

# COM 102

# OBJECT ORIENTED PROGRAMMING

## LAB 0 – Basic Setup

# General Information

- ▶ **Lab Instructor:** Res.Asst. Arzum Karataş
- ▶ **Office:** D-227
- ▶ **Office hours:** Monday : 10.00-11.00  
Monday : 11.00-12.00
- ▶ **Email:** [arzum.karatas@gediz.edu.tr](mailto:arzum.karatas@gediz.edu.tr)
- ▶ **Lab Website:**  
[http://arzumkaratas.weebly.com/com102\\_2016.html](http://arzumkaratas.weebly.com/com102_2016.html)
- ▶ For any general questions about the course, **use my Office Hours** or send me an email for having **an appointment** beyond office hours!

# General Information(cont.)

- ▶ **Lab Assistant:** Res.Asst. Feyza GALİP
- ▶ **Office:** D-227
  
- ▶ **Office hours:** Friday : 14.00-15.00  
Friday : 15.00-16.00
  
- ▶ **Email:** [feyza.galip@gediz.edu.tr](mailto:feyza.galip@gediz.edu.tr)

# Agenda

- ▶ Objectives
- ▶ How to install Java Software Development Kit
- ▶ Introduction to Eclipse
- ▶ Examples
- ▶ Q&A

# Agenda

- ▶ Objectives
- ▶ How to install Java Software Development Kit
- ▶ Introduction to Eclipse
- ▶ Examples
- ▶ Q&A

# Objectives

- ▶ Installing Java SE Development Kit
- ▶ Installing Eclipse
- ▶ Introducing development environment
- ▶ Warm-up with some examples

# Agenda

- ▶ Objectives
- ▶ How to install Java Software Development Kit
- ▶ Introduction to Eclipse
- ▶ Examples
- ▶ Q&A

# How to install Java SDK

- ▶ Java SDK(or JDK) is an application created by Sun Microsystems to create and modify Java programs.
- ▶ It is the first step where to start programming in Java.
- ▶ At the labs ...
  - ▣ It's already installed.
- ▶ At home ...



# How to install Java SDK(cont.)

www.oracle.com/technetwork/java/javase/downloads/index.html

ORACLE

Sign In/Register Help Country ▾ Communities ▾ I am a... ▾ I want to... ▾ Search

Products Solutions Downloads Store Support Training Partner

Oracle Technology Network > Java > Java SE > Downloads

Overview Downloads Documentation Community Technologies Training

### Java SE Downloads

Next Releases (Early Access) Embedded Use Previous Releases

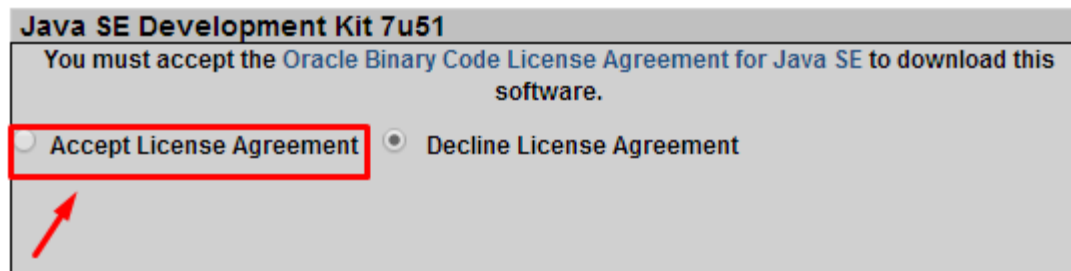
Java Platform (JDK) 7u51

JDK 7u51 & NetBeans 7.4



Download Java JDK 7 from the link  
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

# How to install Java SDK(cont.)

- ▶ Choose Accept Licence Agreement



- ▶ Download the appropriate JDK version according to your OS.

Windows x86	123.64 MB	 <a href="#">jdk-7u51-windows-i586.exe</a>
Windows x64	125.46 MB	 <a href="#">jdk-7u51-windows-x64.exe</a>

- ▶ After downloading ..

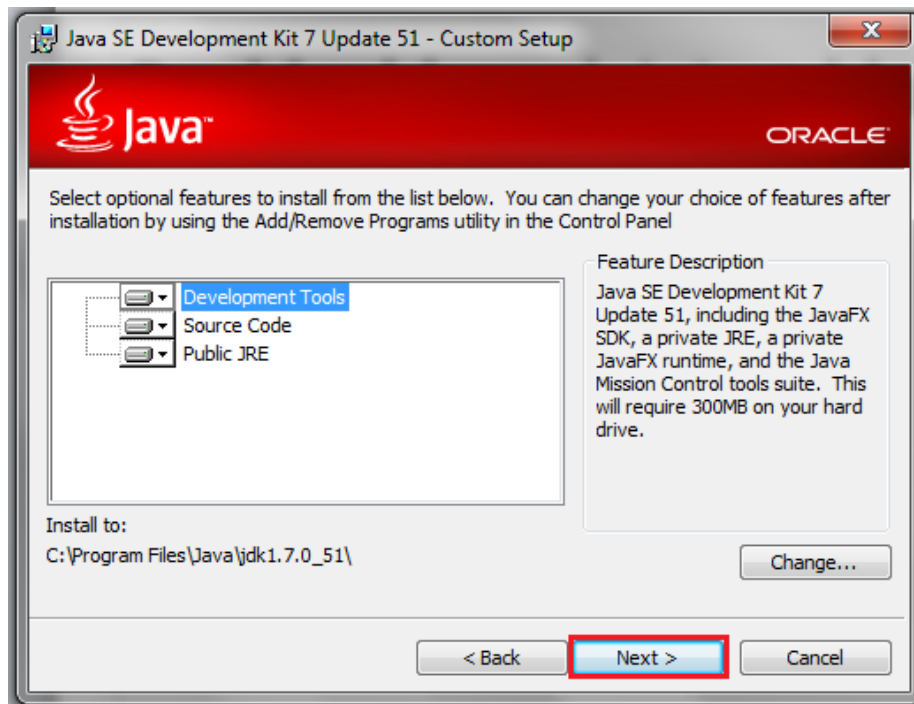
# How to install Java SDK(cont.)

- ▶ Double click on the app downloaded.
- ▶ Click Next button for starting the Installation



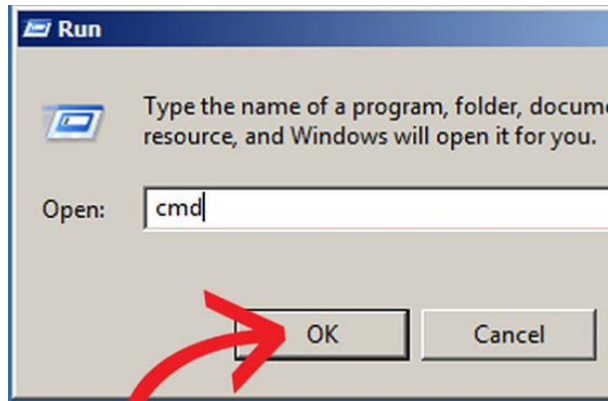
# How to install Java SDK(cont.)

- ▶ Click Next button ..



# How to install Java SDK(cont.)

- ▶ After installation of the JDK, open run by clicking **Start > Run** .

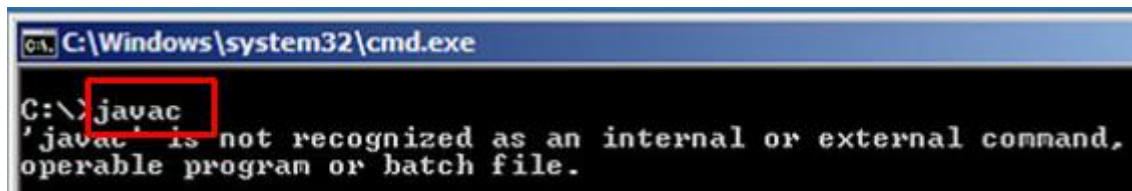


- ▶ Then run «cmd» on click «OK»



# How to install Java SDK(cont.)

- ▶ After focusing the window, type "javac" and press **enter**. If the prompt returns something along the lines of: "'javac' is not recognized as an internal or external command, operable program or batch file«

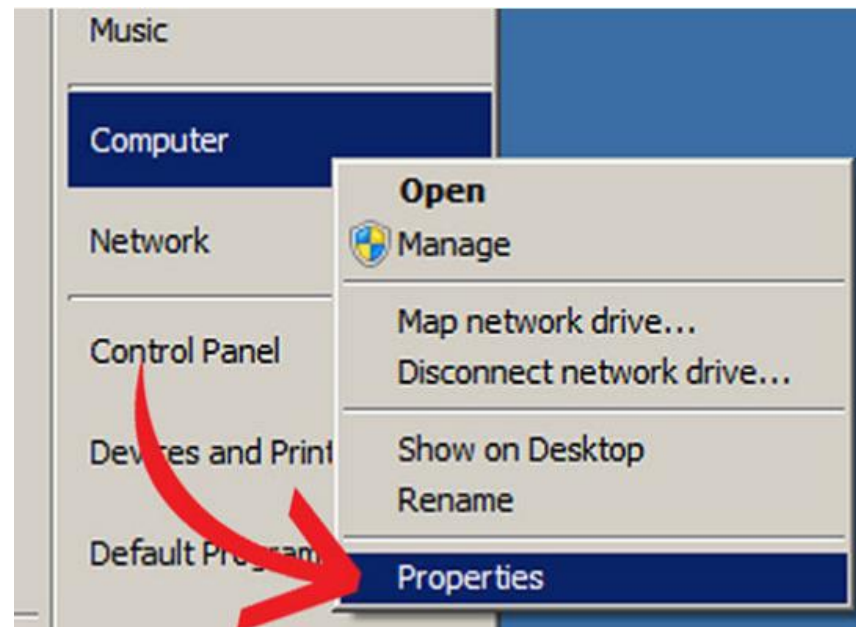


```
C:\Windows\system32\cmd.exe
C:\>javac
'javac' is not recognized as an internal or external command,
operable program or batch file.
```

- ▶ To solve this problem ..

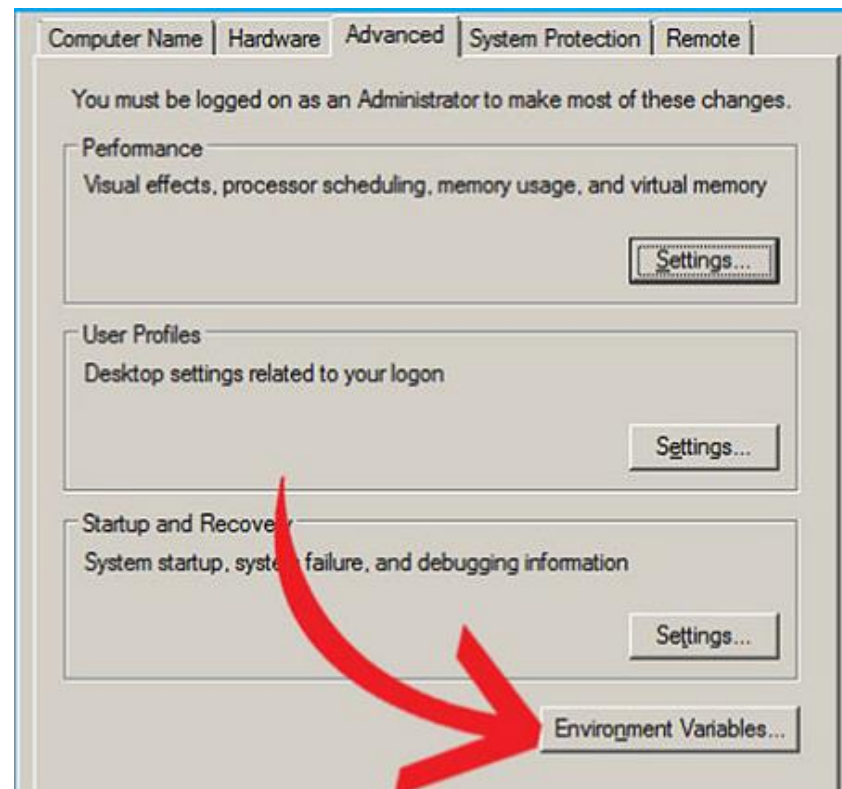
# How to install Java SDK(cont.)

- ▶ Open the properties of **"My Computer"** by either right-clicking the icon on the desktop or **right-clicking Start > My Computer**. When the pop up menu opens, scroll to the bottom and select **"Properties"**.



# How to install Java SDK(cont.)

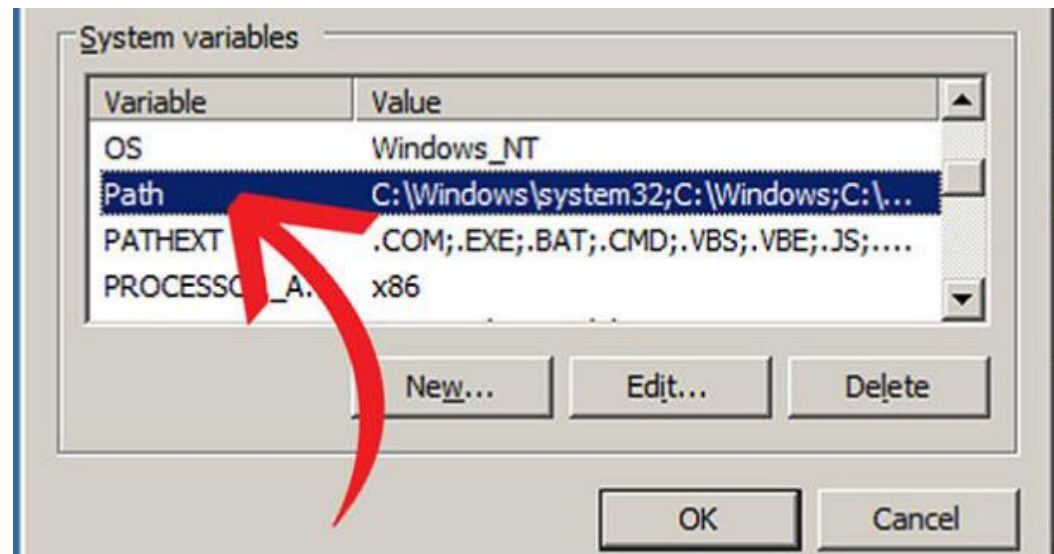
- ▶ This should open a window named "**System Properties**". Click on the "**Advanced**" tab and then click "**Environment Variables**".





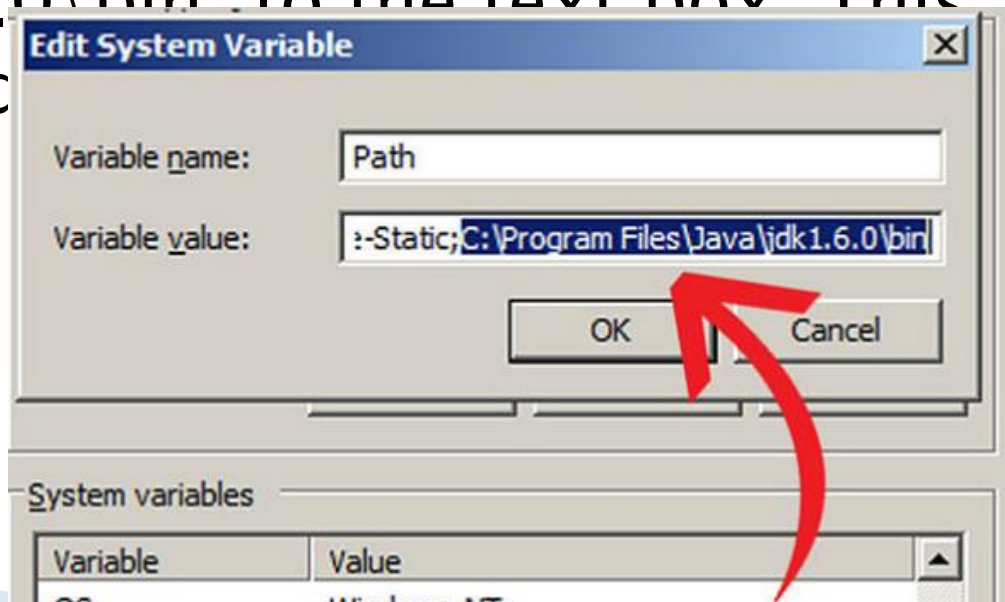
# How to install Java SDK(cont.)

- ▶ Next, another window opens with a lot of confusing sentences and letters. Double-click on the "Path" variable on either of the option boxes. It is recommended to edit the variable in the box "User variables for(your username)".



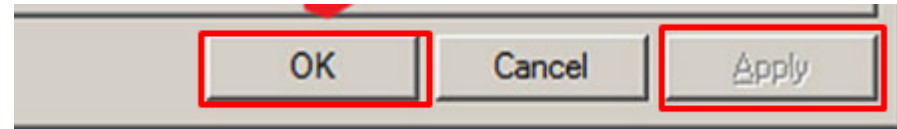
# How to install Java SDK(cont.)

- ▶ Once the variable is opened, a text box in yet another window appears. Careful not to delete anything in this box. At the end of the text box, add a semi-colon if there is not one already, and add "C:\Program Files\Java\jdk1.6.0\bin" to the text box. This is assuming you chose the default installation path of the installation

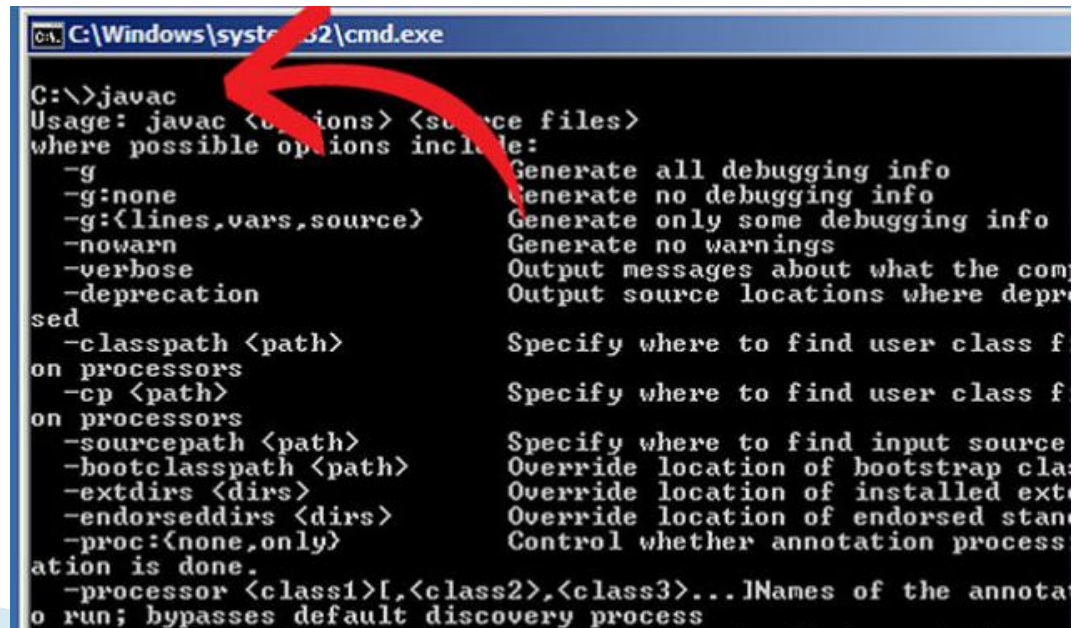


# How to install Java SDK(cont.)

- ▶ Click "Apply" and "OK" to all the windows you have just opened.



- ▶ Open the command prompt again, and then write «javac» again.

A screenshot of a Windows command prompt window. The title bar reads "C:\Windows\system32\cmd.exe". The command prompt shows the command "javac" and its usage information. A red arrow points from the top of the window down to the "javac" command. The text in the window is as follows:

```
C:\Windows\system32\cmd.exe
C:\>javac
Usage: javac <options> <source files>
where possible options include:
-g Generate all debugging info
-g:none Generate no debugging info
-g:<lines,vars,source> Generate only some debugging info
-nowarn Generate no warnings
-verbose Output messages about what the compiler is doing
-deprecation Output source locations where deprecated
code is used
-classpath <path> Specify where to find user class files
on processors
-cp <path> Specify where to find user class files
on processors
-sourcepath <path> Specify where to find input source files
-bootclasspath <path> Override location of bootstrap class files
-extdirs <dirs> Override location of installed extensions
-endorseddirs <dirs> Override location of endorsed standard
classes
-processor:<none,only> Control whether annotation processing
is done.
-processor <class1>[,<class2>,<class3>...]Names of the annotation
processors to run; bypasses default discovery process
```

# How to install Java SDK(cont.)

- ▶ See the java version
- ▶ From command line write «java -version»

```
C:\Users\arzum.karatas>java -version
java version "1.7.0_45"
Java(TM) SE Runtime Environment (build 1.7.0_45-b18)
Java HotSpot(TM) Client VM (build 24.45-b08, mixed mode, sharing)
C:\Users\arzum.karatas>
```

# How to install Java SDK(cont.)

## Sample.java

```
public class Sample {  
  
    public static void main (String[] args) {  
  
        System.out.println("\n\n Congratulations! Your  
        Java Installation worked!\n");  
  
    }  
}
```

<http://pastebin.com/x7t9NNUf>

# How to install Java SDK(cont.)

## ▶ Verifying Your Java Installation

- Open a DOS Command window
- use the '*cd*' command to go to the directory where Sample.java is located.

```
C:\Users\arzum.karatas>cd Desktop
```

- Compile java file to generate bytecode

```
C:\Users\arzum.karatas\Desktop>javac Sample.java
```

- Run java file compiled and obtain output

```
C:\Users\arzum.karatas\Desktop>java Sample
```

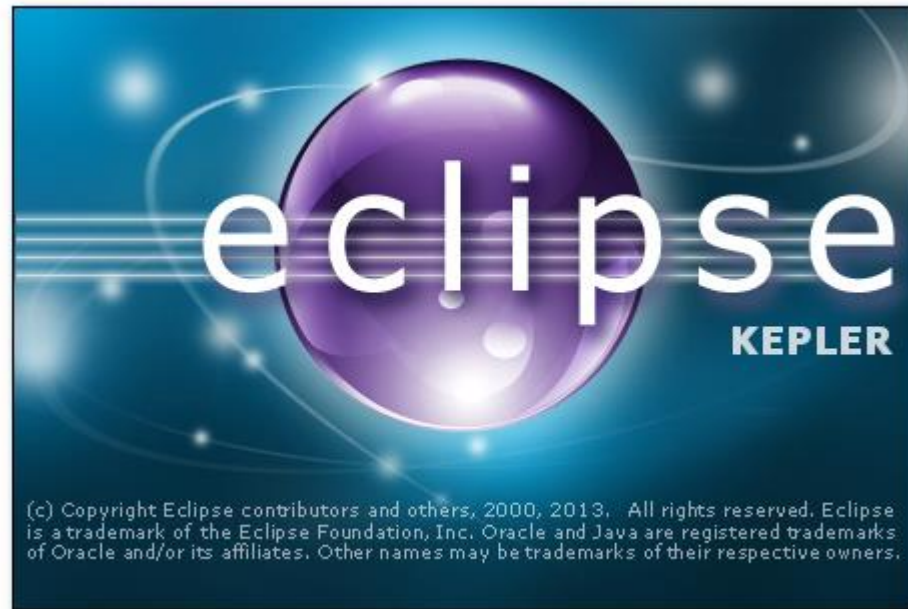
```
Congratulations! Your Java Installation worked!
```

```
Output
```

# Agenda

- ▶ Objectives
- ▶ How to install Java Software Development Kit
- ▶ **Introduction to Eclipse**
- ▶ Examples
- ▶ Q&A

# Introduction to Eclipse





# Introduction to Eclipse(cont.)

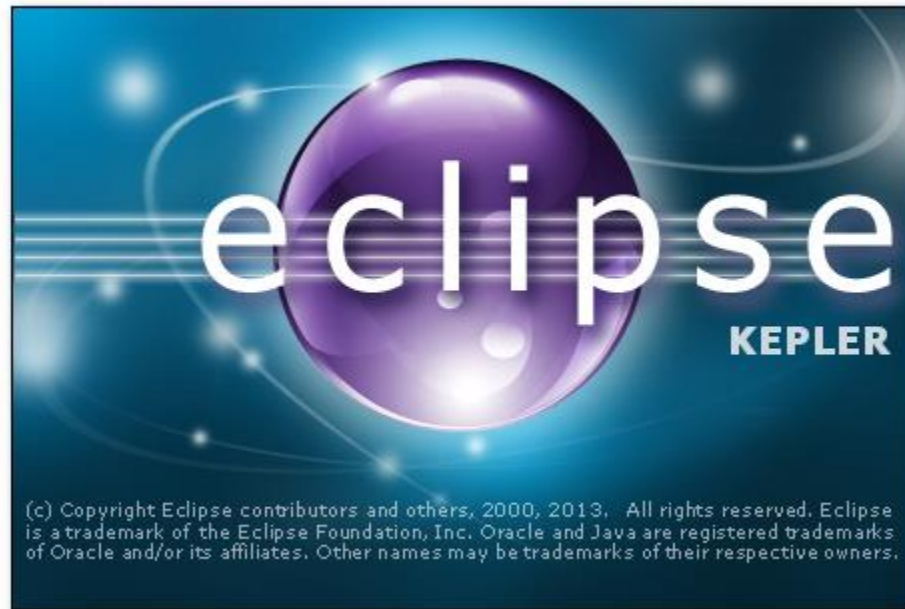
## Getting Eclipse

- ▶ At the labs...
  - It's already installed
- ▶ At home...
  - Download the latest version at:  
<http://www.eclipse.org/>
  - Installation is very simple  
Just unpack the downloaded package
  - Running Eclipse  
Then click **eclipse.exe** (under Windows)

# Introduction to Eclipse(cont.)

## ▶ **Launching Eclipse**

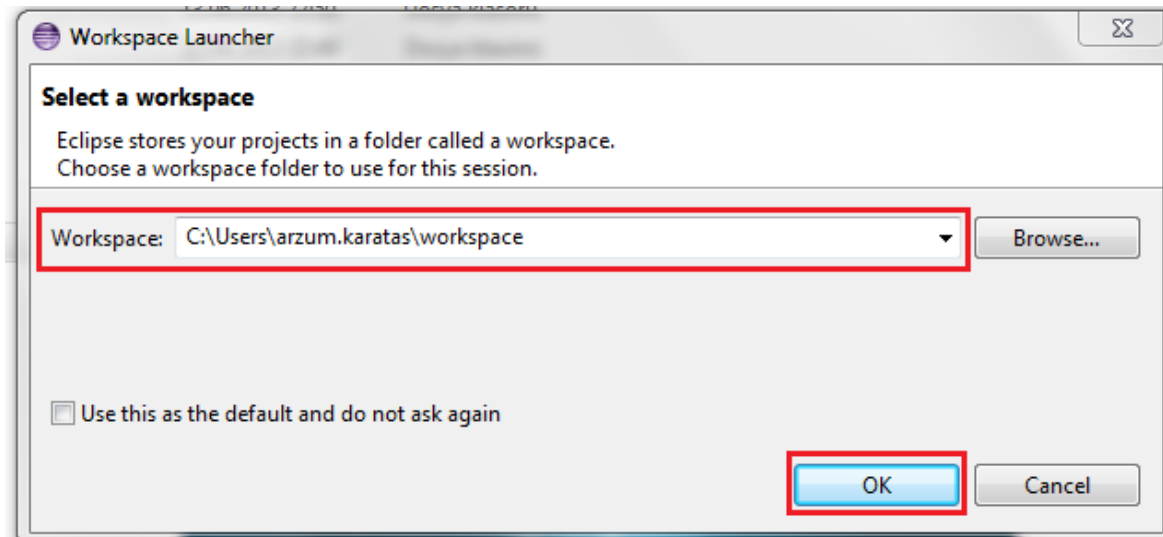
After clicking the eclipse.exe, you should see the following splash screen...



# Introduction to Eclipse(cont.)

## ▶ Selecting Workspace

- In Eclipse, all of your code will live under a workspace
- A workspace is nothing more than a location where we will store our source code and where Eclipse will write out our preferences
- Choose a location where you want to store your files, then click OK



# Introduction to Eclipse(cont.)

## ▶ Eclipse IDE Components

The image shows a screenshot of the Eclipse IDE interface with several callout boxes pointing to different components. The interface includes a menu bar, a toolbar, a Package Explorer, a Hierarchy view, a central Editor Pane, a Task List Pane, an Outline Pane, and a Miscellaneous Pane at the bottom.

**Menubars**  
Full drop down menus plus quick access to common functions

**Perspective Switcher**  
We can switch between various perspectives here

**Task List Pane**  
This contains a list of “tasks” to complete

**Package Explorer Pane**  
This is where our projects/files are listed

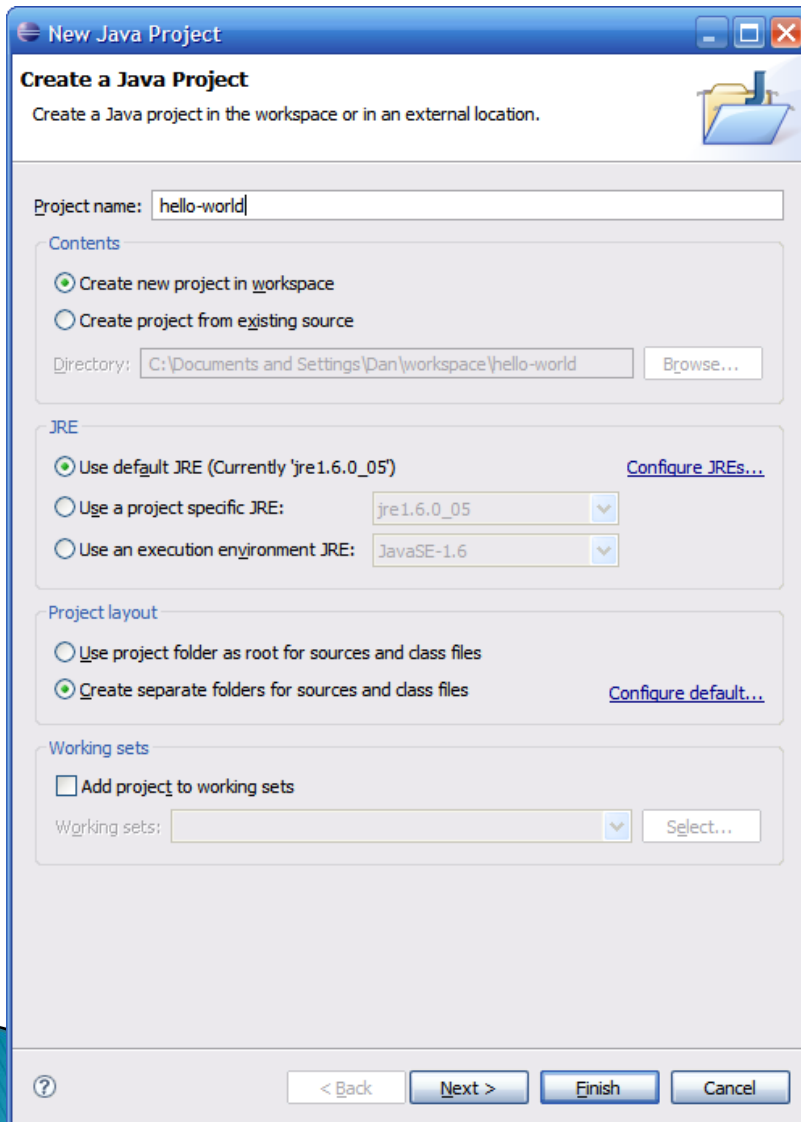
**Editor Pane**  
This is where we edit our source code

**Outline Pane**  
This contains a hierarchical view of a source file

**Miscellaneous Pane**  
Various components can appear in this pane – typically this contains a console and a list of compiler problems



# Introduction to Eclipse(cont.)

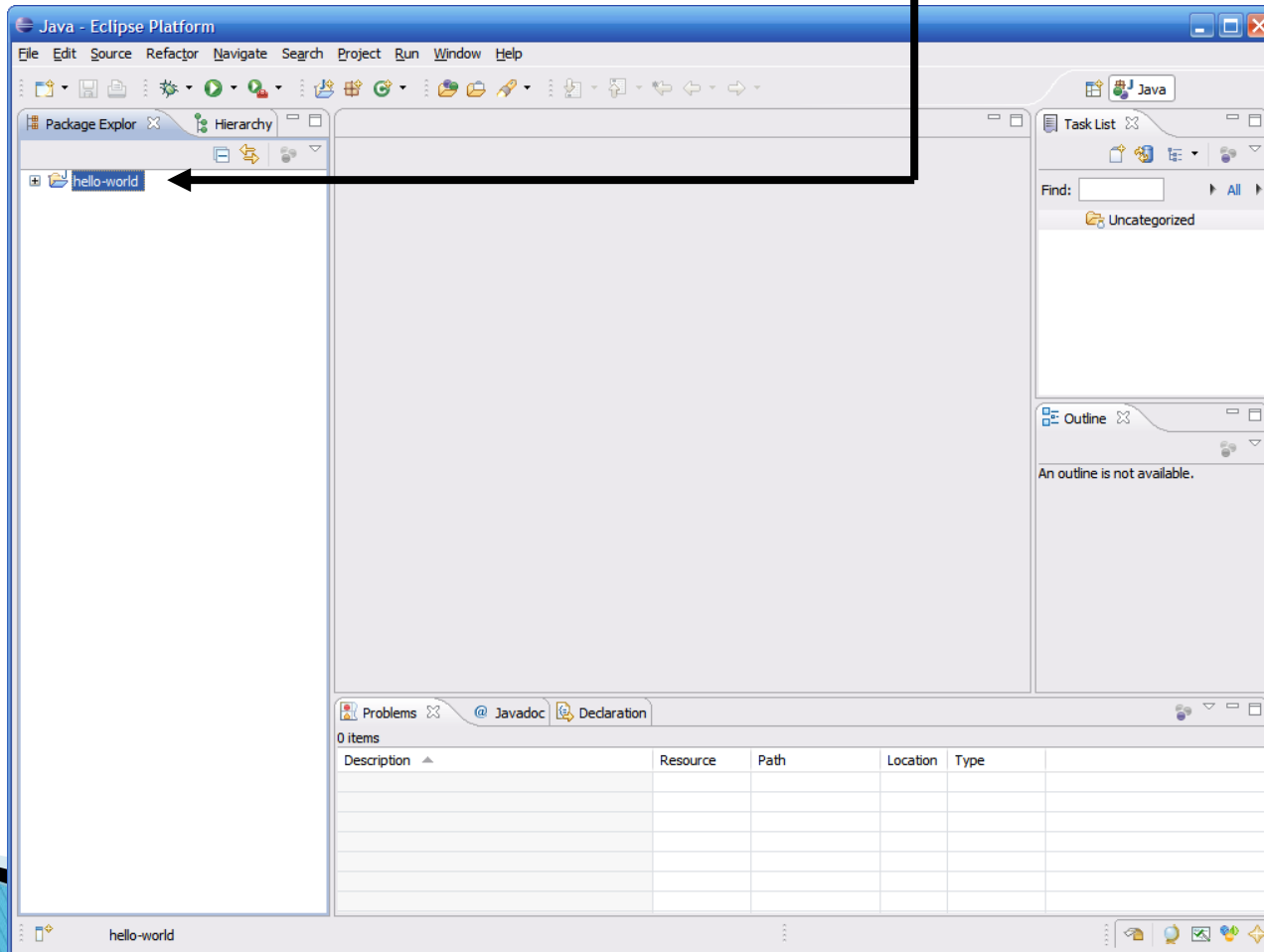


## Creating a New Project (cont.)

- ▶ Enter a name for the project, then click Finish

# Introduction to Eclipse(cont.)

- ▶ The newly created project should then appear under the Package Explorer



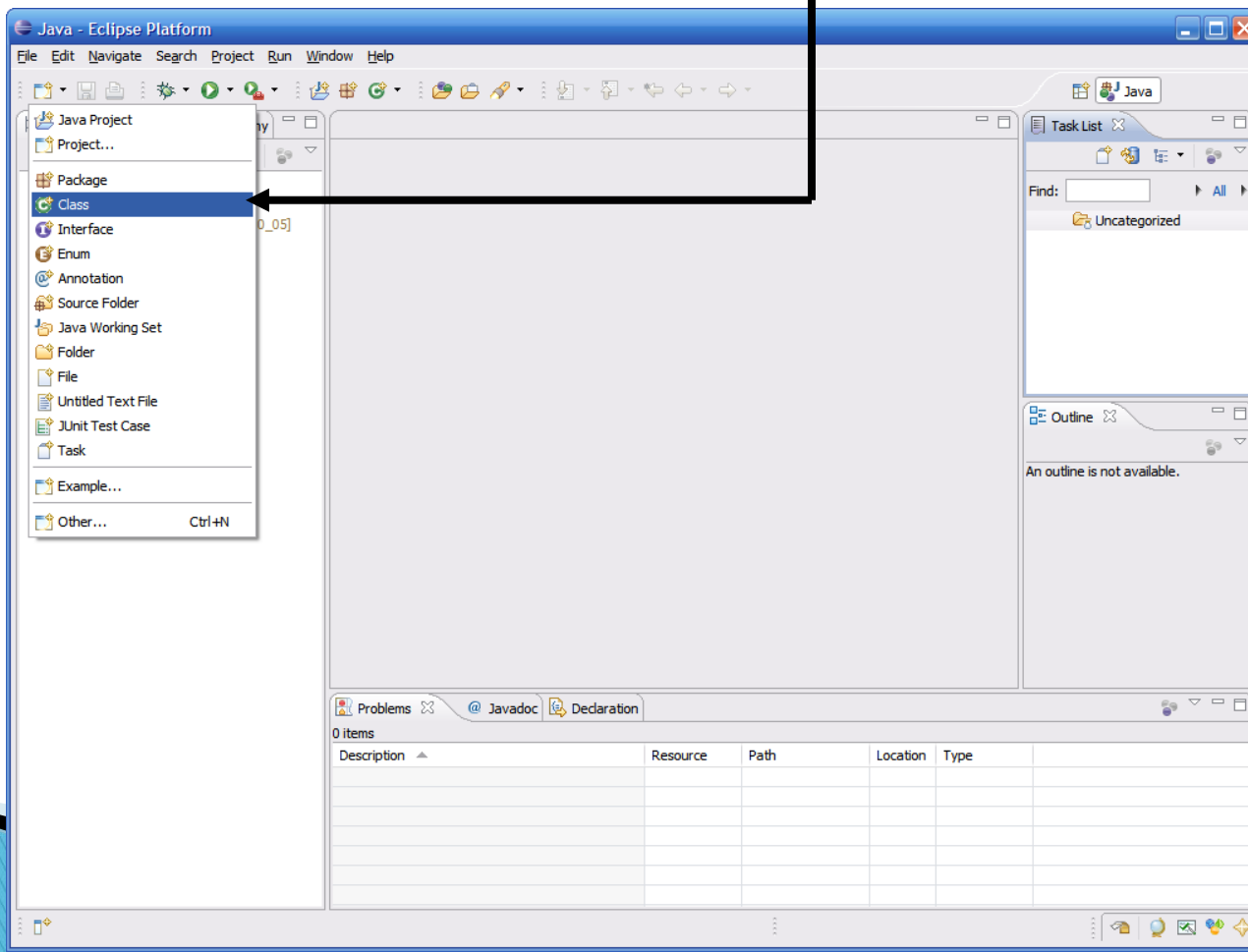




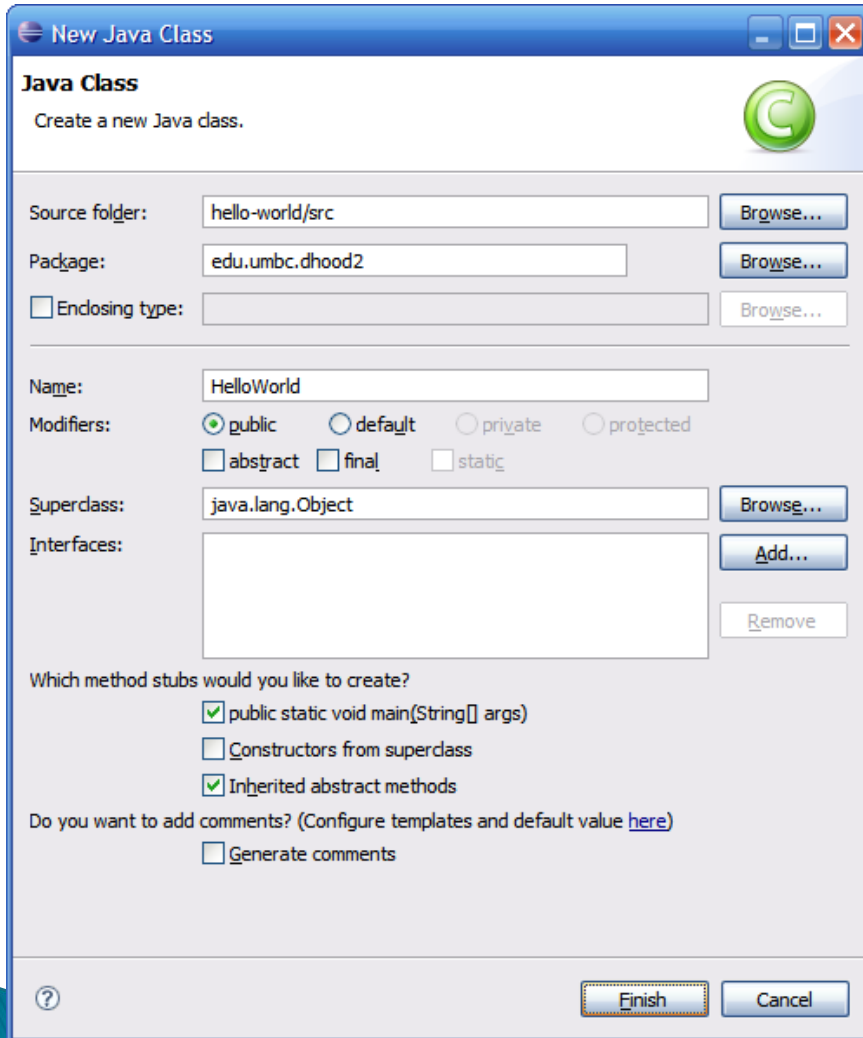
# Introduction to Eclipse(cont.)

## ▶ Creating a Class

To create a class, simply click on the New button, then select Class



# Introduction to Eclipse(cont.)

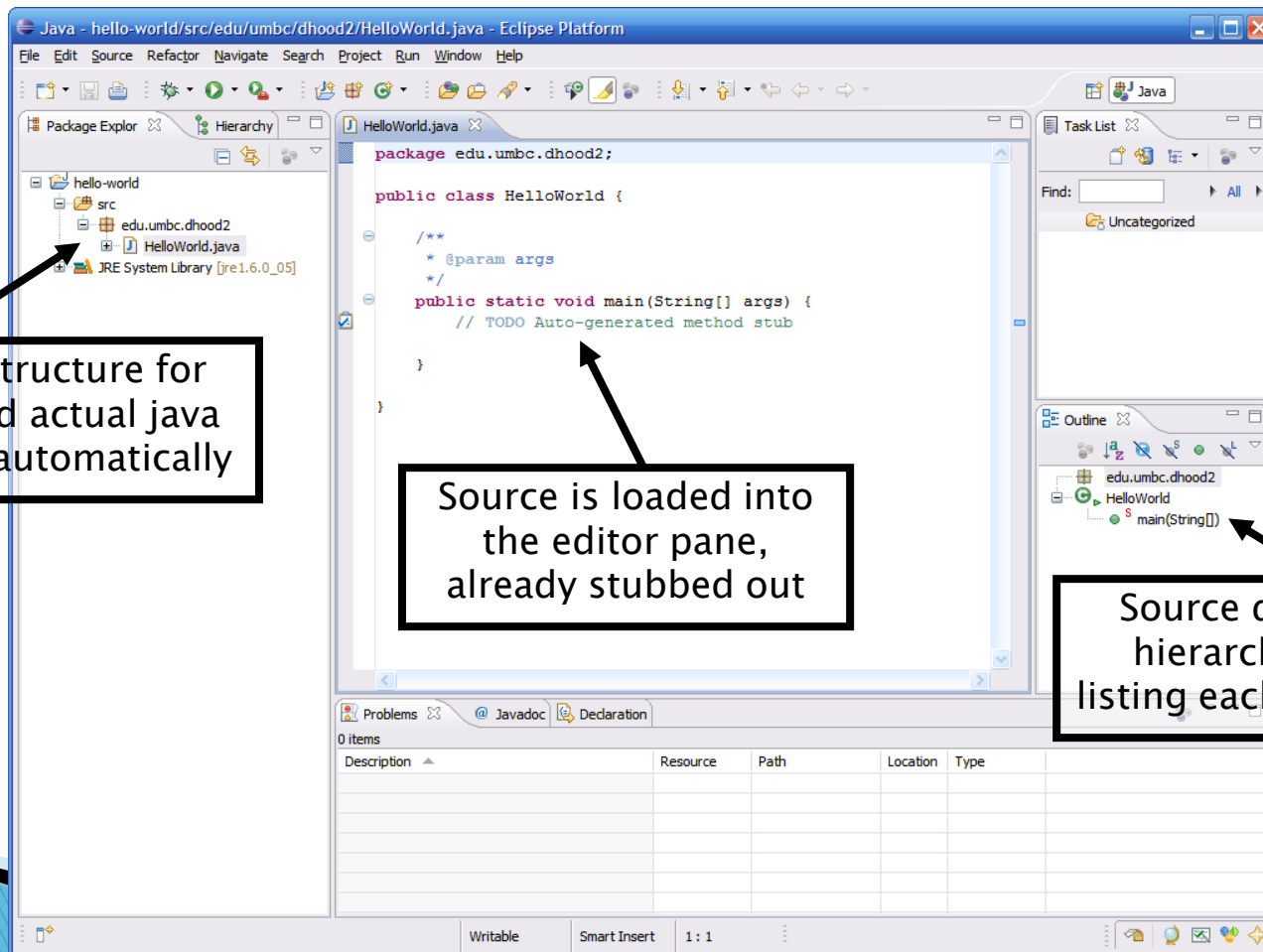


## Creating a Class(cont.)

- ▶ This brings up the new class wizard
- ▶ From here you can specify the following...
  - Package
  - Class name
  - Superclass
  - Whether or not to include a main
  - Etc...
- ▶ Fill in necessary information then click Finish to continue

# Introduction to Eclipse(cont.)

- ▶ As you can see a number of things have now happened...



Directory structure for package and actual java file created automatically

Source is loaded into the editor pane, already stubbed out

Source displayed in a hierarchical fashion listing each method name

# Introduction to Eclipse(cont.)

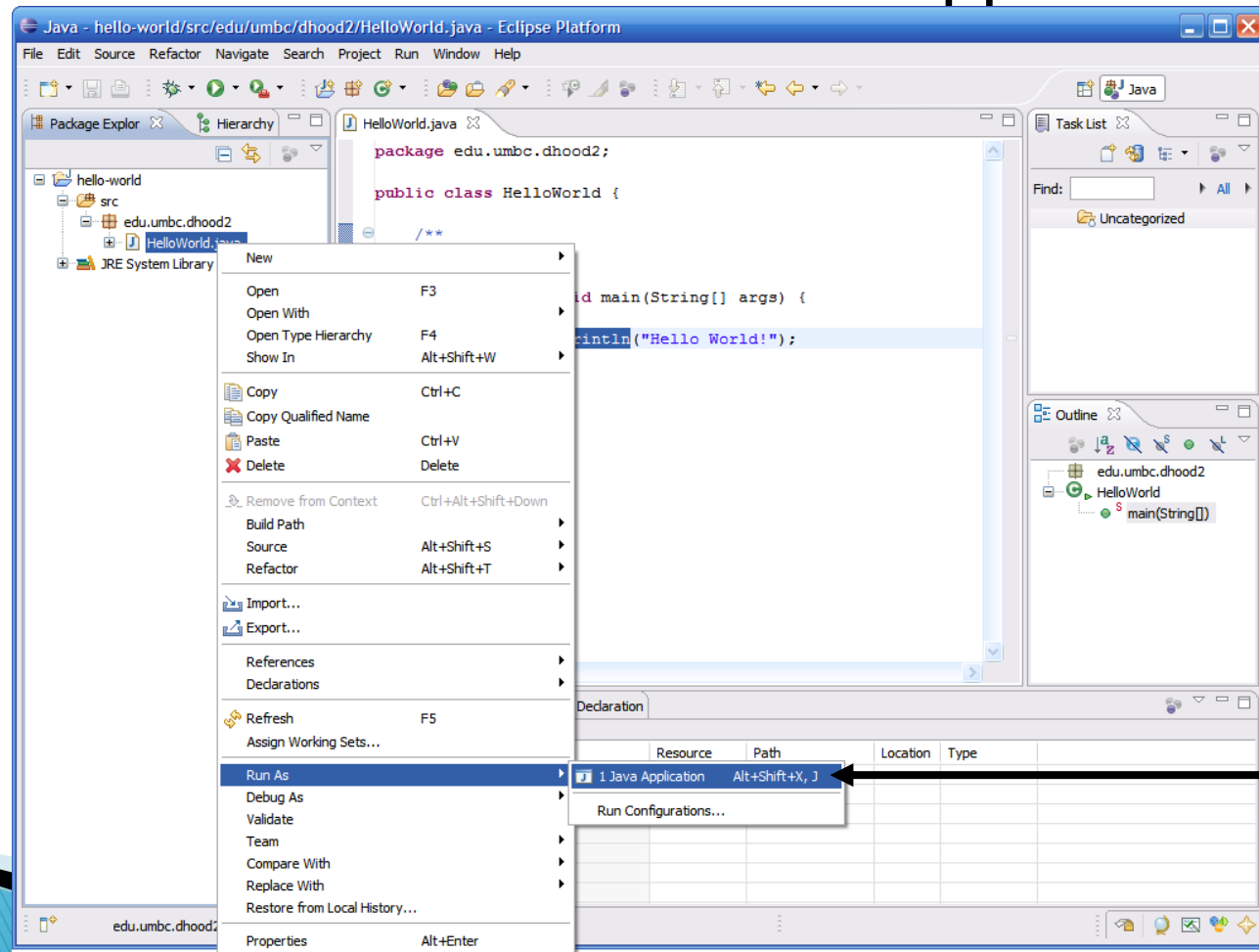
## Compiling Source Code

- ▶ One huge feature of Eclipse is that it automatically compiles your code in the background
  - You no longer need to go to the command prompt and compile code directly
- ▶ This means that errors can be corrected when made
  - We all know that iterative development is the best approach to developing code, but going to shell to do a compile can interrupt the normal course of development
  - This prevents going to compile and being surprised with 100+ errors

# Introduction to Eclipse(cont.)

## Running Code

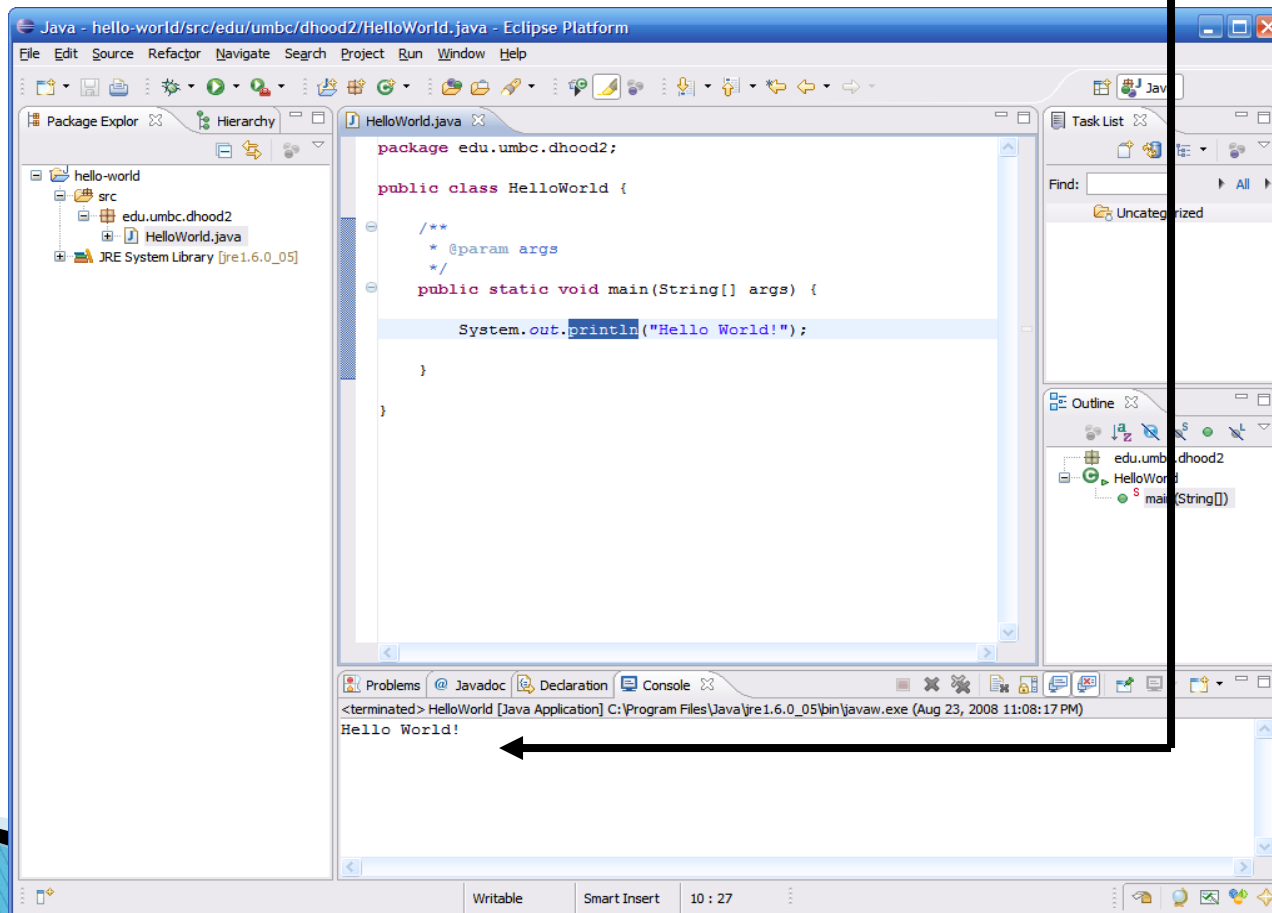
- ▶ An easy way to run code is to right click on the class and select Run As → Java Application



# Introduction to Eclipse(cont.)

## Running Code (continued)

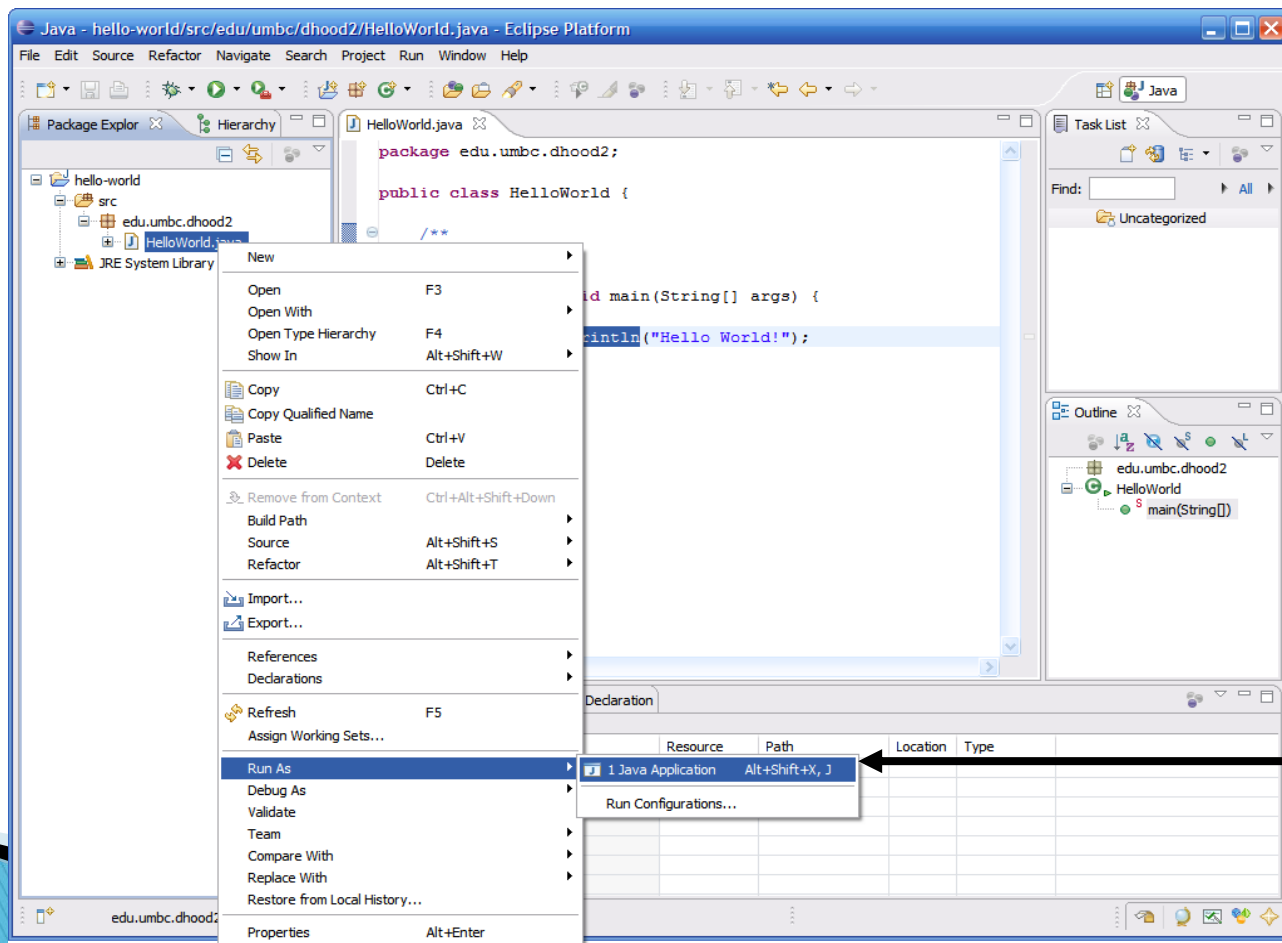
- ▶ The output of running the code can be seen in the Console tab in the bottom pane



# Introduction to Eclipse(cont.)

## Running Code (continued)

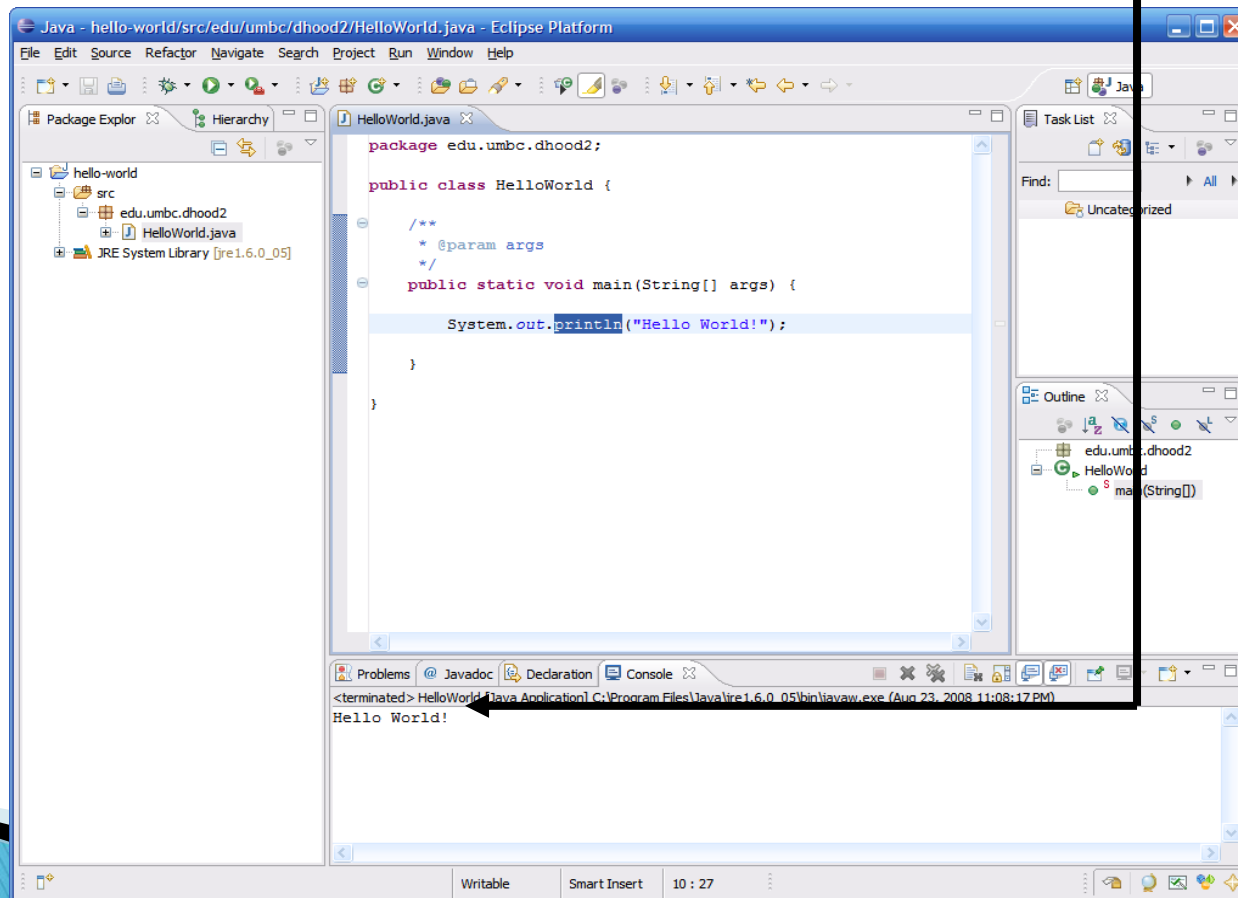
- ▶ An easy way to run code is to right click on the class and select Run As → Java Application



# Introduction to Eclipse(cont.)

## Running Code (continued)

- ▶ The output of running the code can be seen in the Console tab in the bottom pane

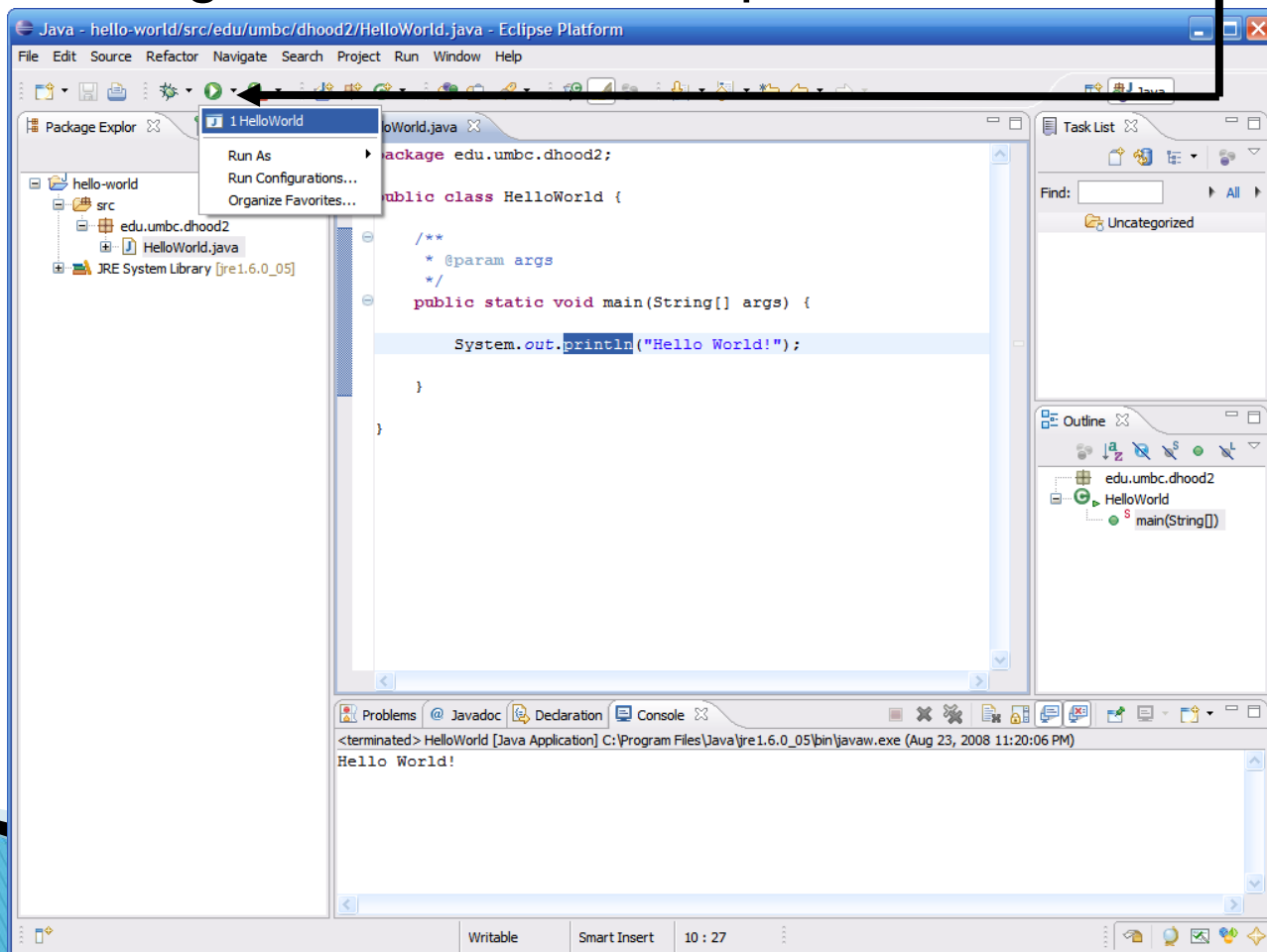




# Introduction to Eclipse(cont.)

## Re-Running Code

- ▶ After you run the code a first time, you can re-run it just by selecting it from the run drop down menu



# Agenda

- ▶ Objectives
- ▶ How to install Java Software Development Kit
- ▶ Introduction to Eclipse
- ▶ **Examples**
- ▶ Q&A

# Example1: Swap Integers

- ▶ Write a C program swapping two integers taken from the user, and then print out the result ..
- ▶ Then, convert the source code into the Java.

# Example2: Find area of a circle

- ▶ Write a C code that calculates the area of a circle whose radius is taken from the user..
- ▶ Then, convert your C code into the Java.

# Example1: Swap Integers

## Sample C code

```
#include <stdio.h>
void main(){
    int firstNumber, secondNumber, temp;

    printf("Enter value of the first integer: ");
    scanf("%d",&firstNumber);
    printf("Enter value of the secondNumber: ");
    scanf("%d",&secondNumber);
    temp = firstNumber;    /* Value of firstNumber is stored in variable temp */
    firstNumber = secondNumber; /* Value of secondNumber is stored in variable firstNumber */
    secondNumber = temp;    /* Value of temp(which contains initial value of firstNumber)
                             is stored in variable secondNumber*/
    printf("\nAfter swapping, value of firstNumber = %d\n", firstNumber);
    printf("After swapping, value of secondNumber = %d", secondNumber);
    ;
}
```

# Example2: Find area of a circle

- ▶ Write a C code that calculates the area of a circle whose a length of a radius is taken from the user..
- ▶ Then, convert your C code into the Java.

# Example2: Find area of a circle

## Sample Code

```
#include<stdio.h>

void main(){

    float radius,area;
    int temp;

    printf("Enter the Length of the radius: ") ;
    scanf("%f\n", &radius);

    area = 3.14*radius*radius;

    printf("\nArea of this circle : %2.2f", area);

}
```



**QUESTIONS**  
**And**  
**Answers**



# Resources

- ▶ <http://www.wikihow.com/Install-the-Java-Software-Development-Kit>
- ▶ <http://www.cs.stanford.edu/people/alee/cs51/penv/installJava.html>
- ▶ <http://www.csee.umbc.edu/courses/undergraduate/341/fall08/Lectures/Eclipse/intro-to-eclipse.ppt>
- ▶ <http://www.cs.bu.edu/~hwxi/academic/courses/C5112/Spring07/slides/EclipseIntroduction.ppt>