



Modul Praktikum

# Dasar Pemrograman

Aris Sudianto, M.Kom

```
def filter(s: Set, p: Int => Boolean): Set = {
```

```
  def forall(s: Set, p: Int => Boolean): Boolean = {
```

```
    def iter(a: Int): Boolean = {
```

```
      if (contains(p,a) && !contains(s,a)) false
```

```
      else if (a > 1000) true
```

```
      else iter(a + 1)
```

```
    }
```

```
    iter(-1000)
```

```
  }
```

```
  def exists(s: Set, p: Int => Boolean): Boolean = {
```

```
    (x: Int) => !(forall(s, !p(x)))
```

```
  }
```

```
  def map(s: Set, f: Int => Int): Set = {
```

```
    (b: Int) => exists(s, (c: Int) => f(c) == b)
```



# Java™

PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNIK UNIVERSITAS HAMZANWADI

Jln. Prof. M.Yamin No.35 Pancor – Selong (83611)

## MODUL 1

### Pengantar Bahasa Pemrograman Java

#### KOMPETENSI

Setelah menyelesaikan modul ini mahasiswa diharapkan dapat memahami macam – macam bahasa pemrograman dan penggunaan masing – masing bahasa pemrograman.

---

#### 1. Tujuan

- ❖ Mahasiswa mengetahui tentang jenis dan macam – macam bahasa pemrograman
- ❖ Mahasiswa mengetahui tentang kelebihan dan kelemahan setiap bahasa pemrograman
- ❖ Mahasiswa mengetahui tentang bahasa pemrograman java
- ❖ Mahasiswa mengetahui tentang kelebihan bahasa pemrograman java dibanding dengan bahasa pemrograman yang lain.

#### 2. Tugas Pendahuluan

- ❖ Jelaskan definisi bahasa pemrograman
- ❖ Sebutkan macam – macam bahasa pemrograman
- ❖ Sebutkan kelebihan bahasa pemrograman java dibanding dengan bahasa pemrograman yang lainnya.
- ❖ Tuliskan syntax dasar pemrograman java

#### 3. Teori

##### KONSEP PEMROGRAMAN

“Bahasa pemrograman adalah bahasa komputer formal atau bahasa yang dirancang untuk mengkomunikasikan instruksi ke mesin, khususnya komputer. Bahasa pemrograman dapat digunakan untuk membuat program untuk mengontrol perilaku mesin atau untuk mengekspresikan algoritme.”

Bahasa pemrograman adalah notasi untuk menulis program, yang merupakan spesifikasi dari suatu komputasi atau algoritma.

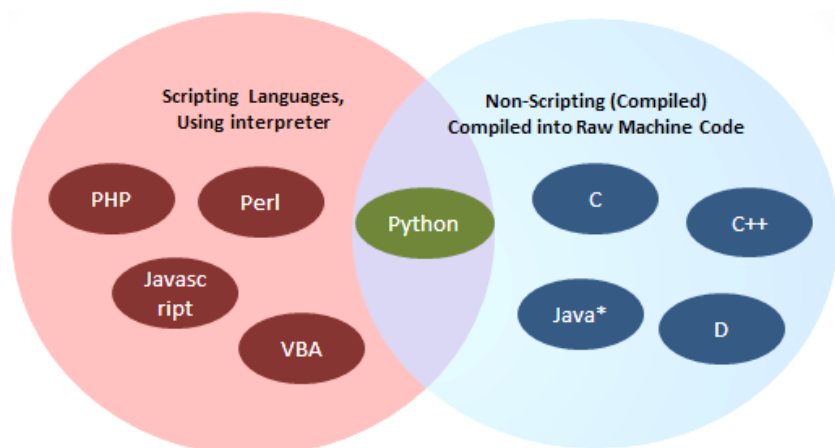
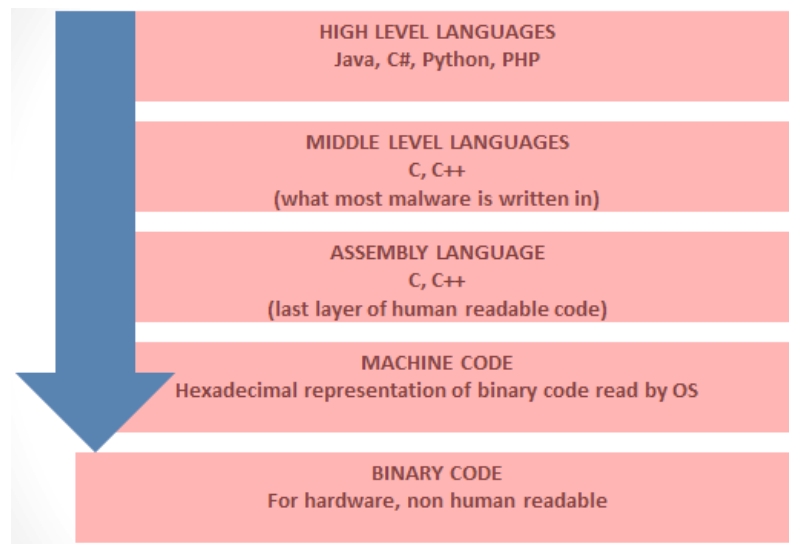
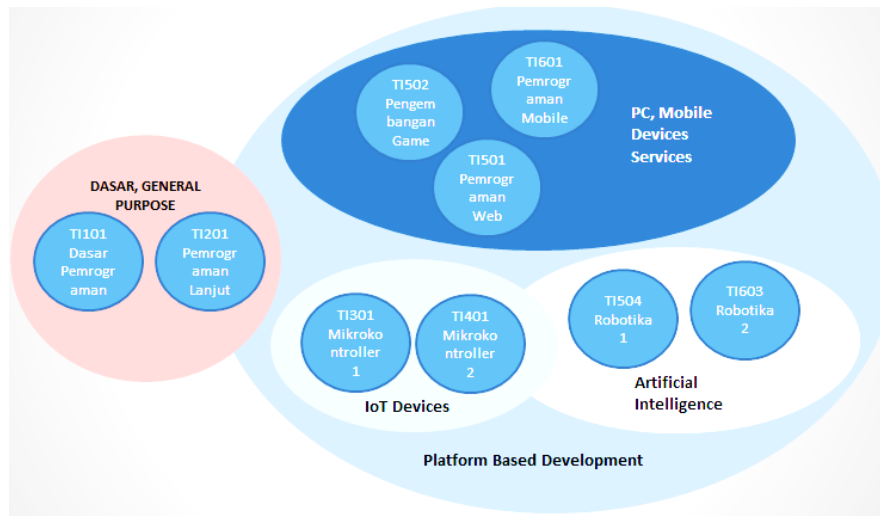
(Aaby, Anthony (2001). Pengantar Bahasa Pemrograman)?

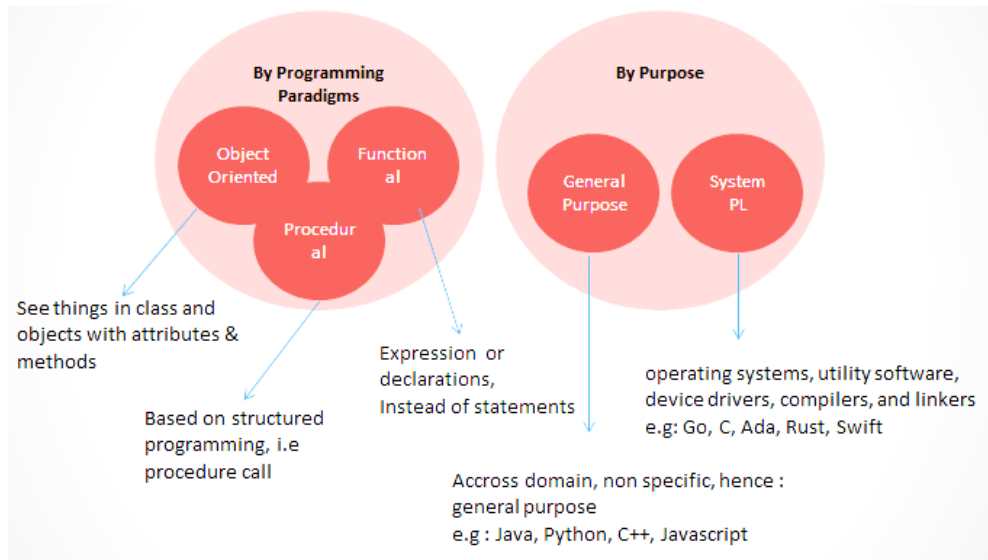
Tujuan: Untuk menulis program

Program komputer adalah kumpulan instruksi [1] yang melakukan tugas tertentu ketika dijalankan oleh komputer. Komputer membutuhkan program untuk berfungsi, dan biasanya menjalankan instruksi program dalam unit pemrosesan pusat.

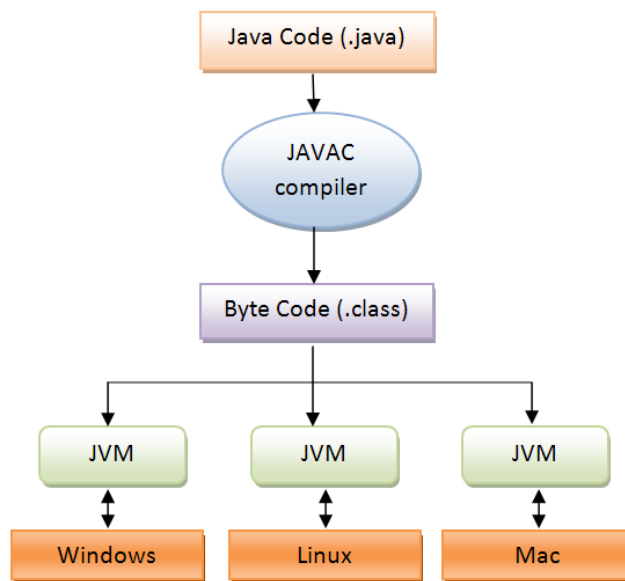
(Silberschatz, Abraham (1994). Konsep Sistem Operasi, Edisi Keempat.)

Bagian dari program komputer yang melakukan tugas yang terdefinisi dengan baik dikenal sebagai algoritma. Kumpulan program komputer, pustaka dan data terkait disebut sebagai perangkat lunak. Program komputer dapat dikategorikan di sepanjang garis fungsional, seperti perangkat lunak aplikasi atau perangkat lunak sistem.





“Write once run anywhere”



#### 4. Praktikum

##### a) Peralatan dan Bahan

- Perangkat komputer/ PC/ Laptop
- Sistem Operasi windows / Linux
- JDK
- Eclipse

##### b) Keselamatan Kerja

- baca dan pahami semua langkah kerja dari praktikum dengan cermat
- jangan membawa makanan dan minuman ke dalam laboratorium, terlebih lagi membawa benda cair.

- Pastikan diri anda terlindung dari efek kejutan listrik saat menyentuh perangkat komputer

**c) Langkah Kerja**

- Siapkan alat dan bahan praktek
- Selalu memperhatikan keselamatan kerja
- Jalankan perangkat menggunakan editor eclipse
- Organisasikan dengan baik direktori penyimpanan anda untuk setiap modul praktikum.

**5. Metode Menuliskan Syntak Dasar Pada Pemrograman Java**

1. Buka aplikasi eclipse kemudian atur workspace untuk menentukan direktory penyimpanan project, pastikan pilih direktory D/E.
2. Setelah jendela eclipse terbuka, buat project baru dengan cara klik menu file → New project.
3. Buat nama project dengan menyesuaikan pembahasan pada modul pertama.
4. Di dalam project buat package untuk membagi project menjadi beberapa bagian dengan tujuan supaya dalam satu project dapat diisi dengan lebih dari satu contoh program.
5. Di dalam package buatlah class baru untuk memulai mengetikkan latihan dasar pemrograman.

**6. Evaluasi Dan Pertanyaan**

Jelaskan maksud penggalan kode program yang telah di praktekkan.

**7. Studi Kasus**

- a. Jelaskan fungsi dari bahasa pemrograman java
- b. Sebutkan macam – macam editor yang digunakan untuk menuliskan syntak dasar bahasa pemrograman java.
- c. Jelaskan kelebihan bahasa pemrograman java.

**8. Kesimpulan**

**9. Hasil Review**

Date	Result (1-5)	Signature

## MODUL 2

### Java, Basic Concepts & Syntax

#### KOMPETENSI

Setelah menyelesaikan modul ini mahasiswa diharapkan dapat memahami konsep bahasa pemrograman java, serta memahami tentang sintak dasar pada pemrograman java.

---

#### 1. Tujuan

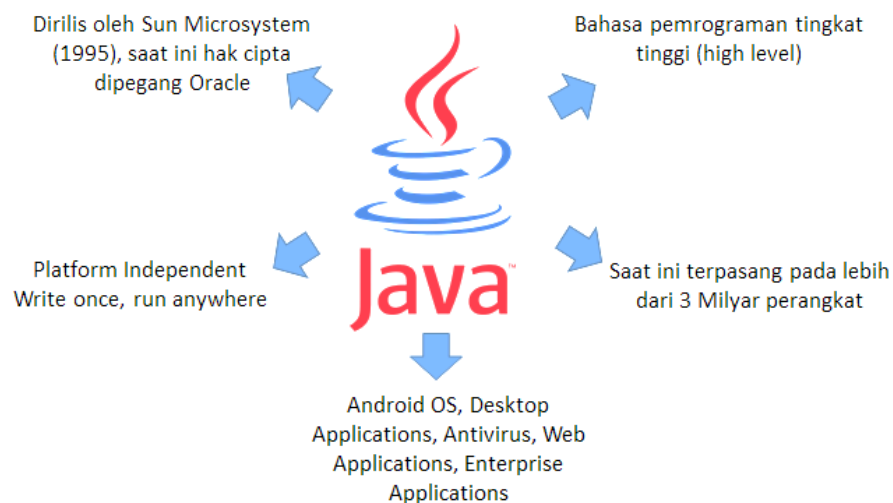
- ❖ Mahasiswa mengetahui tentang konsep bahasa pemrograman java
- ❖ Mahasiswa mengetahui tentang kelebihan dan kelemahan bahasa pemrograman java
- ❖ Mahasiswa mengetahui tentang sintak dasar pemrograman java
- ❖ Mahasiswa mengetahui tentang kelebihan penggunaan sintak dasar pada bahasa pemrograman java.

#### 2. Tugas Pendahuluan

- ❖ Jelaskan konsep bahasa pemrograman java
- ❖ Ketikkan contoh sintak dasar pada pemrograman java berikut ini.
- ❖ Jelaskan maksud sintak dasar pemrograman java

#### 3. Teori

#### Java, Basic Concepts & Syntax



The Java Development Kit (**JDK**) adalah perangkat pengembangan aplikasi berbasis java. JDK berisi :

#### 1. The Java Compiler (**javac**)

2. The **Java Archiving Tool (jar)**
3. The **Java Debugging Tool (jdb)**

Setelah menginstall JDK, kita akan dapat menjalankan **javac**, **jar** dan **jdb** dari *console* (*Command Prompt* pada *Windows*).

JDK dapat diperoleh langsung dari situs oracle :

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

## Java | Instalasi JDK

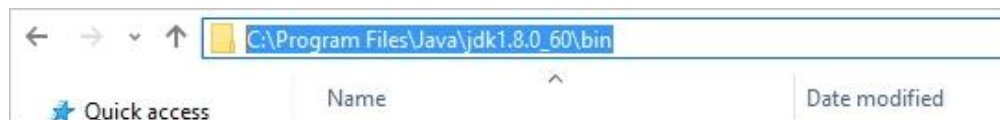
### JDK & PATH

Pada sistem operasi Windows, JDK akan terinstall secara default pada direktori :

"C:\Program Files\Java\jdk1.8.0\_XX"

Dimana **XX** adalah nomor seri versi upgrade

Buka direktori dimaksud, buka folder bin dan salin alamat path yang ada untuk digunakan pada langkah selanjutnya. Lakukan instalasi JDK kemudian ikuti langkah berikut :



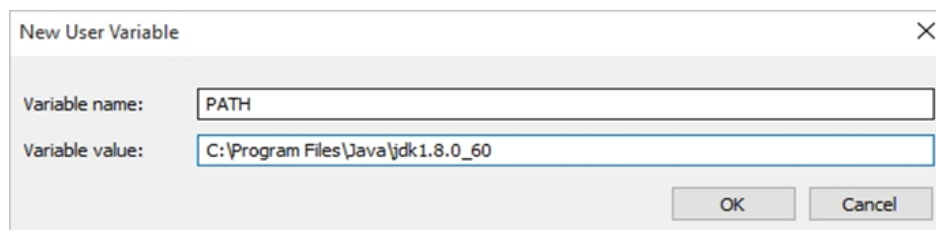
Lakukan instalasi JDK kemudian ikuti langkah berikut :

Klik kanan pada **My Computer** pilih **Properties** -> **Advanced system settings**.

Pilih **Advanced tab** -> **Environment Variables**

Bila **PATH** belum ada pada daftar, pilih **New** kemudian isi seperti pada gambar.

Bila **PATH** sudah ada, klik edit kemudian tambahkan hasil salinan setelah nilai sebelumnya dengan pemisah semicolon (;)



## Java | Java Virtual Machine

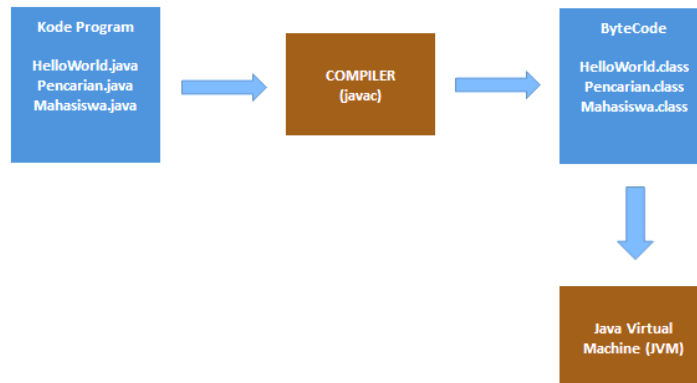
Java Virtual Machine (JVM) adalah 'mesin' virtual yang mampu mengeksekusi aplikasi berbasis Java.

JVM menerjemahkan Java Bytecode agar prosesor pada suatu perangkat dapat menjalankan instruksi dari program berbasis Java.

JVM memungkinkan suatu kode dapat dieksekusi pada platform yang berbeda

- ❖ JVM adalah komponen utama dari arsitektur Java, dan merupakan bagian dari JRE (Java Runtime Environment).

- ❖ JVM bergantung kepada jenis sistem operasi. JVM harus menerjemahkan bytecode ke dalam bahasa mesin, di sisi lain bahasa mesin bergantung kepada sistem operasi yang digunakan sehingga JVM bergantung kepada mesin atau sistem operasi yang digunakan.
- ❖ JVM bertanggung jawab dalam mengalokasikan memori yang dibutuhkan oleh program berbahasa Java



Ketik kode seperti pada gambar kemudian simpan dengan nama HelloWorld.java

```

File Edit Format View Help
public class HelloWorld {
    public static void main(String[] args){
        System.out.println("Hello World!!");
    }
}
  
```

Buka command prompt, jalankan perintah untuk kompilasi (javac) dan kemudian jalankan program HelloWorld

```

C:\Users\STUKTURAL01\Desktop>javac HelloWorld.java
C:\Users\STUKTURAL01\Desktop>java HelloWorld
Hello World!!
C:\Users\STUKTURAL01\Desktop>
  
```

Dalam bahasa pemrograman Java kita dapat menambahkan komentar pada kode yang kita buat. Seluruh karakter/kata yang ditandai sebagai komentar, tidak akan dibaca oleh compiler.

***Contoh komentar dengan baris tunggal (single-line)***

```

File Edit Format View Help
public class HelloWorld {
    // ini program Java pertamaku
    public static void main(String[] args){
        System.out.println("Hello World!!");
    }
}
  
```

***Contoh komentar dengan baris jamak(multi-line)***



```
File Edit Format View Help
/* Ini adalah contoh program Java sederhana
Program ini menuliskan kata Hello World!! ke console
Tulisan ini ada di dalam blok komentar
Tulisan di dalam blok komentar diabaikan oleh compiler */
public class HelloWorld {
    public static void main(String[] args){
        System.out.println("Hello World!!");
    }
}
```

Menuliskan komentar pada kode sangat dianjurkan karena akan memudahkan kita saat harus membaca ulang kode yang telah kita tulis sendiri, dan orang lain saat harus membaca kode kita saat dibutuhkan

```
File Edit Format View Help
public class HelloWorld {
    // ini program Java pertamaku
    public static void main(String[] args){
        System.out.println("Hello World!!");
    }
}
```

VS

```
File Edit Format View Help
public class HelloWorldWithVar {
    public static void main(String[] args){
        String s;
        s = "Hello World!!";
        System.out.println(s);
    }
}
```

#### 4. Praktikum

##### a) Peralatan dan Bahan

- Perangkat komputer/ PC/ Laptop
- Sistem Operasi windows / Linux
- JDK
- Eclipse

##### b) Keselamatan Kerja

- baca dan pahami semua langkah kerja dari praktikum dengan cermat
- jangan membawa makanan dan minuman ke dalam laboratorium, terlebih lagi membawa benda cair.
- Pastikan diri anda terlindung dari efek kejutan listrik saat menyentuh perangkat komputer

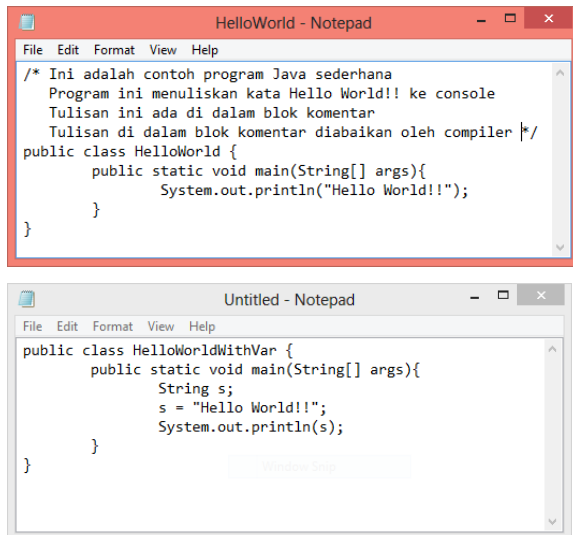
##### c) Langkah Kerja

- Siapkan alat dan bahan praktek
- Selalu memperhatikan keselamatan kerja
- Jalankan perangkat menggunakan editor eclipse
- Organisasikan dengan baik direktori penyimpanan anda untuk setiap modul praktikum.
- Ketikkan contoh syntax dasar pada editor eclipse sesuai dengan contoh.

#### 5. Metode Menuliskan syntax dasar pada pemrograman Java

1. Buka aplikasi eclipse kemudian atur workspace untuk menentukan direktory penyimpanan project, pastikan pilih direktory D/E.
2. Setelah jendela eclipse terbuka, buat project baru dengan cara klik menu file → New project.
3. Buat nama project dengan menyesuaikan pembahasan pada modul pertama.

4. Di dalam project buat package untuk membagi project menjadi beberapa bagian dengan tujuan supaya dalam satu project dapat diisi dengan lebih dari satu contoh program.
5. Di dalam package buatlah class baru untuk memulai mengetikkan latihan dasar pemrograman.
6. Ketikkan contoh seperti di bawah ini pada halaman class yang sudah dibuat.



7. klik icon Run pada toolbar eclipse untuk melihat hasil pada layar konsol.

## 6. Evaluasi Dan Pertanyaan

Jelaskan maksud penggalan kode program yang telah di praktekkan.

## 7. Studi Kasus

Buatlah contoh program menggunakan sintak java untuk menampilkan biodata masing – masing yang terdiri dari :

Nama Mahasiswa : xxxxxxxxxxxxxx  
 NIM : xxxxx  
 Program Studi : xxxxxxxxxxxx  
 Alamat : xxxxxxxxxxx  
 Tempat Lahir : xxxxxxxxxxx  
 Tanggal Lahir : xxxxxxxxxxx

## 8. Kesimpulan

## 9. Hasil Review

Date	Result (1-5)	Signature
------	--------------	-----------

--	--	--

## MODUL 3

### Variabel dan Type Data

#### KOMPETENSI

Setelah menyelesaikan modul ini mahasiswa diharapkan dapat memahami tentang variabel, type data, fungsi variabel dan type data serta cara penulisan dan deklasri pada pemrograman java..

#### 1. Tujuan

- ❖ Mahasiswa mengetahui tentang macam – macam type data pada bahasa pemrograman java
- ❖ Mahasiswa mengetahui tentang penulisan variabel pada bahasa pemrograman java.
- ❖ Mahasiswa mengetahui tentang deklarasi variabel dan type data pada pemrograman java.

#### 2. Tugas Pendahuluan

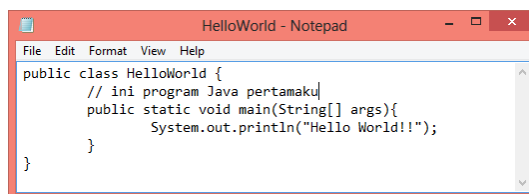
- ❖ Sebutkan dan Jelaskan macam – macam type data pada pemrograman java
- ❖ Tuliskan cara penulisan variabel yang benar pada bahasa pemrograman java
- ❖ Ketikkan cara deklarasi variabel dan type data pada java

#### 3. Teori

## Java | Variabel & Tipe Data

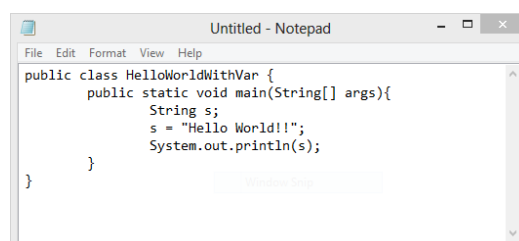
Variabel kita gunakan untuk menyimpan suatu nilai (data) untuk pemrosesan lebih lanjut. Variabel memiliki nama (atau disebut *identifier*), misalnya: nama, usia, alamat dan sejenisnya sesuai dengan data yang diwakili. Nama dari setiap variabel itu harus unik dan akan digunakan saat kita memberi atau mengambil nilai dari variabel tersebut. Pada bahasa pemrograman Java, variabel memiliki tipe data tertentu dan hanya bisa menyimpan nilai sesuai dengan tipe data tersebut.

Contoh sederhana dari penggunaan variabel dan tipe data.



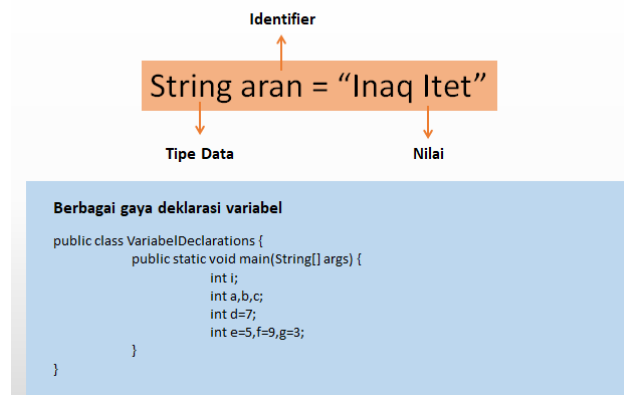
```
public class HelloWorld {
    // ini program Java pertamaku
    public static void main(String[] args){
        System.out.println("Hello World!!");
    }
}
```

VS



```
public class HelloWorldWithVar {
    public static void main(String[] args){
        String s;
        s = "Hello World!!";
        System.out.println(s);
    }
}
```

## Java | Deklarasi Variabel (lanj.)



Tipe data	bit	Min	Max	Default	Example
byte	8	-128	127	0	byte a = 100, byte b = -50
Short	16	-32.768	32,767	0	short s = 10000, short r = -20000
Int	32	2,147,483,648	2,147,483,647	0	int a = 100000, int b = -200000
long	64	$-(2^{63})$	$2^{63}-1$	0L	long a = 100000L, long b = -200000L

### Variables

Variables have types. Some examples:

- `int`: for integers (whole numbers) such as 123 and -456
- `double`: for floating-point or real numbers with optional decimal points and fractional parts in fixed or scientific notations, such as 3.1416, -55.66.
- `String`: for texts such as "Hello" or "Good Morning!". Text strings are enclosed within double quotes.

You can declare a variable of a type and assign it a value. Example:

```
String name = "David";
```

This creates a variable called name of type String, and assigns it the value "David".

It is important to note that a variable is associated with a type, and is only capable of storing values of that particular type. For example, an `int` variable can store integer values, such as 123; but it cannot store real numbers, such as 12.34, or texts, such as "Hello".

```

public class VariabelLokalVsVariabelClass {
    static int i=1000;//cth deklarasi variabel class
    public static void main(String[] args) {
        System.out.println(i);
        int i = 20;//cth deklarasi variabel lokal
        System.out.println(i);
        int x = i + i;
        int y = i + VariabelLokalVsVariabelClass.i;
        System.out.println(x+y);
    }
}

public class VariabelDanTipeData {

    public static void main(String[] args) {

        //cara mendeklarasikan variabel
        int i;//deklarasi variabel bernama i dengan tipe data int (integer : 1,2,3,4,5,6...)
        String s;//deklarasi variabel bernama s dengan tipe data String (teks)
        char c;//deklarasi variabel bernama c dengan tipe data char (karakter, satu huruf)
        boolean b;//deklarasi variabel bernama b dengan tipe data boolean (true/false)
        double d;//deklarasi variabel bernama d dengan tipe data double

        //cara memberikan nilai pada variabel (assignment)
        i = 10; //variabel i diisi dengan nilai 10
        s = "Ini adalah nilai variabel 2"; //variabel String dapat diisi dengan teks yang diapit tanda " " (kutip 2)
        c = 'A'; //variabel char dapat diisi SATU karakter yang diapit tanda ' ' (kutip 1)
        b = true; //variabel boolean dapat diisi nilai true atau false
        d = 3.435; //variabel double dapat diisi nilai pecahan/desimal

        //tampilkan nilai variabel-variabel di atas
        System.out.println("nilai i adalah:"+i);
        System.out.println("nilai s adalah:"+s);
        System.out.println("nilai c adalah:"+c);
        System.out.println("nilai b adalah:"+b);
        System.out.println("nilai d adalah:"+d);
    }
}

```

#### 4. Praktikum

##### a) Peralatan dan Bahan

- Perangkat komputer/ PC/ Laptop
- Sistem Operasi windows / Linux
- JDK
- Eclipse

##### b) Keselamatan Kerja

- baca dan pahami semua langkah kerja dari praktikum dengan cermat
- jangan membawa makanan dan minuman ke dalam laboratorium, terlebih lagi membawa benda cair.
- Pastikan diri anda terlindung dari efek kejutan listrik saat menyentuh perangkat komputer

##### c) Langkah Kerja

- Siapkan alat dan bahan praktek

- Selalu memperhatikan keselamatan kerja
- Jalankan perangkat menggunakan editor eclipse
- Organisasikan dengan baik direktori penyimpanan anda untuk setiap modul praktikum.
- Ketikkan contoh syntax dasar pada editor eclipse sesuai dengan contoh.
- Ketikkan contoh syntax bahasa pemrograman java dengan menggunakan fungsi variabel dan type data, kemudian bandingkan perbedaannya dengan syntax dasar tanpa variabel dan type data.

## 5. Metode Menuliskan Variabel dan Type data pada pemrograman Java

1. Buka aplikasi eclipse kemudian atur workspace untuk menentukan direktory penyimpanan project, pastikan pilih direktory D/E.
2. Setelah jendela eclipse terbuka, buat project baru dengan cara klik menu file → New project.
3. Buat nama project dengan menyesuaikan pembahasan pada modul pertama.
4. Di dalam project buat package untuk membagi project menjadi beberapa bagian dengan tujuan supaya dalam satu project dapat diisi dengan lebih dari satu contoh program.
5. Di dalam package buatlah class baru untuk memulai mengetikkan latihan dasar pemrograman.
6. Ketikkan contoh seperti di bawah ini pada halaman class yang sudah dibuat.

```
public class VariabelDanTipeData {

    public static void main(String[] args) {

        //cara mendeklarasikan variabel
        int i;//deklarasi variabel bernama i dengan tipe data int (integer : 1,2,3,4,5,6....)
        String s;//deklarasi variabel bernama s dengan tipe data String (teks)
        char c;//deklarasi variabel bernama c dengan tipe data char (karakter, satu huruf)
        boolean b;//deklarasi variabel bernama b dengan tipe data boolean (true/false)
        double d;//deklarasi variabel bernama d dengan tipe data double

        //cara memberikan nilai pada variabel (assignment)
        i = 10; //variabel i diisi dengan nilai 10
        s = "Ini adalah nilai variabel 2"; //variabel String dapat diisi dengan teks yang diapit tanda " " (kutip 2)
        c = 'A'; //variabel char dapat diisi SATU karakter yang diapit tanda ' ' (kutip 1)
        b = true; //variabel boolean dapat diisi nilai true atau false
        d = 3.435; //variabel double dapat diisi nilai pecahan/desimal

        //tampilkan nilai variabel-variabel di atas
        System.out.println("nilai i adalah:"+i);
        System.out.println("nilai s adalah:"+s);
        System.out.println("nilai c adalah:"+c);
        System.out.println("nilai b adalah:"+b);
        System.out.println("nilai d adalah:"+d);

    }

}
```

## 6. Evaluasi Dan Pertanyaan

- ❖ Sebutkan dan jelaskan macam – macam type data pada pemrograman java
- ❖ Ketikkan deklarasi dan pengisian variabel dan type data pada pemrograman java..

## 7. Studi Kasus

Buatlah contoh program menggunakan sintak java untuk menampilkan biodata masing – masing yang terdiri dari :

Nama Mahasiswa : xxxxxxxxxxxxxx

NIM : xxxxx

Program Studi : xxxxxxxxxxxxxx

Alamat : xxxxxxxxxxx

Tempat Lahir : xxxxxxxxxxx

Tanggal Lahir : xxxxxxxxxxx

Gunakan variabel dan type data untuk setiap isian biodata yang dibuat.

## 8. Kesimpulan

--

## 9. Hasil Review

Date	Result (1-5)	Signature



## MODUL 4

### Operator, Ekspresi, Assignment, & Statement

#### KOMPETENSI

Setelah menyelesaikan modul ini mahasiswa diharapkan dapat memahami tentang Operator dan jenis – jenisnya, ekspresi, assignment dan statement pada pemrograman java.

#### 1. Tujuan

- ❖ Mahasiswa mampu memahami tentang macam – macam operator dan jenis operator yang digunakan pada pemrograman java
- ❖ Mahasiswa mampu memahami fungsi ekspresi pada bahasa pemrograman java.
- ❖ Mahasiswa mampu memahami fungsi assignment pada bahasa pemrograman java.
- ❖ Mahasiswa mampu memahami fungsi statement pada bahasa pemrograman java.

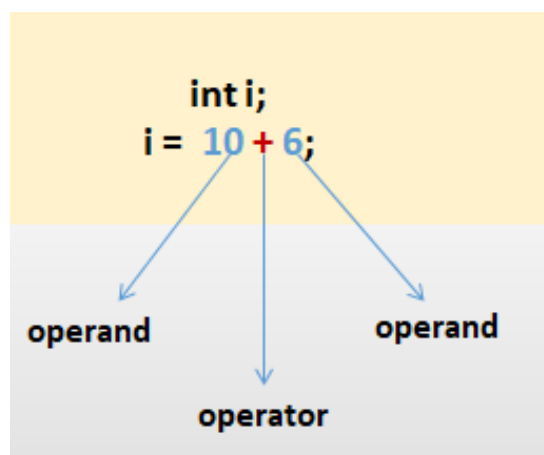
#### 2. Tugas Pendahuluan

- ❖ Sebutkan macam – macam operator pada java
- ❖ Tuliskan deklarasi operator pada pemrograman java
- ❖ Sebutkan fungsi dan cara penggunaan ekspresi dan assignment pada pemrograman java
- ❖ Tuliskan cara penggunaan statement pada bahasa java.

#### 3. Teori

##### Operator | Intro

Operator merupakan symbol pada Java yang digunakan untuk melakukan operasi tertentu secara spesifik. Operator dapat digunakan untuk memanipulasi variabel. Pada contoh di bawah, tanda “+” merupakan operator, sementara nilai 10 dan 6 disebut dengan *operand*.



## Jenis – Jenis Operator pada java



### ❖ Operator Aritmatika

Operator aritmatika terdiri atas simbol-simbol untuk operasi matematis yakni:

- + penambahan, misal `int i = 3 + 4;` maka `i = 7`
- - pengurangan, misal `int i = 3 - 4;` maka `i = -1`
- \* perkalian, misal `int i = 3 * 4;` maka `i = 12`
- / pembagian, misal `int i = 3 / 4;` maka `i = 0`
- % modulus (sisa pembagian), misal `int i = 3 % 4;` maka `i = 3`

Contoh Penerapan

```
double d1,d2,d3,d4,d5;
```

```
d1 = 5.0;
```

```
d2 = 3.0;
```

```
d3 = d1 * d2;
```

```
d4 = d1 / d2;
```

```
d5 = d1 % d2;
```

```
System.out.println("hasil perkalian bil1 dan bil2 adalah:"+d3);
```

```
System.out.println("hasil pembagian bil1 dan bil2 adalah:"+d4);
```

```
System.out.println("sisa pembagian bil1 dan bil2 adalah:"+d5);
```

## Contoh Kasus

```
import java.util.Scanner;
public class OperatorAritmatika {
    public static void main(String[] args) {
        int angka1;
        int angka2;
        int hasil;

        Scanner keyboard = new Scanner(System.in);
        System.out.print("Input angka-1: ");
        angka1 = keyboard.nextInt();
        System.out.print("Input angka-2: ");
        angka2 = keyboard.nextInt();

        // penjumlahan
        hasil = angka1 + angka2;
        System.out.println("Hasil = " + hasil);
        System.out.print("Input angka-1: ");
        angka1 = keyboard.nextInt();
        System.out.print("Input angka-2: ");
        angka2 = keyboard.nextInt();

        // pengurangan
        hasil = angka1 - angka2;
        System.out.println("Hasil = " + hasil);
        System.out.print("Input angka-1: ");
        angka1 = keyboard.nextInt();
        System.out.print("Input angka-2: ");
        angka2 = keyboard.nextInt();

        // perkalian
        hasil = angka1 * angka2;
        System.out.println("Hasil = " + hasil);
        System.out.print("Input angka-1: ");
        angka1 = keyboard.nextInt();
        System.out.print("Input angka-2: ");
        angka2 = keyboard.nextInt();

        // Pembagian
        hasil = angka1 / angka2;
        System.out.println("Hasil = " + hasil);
        System.out.print("Input angka-1: ");
        angka1 = keyboard.nextInt();
        System.out.print("Input angka-2: ");
        angka2 = keyboard.nextInt();

        // Sisa Bagi
        hasil = angka1 % angka2;
        System.out.println("Hasil = " + hasil);    }
```

## ❖ Operator Penugasan

Operator penugasan (*Assignment Operator*) fungsinya untuk memberikan tugas pada variabel tertentu. Biasanya untuk mengisi nilai.

Contoh: `int a = 10;`

Variabel `a` ditugaskan untuk menyimpan nilai `10`.

Operator Penugasan terdiri dari:

Nama Operator	Symbol
Pengisian Nilai	=
Pengisian dan Penambahan	+=
Pengisian dan Pengurangan	-=
Pengisian dan Perkalian	*=
Pengisian dan Pembagian	/=
Pengisian dan Sisa bagi	%=

```
public class OperatorPenugasan {
    public static void main(String[] args) {
        int a;
        int b;
        // Pengisian nilai
        a = 5;
        b = 10;
        // penambahan
        b += a;
        // sekarang b = 15
        System.out.println("Penambahan : " + b);
        // pengurangan
        b -= a;
        // sekarang b = 10 (karena 15-5)
        System.out.println("Pengurangan : " + b);
        // perkalian
        b *= a;
        // sekarang b = 50 (karena 10*5)
        System.out.println("Perkalian : " + b);
        // Pembagian
        b /= a;
        // sekarang b=10
        System.out.println("Pembagian : " + b);
        // Sisa bagi
        b %= a;
        // sekarang b=0
        System.out.println("Sisa Bagi : " + b);
    }
}
```

## ❖ Operator Logika

Nama	Simbol di Java
Logika AND	&&
Logika OR	
Negasi/kebalikan	!

Operator Logika digunakan untuk membuat operasi logika.

Misalnya seperti ini:

- Pernyataan 1: Petani Kode seorang programmer
- Pernyataan 2: Petanikode menggunakan Linux

Jika ditanya, apakah Petani Kode programmer yang menggunakan Linux?

Tentu kita akan cek dulu kebenarannya

- Pernyataan 1: Petani Kode seorang programmer = **true**.
- Pernyataan 2: Petanikode menggunakan Linux = **true**.

Apa petanikode programmer dan menggunakan Linux?

*Pernyataan 1 && Pernyataan 2 = true*

tabel kebenaran untuk logika AND.

Pernyataan 1	Pernyataan 2	Logika AND
true	true	true
true	false	false
false	true	false
false	false	false

## ❖ Operator Bitwise

Operator bitwise merupakan operator yang digunakan untuk operasi bit (biner). Operator bitwise terdiri dari:

Nama	Simbol di Java
AND	&
OR	
XOR	^
Negasi/kebalikan	~
Left Shift	<<
Right Shift	>>
Left Shift (unsigned)	<<<
Right Shift (unsigned)	>>>

Operator ini berlaku untuk tipe data **int**, **long**, **short**, **char**, dan **byte**. Operator ini akan menghitung dari bit-ke-bit.

Misalnya, kita punya variabel **a = 60** dan **b = 13**. Bila dibuat dalam bentuk biner, akan menjadi seperti ini:

$$a = 00111100$$

$b = 00001101$

Kemudian, dilakukan operasi bitwise Operasi AND

$a = 00111100$

$b = 00001101$

$a \& b = 00001100$

Contoh dalam Pemrograman Java

```
public class OperatorBitwise {
    public static void main(String[] args) {
        int a = 60; /* 60 = 0011 1100 */
        int b = 13; /* 13 = 0000 1101 */
        int c = 0;

        c = a & b; /* 12 = 0000 1100 */
        System.out.println("a & b = " + c);
        c = a | b; /* 61 = 0011 1101 */
        System.out.println("a | b = " + c);

        c = a ^ b; /* 49 = 0011 0001 */
        System.out.println("a ^ b = " + c);

        c = ~a; /* -61 = 1100 0011 */
        System.out.println("~a = " + c);

        c = a << 2; /* 240 = 1111 0000 */
        System.out.println("a << 2 = " + c);

        c = a >> 2; /* 215 = 1111 */
        System.out.println("a >> 2 = " + c);
        c = a >>> 2; /* 215 = 0000 1111 */
        System.out.println("a >>> 2 = " + c);
    }
}
```

### ❖ Operator Ternary

Operator ini unik, seperti membuat pertanyaan.

Simbolnya menggunakan tanda tanya (?) dan titik-dua (:) untuk memisah jawabannya.

```
public class OperatorTernary {
    public static void main(String[] args) {
        boolean suka = true;
        String jawaban;

        // menggunakan operator ternary
        jawaban = suka ? "iya" : "tidak";

        // menampilkan jawaban
        System.out.println(jawaban);
    }
}
```

#### 4. PRAKTIKUM

##### a) Peralatan dan Bahan

- Perangkat komputer/ PC/ Laptop
- Sistem Operasi windows / Linux
- JDK
- Eclipse

##### b) Keselamatan Kerja

- baca dan pahami semua langkah kerja dari praktikum dengan cermat
- jangan membawa makanan dan minuman ke dalam laboratorium, terlebih lagi membawa benda cair.
- Pastikan diri anda terlindung dari efek kejutan listrik saat menyentuh perangkat komputer

##### c) Langkah Kerja

- Siapkan alat dan bahan praktek
- Selalu memperhatikan keselamatan kerja
- Jalankan perangkat menggunakan editor eclipse
- Organisasikan dengan baik direktori penyimpanan anda untuk setiap modul praktikum.
- Ketikkan contoh sintak dasar pada editor eclipse sesuai dengan contoh.
- Ketikkan contoh sintak bahasa pemrograman java dengan menggunakan fungsi variabel dan type data, kemudian bandingkan perbedaannya dengan sintak dasar tanpa variabel dan type data.

#### 5. Metode penggunaan Operator pada java

- ❖ Buka aplikasi eclipse kemudian atur workspace untuk menentukan direktory penyimpanan project, pastikan pilih direktory D/E.
- ❖ Setelah jendela eclipse terbuka, buat project baru dengan cara klik menu file → New project.
- ❖ Buat nama project dengan menyesuaikan pembahasan pada modul pertama.
- ❖ Di dalam project buat package untuk membagi project menjadi beberapa bagian dengan tujuan supaya dalam satu project dapat diisi dengan lebih dari satu contoh program.
- ❖ Di dalam package buatlah class baru untuk memulai mengetikkan latihan dasar pemrograman.
- ❖ Ketikkan contoh program penggunaan operator pada halaman class yang sudah dibuat.

#### 6. Evaluasi dan pertanyaan

- ❖ Jelaskan macam – macam operator pada java
- ❖ Ketikkan contoh penggunaan operator pada java

#### 7. Studi kasus

- ❖ Ketikkan contoh penggunaan operator pada java sesuai dengan contoh sintak yang sudah dipaparkan pada materi praktikum.

**8. Kesimpulan**

--

**9. Hasil Review**

Date	Result (1-5)	Signature



## MODUL 5

### Input & Output

#### KOMPETENSI

Setelah menyelesaikan modul ini, mahasiswa diharapkan dapat mengetahui manfaat fungsi scanner untuk input data melalui media keyboard, serta dapat menerapkannya pada pemrograman java untuk input dan output sederhana.

---

#### 1. TUJUAN

- ❖ Mahasiswa mampu memahami fungsi scanner
- ❖ Mahasiswa mampu menerapkan fungsi scanner sebagai input data pada pemrograman java
- ❖ Mahasiswa mampu membuat contoh program sederhana dengan menerapkan input dan output sederhana.

#### 2. TUGAS PENDAHULUAN

- ❖ Jelaskan fungsi perintah scanner pada pemrograman java.
- ❖ Tuliskan sintak untuk penggunaan fungsi scanner pada java
- ❖ Ketikkan contoh sederhana untuk menampilkan biodata dengan memanfaatkan fungsi scanner sebagai input data.

#### 3. TEORI

##### Input dan Output sederhana

##### Perintah Scanner pada bahasa pemrogram Java

---

*Scanner* merupakan *class* yang menyediakan fungsi-fungsi untuk mengambil input dari keyboard.

Agar kita bisa menggunakan Scanner, kita perlu mengimpornya ke dalam kode:

```
import java.util.Scanner;
```

Perlu diperhatikan, penggunaan fungsi untuk mengambil data bergantung dari tipe data yang digunakan.

Misal, tipe datanya adalah *String*, maka fungsi atau *method* yang dipakai adalah *next()*. Begitu juga dengan tipe data lain, *Integer* menggunakan *nextInt()*, *Double* menggunakan *nextDouble()*, dsb.

```
package pertemuan2;
// mengimpor Scanner ke program
import java.util.Scanner;

public class DataKaryawan {
    public static void main(String[] args) {
        // deklarasi variabel
        String nama, alamat;
        int usia, gaji;
    }
}
```

```

// membuat scanner baru
Scanner keyboard = new Scanner(System.in);

// Tampilkan output ke user
System.out.println("### Pendataan Karyawan PT. Petani Kode ###");
System.out.print("Nama karyawan: ");
// menggunakan scanner dan menyimpan apa yang diketik
nama = keyboard.next();
// Tampilkan output lagi
System.out.print("Alamat: ");
// menggunakan scanner lagi
alamat = keyboard.next();
System.out.print("Usia: ");
usia = keyboard.nextInt();
System.out.print("Gaji: ");
gaji = keyboard.nextInt();

// Menampilkan apa yang sudah disimpan di variabel
System.out.println("-----");
System.out.println("Nama Karyawan: " + nama);
System.out.println("Alamat: " + alamat);
System.out.println("Usia: " + usia + " tahun");
System.out.println("Gaji: Rp " + gaji);
}
}

```

Perlu diperhatikan, menggunakan fungsi untuk mengambil data bergantung dari tipe data yang digunakan. Misal, tipe datanya adalah *String*, maka fungsi atau *method* yang dipakai adalah `next()`. Begitu juga dengan tipe data lain, *Integer* menggunakan `nextInt()`, *Double* menggunakan `nextDouble()`, dsb.

### Menampilkan Output

Kita sudah mengenal beberapa cara mengambil input dari keyboard untuk program berbasis teks.

Sekarang bagaimana dengan outputnya?

Ada beberapa fungsi yang sudah disediakan oleh Java:

1. Fungsi `System.out.print()`
2. Fungsi `System.out.println()`
3. Fungsi `System.out.format()`

Apa saja perbedaan dari fungsi-fungsi tersebut?

Fungsi `print()` vs `println()`

Fungsi `print()` dan `println()` sama-sama digunakan untuk menampilkan teks.

Lalu apa bedanya?

Fungsi `print()` akan menampilkan teks apa adanya. Sedangkan `println()` akan menampilkan teks dengan ditambah baris baru.

### Menggabungkan String

Ketika menggunakan fungsi `print()` maupun `println()`, kadang kita perlu mengambil teks dari variabel dan menggabungkannya dengan teks yang lain.

#### 4. PRAKTIKUM

##### a) Peralatan dan Bahan

- Perangkat komputer/ PC/ Laptop
- Sistem Operasi windows / Linux
- JDK
- Eclipse

##### b) Keselamatan Kerja

- baca dan pahami semua langkah kerja dari praktikum dengan cermat
- jangan membawa makanan dan minuman ke dalam laboratorium, terlebih lagi membawa benda cair.
- Pastikan diri anda terlindung dari efek kejutan listrik saat menyentuh perangkat komputer

##### c) Langkah Kerja

- Siapkan alat dan bahan praktek
- Selalu memperhatikan keselamatan kerja
- Jalankan perangkat menggunakan editor eclipse
- Organisasikan dengan baik direktori penyimpanan anda untuk setiap modul praktikum.
- Ketikkan contoh syntax dasar pada editor eclipse sesuai dengan contoh.
- Ketikkan contoh syntax bahasa pemrograman java dengan menggunakan fungsi Scanner sebagai fungsi untuk memasukkan data serta gunakan fungsi untuk menampilkan data menggunakan print dan println.

#### 5. Metode Menuliskan Perintah Scanner, Print() dan Println() dasar pada pemrograman Java

- Buka aplikasi eclipse kemudian atur workspace untuk menentukan direktory penyimpanan project, pastikan pilih direktory D/E.
- Setelah jendela eclipse terbuka, buat project baru dengan cara klik menu file → New project.
- Buat nama project dengan menyesuaikan pembahasan pada modul pertama.
- Di dalam project buat package untuk membagi project menjadi beberapa bagian dengan tujuan supaya dalam satu project dapat diisi dengan lebih dari satu contoh program.
- Di dalam package buatlah class baru untuk memulai mengetikkan latihan dasar pemrograman.
- Ketikkan contoh seperti di bawah ini pada halaman class yang sudah dibuat.

#### 6. Evaluasi dan pertanyaan

- Jelaskan fungsi dari perintah scanner pada java
- Jelaskan perbedaan fungsi print dengan println pada java
- Ketikkan contoh program menggunakan fungsi scanner dan juga perintah output print dan println
- Amati kemudian jelaskan perbedaan fungsi print dan println.

## 7. Studi kasus

Tulis program ini pada IDE favorit anda kemudian jalankan

```
1 package org.stth.dasprog.io;
2
3 import java.util.Scanner;
4
5 public class SimpleConsoleInput {
6     public static void main(String[] args) {
7         Scanner scan = new Scanner(System.in);
8         System.out.print("Masukkan nama anda");
9         String nama = scan.nextLine();
10        System.out.println("Nama anda adalah "+ nama);
11        scan.close();
12    }
13 }
14 }
15 }
```

Import class Scanner dari java.util

Menyiapkan scanner untuk membaca input dari user

Membaca baris input dari user kemudian menyimpan nilainya pada variabel name

```
1 package org.stth.dasprog.io;
2
3 import java.util.Scanner;
4
5 public class SimpleConsoleInputDua {
6     public static void main(String[] args) {
7         Scanner scan = new Scanner(System.in);
8         System.out.print("Masukkan nama anda ");
9         String nama = scan.nextLine();
10
11        System.out.print("Masukkan tinggi badan anda dalam centimeter ");
12        String stringTinggiBadan = scan.nextLine();
13
14        System.out.print("Masukkan berat badan anda dalam kilogram ");
15        String stringBeratBadan = scan.nextLine();
16
17        scan.close();
18
19        Double tinggiBadan = Double.parseDouble(stringTinggiBadan)/100; // karena rumus BMI menggunakan meter, bukan cm
20        Double beratBadan = Double.parseDouble(stringBeratBadan);
21
22        double bmi = beratBadan/(tinggiBadan*tinggiBadan);
23        System.out.println("Nama anda adalah "+ nama);
24        System.out.println("Tinggi badan anda adalah "+ stringTinggiBadan + " cm");
25        System.out.println("Berat badan anda adalah "+ stringBeratBadan + " kg");
26        System.out.println("Nilai indeks BMI anda adalah "+ bmi );
27    }
28 }
29 }
30 }
31 }
```

**Soal 1 : Buatlah program yang menerima user input berupa :**

1. Nama
2. Nama mata kuliah
3. Nilai tugas dan PR
4. Nilai UTS
5. Nilai UAS
6. Persentase kehadiran

Seluruh nilai tersebut dalam skala 0-100

Simpan input tersebut pada variabel yang bersesuaian kemudian lakukan perhitungan nilai akhir dengan rumus :

**Nilai akhir = (10 \* nilai kehadiran + 25 \* nilai tugas dan PR + 30 \* nilai UTS + 35 \* nilai UAS) /100**

Kemudian tampilkan nilai akhir tersebut di layar dengan format : nama mahasiswa mendapatkan nilai akhir XXX pada mata kuliah YYYY

**Soal 2 : Buatlah program yang menerima user input berupa :**

1. Nama Pegawai
2. Jabatan
3. Masa Kerja
4. Gaji Pokok
5. Gaji Lembur
6. Bonus Bulanan

Seluruh Gaji tersebut dalam satuan Rupiah

Simpan input tersebut pada variabel yang bersesuaian kemudian lakukan perhitungan Total Gaji dengan rumus :

**Total Gaji = (Gaji Pokok + Gaji Lembur + Bonus Bulanan)**

Kemudian tampilkan Total Gaji tersebut di layar dengan format : nama pegawai dengan masa Kerja XXX mendapatkan Total Gaji XXX

**8. Kesimpulan**

--

**9. Hasil Review**

Date	Result (1-5)	Signature

## MODUL 6

### Struktur Kondisi dan Iterasi

#### KOMPETENSI

Setelah menyelesaikan modul ini, mahasiswa diharapkan dapat mengetahui fungsi struktur kondisional pada java, serta dapat memahami fungsi iterasi pada bahasa pemrograman java.

---

#### 1. TUJUAN

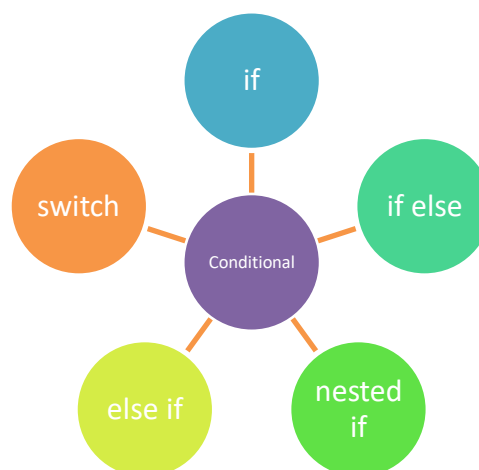
- ❖ Mahasiswa mampu memahami fungsi struktur kondisi ( if, if else, if elseif, switch case )
- ❖ Mahasiswa mampu menerapkan fungsi struktur kondisi pada pemrograman java
- ❖ Mahasiswa mampu membuat contoh program sederhana dengan menerapkan fungsi kondisi pada java
- ❖ Mahasiswa mampu memahami fungsi iterasi pada bahasa pemrograman java.
- ❖ Mahasiswa mampu menerapkan fungsi iterasi pada pemrograman java.

#### 2. TUGAS PENDAHULUAN

- Sebutkan macam – macam fungsi kondisional pada java
- Jelaskan perbedaan masing – masing fungsi kondisional pada java.
- Sebutkan macam – macam fungsi iterasi pada bahasa pemrograman java.
- Jelaskan dan berikan contoh dengan menggunakan syntak java untuk fungsi kondisional dan iterasi.

#### 3. TEORI

##### Struktur Kondisi dan Iterasi



##### a) Fungsi kondisi “ if “

“if statement” adalah statement conditional yang paling sederhana dan sering digunakan.

Jika kondisi pada ekspresi if menghasilkan nilai **true**, maka kode dalam blok akan dieksekusi, bila tidak program akan dilanjutkan ke baris kode setelah blok if.

Syntax

```
if (kondisi) {
    // kode dalam blok ini akan dieksekusi bila kondisi = true
    System.out.println("kondisi bernilai true");
}
//contoh
int usia = 15;
boolean sudahKawin = true;
if (usia > 17 || sudahKawin) {
    System.out.println("Sudah berhak memilih pada pemilu");
}
```

#### b) Fungsi kondisi “ if else “

Variasi lain adalah “**if else statement**” yang menyajikan blok kode untuk dieksekusi bila kondisi bernilai **false**.

Syntax

```
if (kondisi) {
    // kode dalam blok ini akan dieksekusi bila kondisi = true
    System.out.println("kondisi bernilai true");
} else {
    // kode dalam blok ini dieksekusi bila kondisi = false
    System.out.println("kondisi bernilai false");
}
//Contoh
boolean sudahLulus = true;
if (sudahLulus) {
    System.out.println("Boleh nikah");
} else {
    System.out.println("Nikahnya nanti dulu, tunggu wisuda");
}
```

#### c) Nested If

Java memungkinkan kita menggunakan “nested if” atau if bersarang, yang merupakan struktur kondisi if yang diletakkan di dalam struktur kondisi if lainnya

**Syntax**

```
if (kondisi) {
    System.out.println("kondisi bernilai true");
    if (kondisilain) {
```

```

        System.out.println("kondisi dan kondisi lain bernilai true");
    }
} else {
    System.out.println("kondisi bernilai false");
}

```

#### **//Contoh**

```

int jumlahEmas = 100; //jumlah dalam gram
int masaKepemilikan = 2; //periode dalam tahun
if (jumlahEmas > 85) {
    System.out.println("Cukup nisabnya");
    if (masaKepemilikan > 1) {
        System.out.println("Cukup haulnya");
    }
}

```

#### **d) Kondisional Else if**

Statement "else if" dapat digunakan untuk memeriksa beberapa kondisi sekaligus.

##### **Syntax**

```

if (kondisi) {
    //blok ini dijalankan bila kondisi = true
} else if (kondisilain) {
    //blok ini dijalankan bila kondisi = false dan kondisilain = true
} else {
    //blok ini dijalankan bila kondisi sebelumnya tidak ada yang bernilai true
}

```

#### **//Contoh**

```

double nilaiAkhir = 75.0;
char grade;
if (nilaiAkhir >= 80) {
    grade = 'A';
} else if (nilaiAkhir >= 70) {
    grade = 'B';
} else if (nilaiAkhir >= 60) {
    grade = 'C';
} else if (nilaiAkhir >= 50) {
    grade = 'D';
} else {
    grade = 'E';
}

```



#### 4. PRAKTIKUM

##### a) Peralatan dan Bahan

- Perangkat komputer/ PC/ Laptop
- Sistem Operasi windows / Linux
- JDK
- Eclipse

##### b) Keselamatan Kerja

- baca dan pahami semua langkah kerja dari praktikum dengan cermat
- jangan membawa makanan dan minuman ke dalam laboratorium, terlebih lagi membawa benda cair.
- Pastikan diri anda terlindung dari efek kejutan listrik saat menyentuh perangkat komputer

##### c) Langkah Kerja

- Siapkan alat dan bahan praktek
- Selalu memperhatikan keselamatan kerja
- Jalankan perangkat menggunakan editor eclipse
- Organisasikan dengan baik direktori penyimpanan anda untuk setiap modul praktikum.
- Ketikkan contoh sintak dasar pada editor eclipse sesuai dengan contoh.
- Ketikkan contoh sintak bahasa pemrograman java dengan menggunakan fungsi Kondisional seperti if, if else, if elseif, dan switch case pada bahasa pemrograman java.

#### 5. Metode Menuliskan Perintah if, if else, if elseif dan switch case pada pemrograman Java

- Buka aplikasi eclipse kemudian atur workspace untuk menentukan direktory penyimpanan project, pastikan pilih direktory D/E.
- Setelah jendela eclipse terbuka, buat project baru dengan cara klik menu file → New project.
- Buat nama project dengan menyesuaikan pembahasan pada modul pertama.
- Di dalam project buat package untuk membagi project menjadi beberapa bagian dengan tujuan supaya dalam satu project dapat diisi dengan lebih dari satu contoh program.
- Di dalam package buatlah class baru untuk memulai mengetikkan latihan dasar pemrograman untuk pengenalan fungsi kondisional pada java
- Ketikkan contoh seperti di bawah ini pada halaman class yang sudah dibuat.

#### 6. Evaluasi dan pertanyaan

- Sebutkan macam – macam fungsi kondisional pada java
- Jelaskan perbedaan masing – masing diantara fungsi kondisional yang telah disebutkan
- Buatlah contoh sederhana dari fungsi kondisional pada java

## 7. Studi kasus

Buatlah program sederhana dengan menerapkan fungsi kondisional pada java dengan contoh kasus mencari index masa tubuh dengan ketentuan sebagai berikut :

$$\text{IMT} = \frac{\text{Berat Badan (kg)}}{\text{Tinggi Badan (m)} \times \text{Tinggi Badan (m)}}$$

Nilai IMT	Artinya
18,4 ke bawah	Berat Badan Kurang
18,5 – 24,9	Berat Badan Ideal
25 – 29,9	Berat Badan Lebih
30 – 39,9	Gemuk
40 ke atas	Sangat Gemuk

## 8. Kesimpulan

--

## 9. Hasil Review

Date	Result (1-5)	Signature

## MODUL 7

### Fungsi Perulangan

#### KOMPETENSI

Setelah menyelesaikan modul ini, mahasiswa diharapkan dapat mengetahui fungsi perulangan pada java, yang meliputi penggunaan perintah for, while, do while, serta dapat memahami fungsi perulangan pada bahasa pemrograman java.

---

#### 1. TUJUAN

- ❖ Mahasiswa mampu memahami fungsi perulangan ( for, while, do while )
- ❖ Mahasiswa mampu menerapkan fungsi perulangan pada pemrograman java
- ❖ Mahasiswa mampu membuat contoh program sederhana dengan menerapkan fungsi perulangan pada java

#### 2. TUGAS PENDAHULUAN

- ❖ Sebutkan macam – macam fungsi perulangan pada java
- ❖ Jelaskan perbedaan masing – masing fungsi perulangan pada java
- ❖ Buat contoh sederhana untuk penggunaan fungsi perulangan pada java.

#### 3. TEORI



##### a. Fungsi Perulangan For

“for” perulangan for digunakan saat kita mengetahui berapa banyak perulangan yang akan kita lakukan.

##### Syntax

```
For ( int hitungan = 0; hitungan <= 10; hitungan++ )  
{ // blok kode yang akan diulang  
}
```

### Penjelasan:

- variabel hitungan tugasnya untuk menyimpan hitungan pengulangan.
- hitungan  $\leq 10$  artinya selama nilai hitungannya lebih kecil atau sama dengan 10, maka pengulangan akan terus dilakukan. Dengan kata lain, perulangan ini akan mengulang sebanyak 10 kali.
- hitungan++ fungsinya untuk menambah satu (+1) nilai hitungan pada setiap pengulangan.
- Blok kode *For* dimulai dengan tanda '{' dan diakhiri dengan '}'.

### Contoh Sederhana Penerapan fungsi For

```
class Bintang {  
    public static void main(String[] args) {  
        for(int i=0; i <= 5; i++) {  
            System.out.println("*****");  
        }  
    }  
}
```

### b. Fungsi Perulangan While

**while** : perulangan while digunakan pada saat kita tidak mengetahui jumlah perulangan yang akan dilakukan tapi mengetahui kondisi atau syarat dilakukan perulangan

#### Syntax

```
while ( kondisi ) {  
    // blok kode yang akan diulang  
}
```

### Penjelasan:

- *kondisi* bisa kita isi dengan perbandingan maupun variabel boolean. *Kondisi* ini hanya memiliki nilai true dan false.
- Perulangan while akan berhenti sampai *kondisi* bernilai false.

### Contoh Sederhana Penerapan fungsi while

```
public class perulangan {  
    public static void main(String args[]){  
        int x=0;  
        while (x<10) {  
            System.out.println("Nilai x =" +x);  
            x++;  
        }  
    }  
}
```

### c. Fungsi Perulangan do while

**do-while** : perulangan yang hampir sama dengan while namun, pernyataan akan dilakukan terlebih dahulu kemudian dilakukan pengecekan

#### **Syntax**

```
do {  
    // blok kode yang akan diulang  
} while (kondisi);
```

#### **Penjelasan:**

Jadi kerjakan dulu (Do), baru di cek kondisinya while( kondisi ). Kalau kondisi bernilai true, maka lanjutkan perulangan.

#### **Contoh Sederhana Penerapan fungsi do while**

```
public class PerulanganDoWhile {  
    public static void main(String[] args) {  
        // membuat variabel  
        int i = 0;  
        do {  
            System.out.println("perulangan ke-" + i);  
            i++;  
        } while ( i <= 10);  
    }  
}
```

## 4. PRAKTIKUM

### a) Peralatan dan Bahan

- Perangkat komputer/ PC/ Laptop
- Sistem Operasi windows / Linux
- JDK
- Eclipse

### b) Keselamatan Kerja

- baca dan pahami semua langkah kerja dari praktikum dengan cermat
- jangan membawa makanan dan minuman ke dalam laboratorium, terlebih lagi membawa benda cair.
- Pastikan diri anda terlindung dari efek kejutan listrik saat menyentuh perangkat komputer

### c) Langkah Kerja

- Siapkan alat dan bahan praktek
- Selalu memperhatikan keselamatan kerja
- Jalankan perangkat menggunakan editor eclipse

- Organisasikan dengan baik direktori penyimpanan anda untuk setiap modul praktikum.
- Ketikkan contoh sintak dasar pada editor eclipse sesuai dengan contoh.
- Ketikkan contoh sintak bahasa pemrograman java dengan menggunakan fungsi Perulangan yaitu for, while dan do while

#### 5. Metode Menuliskan Perintah for, while dan do while pada pemrograman Java

- Buka aplikasi eclipse kemudian atur workspace untuk menentukan direktory penyimpanan project, pastikan pilih direktory D/E.
- Setelah jendela eclipse terbuka, buat project baru dengan cara klik menu file → New project.
- Buat nama project dengan menyesuaikan pembahasan pada modul pertama.
- Di dalam project buat package untuk membagi project menjadi beberapa bagian dengan tujuan supaya dalam satu project dapat diisi dengan lebih dari satu contoh program.
- Di dalam package buatlah class baru untuk memulai mengetikkan latihan dasar pemrograman untuk pengenalan fungsi perulangan pada java
- Ketikkan contoh seperti di bawah ini pada halaman class yang sudah dibuat

#### 6. Evaluasi dan pertanyaan

- Sebutkan macam – macam fungsi perulangan pada java.
- Jelaskan perbedaan fungsi for, while dan do while .

#### 7. Studi kasus

Ketikkan contoh script di bawah ini, amati hasilnya dan berikan penjelasan.

##### Contoh 1

```
class CetakAngka{
    public static void main(String[] argumen){
        for(int i=0; i <= 10; i++){
            System.out.print( i + " ");
        }
    }
}
```

##### Contoh 2

```
class CetakBilanganGanjil{
    public static void main(String[] argumen){
        for(int i = 1; i <= 20; i += 2){
            System.out.print( i + " ");
        }
    }
}
```

### Contoh 3

```
import java.util.Scanner;
public class PerulanganWhile {
    public static void main(String[] args) {
        boolean running = true;
        int counter = 0;
        String jawab;
        Scanner scan = new Scanner(System.in);
        while( running ) {
            System.out.println("Apakah anda ingin keluar?");
            System.out.print("Jawab [ya/tidak]> ");
            jawab = scan.nextLine();
            // cek jawabannya, kalau ya maka berhenti mengulang
            if( jawab.equalsIgnoreCase("ya") ){
                running = false;
            }
            counter++;
        }
        System.out.println("Anda sudah melakukan perulangan sebanyak " + counter + " kali");
    }
}
```

### 8. Kesimpulan

--

### 9. Hasil Review

Date	Result (1-5)	Signature

## MODUL 8

### Array

#### KOMPETENSI

Setelah menyelesaikan modul ini, mahasiswa diharapkan dapat mengetahui penggunaan fungsi array pada java, yang meliputi array dimensi satu dan array dimensi dua.

---

#### 1. TUJUAN

- ❖ Mahasiswa mampu memahami fungsi array pada java
- ❖ Mahasiswa mampu fungsi array dimensi satu pada pemrograman java
- ❖ Mahasiswa mampu fungsi array dimensi dua pada pemrograman java

#### 2. TUGAS PENDAHULUAN

- ❖ Jelaskan fungsi array pada java
- ❖ Sebutkan macam – macam array pada java
- ❖ Jelaskan perbedaan array dimensi satu dan array dimensi dua.

#### 3. TEORI

##### Array

Array merupakan kumpulan variabel-variabel dengan tipe data yang sama

Array digunakan saat terdapat kebutuhan untuk menyimpan rangkaian nilai tertentu

##### Contoh kasus :

Satu kelas berisi 30 mahasiswa,

Nilai UAS setiap mahasiswa tidak perlu disimpan dalam 30 variabel berbeda, namun cukup dalam array berukuran 30 dengan tipe data **double**

##### Cara deklarasi

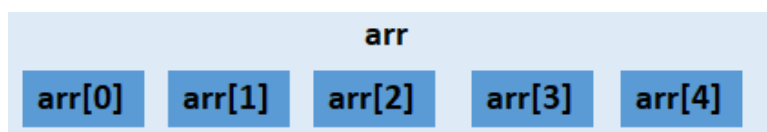
```
double[ ] nilaiUAS = new double[30];
```

Artinya : nilaiUAS merupakan array dengan tipe data double, yang mampu menampung 30 nilai

Elemen pada array diakses menggunakan indeks (nomor urut)

Indeks tersebut dimulai dari angka 0 hingga (n-1) dimana n adalah ukuran array bersangkutan

```
int[ ] arr = new int[5] → n
```





### Contoh

```
double[] nilaiUAS = new double[30]; //nilaiUAS merupakan array dengan tipe data double dengan 30 nilai
nilaiUAS[0] = 90; //nilaiUAS pada indeks ke-0 diisi dengan nilai int 90
nilaiUAS[29] = 85; //nilaiUAS pada indeks ke-29 diisi dengan nilai int 85
System.out.println(nilaiUAS[3]); // Mencetak nilai UAS ke-4 pada console
```

### Inisialisasi Array

Java menyediakan jalan pintas untuk instansiasi array dengan tipe data primitif dan string.

Jika nilainya sudah diketahui sebelumnya, kita dapat menggunakan **array literal** seperti contoh dibawah ini.

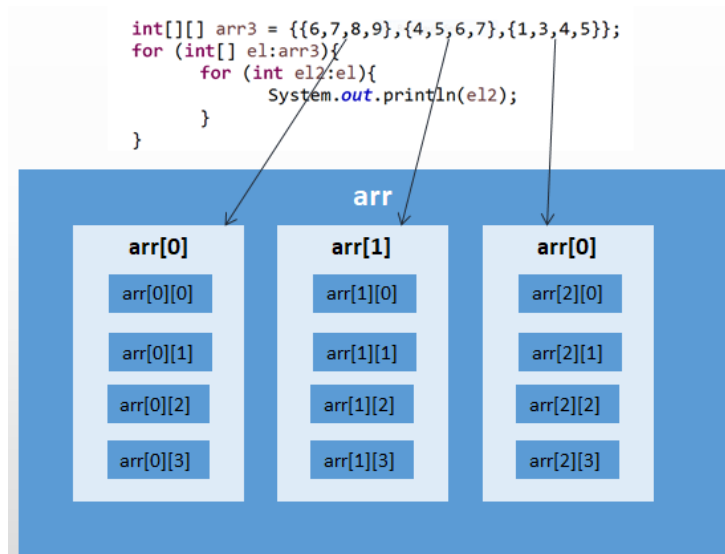
```
String[] jaje = {"keciput", "tareq", "sagon", "lepit", "tujaq"};
```

Ukuran sebuah array dapat diakses dengan menggunakan properti *length*

```
System.out.println(jaje.length); // akan menghasilkan nilai 5
```

### Array Multidimensi

Kita dapat pula memanfaatkan array multidimensi. Array multidimensi adalah array berisikan array di dalamnya (array di dalam array).



## 4. PRAKTIKUM

### a) Peralatan dan Bahan

- Perangkat komputer/ PC/ Laptop
- Sistem Operasi windows / Linux
- JDK
- Eclipse

### b) Keselamatan Kerja

- baca dan pahami semua langkah kerja dari praktikum dengan cermat
- jangan membawa makanan dan minuman ke dalam laboratorium, terlebih lagi membawa benda cair.

- Pastikan diri anda terlindung dari efek kejutan listrik saat menyentuh perangkat komputer

### c) Langkah Kerja

- Siapkan alat dan bahan praktek
- Selalu memperhatikan keselamatan kerja
- Jalankan perangkat menggunakan editor eclipse
- Organisasikan dengan baik direktori penyimpanan anda untuk setiap modul praktikum.
- Ketikkan contoh syntax dasar pada editor eclipse sesuai dengan contoh.
- Ketikkan contoh syntax bahasa pemrograman java dengan menggunakan fungsi Array

## 5. Metode Menuliskan fungsi array pada pemrograman Java

- Buka aplikasi eclipse kemudian atur workspace untuk menentukan direktory penyimpanan project, pastikan pilih direktory D/E.
- Setelah jendela eclipse terbuka, buat project baru dengan cara klik menu file → New project.
- Buat nama project dengan menyesuaikan pembahasan pada modul pertama.
- Di dalam project buat package untuk membagi project menjadi beberapa bagian dengan tujuan supaya dalam satu project dapat diisi dengan lebih dari satu contoh program.
- Di dalam package buatlah class baru untuk memulai mengetikkan latihan dasar pemrograman untuk pengenalan fungsi array pada java
- Ketikkan contoh seperti di bawah ini pada halaman class yang sudah dibuat

## 6. Evaluasi dan pertanyaan

- Sebutkan macam – macam fungsi array pada pemrograman java
- Jelaskan perbedaan dari array dimensi satu dan array dimensi dua
- Tuliskan contoh syntax dasar fungsi array pada java

## 7. Studi kasus

Ketikkan contoh perintah di bawah ini, kemudian berikan penjelasan maksud dari syntax tersebut.

```
double[] arr = {1.9, 2.9, 3.4, 3.5};
for (int i = 0; i < arr.length; i++) {
    System.out.println(arr[i]);
}
```

```
double[] arr2 = {1.9, 2.9, 3.4, 3.5};
for (double element: arr2) {
    System.out.println(element);
}
```

Case 1

Diketahui sebuah array dengan elemen : 1,3,5,3,7,6,4,5,7,8,9,2,3,4,5,6,7

1. Buat program untuk mencari nilai maksimal bilangan pada array
2. Buat program untuk mencari nilai rata-rata bilangan pada array
3. Buat program untuk mencetak bilangan pada array yang habis dibagi 2
4. Buat program untuk mencetak bilangan pada array yang juga merupakan bilangan prima
5. Buat program untuk mencari nilai 7 pada array dan berapa kali kemunculannya
6. Buat program untuk mencari frekuensi kemunculan tiap bilangan pada array
7. Buat program untuk mengurutkan nilai pada array (kecil ke besar)

Case 2

Diketahui dua buah matriks 3 X 3 sbb

1 3 5	2 0 5	Gunakan array untuk mendemonstrasikan operasi penjumlahan kedua matriks tersebut
3 4 0	7 4 1	
5 0 2	0 2 3	

8. Kesimpulan

9. Hasil Review

Date	Result (1-5)	Signature