

**LAPORAN PRATIKUM PEMROGRAMAN ALGORITMA DAN
PEMROGRAMAN**

PEKAN 7

disusun Oleh:

Habib Al Faruq

2511532010

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 atas kehadiran Tuhan Yang Maha Esa, karena berkat rahmat dan karunia-Nya, penulis dapat menyelesaikan laporan praktikum algoritma pemrograman mengenai *string* di Java. Penulisan laporan ini bertujuan untuk membuat sebuah laporan tentang materi pemrograman mengenai penggunaan *string*.

Penyusunan laporan praktikum ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu, penulis ingin menyampaikan terima kasih kepada dosen pengampu mata kuliah praktikum pemrograman algoritma dan pemrograman, Bapak Dr. Wahyudi. S.T.M.T dan asisten praktikum, Uda Muhammad Zaki Al Hafiz yang telah memberikan arahan dan masukan dalam proses penyusunan laporan praktikum ini. Penulis berharap laporan praktikum ini dapat memberikan manfaat bagi pembaca dalam mempelajari konsep penggunaan *string*.

DAFTAR ISI

| | |
|---|---------------|
| KATA PENGANTAR..... | i |
| DAFTAR ISI | ii |
| BAB I PENDAHULUAN..... | 1 |
| 1.1 Latar Belakang Praktikum | 1 |
| 1.2 Tujuan Praktikum..... | 1 |
| 1.3 Manfaat Praktikum | 2 |
| BAB II PEMBAHASAN..... | 4 |
| 2.1 Perulangan While..... | 4 |
| 2.1.1 Penjelasan Kode Program..... | 4 |
| 2.1.2 Contoh Output | 4 |
| 2.2 Lempar Dadu | 5 |
| 2.2.1 Penjelasan Kode Program..... | 6 |
| 2.3 Do While | 7 |
| 2.3.1 Penjelasan Kode Progam | 8 |
| BAB III KESIMPULAN..... | 10 |
| 3.1 Ringkasan Hasil Praktikum..... | 10 |
| DAFTAR PUSTAKA..... | 11 |

BAB I

PENDAHULUAN

1.1 Latar Belakang Praktikum

Pemrograman Java merupakan salah satu fondasi utama dalam bidang teknologi informasi karena membantu mahasiswa mengasah kemampuan berpikir logis, terstruktur, dan sistematis. Melalui praktikum ini, mahasiswa mempelajari konsep-konsep dasar seperti penggunaan *method*, tipe data, perulangan, percabangan, serta manipulasi *string*. Pemahaman terhadap konsep tersebut menjadi bekal penting dalam pengembangan program yang lebih kompleks, misalnya sistem manajemen data atau aplikasi berbasis objek. Dengan menguasai dasar-dasar ini, mahasiswa diharapkan mampu memahami alur logika pemrograman dan bagaimana suatu kode dapat menghasilkan keluaran sesuai dengan masukan yang diberikan.

Selain itu, praktikum ini juga memperkenalkan konsep *Object-Oriented Programming (OOP)* melalui pembuatan *class* dan *object*, seperti pada contoh *class Mahasiswa*. Mahasiswa dilatih untuk memahami bagaimana variabel dan *method* dalam suatu *class* dapat saling berinteraksi membentuk struktur program yang utuh. Pengenalan konsep ini penting karena OOP menjadi salah satu paradigma yang paling banyak diterapkan dalam pengembangan perangkat lunak modern. Dengan mengintegrasikan logika dasar, penggunaan *method*, serta pengolahan *string*, mahasiswa akan memiliki landasan yang kuat untuk melanjutkan ke tahap pemrograman yang lebih lanjut dan aplikatif.

1.2 Tujuan Praktikum

Tujuan utama dari praktikum ini adalah membantu mahasiswa memahami serta menerapkan dasar-dasar pemrograman Java secara tepat. Melalui kegiatan ini,

mahasiswa dilatih untuk membuat fungsi atau *method* seperti *isPrime()* yang berfungsi menentukan apakah suatu bilangan merupakan bilangan prima. Selain itu, mahasiswa juga diharapkan mampu menggunakan struktur kontrol seperti perulangan (*looping*), percabangan (*if-else*), dan operasi aritmetika guna menyelesaikan permasalahan sederhana dalam bentuk program. Dengan latihan tersebut, mahasiswa belajar bagaimana sebuah logika dapat diterjemahkan menjadi barisan kode yang dapat dijalankan komputer untuk menghasilkan keluaran yang diinginkan.

Lebih jauh, praktikum ini bertujuan agar mahasiswa memahami konsep dasar *Object-Oriented Programming (OOP)* melalui penggunaan *class*, *object*, serta penerapan *setter* dan *getter method*. Mahasiswa juga diperkenalkan pada berbagai *method* bawaan dari *classString*, seperti *length()*, *toUpperCase()*, *toLowerCase()*, dan *indexOf()*, yang berfungsi mempermudah pengolahan data berbasis teks. Dengan penguasaan konsep-konsep tersebut, mahasiswa diharapkan mampu menulis program yang tidak hanya berjalan dengan baik, tetapi juga efisien, terstruktur, serta mudah dikembangkan lebih lanjut.

1.3 Manfaat Praktikum

Manfaat utama dari praktikum ini adalah membantu mahasiswa mengembangkan kemampuan berpikir logis serta keterampilan pemecahan masalah melalui penerapan langsung konsep-konsep pemrograman. Dengan memahami prinsip kerja perulangan, percabangan, dan pengolahan data menggunakan fungsi, mahasiswa menjadi lebih siap menghadapi tantangan pemrograman yang lebih kompleks. Selain itu, mahasiswa juga dilatih untuk menulis kode secara terstruktur, rapi, dan mudah dipahami—suatu kemampuan yang sangat dibutuhkan dalam dunia profesional di bidang teknologi informasi.

Selain itu, praktikum ini memberikan pengalaman langsung dalam penerapan konsep *Object-Oriented Programming (OOP)* yang menjadi dasar utama pengembangan aplikasi modern. Melalui latihan pembuatan *class Mahasiswa* serta penggunaan berbagai *String method*, mahasiswa belajar bagaimana data dapat disimpan, diproses, dan ditampilkan dengan efisien. Pengalaman ini menjadi bekal penting dalam pengembangan aplikasi berbasis data, seperti sistem informasi akademik, aplikasi web, maupun perangkat lunak lainnya di masa mendatang.

BAB II

PEMBAHASAN

2.1. Bilangan Prima

```
1 package pekan7_2511532010;
2 import java.util.Scanner;
3
4 public class BilanganPrima_2511532010 {
5     public static boolean isPrime( int n ) {
6         int factors = 0;
7         for (int i = 1; i <= n ; i ++ ) {
8             if (n%i == 0) {factors++;}
9         }
10        return (factors == 2);}
11
12    public static void main(String[] args) {
13        Scanner input = new Scanner (System.in);
14        System.out.print ("Input nilai n = ");
15        int a = input.nextInt();
16        if (isPrime(a)) {
17            System.out.println (a + " bilangan prima");
18        }else {System.out.println (a + " bukan bilangan prima");}}
```

Gambar 2.1

2.1.1 Penjelasan Kode Program

Pada gambar 3.1, kita buat dulu method boolean `isPrime (int n)`, dengan mendeklarasikan variabel faktor sebagai integer dengan nilai 0. *For* `int i = 1, <=n, i++` yang artinya untuk integer `i = 0`, `i` harus lebih kecil atau sama dengan `n`, dan setiap perulangan `i` ditambah 1. ketika `i` masih kecil atau sama dengan `n`, jika `n mod i == 0`, maka faktor ++. Keluar dari `for`, kita buat `return factors == 2`.

Masuk ke *main method* kita agar kita bisa menginputkan `a` kedalam keluaran "Input nilai n = ". Jika `a` adalah bilangan prima maka akan dicetak `a + " bilangan prima"`. Namun jika `a` bukan bilangan prima, maka akan dicetak `a + " bukan bilangan prima"`.

2.1.2 Contoh Output

```
Input nilai n = 8
8 bukan bilangan prima
```

Gambar 2.2

```
Input nilai n = 11
11 bilangan prima
```

Gambar 2.3

2.2 Mahasiswa dan Panggil Mahasiswa

```
1 package pekan7_2511532010;
2
3 public class Mahasiswa_2511532010 {
4
5     private int nim;
6     private String nama,nim2;
7
8     public void setNim (int nim) {
9         this.nim = nim;
10    }
11
12    public void setNim2 (String nim2) {
13        this.nim2 = nim2;
14    }
15
16    public void setNama (String nama) {
17        this.nama = nama;
18    }
19
20    public int getNim () {
21        return nim;
22    }
23
24    public String getNim2 () {
25        return nim2;
26    }
27
28    public String getNama () {
29        return nama;
30    }
31
32    public void Cetak() {
33        System.out.println ("Nim = " + nim);
34        System.out.println ("Nama = " + nama);
35    }
36
37    public void Cetak2 () {
38        System.out.println ("Nim = " + nim2);
39        System.out.println ("Nama = " + nama);
40    }
41 }
```

Gambar 2.4


```
1 package pekan7_2511532010;
2
3 public class PanggilMahasiswa_2511532010 {
4
5     public static void main(String[] args) {
6         Mahasiswa_2511532010 a = new Mahasiswa_2511532010 ();
7         a.setNim(2511);
8         a.setNama("HABIB");
9         System.out.println (a.getNim ());
10        System.out.println (a.getNama ());
11        a.Cetak();}}
12
```

Problems Javadoc Declaration Console X Git Staging Coverage Eclipse IDE for Java Developers 2025-09 Release
<terminated> PanggilMahasiswa_2511532010 [Java Application] /Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java (12 Nov 2025, 10.43.28 – 10.43.29 elapsed: 0:00:00.208)
2511
HABIB
Nim = 2511
Nama = HABIB

Gambar 2.5

```
1 package pekan7_2511532010;
2 import java.util.Scanner;
3 public class PanggilMahasiswa2_2511532010 {
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_2511532010 a = new Mahasiswa_2511532010 ();
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();}}
```

Problems Javadoc Declaration Console X Git Staging Coverage Eclipse IDE for Java Developers 2025-09 Release
<terminated> PanggilMahasiswa2_2511532010 [Java Application] /Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java (12 Nov 2025, 10.45.01 – 10.45.11 elapsed: 0:00:10.395) [p
NIM: 2511532010
Nama: HABIB AL FARUQ
HABIB AL FARUQ anda angkatan 2025
Anda Mahasiswa Informatika
Nim = 2511532010
Nama = HABIB AL FARUQ

Gambar 2.6

2.2.1 Penjelasan Kode Program

Pada gambar 2.4, kita deklarasikan variabel `nim` sebagai integer dan variabel `nim`, `nim2` sebagai String yang mana keduanya bersifat `private`. Lalu, kita buat method setter untuk memasukkan nilai ke variabel, setiap setter kita ikuti `this`. seperti gambar 2.4. Setelah selesai dengan setter method, kita lanjutkan ke method getter untuk mengambil dan mengembalikan nilai dari setiap variabel. Setiap method getter diikuti dengan `return` seperti gambar 2.4. Setelah itu kita buat method cetak yang akan mencetak persis seperti gambar 2.4. Terakhir, kita buat method cetak2 yang akan mencetak persis seperti gambar 2.4.

Pada gambar 2.5 kita membuat objek dari program pada gambar 2.4 dan memanggil method nya, kita membuat objek `a` dari class `Mahasiswa_2511532010`. `a.setNim(2511)` untuk memanggil method `setNim` pada objek `a` untuk menyimpan nilai 2511. `a.setNama("HABIB")` untuk memanggil method `setNama` pada objek `a` untuk menyimpan nilai "HABIB". Lalu kita panggil method `a.getNim` untuk memanggil nilai `nim` dan mencetaknya, dan kita panggil method `a.getNama` untuk memanggil nilai `nama` dan mencetaknya. `a.cetak` berfungsi untuk memanggil method `Cetak()` yang telah kita buat pada program di gambar 2.4

Pada gambar 2.6 kita membuat objek dari program pada gambar 2.4 dan memanggil method nya, kita buat pernyataan input agar kita bisa menginputkan variabel `x` dan `y`. Lalu, kita membuat objek `a` dari class `Mahasiswa_2511532010`. `a.setNim2(x)` untuk memanggil method `setNim2` pada objek `a` untuk menyimpan nilai `x`. `a.settNama(y)` untuk memanggil method `setNim` pada objek `a` untuk menyimpan nilai `y`. Kita buat percabangan jika `x` dimulai dengan "25" maka `print y + " anda angkatan 2025"`, jika `x` mengandung "1153" maka `print "Anda Mahasiswa Informatika"`. `a.cetak` berfungsi untuk memanggil method `Cetak2()` yang telah kita buat pada program di gambar 2.4. Terakhir, kita tutup inputnya.

2.3 String

```

1 package pekan7_2511532010;
2
3 public class String1_2511532010 {
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 ());
9         System.out.println (salam.toLowerCase ());
10        System.out.println (salam.indexOf ("salam"));}

```

<terminated> String1_2511532010 [Java Application] /Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java (12 Nov 2025, 11.23.29 - 11.23.30 elapsed: 0:00:00.161) [pid: 22773]
 panjang salam adalah: 15
 ASSALAMUALAIKUM
 assalamualaikum
 2

Gambar 2.7

```

1 package pekan7_2511532010;
2
3
4 public class String2_2511532010 {
5
6     public static void main(String[] args) {
7         String firstName = "Syifa";
8         String lastName = "Muhassanah";
9         String txt1 = "Dosen \"intelektual\" kampus";
10        System.out.println ("Nama Lengkap : " + firstName + " " + lastName);
11        System.out.println ("Nama Legkap : " + firstName.concat (lastName));
12        System.out.println (txt1);
13        int x = 10;
14        int y = 20;
15        int z = x + y;
16        System.out.println ("x + y = " + z);
17        String a = "10";
18        String b = "20";
19        String c = a + b;
20        System.out.println ("String a + String b = " + c);
21        String v = a + y;
22        System.out.println ("String a + Integer y = " + v);}

```

<terminated> String2_2511532010 [Java Application] /Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java (12 Nov 2025, 11.28.39 - 11.28.39 elapsed: 0:00:00.196) [pid: 22896]
 Nama Lengkap : Syifa Muhassanah
 Nama Legkap : SyifaMuhassanah
 Dosen "intelektual" kampus
 x + y = 30
 String a + String b = 1020
 String a + Integer y = 1020

Gambar 2.8

2.3.1 Penjelasan Kode Program

Pada gambar 2.7, kita mendeklarasikan variabel salam sebagai String dengan nilai "Assalamualaikum". Kita mencetak "panjang salam adalah: " + salam.length() yang mana akan menghitung jumlah karakter dalam string salam. salam.toUpperCase() bermakna akan mencetak nilai dari variabel salam dengan semuanya huruf ditukar menjadi *upper case*. salam.toLowerCase() bermakna akan mencetak nilai dari variabel

salam dengan semuanya huruf ditukar menjadi *lower case*. `salam.indexOf("salam")` bermakna mencari awal posisi substring "salam" di dalam string. Pada "Assalamualaikum", huruf s pertama dari "salam" berada pada indeks ke-2 (karena indeks dimulai dari 0).

Pada gambar 2.8, kita mendeklarasikan variabel `firstName` sebagai String dengan nilai "Syifa", variabel `lastName` sebagai String dengan nilai "Muhassanah", variabel `txt1` sebagai String dengan nilai "Dosen \"intelektual\" kampus". Lalu, kita akan mencetak "Nama Lengkap : " + `firstName` + " " + `lastName`. Cetak juga ("Nama Legkap : " + `firstName.concat(lastName)`), dengan memanggil method `concat` untuk mencetak `lastName` langsung di belakang `firstName` tanpa spasi. Terakhir kita mencetak `txt1`.

Selanjutnya pada proram gambar 2.8, kita deklarasikan variabel `x` sebagai integer denagn nilai 10, variabel `y` sebagai integer dengan nilai 20, variabel `z` sebagai integer dengan nilai `x + y`, dan kita cetak "`x + y =` " `z`. Kemudian kita deklarasikan variabel `x` sebagai String dengan nilai "10", variabel `b` sebagai String dengan nilai "20", variabel `c` sebagai String dengan nilai `a + b`. Kita akan mencetak "String a + String b = " + `c`. Lalu, kita deklarasikan variabel `v` sebagai string dengan nilai `a + y` dan kita cetak "String a + Integer y = " + `v`.

BAB III

KESIMPULAN

3.1 Ringkasan Hasil Praktikum

Berdasarkan hasil praktikum yang telah dilakukan, dapat disimpulkan bahwa kegiatan ini memberikan pemahaman dasar mengenai cara kerja bahasa pemrograman Java. Melalui latihan seperti pembuatan fungsi untuk mengecek bilangan prima, penggunaan struktur perulangan (*looping*), dan percabangan (*if-else*), mahasiswa dapat memahami bagaimana logika berpikir diubah menjadi kode program yang sistematis dan terarah. Pemahaman ini menjadi landasan penting dalam mengembangkan kemampuan analisis serta berpikir logis dalam penyusunan algoritma yang efisien.

Selain itu, praktikum ini juga memperkenalkan mahasiswa pada berbagai operasi dasar dalam pengolahan *string*, seperti *length()*, *toUpperCase()*, *toLowerCase()*, dan *indexOf()*. Melalui penerapan fungsi-fungsi tersebut, mahasiswa belajar memahami bagaimana data teks dapat dimanipulasi di dalam program untuk menghasilkan informasi yang sesuai dengan kebutuhan. Kegiatan ini tidak hanya meningkatkan keterampilan teknis, tetapi juga menumbuhkan ketelitian serta kemampuan memahami struktur logika agar program berjalan dengan hasil yang benar.

Secara keseluruhan, praktikum ini melatih mahasiswa untuk berpikir secara rasional dan terstruktur dalam menulis kode, serta memahami alur kerja program dari tahap input hingga output. Pengalaman ini menjadi bekal penting untuk mempelajari konsep pemrograman yang lebih kompleks pada tahap berikutnya. Selain itu, kemampuan pemecahan masalah yang terasah melalui praktikum ini juga menjadi modal utama bagi mahasiswa dalam menghadapi tantangan di bidang teknologi informasi di masa depan.

DAFTAR PUSTAKA

- [1] Oracle, *The Java® Tutorials — Primitive Data Types*. [Online]. Available: <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html>
- [2] GeeksforGeeks, *Java Data Types – GeeksforGeeks*. [Online]. Available: <https://www.geeksforgeeks.org/data-types-in-java>