

# Dasar-Dasar Pemrograman 2

## Lab 01

### Pengenalan Java



FAKULTAS  
ILMU  
KOMPUTER

Di DDP 1 kita telah belajar konsep-konsep dasar pemrograman dengan menggunakan bahasa Python. Di DDP 2 ini kita akan memperdalam pemahaman pemrograman kita dengan konsep-konsep yang menarik seperti Object Oriented Programming dengan menggunakan bahasa pemrograman Java.

Pasti timbul pertanyaan di benak masing-masing, kenapa memakai Java? Kenapa tidak pakai Python saja?

Singkatnya, Java mempunyai kelebihan yaitu kejelasan struktur dalam pemrogramannya, sehingga membuat mudah untuk memahami konsep Object Oriented. Selain itu, Java juga termasuk bahasa pemrograman yang paling populer di dunia. Bahasanya dapat digunakan untuk membuat banyak hal di berbagai platform. Patut untuk dipelajari, bukan?

### Perbedaan Java dan Python

#### 1. Interpreted vs Compiled

Python merupakan salah satu *Interpreted Language*, di mana saat program dijalankan, komputer akan mengeksekusi tiap instruksi satu per satu. Oleh karena itu, kita bisa langsung menjalankan program Python. Java merupakan salah satu *Compiled Language*, di mana setiap program harus diubah menjadi bahasa lain yang dimengerti komputer, lalu setelah itu baru bisa dijalankan.

#### 2. Struktur Kode

Perbedaan struktur kode Python dan Java cukup mencolok. Namun, pada dasarnya semua bahasa pemrograman memiliki tujuan yang sama, yaitu sebagai bahasa penghubung antara manusia yang membuat

program dan komputer yang akan menjalankan program. Sehingga, jika sudah memahami sebuah bahasa pemrograman, maka akan lebih mudah untuk memahami bahasa yang lain.

Berikut perbedaan *syntax* di Python dan Java:

- **Declaring & Assigning Variables**

Variabel di **Python** bersifat *dynamically-typed*, sehingga kita bisa memasukkan nilai dengan tipe data apapun ke sebuah variabel dan kita bisa mengubah nilai variabel tersebut dengan tipe data yang berbeda.

Variabel di **Java** bersifat *statically-typed*. Di Java, kita harus mendeklarasikan (memberi informasi) tipe data dari sebuah variabel secara eksplisit. Variabel tersebut juga hanya bisa diisi dengan nilai dengan tipe data yang sudah kita deklarasikan tadi.

### Python

```
var1 = 3
var2 = "Dek Depe"
var3 = True
var1 = False
```

### Java

```
int var1 = 3;
String var2 = "Dek Depe";
boolean var3 = true;
var1 = false; // error, var1 bertipe int
```

Tipe data pada Java antara lain: **int** (bil. bulat), **double** (bil. pecahan), **boolean** (true/false), **String** (kalimat), **char** (satu huruf), dan masih banyak lagi yang akan kita pelajari nanti.

**Catatan:** Setiap instruksi pada java *harus* diakhiri titik koma / semicolon ( ; ). Berbeda dengan Python yang setiap instruksinya dipisah oleh baris. (Ya, kita bisa menulis lebih dari satu instruksi pada baris yang sama -- tetapi tidak dianjurkan karena bisa membuat kode menjadi sulit dibaca oleh manusia.)

**Catatan (2):** String di Java *wajib* menggunakan tanda kutip ( " ... " ). Tidak seperti Python yang bisa menggunakan tanda kutip maupun apostrof ( ' ... ' )

- **And-or operations**

Pada Java, gunakan && untuk and, dan gunakan || untuk or.

### Python

```
a = True
b = True
if a and b:
    print("True and True")

a = True
b = False:
if a or b:
    print("True or False")
```

### Java

```
boolean a = true;
boolean b = true;
if (a && b) {
    System.out.println("true and true");
}

a = true;
b = false;
if (a || b) {
    System.out.println("true or false");
}
```

- **If-Else Conditions**

Struktur kode If-Else pada Python dan Java kurang lebih sama. Perbedaannya adalah **Java** menggunakan **else if** sedangkan **Python** menggunakan **elif**.

## Python

```
nilai = 78
huruf = ''

if nilai > 85:
    huruf = 'A'
elif nilai > 70:
    huruf = 'B'
elif nilai > 55:
    huruf = 'C'
else:
    huruf = 'D'

print(huruf)
```

## Java

```
int nilai = 78;
char huruf;

if (nilai > 85) {
    huruf = 'A';
} else if (nilai > 70) {
    huruf = 'B';
} else if (nilai > 55) {
    huruf = 'C';
} else {
    huruf = 'D';
}

System.out.println(huruf);
```

- **Equals**

Sama seperti Python, jika Anda ingin mengetahui apakah suatu nilai bertipe data primitif dengan nilai bertipe data primitif lainnya (misal: int dengan int) sama atau tidak, kita dapat menggunakan '=='. Namun, jika kita ingin membandingkan suatu string dengan string yang lain, kita harus menggunakan method .equals(...) di Java. Hal ini karena String bukan tipe data primitif di Java.

Menggunakan `==` pada variabel bertipe non-primitif akan membandingkan *reference*-nya (sama seperti `is` pada Python).

## Python

```
kata = 'kata'
kata_lain = 'kata lain'
print(kata == kata_lain) # False
kata_lain = 'kata'
print(kata == kata_lain) # True
```

## Java

```
String kata = "kata";
String kataLain = "kata lain";
System.out.println(kata.equals(kataLain)); // false

kataLain = "kata";
System.out.println(kata.equals(kataLain)); // true
```

- **Input**

Struktur input pada Java memiliki bentuk yang lebih kompleks dibanding pada Python. Pada Java, kita bisa menggunakan objek `Scanner` untuk membaca input. Untuk dapat menggunakan `Scanner`, yang harus dilakukan pertama kali adalah meng-*import* **`java.util.Scanner`**. Kemudian, kita membuat objek `Scanner` untuk membaca input dan menyimpannya di suatu variabel. Objek `Scanner` memiliki berbagai *method* untuk membaca input berdasarkan tipe data. Beberapa *method* yang sering digunakan:

<i>Method</i>	Deskripsi
<code>nextInt()</code>	membaca input bertipe int
<code>nextLong()</code>	membaca input bertipe long
<code>nextFloat()</code>	membaca input bertipe float
<code>nextDouble()</code>	membaca input bertipe double
<code>nextBoolean()</code>	membaca input bertipe boolean
<code>nextLine()</code>	membaca input satu baris dalam bentuk string

Berikut perbandingan antara input di Python dengan di Java.

### Python

```
var = input()
```

### Java

```
import java.util.Scanner;

Scanner input = new Scanner(System.in);
String var = input.nextLine();
```

**Hati-hati!** Apabila kamu ingin membaca input yang bertipe data selain String (misal int), dan kamu ingin input tersebut diakhiri dengan baris baru (*user* menekan Enter), sebaiknya tetap panggil *method* `nextLine()` setelah memanggil `nextInt()`. Ini untuk mencegah masalah yang muncul apabila kamu ingin membaca String dengan `nextLine()` setelah meminta input non-String dan diakhiri dengan baris baru.

Perhatikan contoh berikut.

```
int angka = input.nextInt();
String pesan = input.nextLine();
System.out.println("Angka: " + angka);
System.out.println("Pesan: " + pesan);
```

Apa *output* yang dihasilkan?

Sekarang bandingkan dengan contoh berikut.

```
int angka = input.nextInt();
input.nextLine();
String pesan = input.nextLine();
System.out.println("Angka: " + angka);
System.out.println("Pesan: " + pesan);
```

Apa *output* yang dihasilkan? **Mengapa demikian?** Silakan coba cari tahu konsep token dan bagaimana cara Scanner bekerja.

Apabila sudah selesai menggunakan Scanner, jangan lupa untuk **menutupnya** dengan memanggil *method* `.close()`.

- **While Loop**

Struktur While-loop pada java dan Python sama, yaitu dalam format `while <kondisi>`. Namun, tidak seperti Python, kondisi di Java harus dibungkus dengan tanda kurung.

#### Python

```
# print angka 0 sampai 9
number = 0
while number < 10:
    print(number)
    number += 1
```

#### Java

```
// print angka 0 sampai 9
int number = 0;
while (number < 10) {
    System.out.println(number);
    number++;
}
```

- **For Loop**

Struktur For-loop pada Java dan Python sedikit berbeda, tidak seperti Python yang menggunakan `range`, di Java kita harus menspesifikasikan inisialisasi loop seperti apa, akan terus mengiterasi dalam kondisi bagaimana, dan apa yang dilakukan setiap kali selesai iterasi.

#### Python

```
for i in range(0, 5, 1):
    print(i)
```

#### Java

```
for (int i = 0; i < 5; i++) {
    System.out.println(i);
}
```

## Membuat Program Java

Berikut adalah contoh program “Hello World” yang kita buat dengan menggunakan Java.

```
// disimpan dengan nama file HelloWorld.java
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

Hal pertama yang kita lakukan adalah membuat file yang bernama **HelloWorld.java**. Ekstensi **.java** ini berguna untuk menunjukkan bahwa file ini adalah suatu kode program yang dapat di-*compile* oleh Java Compiler.

Kemudian, kita membuat sebuah *class*. Perhatikan bahwa **nama class HARUS sama dengan nama file di mana class tersebut dibuat**. Lalu, kita membuat sebuah *method* bernama “main”. Method ini berfungsi sebagai *entry-point* (titik masuk) dari program Java sehingga saat program dijalankan, sistem akan langsung mencari fungsi “main” ini dan menjalankan kode yang ada di dalamnya.

**System.out.println()** digunakan untuk mencetak ke layar, seperti fungsi **print()** di Python.

Untuk menjalankan program, pertama *save file* terlebih dahulu, lalu buka Command Prompt (Terminal) di folder tempat menyimpan file java tersebut dengan cara mengetik “cmd” di *address bar* Windows Explorer tempat menyimpan file. Kita juga bisa membuka Command Prompt secara langsung dan menavigasikannya ke folder penyimpanan *file* Java dengan menggunakan perintah “**cd [path direktori penyimpanan]**”.

Seperti yang telah disebutkan sebelumnya, Java adalah *compiled language*. Jadi, sebelum menjalankan program diharuskan meng-*compile*-nya terlebih dahulu, yaitu dengan perintah: **javac [nama\_file].java**

Langkah terakhir adalah menjalankan program yang sudah di-*compile* dengan perintah:

**java [nama\_file]** (Perhatikan bahwa tidak ada ekstensi **.java** saat menjalankan program).

**Selamat, kamu sudah berhasil menjalankan program “Hello World” di Java!**



**Catatan:** Karena kita akan menggunakan Eclipse IDE dalam beberapa lab selanjutnya, kamu disarankan untuk membaca basic tutorial Eclipse IDE untuk membuat program “Hello World”. Kamu bisa membacanya di sini <https://www.codejava.net/ides/eclipse/how-to-create-build-and-run-a-java-hello-world-program-with-eclipse> atau melalui sumber-sumber yang lain.

*(See the next page to do the task...)*

# Soal Lab 01

---

## Arin, a Diligent Fasilkom Student



Source: <https://www.google.com>

Arin adalah seorang mahasiswa baru Fasilkom yang tertarik untuk mempelajari Java. Di waktu liburannya, Arin mengunjungi suatu museum dan melihat daftar **harga tiket masuk yang dikelompokkan berdasarkan usia**. Setelah melihat hal tersebut, Arin tertarik untuk membuat program sederhana yang menentukan berapa **harga tiket masuk** untuk orang dengan **usia tertentu**.

Sepulangnya dari museum, adik Arin meminta Arin untuk membantu menyelesaikan tugas matematika sekolahnya yang terdiri dari soal-soal perpangkatan. Karena Arin sangat senang *ngoding*, dia akhirnya tertarik untuk membuat program sederhana yang **menghitung hasil dari pangkat bilangan tertentu sesuai yang diinput** untuk membantu tugas adiknya.

Dia sudah pernah mempelajari Python sebelumnya, sehingga dia sudah berhasil membuat program dari kedua masalah di atas dengan menggunakan Python. Karena Arin sangat senang belajar, dia tertarik untuk mengubah program yang sudah ia buat menggunakan Python ke dalam Java. Karena Arin baru saja mulai mempelajari Java, Arin membutuhkan bantuan dari kakak tingkatnya untuk menyelesaikan permasalahan tersebut. Sebagai kakak tingkat yang baik hati, ayo bantu Arin **mengubah barisan kode Python ke dalam barisan kode Java!**

### **Soal Lab:**

Buatlah dua buah file Java yang bernama Tiket.java dan PangkatN.java, yang masing-masing memiliki satu class dengan mengikuti nama file yang telah ditentukan dan melakukan hal serupa dengan apa yang dikerjakan oleh potongan kode Python pada masing-masing program di bawah.

Dalam menuliskan kode Java, gunakan tipe data yang sesuai dan perhatikan coding style yang benar (indentation, penamaan variabel, dll) dan jangan lupa untuk menambahkan komentar pada bagian-bagian penting di kode masing-masing.

**Catatan:** Kamu bebas untuk mengerjakan dengan menggunakan Eclipse IDE atau Text Editor yang biasa kamu pakai.

### **Program tiket.py**

```
harga_dasar = int(input('Masukkan harga dasar tiket: Rp'))
nama = input('Masukkan nama: ')
usia = int(input('Masukkan usia: '))

if usia < 15:
    harga = harga_dasar
elif usia >= 15 and usia < 23:
    harga = 2*harga_dasar
elif usia >= 23 and usia < 30:
    harga = 3*harga_dasar
else:
    harga = 4*harga_dasar

print(
    nama + ', yang berusia ' + str(usia)
    + ' tahun, harus membayar tiket sebesar Rp' + harga + '.'
)
```

## Program pangkat\_n.py

```
angka = int(input('Masukkan angka untuk dipangkatkan n: '))
n = int(input('Masukkan n: '))
angka_awal = angka

i = 1
while i < n:
    angka = angka*angka_awal
    i += 1 # gunakan increment di Java
print(str(angka_awal) + ' pangkat ' + str(n) + ' adalah ' + str(angka))

if angka % 2 == 0:
    print('Hasilnya genap')
else:
    print('Hasilnya ganjil')
```

### Catatan:

- Untuk tutorial kali ini, **tidak perlu memvalidasi input** dari *user* (asumsikan tipe data input yang diberikan sudah sesuai).
- Usahakan untuk **mengikuti konvensi penamaan variabel** dalam bahasa pemrograman yang digunakan. Pada Python, konvensi penamaan variabel menggunakan aturan **snake\_case**. Pada Java, konvensi penamaan variabel menggunakan aturan **camelCase**. Untuk penamaan *class*, Python dan Java sama-sama menggunakan aturan UpperCamelCase atau yang dikenal dengan istilah **PascalCase**.

### Submisi File

Satukan semua file Java (yang berekstensi **.java** saja) ke dalam format **zip** dengan format penamaan:

DDP2\_[Kelas]\_[KodeAsdos]\_[NPM]\_[NamaLengkap].zip, kemudian *upload* ke slot submisi yang ada di SCeLE.

Contoh: DDP2\_G\_SMA\_1234567890\_KefasSatria.zip

Selamat mengerjakan,  
*God bless you!*