



**MODUL PERKULIAHAN**

# **PEMROGRAMAN IV (Object I)**

Konsep Pemrograman Java, Java Platform,  
Keuntungan menggunakan Java, dan Tipe-tipe  
data standar

**Fakultas**  
Teknik

**Program Studi**  
Informatika

**Tatap Muka**  
**06**

**Kode MK**  
06410006

**Disusun Oleh**  
Ardiles Sinaga, S.T., M.T.  
Kurnia Jaya Eliazar, S.T., M.T.

## **Abstract**

Modul Pertemuan 04 Berisi Mengenai Konsep Pemrograman Java, Java Platform, Keuntungan menggunakan Java, dan Tipe-tipe data standar.

## **Kompetensi**

Mahasiswa memiliki kemampuan untuk menjelaskan tentang Konsep Pemrograman Java, Java Platform, Keuntungan menggunakan Java, dan Tipe-tipe data standar.

## **Pertemuan 06 – Konsep Pemrograman Java, Java Platform, Keuntungan menggunakan Java, dan Tipe-tipe data standar.**

### **Konsep Pemrograman Java**

#### **Sejarah Java**

Proyek Java dimulai pada tahun 1991, ketika sejumlah insinyur perusahaan Sun yang dimotori oleh James Gosling mempunyai keinginan untuk mendesain sebuah bahasa komputer kecil yang dapat dipergunakan untuk peralatan konsumen seperti kotak tombol saluran TV.

Proyek ini kemudian diberi nama sandi Green.

Keharusan untuk membuat bahasa yang kecil, dan kode yang ketat mendorong mereka untuk menghidupkan kembali model yang pernah dicoba oleh bahasa UCSD Pascal, yaitu mendesain sebuah bahasa yang portable yang menghasilkan kode intermediate. Kode intermediate ini kemudian dapat digunakan pada banyak komputer yang interpretnya telah disesuaikan.

Karena orang-orang Sun memiliki latar belakang sebagai pemakai unix sehingga mereka lebih menggunakan C++ sebagai basis bahasa pemrograman mereka, maka mereka secara khusus mengembangkan bahasa yang berorientasi objek bukan berorientasi prosedur. Seperti yang dikatakan Gosling "Secara keseluruhan, bahasa hanyalah sarana, bukan merupakan tujuan akhir". Dan Gosling memutuskan menyebut bahasanya dengan nama "Oak" (diambil dari nama pohon yang tumbuh tepat diluar jendela kantornya di Sun), tetapi kemudian nama Oak diubah menjadi java, karena nama Oak merupakan nama bahasa komputer yang sudah ada sebelumnya.

Pada tahun 1994 sebagian besar orang menggunakan mosaic, browser web yang tidak diperdagangkan yang berasal dari pusat Supercomputing Universitas Illinois pada tahun 1993. (Mosaic sebagian ditulis oleh Marc Andreessen dengan bayaran \$6.85 per jam, sebagai

mahasiswa yang melakukan studi praktek. Di kemudian hari ia meraih ketenaran sebagai salah seorang pendiri dan pemimpin teknologi di netscape)

Browser yang sesungguhnya dibangun oleh Patrick Naughton dan Jonathan Payne dan berkembang ke dalam browser HotJava yang kita miliki saat ini. Browser HotJava ditulis dalam Java untuk menunjukkan kemampuan Java. Tetapi para pembuat juga memiliki ide tentang suatu kekuatan yang saat ini disebut dengan applet, sehingga mereka membuat browser yang mampu menerjemahkan kode byte tingkat menengah. “Teknologi yang Terbukti” ini diperlihatkan pada SunWorld '95 pada tanggal 23 mei 1995, yang mengilhami keranjingan terhadap Java terus berlanjut.

### **Kriteria “Kertas Putih” Java**

Penulis Java telah menulis pengaruh “Kertas Putih” yang menjelaskan tujuan rancangan dan keunggulannya. Kertas mereka disusun lewat 11 kriteria berikut :

#### **Sederhana (Simple)**

Syntax untuk Java seperti syntax pada C++ tetapi syntax Java tidak memerlukan header file, pointer arithmetic (atau bahkan pointer syntax), struktur union, operator overloading, class virtual base, dan yang lainnya. Jika anda mengenal C++ dengan baik, maka anda dapat berpindah ke syntax Java dengan mudah tetapi jika tidak, anda pasti tidak berpendapat bahwa Java sederhana.

#### **Berorientasi Objek (Object Oriented)**

Rancangan berorientasi objek merupakan suatu teknik yang memusatkan rancangan pada data (objek) dan interface. Fasilitas pemrograman berorientasi objek pada Java pada dasarnya adalah sama dengan C++. Feature pemrograman berorientasi objek pada Java benarbenar sebanding dengan C++, perbedaan utama antara Java dengan C++ terletak pada penurunanberganda (multiple inheritance), untuk ini Java memiliki cara penyelesaian yang lebih baik.



### **Terdistribusi (Distributed)**

Java memiliki library rutin yang luas untuk dirangkai pada protokol TCP/IP seperti HTTP dan FTP dengan mudah. Aplikasi Java dapat membuka dan mengakses objek untuk segala macam NET lewat URL sama mudahnya seperti yang biasa dilakukan seorang programmer ketika mengakses file sistem secara lokal.

### **Kuat (Robust)**

Java dimaksudkan untuk membuat suatu program yang benar-benar dapat dipercaya dalam berbagai hal. Java banyak menekankan pada pengecekan awal untuk kemungkinan terjadinya masalah, pengecekan pada saat runtime dan mengurangi kemungkinan timbulnya kesalahan (error). Perbedaan utama antara Java dan C++ adalah Java memiliki sebuah model pointer yang mengurangi kemungkinan penimpaan (overwriting) pada memory dan kerusakan data (data corrupt).

### **Aman (Secure)**

Java dimaksudkan untuk digunakan pada jaringan terdistribusi. Sebelum sampai pada bagian tersebut, penekanan terutama ditujukan pada masalah keamanan. Java memungkinkan penyusunan program yang bebas virus, sistem yang bebas dari kerusakan.

### **Netral Arsitektur (Architecture Neutral)**

Kompiler membangkitkan sebuah format file dengan objek arsitektur syaraf, program yang di kompilasi dapat dijalankan pada banyak prosesor, disini diberikan sistem run time dari Java. Kompiler Java melakukannya dengan membangkitkan instruksi-instruksi kode byte yang tidak dapat dilakukan oleh arsitektur komputer tertentu. Dan yang lebih baik Java dirancang untuk mempermudah penterjemahan pada banyak komputer dengan mudah dan diterjemahkan pada komputer asal pada saat run-time.

### **Portabel (Portable)**

Tidak seperti pada C dan C++, di Java terdapat ketergantungan pada saat implementasi (implement dependent). ukuran dari tipe data primitif ditentukan, sebagaimana kelakuan

aritmatik padanya. Librari atau pustaka merupakan bagian dari sistem yang mendefinisikan interface yang portabel.

### **Interpreter**

Interpreter Java dapat meng-eksekusi kode byte Java secara langsung pada komputer-komputer yang memiliki interpreter. Dan karena proses linking dalam Java merupakan proses yang kenaikannya tahap demi tahap dan berbobot ringan, maka proses pengembangan dapat menjadi lebih cepat dan masih dalam penelitian.

### **Kinerja Yang Tinggi (High Performance)**

Meskipun kinerja kode byte yang di interpretasi biasanya lebih dari memadai, tetapi masih terdapat situasi yang memerlukan kinerja yang lebih tinggi. Kode byte dapat diterjemahkan (pada saat run-time) ke dalam kode mesin untuk CPU tertentu dimana aplikasi sedang berjalan.

### **Multithreaded**

Multithreading adalah kemampuan sebuah program untuk melakukan lebih dari satu pekerjaan sekaligus. Keuntungan dari multithreading adalah sifat respons yang interaktif dan real-time.

### **Dinamis**

Dalam sejumlah hal, Java merupakan bahasa pemrograman yang lebih dinamis dibandingkan dengan C atau C++. Java dirancang untuk beradaptasi dengan lingkungan yang terus berkembang. Librari dapat dengan mudah menambah metode dan variabel contoh yang baru tanpa banyak mempengaruhi klien. Informasi tipe run-time dalam Java adalah langsung (straightforward).

- **EDISI JAVA**

Java adalah bahasa yang dapat di jalankan dimanapun dan disembarang platform apapun, diberagam lingkungan : internet, intranet, consumer Electronic products dan computer

Applications. The Java 2 platform tersedia dalam 3 edisi untuk keperluan berbeda. Untuk beragam aplikasi yang dibuat dengan bahasa java, java dipaketkan dalam edisi2 berikut :

1. Java 2 Standard Edition ( J2SE )
2. Java 2 Enterprise Edition ( J2EE )
3. Java 2 Micro Edition ( J2ME )

Masing – masing edisi berisi java 2 Software Development Kit ( J2SDK ) untuk mengembangkan aplikasi dan java 2 Runtime Environment ( J2RE ) untuk menjalankan aplikasi.

- Kelebihan Java dibandingkan dengan C++
  - Pembuat program java telah merancang java untuk menghilangkan pengalokasian dan dealokasi memori secara manual, karena java memiliki Garbage Collection.
  - Diperkenalkannya deklarasi array yang sebenarnya dan menghilangkan aritmatika pointer. Hal ini yang sering menyebabkan memori overwrite.
  - Di hilangkannya multiple inheritance, mereka menggantinya dengan interface.

## **Pengenalan Java**

### **Apa itu Java ?**

Java adalah salah satu bahasa pemrograman berorientasi objek (OOObject Oriented Programming). Paradigma OOP menyelesaikan masalah dengan merepresentasikan masalah ke model objek.

### **Keutamaan Java dibanding bahasa pemrograman lain:**

- Cross platform, dengan adanya Java Virtual Machine(JVM)
- Pengembangannya didukung oleh programmer secara luas
- Automatic Garbage Collection, membebaskan programmer dari tugas manajemen memori

## **Pemrograman Berorientasi Obyek (OOP)**

### **Pemisalan Objek dalam OOP**



Objek-objek dalam dunia nyata, mempunyai 2 karakteristik khusus : Status dan Perilaku. Contohnya, sepeda punya status(jumlah gir, jumlah pedal, dua buah ban) dan perilaku(mengerem, mempercepat, ubah gir). Bahasa yang berorientasi pada objek pun mempunyai karakteristik yang sama dengan objek-objek di dunia nyata. Yaitu status yang dalam bahasa pemrograman biasanya disimpan sebagai Variabel dan perilaku yang diimplementasikan sebagai Method.

## Karakteristik OOP

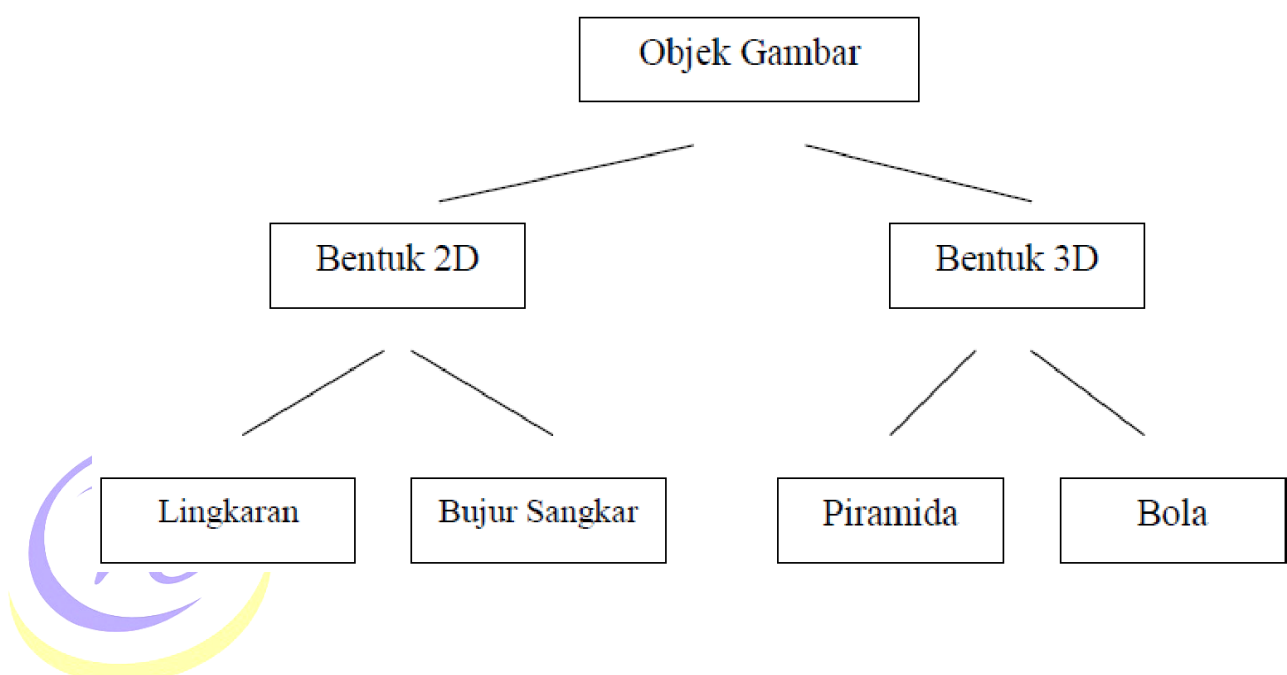
### 1. Enkapsulasi(Pembungkusan)

Enkapsulasi adalah pelindung program dan data yang sedang diolah. Enkapsulasi mendefinisikan perilaku dan melindungi program dan data agar tidak diakses secara sembarangan oleh program lain.

Dalam Java, dasar enkapsulasi adalah class. Anda membuat suatu class yang menyatakan bahwa variable atau method sebuah class tidak dapat diakses oleh class lain dengan menjadikan class tersebut private, atau menjadikan class tersebut protected – yaitu hanya bisa diakses oleh turunannya, atau menjadikan class tersebut public – yaitu bisa diakses oleh sembarang class.

### 2. Inheritansi

Objek-objek yang berada di sekitar kita adalah objek-objek yang saling terhubung secara hirarkis. Misalnya :



Lingkaran dan Bujur Sangkar adalah turunan dari bentuk 2D dan Bentuk 2D adalah turunan dari Objek Gambar Objek Gambar

Bentuk 2D Bentuk 3D Lingkaran Bujur Sangkar Piramida Bola

Lingkaran dan Bujur Sangkar mewarisi(inherit) sifat-sifat dari bentuk 2D, juga mewarisi sifat-sifat dari objek gambar Lingkaran dan Bujur Sangkar dapat dikatakan subclass dari bentuk 2D. Bentuk 3D adalah superclass dari Bola dan Piramida, dan seterusnya.

### 3. Polimorfisme

Walaupun Lingkaran dan Bujur Sangkar sama-sama turunan dari Bentuk 2D, tetapi cara menubah ukuran masing-masing berbeda, untuk lingkaran anda harus merubah besar jari-jarinya, sedang untuk bujur sangkar anda harus mengubah panjang sisinya.

Dalam Java implementasi, method suatu parent-class dapat diubah oleh sub-class, hal ini dikenal dengan overriding method. Deklarasi method sama tetapi implementasi atau definisinya berbeda(Method atau perilaku yang sama tapi implementasinya/caranya yang berbeda-beda inilah yang disebut dengan Polimorfisme).

Menulis Program Java

Aturan penulisan program di Java

- Java adalah turunan dari C, sehingga Java memiliki sifat C yaitu Case sensitive, yaitu membedakan antara huruf besar dan kecil
- Dalam sebuah file program di Java, hanya diijinkan memiliki 1 buah class yang bersifat public
- Dalam sebuah file program Java, hanya ada satu method main(method yang pertama kali dibaca oleh interpreter Java)
- Nama sebuah file program Java harus sama dengan nama class yang memiliki method main() di dalam tubuhnya. Perhatikan bahwa tulisan nama file dengan nama class (huruf besar maupun kecilnya) haruslah persis sama. .

Berikut adalah contoh membuat program dengan menggunakan Java. Pada program akan ditampilkan tulisan “Hello World !“

Hello.java



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

Langkah selanjutnya :

1. Simpan dengan nama: Hello.java
2. compile Hello.java : javac Hello.java
3. hasilnya akan menghasilkan : Hello.class
4. jalankan Hello.class: java Hello.class atau java Hello
5. akan keluar hasil :

Hello world!

Programming Tip:

Dalam penulisan bahasa program, disarankan :

1. Huruf depan dari sebuah class atau method menggunakan huruf besar
2. Menulis Komentar pada sebuah class atau method untuk memudahkan debug (pencarian kesalahan). Serta mempermudah orang lain membaca program kita. Ingat !!! Dalam dunia nyata, programmer bekerja secara team, jadi usahakan partner team mengerti apa yang kita buat dengan memberikan komentar(tentang pembuatan komentar akan dibahas dibawah)
3. Membuat indentasi(jarak antara induk perintah dan anak perintah). Identasi sebisa mungkin dibuat standard, semisal pada contoh diatas, jarak antara tulisan “class Hello” sebagai induk perintah dengan tulisan “public.....” sebagai anak perintah adalah 5 spasi. Sekali lagi, ini untuk mempermudah dalam pengertian program.

## Deklarasi Variabel

Bahasa pemrograman pada umumnya mengenal adanya variabel yang digunakan untuk menyimpan nilai atau data. Java dikenal dengan bahasa pemrograman yang bersifat strongly

typed yang artinya diharuskan mendeklarasikan tipe data dari semua variabel dan apabila lupa atau salah mengikuti aturan pendeklarasian variabel maka akan mendapat error pada saat proses kompilasi.

## Tipe data

Java memiliki dua jenis tipe data yang dikategorikan menjadi dua yaitu tipe data primitif dan tipe data referensi.

### 1. Tipe Data Primitif

Macam tipe data primitif diantaranya :

- Integer (Bilangan bulat)

Integer merupakan tipe data numerik yang digunakan untuk mendefinisikan bilangan bulat. Tipe data numeric yang termasuk integer diantaranya:

Tipe	Deskripsi
Byte	-128 s/d +127 menempati 8 bits di memori
Short	-32768 s/d +32767 menempati 16 bits di memori
Int	-2147483648 s/d +2147483647 menempati 32 bits di memori
Long	-9223372036854775808 s/d +9223372036854775807 menempati 64 bits di memori

Tipe Data Primitif Integer

- Floating Point (Bilangan pecahan)

Floating point digunakan untuk menangani bilangan decimal atau perhitungan yang lebih detail dibanding integer.

Tipe	Deskripsi
Byte	-128 s/d +127 menempati 8 bits di memori
Short	-32768 s/d +32767 menempati 16 bits di memori
Int	-2147483648 s/d +2147483647 menempati 32 bits di memori
Long	-9223372036854775808 s/d +9223372036854775807 menempati 64 bits di memori

Tipe Data Primitif Floating Point

- Floating Point (Bilangan pecahan)

Floating point digunakan untuk menangani bilangan decimal atau perhitungan yang lebih detail dibanding integer.

Tipe	Deskripsi
Float	$-3.4 \times 10^8$ s/d $+3.4 \times 10^8$
Double	$-1.7 \times 10^{308}$ s/d $+1.7 \times 10^{308}$

Tipe Data Primitif Floating Point

- Char

Char adalah karakter tunggal yang pendefinisian di awal dan akhir menggunakan tanda petik tunggal ( ' ). Tipe char mengikuti aturan Unicode, sehingga bisa dapat menggunakan kode untuk kemudian diikuti bilangan dari 0 sampai 65535, tetapi yang biasa digunakan adalah bilangan heksadesimal dari 0000 sampai FFFF.

- Boolean

Tipe data Boolean terdiri dari dua nilai saja, yaitu true dan false. Boolean sangat penting untuk mengevaluasi suatu kondisi.

## 2. Tipe Data Referensi

Kelebihan pemrograman dengan orientasi objek adalah dapat mendefinisikan tipe data baru yang merupakan objek dari class tertentu. Tipe data ini digunakan untuk mereferensikan objek atau class tertentu, seperti String.

- Variabel

Variabel merupakan container yang digunakan untuk menyimpan suatu nilai pada sebuah program dengan tipe tertentu. Untuk mendefinisikan variabel, suatu identifier dapat digunakan untuk menamai variabel tersebut.

- Identifier

Identifier adalah kumpulan karakter yang dapat digunakan untuk menamai variabel, method, class, interface, dan package. Dalam pemrograman Java identifier bisa disebut sah apabila diawali dengan :

- Huruf / abjad
- Karakter Mata Uang
- Underscore(\_)

Identifier dapat terdiri dari :

- Huruf / abjad
- Angka
- Underscore (\_)

Identifier tidak boleh mengandung @, spasi atau diawali dengan angka serta tidak boleh menggunakan keyword yang telah digunakan di pemrograman java. Selain karakter, Unicode juga dapat digunakan sebagai identifier.

## Mendeklarasikan Variabel

Sintaks dasar :

**[tipe data] [nama variabel]**

Menuliskan tipe data dari variabel, contoh:

```
int bilangan;
char karakter;
float bildesimal;
boolean status;
```

{Setelah dideklarasikan sesuai dengan tipe data, selanjutnya memberi nilai variabel tersebut dengan tanda =}

```
Bilangan = 20;
Karakter = 'k';
Bildesimal = 22.2f;
Status = true;
```

{melakukan deklarasi tipe data dan memberi nilai variabel dalam satu baris}

```
int bilangan = 20;  
char karakter = 'k';  
boolean status = true;
```

{membuat variabel menjadi konstanta sehingga tidak bisa diubah lagi nilainya dengan menambahkan keyword sebelum tipe data}

```
Final int a = 10;  
Final float pajak = 15.5;
```

{membuat agar konstanta dapat diakses oleh kelas lain tanpa harus membuat objek terlebih dulu dilakukan penambahan modifier public dan keyword static}

```
Public static final a = 10;
```

## Daftar Pustaka

- [1] Deitel P.J., Deitel H.M., “Java: How to Program”, Prentice Hall, 2004
- [2] Holmes B. J., Joyce D.T., “Object-Oriented Programming With Java, Second Edition”, JONES AND BARTLETT PUBLISHERS, 2001
- [3] Keogh J., “JAVA DEMYSTIFIED”, McGraw-Hill/Osborne, 2004
- [4] Wu C.T., “AN INTRODUCTION TO OBJECT-ORIENTED PROGRAMMING WITH JAVA™, FIFTH EDITION”, McGraw-Hill, 2010

- [5] Wu C.T., “A COMPREHENSIVE INTRODUCTION TO OBJECT-ORIENTED PROGRAMMING WITH JAVA”, McGraw-Hill, 2008
- [6] Poo D., Kiong D., Ashok S., “Object-Oriented Programming and Java Second edition”, Springer, 2008
- [7] Sintes T., “Sams Teach Yourself Object Oriented Programming in 21 Days”, Sam Publishing, 2002

