

# Package

---



*Object Oriented Programming*



# Packages

---

- Package adalah koleksi dari beberapa class dan interface yang berhubungan, dan menyediakan proteksi akses dan pengelolaan namespace
- 1 package adalah 1 subfolder di file system
- Package sangat berguna untuk mengorganisir file dalam suatu project atau library
- Keyword: *package name;*



# Budi.java

---

```
package kelasku;  
  
public class Budi{  
    public void info(){  
        System.out.println("Kelas Budi");  
    }  
}
```



# Joko.java

---

```
package kelasku;  
  
public class Joko{  
    public void info(){  
        System.out.println("Kelas Joko");  
    }  
}
```



# TesPaket.java

```
import kelasku.Budi;  
import kelasku.Joko;  
  
public class TesPaket{  
    public static void main(String[] args){  
        Budi objectBudi = new Budi();  
        objectBudi.info();  
        Joko objectJoko = new Joko();  
        objectJoko.info();  
    }  
}
```



ml ▾ code ▾ package ▾



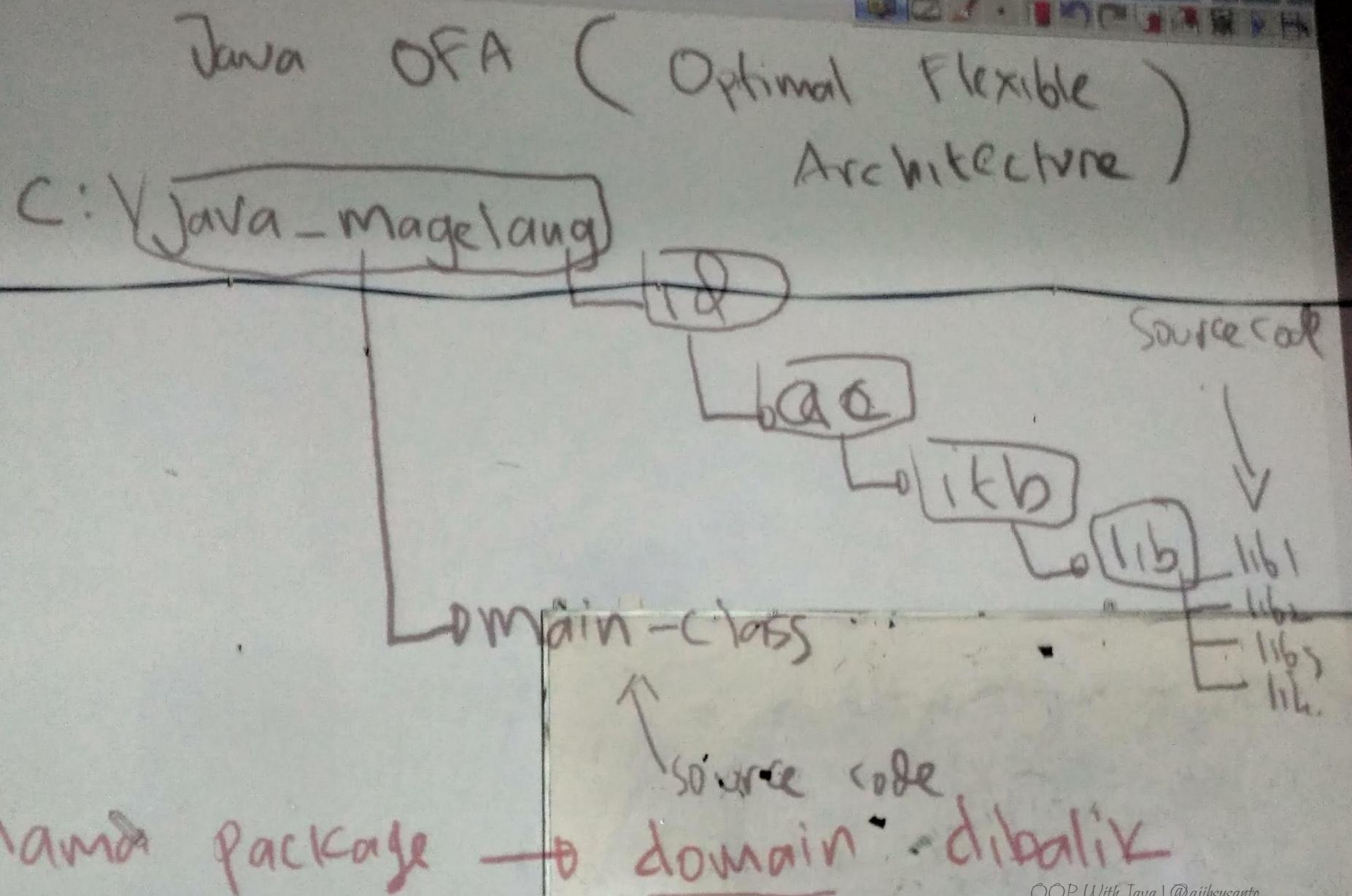
Search

Name ▾

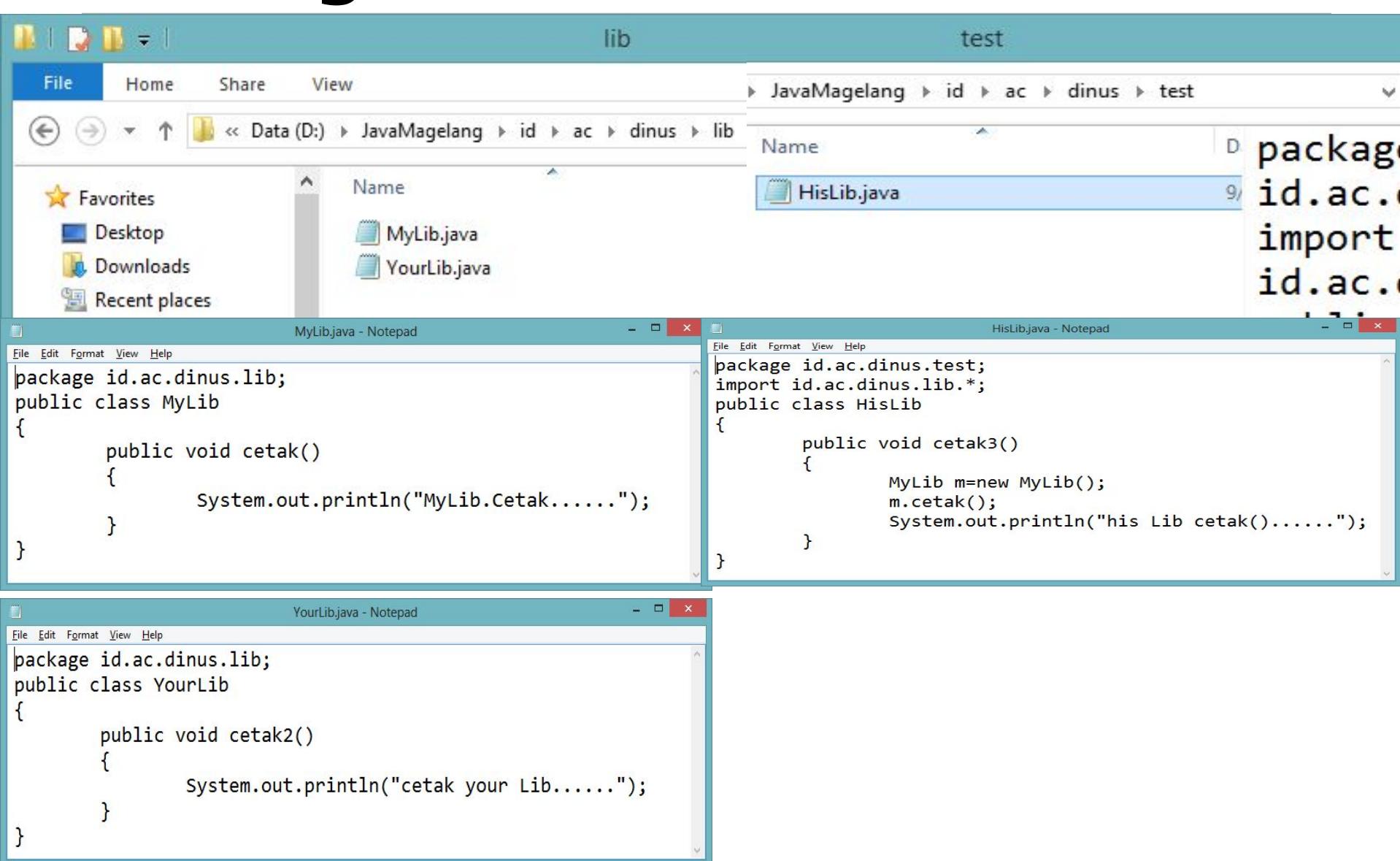
- kelasku
- TesPaket.class
- TesPaket.java



# Packages



# Packages



# Packages

The image shows a Windows desktop environment. In the foreground, there is a Notepad window titled "AksesLib.java - Notepad" containing Java code. Below it, an open Command Prompt window titled "C:\Windows\system32\cmd.exe" displays the output of running the Java program.

```
import id.ac.dinus.lib.*;
import id.ac.dinus.test.*;
public class AksesLib
{
    public static void main(String[] a)
    {
        MyLib m=new MyLib();
        m.cetak();
        YourLib y=new YourLib();
        y.cetak2();
        HisLib h=new HisLib();
        h.cetak3();
    }
}
```

```
D:\JavaMageLang>java AksesLib
MyLib.Cetak.....
cetak your Lib.....
MyLib.Cetak.....
his Lib cetak().....
D:\JavaMageLang>
```



# Packages

The screenshot shows a Windows desktop environment. In the foreground, there is a Notepad window titled "AksesLib2.java - Notepad" containing Java code. Below it is a Command Prompt window titled "C:\Windows\system32\cmd.exe".

**Notepad Content (AksesLib2.java):**

```
import id.ac.dinus.lib.MyLib; //lsg ke class
import id.ac.dinus.lib.YourLib;
import id.ac.dinus.test.*;
public class AksesLib2
{
    public static void main(String[] a)
    {
        MyLib m=new MyLib();
        m.cetak();
        YourLib y=new YourLib();
        y.cetak2();
        HisLib h=new HisLib();
        h.cetak3();
    }
}
```

**Command Prompt Output:**

```
D:\JavaMagelang>java AksesLib2
MyLib.Cetak......
cetak your Lib......
MyLib.Cetak......
his Lib cetak()......

D:\JavaMagelang>
```



# Tugas

---

- Paketkan berbagai class yang kita buat (**Mobil** dan **Bicycle**) dalam package **Transportasi**
- Panggil dari class lain (**TransportasiDemo.java**) dengan **import**



# Rehat Sejenak

---

- Shower Gel



# Java Class - Library



*Object Oriented Programming*



# Class Libraries

---

- sebuah kumpulan dari program yang disertakan dalam Java.
- Pemrograman menjadi lebih mudah.
- Kelengkapan library semakin beragam jika ditambah dengan karya komunitas Java



# Class Libraries

Package	Nama Package	Keterangan
Language	java.lang	Class-class utama yang merupakan inti dari bahasa Java
Utilities	java.util	Class-class yang mendukung utilitas struktur Java
I/O	java.io	Class yang mendukung berbagai macam tipe input dan output
Text	java.text	Class yang mendukung lokalisasi penanganan teks, tanggal, bilangan, dan message
Math	java.math	Class untuk melakukan perhitungan aritmatik arbitrary-procesion, baik integer atau floating point.
AWT	java.awt	Class untuk perancangan user-interface dan event-handling
Swing	javax.swing	Class untuk membuat berbagai komponen dalam Java yang bertingkah laku sama dengan berbagai platform.
Javax	javax	Perluasan dari bahasa Java
Applet	java.applet	Class untuk membuat applet
Beans	java.beans	Class untuk membuat Java Beans
Reflection	java.lang.reflect	Class untuk memperoleh informasi run-time
SQL	java.sql	Class untuk mendukung akses dan pengolahan data dalam database.
RMI	java.rmi	Class untuk mendukung distributed programming.
Network	java.net	Class untuk mendukung dalam membangun aplikasi jaringan.
Security	java.security	Class untuk mendukung keamanan kriptografi



# Library Math

---

- `Math.abs(x)`, menghitung nilai mutlak (absolut) dari  $x$ . Nilai mutlak bilangan negatif adalah bilangan positif, dan bilangan positif tetap bilangan positif.
- Fungsi trigonometri `Math.sin(x)`, `Math.cos(x)`, and `Math.tan(x)`  
*(Untuk semua fungsi trigonometri, sudut memiliki satuan radian, bukan derajat)*



# Library Math

---

- Fungsi trigonometri inverse, yang mencari sudut dari suatu nilai trigonometric, kebalikan dari fungsi trigonometri, seperti arcus sin, arcus cos, dan arcus tangen. `Math.asin(x)`, `Math.acos(x)`, and `Math.atan(x)`.
- `Math.exp(x)`, menghitung pangkat dari bilangan natural e, atau  $e^x$ . Dan logaritma natural  $\log_e x$  atau  $\ln x$  bisa dihitung dengan menggunakan fungsi `Math.log(x)`.



# Library Math

---

- `Math.pow(x,y)` menghitung  $x^y$  atau  $x$  pangkat  $y$
- `Math.floor(x)` menghitung pembulatan ke bawah dari suatu bilangan riil, misalnya  $3.84$  akan dibulatkan ke bawah menjadi  $3.0$
- `Math.random()` memilih bilangan acak di antara  $0.0$  dan  $1.0$ . Komputer memiliki algoritma perhitungan tertentu yang hasilnya bilangan acak (meskipun bukan bilangan yang betul-betul acak, tetapi cukup untuk kebanyakan fungsi)



# Library Math

PI	double	double
IEEEremainder(double f1, double f2)	double	double
abs(double a)	double	double
abs(float a)	float	float
abs(int a)	int	int
abs(long a)	long	long
acos(double a)	double	double
asin(double a)	double	double
atan(double a)	double	double
atan2(double y, double x)	double	double
cbrt(double a)	double	double
ceil(double a)	double	double
copySign(double magnitude, double sign)	double	double
copySign(float magnitude, float sign)	float	float
cos(double a)	double	double
cosh(double x)	double	double
exp(double a)	double	double
expm1(double x)	double	double
floor(double a)	double	double
getExponent(double d)	int	int
getExponent(float f)	int	int
hypot(double x, double y)	double	double
log(double a)	double	double
log10(double a)	double	double
log1p(double x)	double	double
max(double a, double b)	double	double
max(float a, float b)	float	float
max(int a, int b)	int	int
max(long a, long b)	long	long
min(double a, double b)	double	double
min(float a, float b)	float	float
min(int a, int b)	int	int
min(long a, long b)	long	long
nextAfter(double start, double direction)	double	double
nextAfter(float start, double direction)	float	float
scalb(float f, int scaleFactor)	float	float
signum(double d)	double	double
signum(float f)	float	float
sin(double a)	double	double
sinh(double x)	double	double
sqrt(double a)	double	double
tan(double a)	double	double
tanh(double x)	double	double
toDegrees(double angrad)	double	double
toRadians(double angdeg)	double	double
ulp(double d)	double	double
ulp(float f)	float	float



# Library Date

- sebuah class yang dapat digunakan untuk mendapatkan tanggal.
- Terletak di package `java.util.Date`.
- Sebelum menggunakan :
  - `import java.util.Date;`
- Penerapan cukup sederhana, cukup dengan membuat obyek dari class `Date`.

```
import java.util.Date;
public class Tanggal {
    public static void main(String[] args) {
        Date tanggal=new Date();
        System.out.println(tanggal);
    }
}
```

Mon Apr 14 21:59:53 ICT 2014



# Format Tanggal

```
import java.util.Date;  
import java.text.SimpleDateFormat;  
public class Tanggal {  
    public static void main(String[] args) {  
        Date tanggal=new Date();  
        SimpleDateFormat sdf=new SimpleDateFormat("dd-MM-yyyy");  
        SimpleDateFormat sdf1=new SimpleDateFormat("dd MMMM yyyy");  
        System.out.println(tanggal);  
        System.out.println(sdf.format(tanggal));  
        System.out.println(sdf1.format(tanggal));  
    }  
}
```

Mon Apr 14 22:04:31 ICT 2014

14-04-2014

14 April 2014



# Format Tanggal

```
import java.util.Date;
import java.text.SimpleDateFormat;
import java.util.Calendar;
public class Tanggal {
    public static void main(String[] args) {
        Date tanggal=new Date();
        SimpleDateFormat sdf=new SimpleDateFormat("dd-MM-yyyy");
        SimpleDateFormat sdf1=new SimpleDateFormat("dd MMMM yyyy");
        System.out.println(tanggal);
        System.out.println(sdf.format(tanggal));
        System.out.println(sdf1.format(tanggal));
        Calendar cal=Calendar.getInstance();
        cal.add(Calendar.DATE, 3);
        Date tigaHariLagi=cal.getTime();
        System.out.println("3 Hari lagi : "+sdf.format(tigaHariLagi));
    }
}
```

Mon Apr 14 22:04:31 ICT 2014

14-04-2014

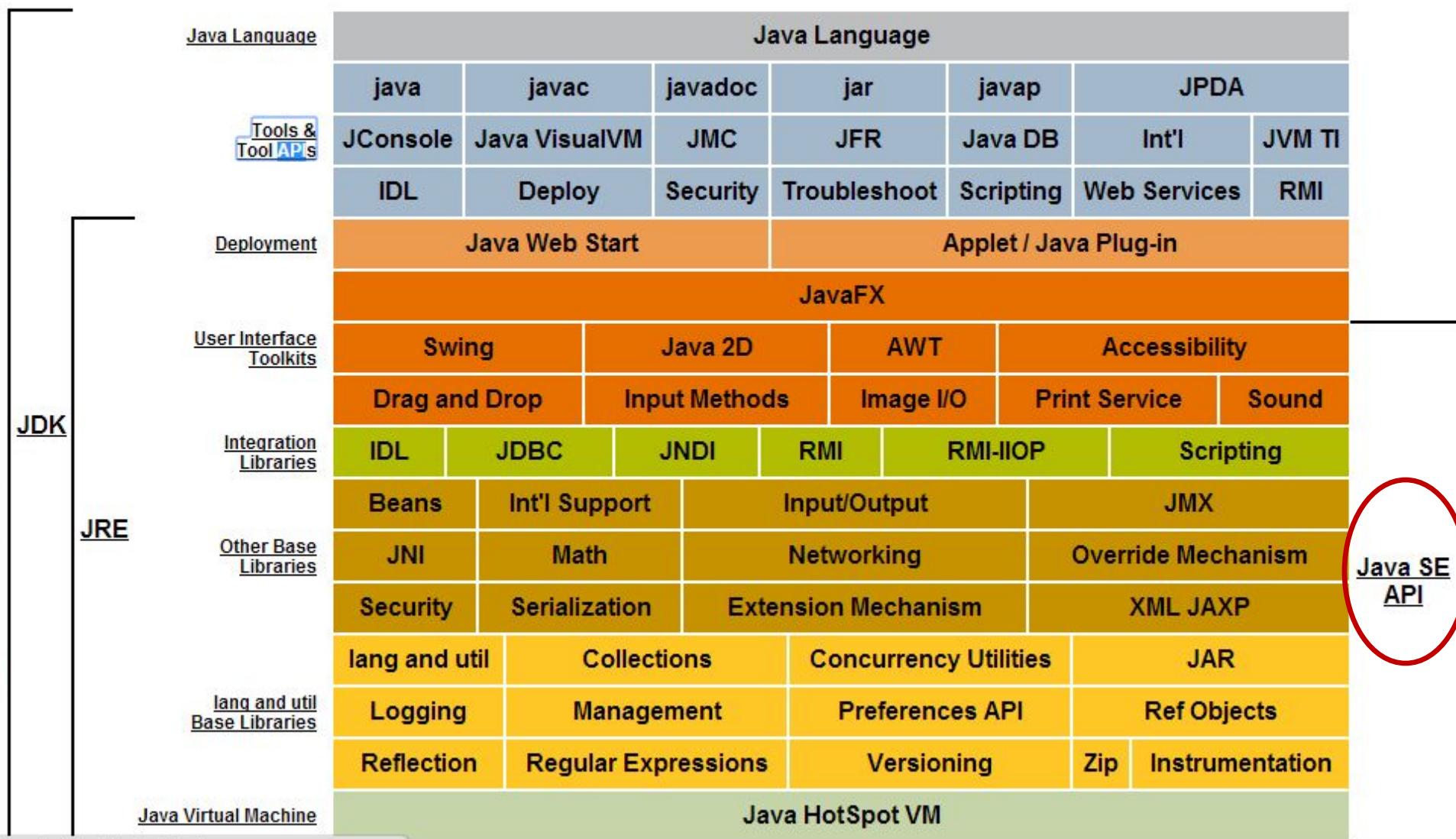
14 April 2014

3 Hari lagi : 17-04-2014



# Java SE API - <http://docs.oracle.com/javase/7/docs/>

## Description of Java Conceptual Diagram



# Java SE API

<http://docs.oracle.com/javase/7/docs/api/>

Java™ Platform Standard Ed. 7

Overview Package Class Use Tree Deprecated Index Help

Prev Next Frames No Frames

## Java™ Platform, Standard Edition 7 API Specification

This document is the API specification for the Java™ Platform, Standard Edition.

See: Description

### Packages

Package	Description
java.applet	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating user interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.datatransfer	Provides interfaces and classes for transferring data between and within applications.
java.awt.dnd	Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities logically associated with presentation elements in the GUI.
java.awt.event	Provides interfaces and classes for dealing with different types of events fired by AWT components.
java.awt.font	Provides classes and interface relating to fonts.
java.awt.geom	Provides the Java 2D classes for defining and performing operations on objects related to two-dimensional geometry.
java.awt.im	Provides classes and interfaces for the input method framework.
java.awt.im.spi	Provides interfaces that enable the development of input methods that can be used with any Java runtime environment.
java.awt.image	Provides classes for creating and modifying images.
java.awt.image.renderable	Provides classes and interfaces for producing rendering-independent images.
java.awt.print	Provides classes and interfaces for a general printing API.



# Jar (Java Archive)

---



*Object Oriented Programming*



# JAR

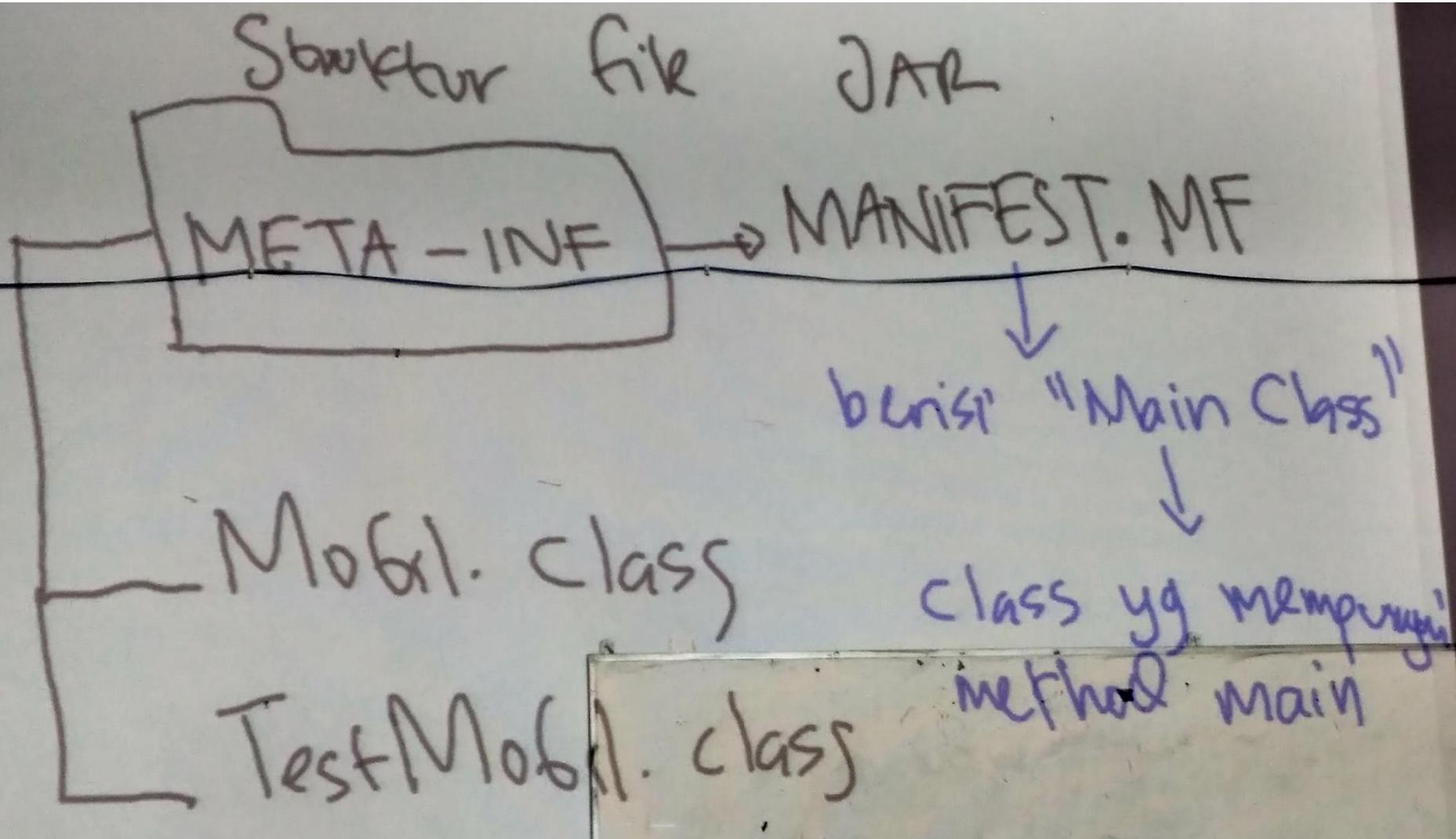
JAR → Java Archive

Teknik yang digunakan  
untuk mendistribusikan  
Software Java, ke end user

→ sebuah file .Zip, namanya saja  
extensi diubah menjadi .jar

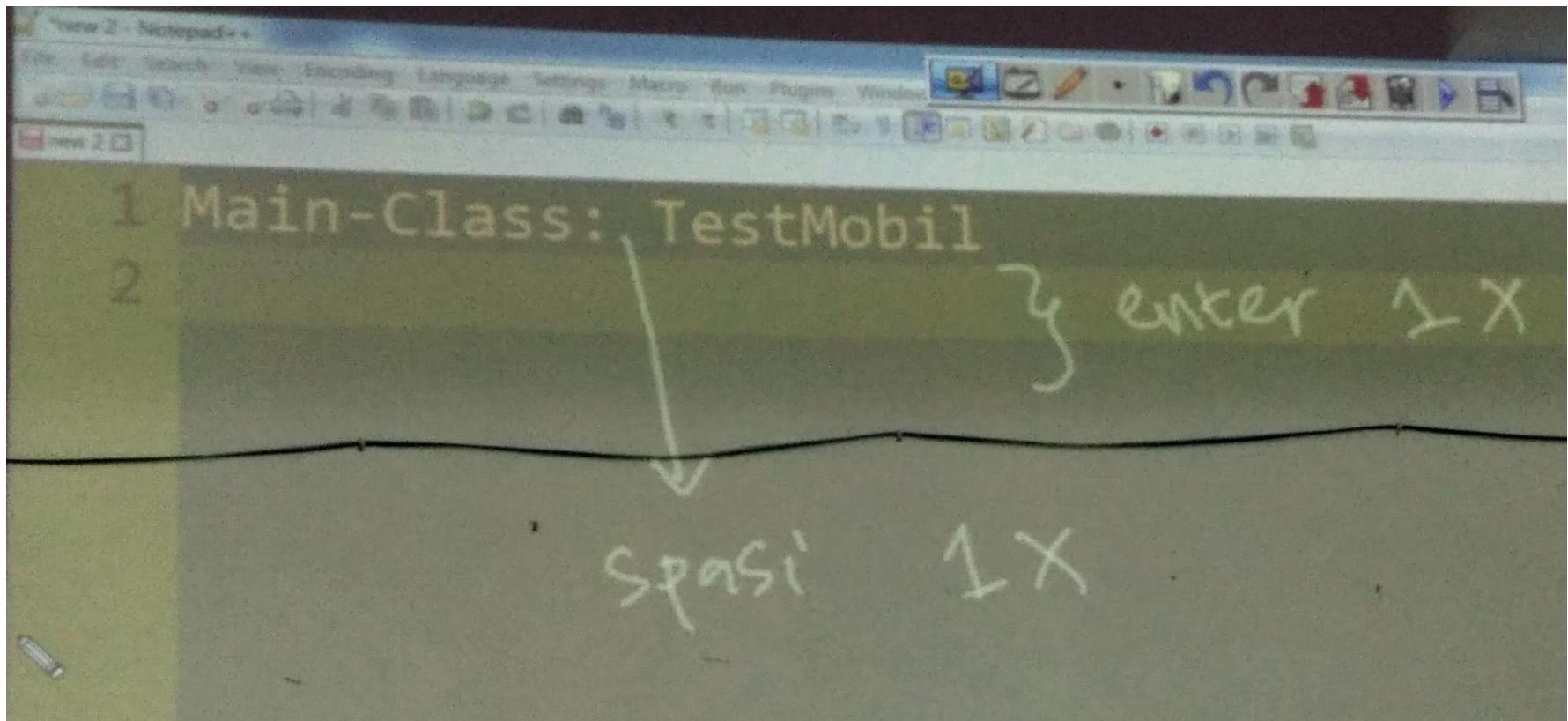


# JAR

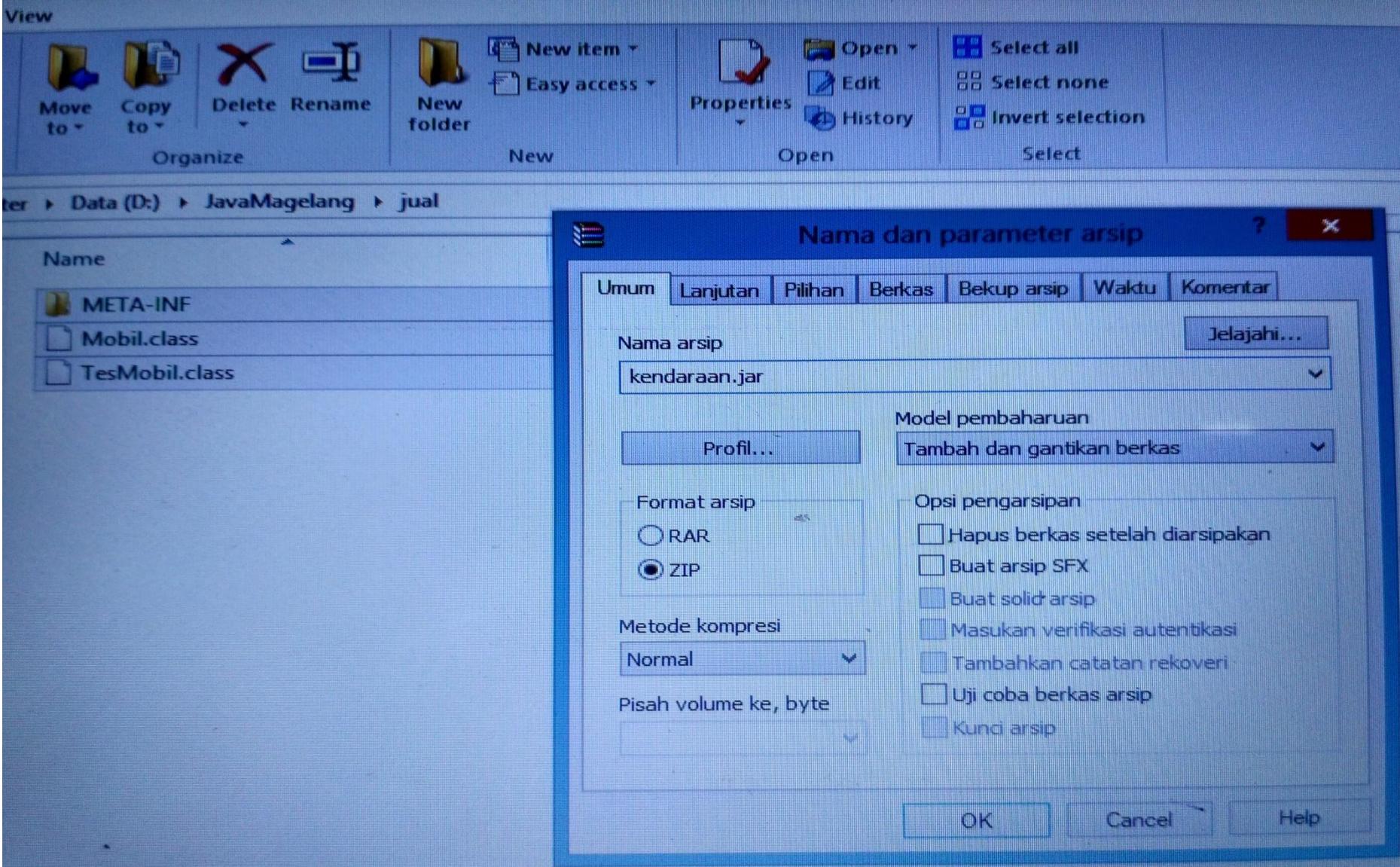


# JAR

---



# JAR



# JAR

jual

iew Move to Copy to Delete Rename New folder New item Easy access Properties Open Select all Select none Invert selection Organize New Open Select

er > Data (D:) > JavaMagelang > jual >

Name	Date modified	Type	Size
META-INF	9/26/2017 9:49 AM	File folder	
kendaraan.jar	9/26/2017 9:55 AM	Exe	
Mobil.class	9/26/2017 9:29 AM	CL	
TesMobil.class	9/26/2017 9:29 AM	CL	

Command Prompt

```
location: variable avanza of type Mobil
1 error
D:\JavaMagelang>javac TesMobil.java

D:\JavaMagelang>java TesMobil
Maju.....
Mundur.....
Belok.....
Roda    : 4
Mesin   : 1

D:\JavaMagelang>cd jual

D:\JavaMagelang\jual>java -jar kendaraan.jar
Error: Could not find or load main class TestMobil

D:\JavaMagelang\jual>java -jar kendaraan.jar
Maju.....
Mundur.....
Belok.....
Roda    : 4
Mesin   : 1

D:\JavaMagelang\jual>_
```





Ajib Susanto

[ajibsusanto@gmail.com](mailto:ajibsusanto@gmail.com)

[ajib.susanto@dsn.dinus.ac.id](mailto:ajib.susanto@dsn.dinus.ac.id)

<http://ajibsusanto.net>

@ajibsusanto / 085876247118



# Referensi

---

- *Object First With Java, Fifth edition, David J. Barnes & Michael Kölling, Prentice Hall / Pearson Education, 2012.*
- *The Java<sup>TM</sup> Tutorial,*  
<http://docs.oracle.com/javase/tutorial/java/nutsandbolts/>, Oracle, 1995-2014.
- *Java SE Tutorial,*  
<http://www.oracle.com/technetwork/java/javase/downloads/javase7-tutorial-2012-02-28-1536013.html>, Oracle, 2014.
- *Java Platform, SE Documentation,*  
<https://docs.oracle.com/en/java/javase/index.html>
- *SCJP Sun Certified Programmer for Java<sup>TM</sup> 6 Study Guide Exam (310-065), Kathy Sierra & Bert Bates, Mc Graw Hill, 2008.*
- *Object Oriented Programming with Java, Romi Satria Wahono, 2008.*

