

# Java For Fun

## Download paket JDK versi 8

Paket *java development kit (JDK)* versi *stable* 8 update 261 (saat ini) atau lebih dapat didonlot di *website Oracle*:

<https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>

Tergantung sistem operasi dan moda bit (32 atau 64)

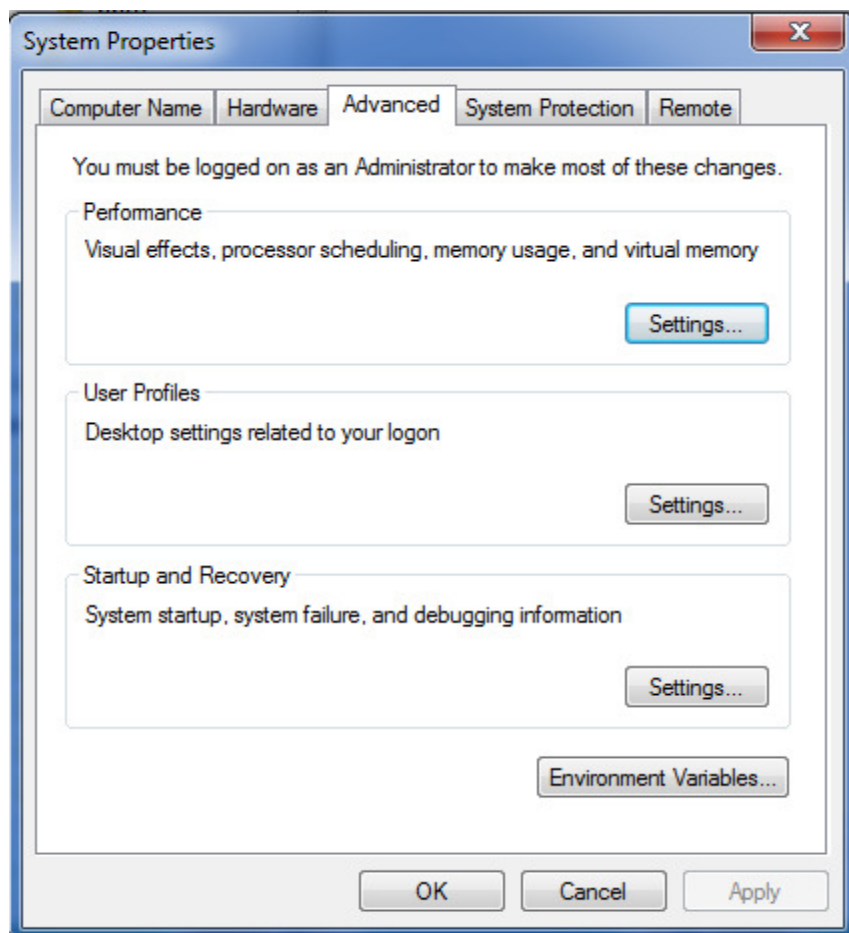
## Panduan instalasi

[https://docs.oracle.com/javase/8/docs/technotes/guides/install/windows\\_jdk\\_install.html#CHDEBCCJ](https://docs.oracle.com/javase/8/docs/technotes/guides/install/windows_jdk_install.html#CHDEBCCJ)

Ekstrak atau eksekusi aplikasi installer tersebut di atas dan ikuti sesuai panduan

## Konfigurasi (Windows 7)

Buka **Control Panel\System and Security\System** kemudian pilih **Advanced system settings** seperti

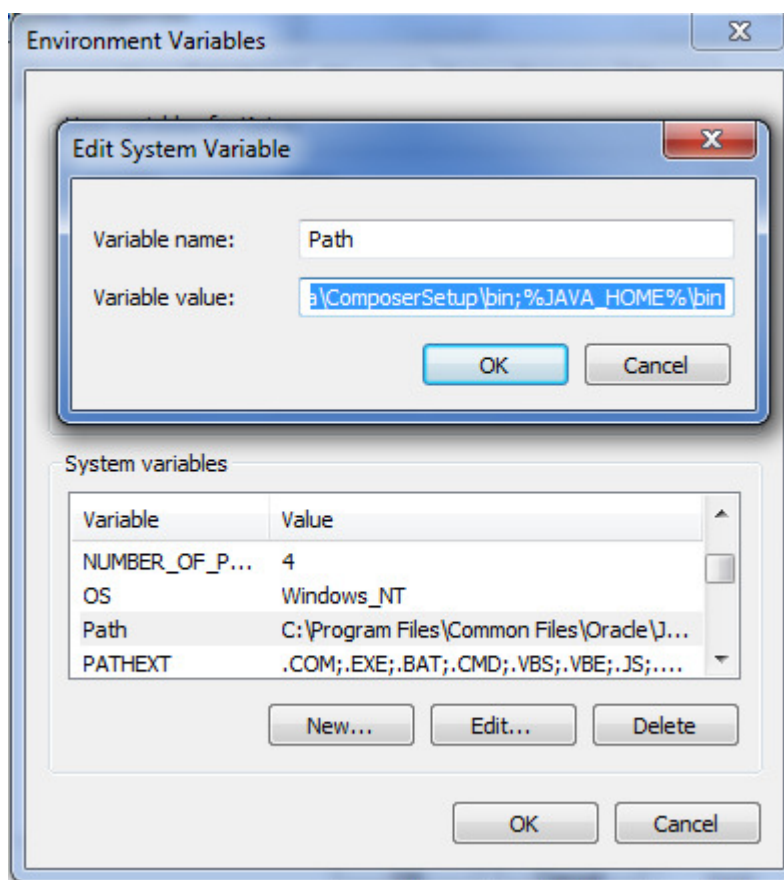
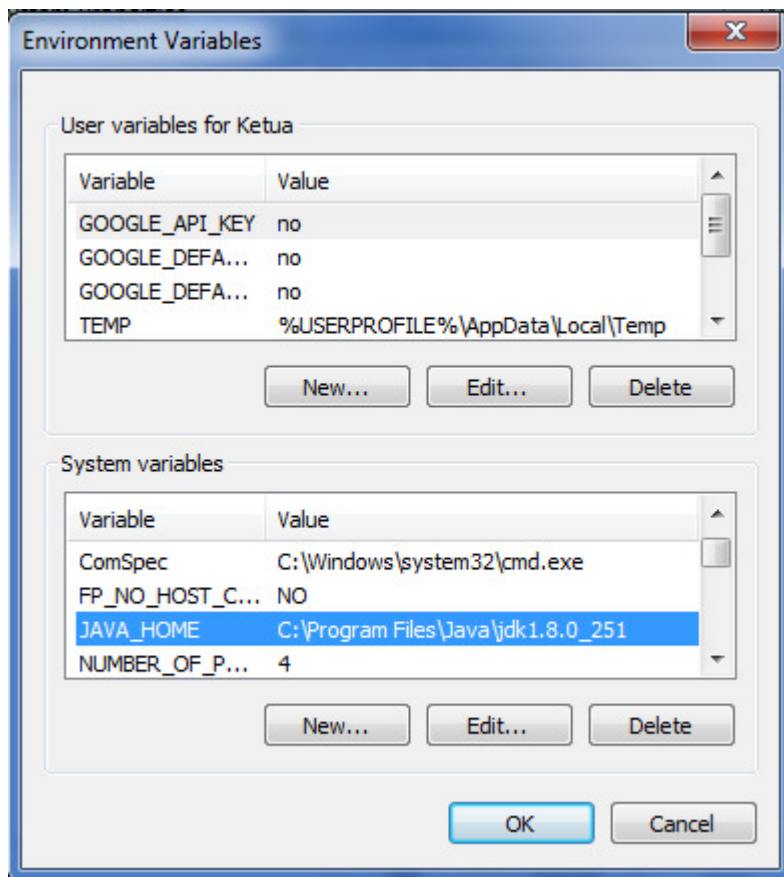


tampilan: **System Properties**

Pilih klik tombol **Environment Variables**

Buat variabel baru (klik **New**) beri nama **JAVA\_HOME** kemudian isikan *path* direktori program java diinstal, seperti contoh: **C:\Program Files\Java\jdk1.8.0\_251**

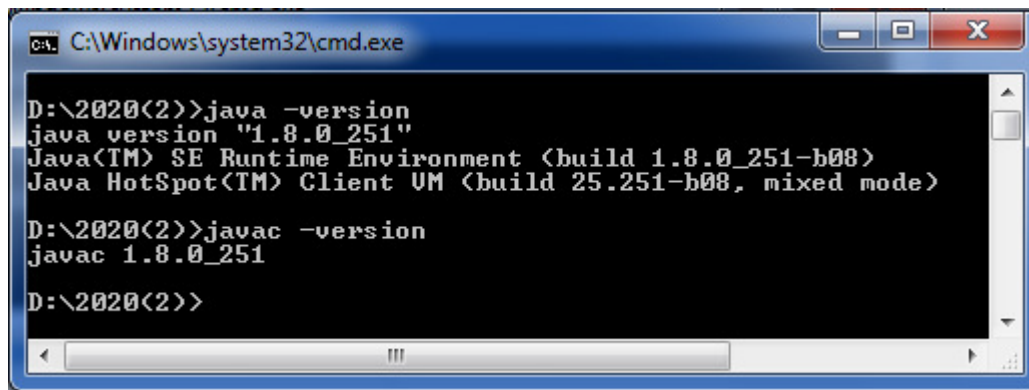
Edit **path** (di *append*) dengan menambahkan: **%JAVA\_HOME%\bin**



Klik tombol OK sampai semua tampilan *window setting* pada *control panel* tertutup

## Test

Buka console (cmd) dan ketikkan perintah: (1) **java -version** [ENTER] dan (2) **javac -version** [ENTER], seperti contoh: tampilan versi **1.8.0\_251**



```
C:\Windows\system32\cmd.exe

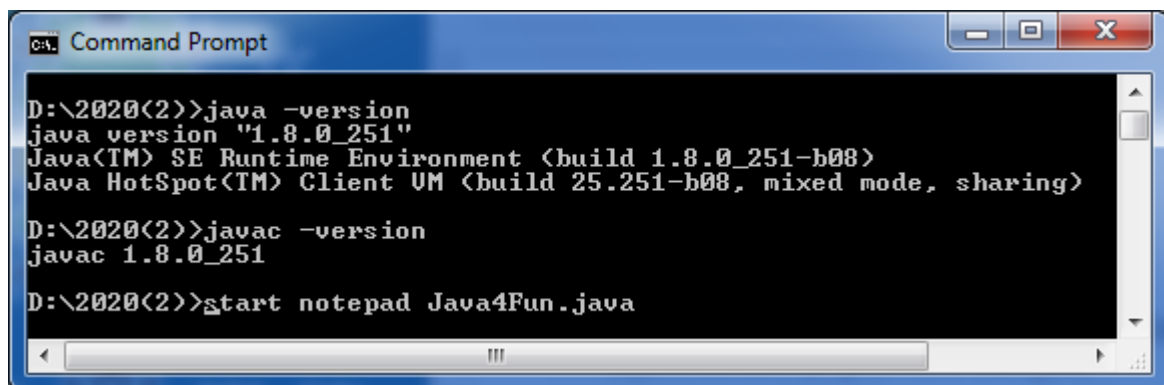
D:\2020(2)>java -version
java version "1.8.0_251"
Java(TM) SE Runtime Environment (build 1.8.0_251-b08)
Java HotSpot(TM) Client VM (build 25.251-b08, mixed mode)

D:\2020(2)>javac -version
javac 1.8.0_251

D:\2020(2)>
```

## Koding

Koding dapat dilakukan dengan program IDE seperti **Netbeans** atau **Eclipse**. Jika belum tersedia dapat menggunakan aplikasi editor teks seperti **Notepad**, seperti contoh:



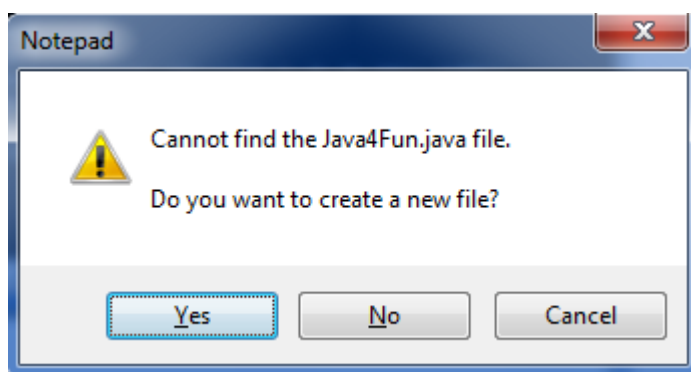
```
Command Prompt

D:\2020(2)>java -version
java version "1.8.0_251"
Java(TM) SE Runtime Environment (build 1.8.0_251-b08)
Java HotSpot(TM) Client VM (build 25.251-b08, mixed mode, sharing)

D:\2020(2)>javac -version
javac 1.8.0_251

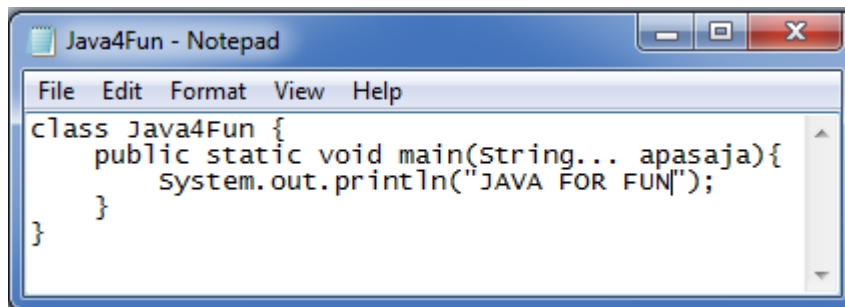
D:\2020(2)>start notepad Java4Fun.java
```

Tampilan menjalankan **Notepad** (start) dengan sekaligus membuat file baru (**Java4Fun.java**)



Tampilan konfirmasi membuat file baru (Yes)

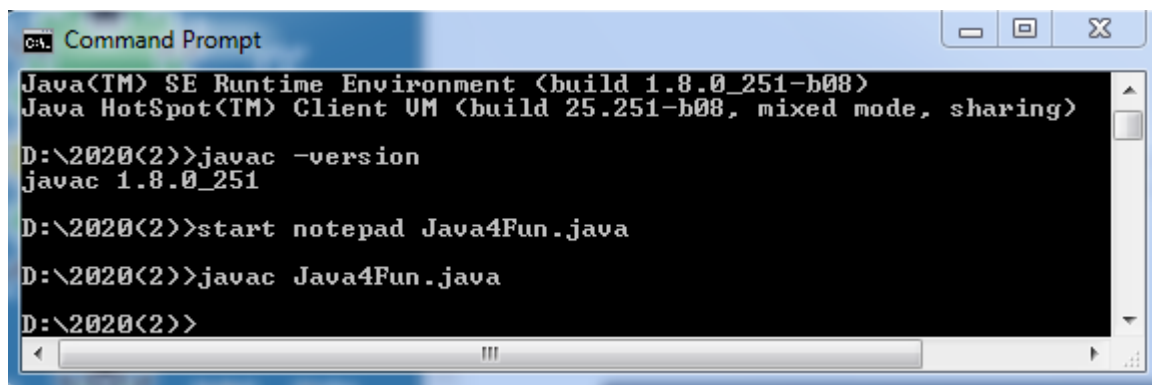
Program **Java** yang dapat dieksekusi harus memiliki prosedur utama ( **public static void main** ) dengan parameter *array* dari *class String* (**String[]**) atau dengan notasi *ellipsis* (**String...**) dengan nama obyek apasaja (bebas). Sebagai contoh *print* teks ("JAVA FOR FUN") ke layar *console* (*command prompt*) melalui prosedur *static* : **System.out.println(String)** :



```
File Edit Format View Help
class Java4Fun {
    public static void main(String... apasaja){
        System.out.println("JAVA FOR FUN!");
    }
}
```

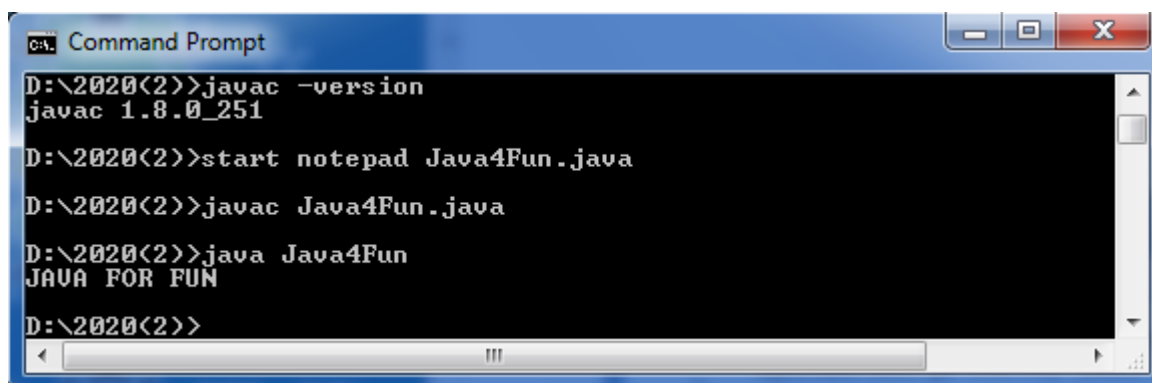
Tampilan program **Java4Fun** (diktik pada **Notepad**), jangan lupa **SAVE** dengan **Ctrl-S**

**Kompilasi** (javac namaFileDenganExtension.Java)



```
Ca. Command Prompt
Java(TM) SE Runtime Environment (build 1.8.0_251-b08)
Java HotSpot(TM) Client VM (build 25.251-b08, mixed mode, sharing)
D:\2020(2)>javac -version
javac 1.8.0_251
D:\2020(2)>start notepad Java4Fun.java
D:\2020(2)>javac Java4Fun.java
D:\2020(2)>
```

**Eksekusi** (java namaFileTanpaExtension)



```
Ca. Command Prompt
D:\2020(2)>javac -version
javac 1.8.0_251
D:\2020(2)>start notepad Java4Fun.java
D:\2020(2)>javac Java4Fun.java
D:\2020(2)>java Java4Fun
JAVA FOR FUN
D:\2020(2)>
```

## **Dokumen API**

Dokumen API (*application programming interface*) atau di singkat **API-Docs** dapat dilihat pada

<https://docs.oracle.com/javase/8/docs/api/index.html>

**API-Docs** berfungsi sebagai panduan dan referensi sintaksis *class* dan *interface* pada paket java yang telah ada. Setiap class terdiri dari atribut (*fields*) dan metoda termasuk konstruktor (*constructor*) sebagai metoda khusus yang namanya sama dengan nama *class*). Pada **API-Docs** ini dapat ditelusuri hirarki (relasi **Whole-Part**) dari *class* (termasuk *abstract class*) dan *interface* mulai dari *class* **Object** (sebagai “Nabi Adam”) yang menurunkan (paradigma *inheritance*) menjadi *subclass* (*class* turunan) secara berjenjang. Sebagai contoh tampilan **API-Docs** untuk *class* **Button** yang merupakan subclass dari *class* **Component** pada paket **java.awt** dan *implements interface* **Accessible** dapat dilihat di bawah:

Java™ Platform Standard Ed. 8

OVERVIEW PACKAGE **CLASS** USE TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES

SUMMARY: NESTED | FIELD | CONSTR | METHOD  
DETAIL: FIELD | CONSTR | METHOD

java.awt

## Class Button

java.lang.Object  
    java.awt.Component  
        java.awt.Button

**All Implemented Interfaces:**  
ImageObserver, MenuContainer, Serializable, Accessible

---

```
public class Button
    extends Component
    implements Accessible
```

This class creates a labeled button. The application can cause some action to happen when the button is pushed. This image depicts three views of a "Quit" button as it appears under the Solaris operating system:

Contoh tampilan **API-Docs** paket **java.awt** class **Button** (*inheritance (extends)* dari class **Component** dan *implements interface **Accesible***)