, bukan angka. Operator “+” pada string akan menggabungkan teks. "10" + "20" menghasilkan "1020".

Output:

```
Nama Depan Ihsanul
Nama Belakang Zaky
Nama Lengkap: Ihsanul Zaky
Nama Lengkap: Ihsanul Zaky
Dosen"intelektual" kampus
x+ y= 30
String a + String b= 1020
String a + String y= 1020
```

BAB III

KESIMPULAN

3.1 Kesimpulan

Dari seluruh pepercobaan yang dilakukan dalam praktikum string, dapat kita simpulkan bahwa string merupakan komponen penting dalam pengolahan data berbasis teks. Sebagai objek yang bersifat immutable, String tidak dapat diubah setelah didefinisikan, sehingga memberikan kestabilan dalam memanipulasi data.

Java menyediakan berbagai method bawaan seperti `length()`, `toUpperCase()`, `toLowerCase()`, `indexof()`, `startsWith()`, `contains()`, `concat()`, dan `equals()` yang sangat berguna dalam proses validasi, pencarian, dan pemggabungan teks. Penggunaan operator `+` juga memungkinkan penggabungan string maupun penjumlahan bilangan, tergantung pada tipe data yang digunakan. Perlu diperhatikan bahwa perbandingan isi string sebaiknya menggunakan method `equals()`, bukan operator `==`, karena `==` hanya membandingkan referensi objek di memori.

Dalam implementasinya, objek String dapat dikombinasikan dengan prinsip *encapsulation* melalui penggunaan atribut private yang diakses menggunakan method getter dan setter. Selain itu, kelas Scanner berperan penting dalam menerima input dari pengguna, yang kemudian dapat diproses menggunakan berbagai method dari kelas String.

Secara keseluruhan, pemahaman terhadap konsep dan fitur dari kelas String sangat membantu dalam pengembangan program Java yang efisien dan terstruktur, khususnya dalam pengelolaan input-output serta manipulasi data teks, seperti pada validasi NIM mahasiswa dan aplikasi serupa lainnya.

3.2 Saran

Sebaiknya dosen menyelenggarakan sesi pra-praktikum di kelas sebagai pengantar materi, agar mahasiswa memperoleh pemahaman awal yang lebih baik. Dengan demikian, potensi kepanikan atau kesalahan selama praktikum dapat diminimalkan. Disarankan agar materi praktikum dibagikan terlebih dahulu melalui platform iLearn, sehingga mahasiswa memiliki waktu yang cukup untuk mempelajari dan mempersiapkan diri sebelum pelaksanaan praktikum.

DAFTAR PUSTAKA

- [1] Oracle America, Inc., “Class String (Java Platform SE 8),” Oracle Documentation, 2014. [Online]. Available: <https://docs.oracle.com/javase/8/docs/api/java/lang/String.html>. [Accessed: 13-Nov-2025].
- [2] Oracle Corporation, “The Java™ Tutorials – Strings,” Oracle, 2019. [Online]. Available: <https://docs.oracle.com/javase/tutorial/java/data/strings.html>. [Accessed: 13-Nov-2025].