

Object – Oriented Programming

Lab #0

● Introduction

- **Assistant #1 : 장윤환**
 - Office : IT/BT관 808호
 - E-mail : dbsgghksdlwkd@hanyang.ac.kr
- **Assistant #2 : 박주현**
 - Office : IT/BT관 808호
 - E-mail : hdhyun216@gmail.com

CSLAB

● Schedule

Week	Lecture	Lab
1		Course Overview ,Intro (Lab 00), Git 사용법
2	Getting Started (Ch. 1) Console Input and Output (Ch. 2)	Variables and String (Lab 01)
3	Defining Classes I (Ch. 4)	Flow Control (Lab 02)
4	Defining Classes II (Ch. 5)	Intro to Classes (Lab 03)
5	Defining Classes II (Ch. 5)	Classes (Lab 04, 05)
6	Arrays (Ch. 6)	Arrays and Enumerated (Lab 06)
7	Inheritance (Ch. 7)	Inheritance (Lab 07)
8	Midterm Exam	
9	Polymorphism and Abstract Classes (Ch. 8)	Polymorphism and Abstract Classes (Lab 08)
10	Exception Handling (Ch. 9)	Exception Handling (Lab 09)
11	Interface and Inner Classes (Ch. 13)	Input and Output Files (Lab 10)
12	Generics and the ArrayList Class (Ch. 14)	Interfaces InnerClasses and Threading (Lab 11)
13	Design Patterns: Singleton, Strategy	Generics and ArrayLists (Lab 12)
14	Design Patterns: Template, Command	Design Patterns: Singleton (Lab 13)
15	Design Patterns: Observer, Decorator	Design Patterns: Observer, Decorator (Lab 14)
16	Final Exam	

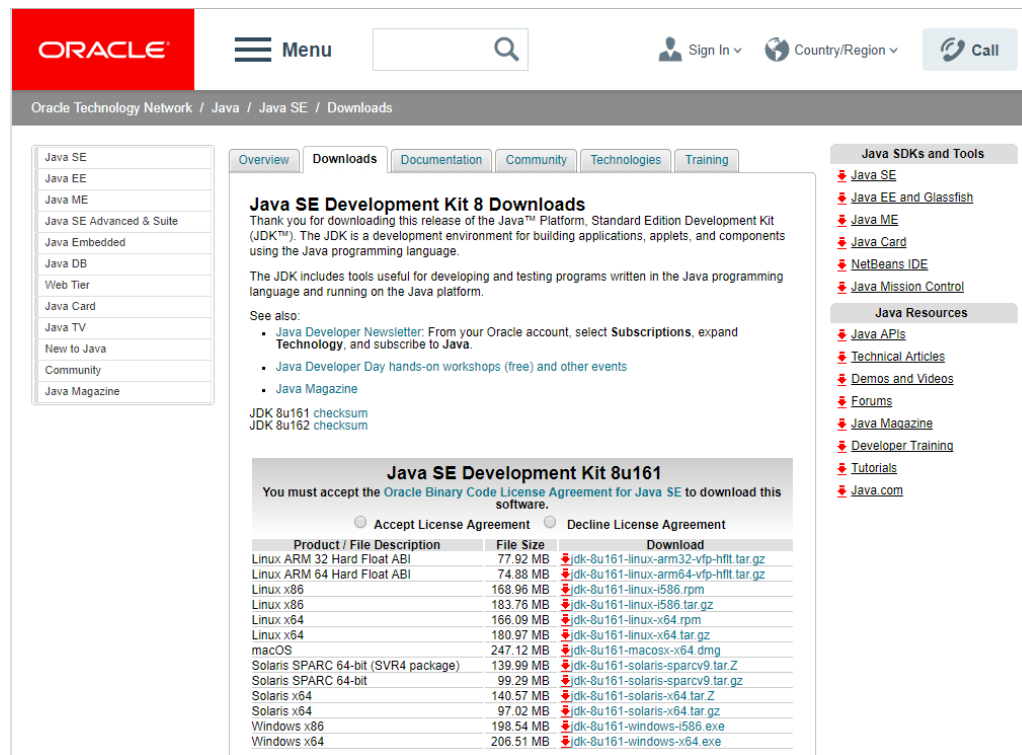
- **Contents**

- Introduction
- Intro to Eclipse

CSLAB

● JDK 설치 (1)

- Java SE Development Kit 8 (JDK, 자바 개발 도구):
<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>



ORACLE

Menu

Sign In Country/Region Call

Oracle Technology Network / Java / Java SE / Downloads

Overview Downloads Documentation Community Technologies Training

Java SE Development Kit 8 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter: From your Oracle account, select **Subscriptions**, expand **Technology**, and subscribe to **Java**.
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK 8u161 checksum
JDK 8u162 checksum

Java SE Development Kit 8u161

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software.

☐ Accept License Agreement ☐ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.92 MB	jdk-8u161-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.88 MB	jdk-8u161-linux-arm64-vfp-hflt.tar.gz
Linux x86	168.96 MB	jdk-8u161-linux-i586.rpm
Linux x86	183.76 MB	jdk-8u161-linux-i586.tar.gz
Linux x64	166.09 MB	jdk-8u161-linux-x64.rpm
Linux x64	180.97 MB	jdk-8u161-linux-x64.tar.gz
macOS	247.12 MB	jdk-8u161-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	139.99 MB	jdk-8u161-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.29 MB	jdk-8u161-solaris-sparcv9.tar.gz
Solaris x64	140.57 MB	jdk-8u161-solaris-x64.tar.Z
Solaris x64	97.02 MB	jdk-8u161-solaris-x64.tar.gz
Windows x86	198.54 MB	jdk-8u161-windows-i586.exe
Windows x64	206.51 MB	jdk-8u161-windows-x64.exe

Java SDKs and Tools

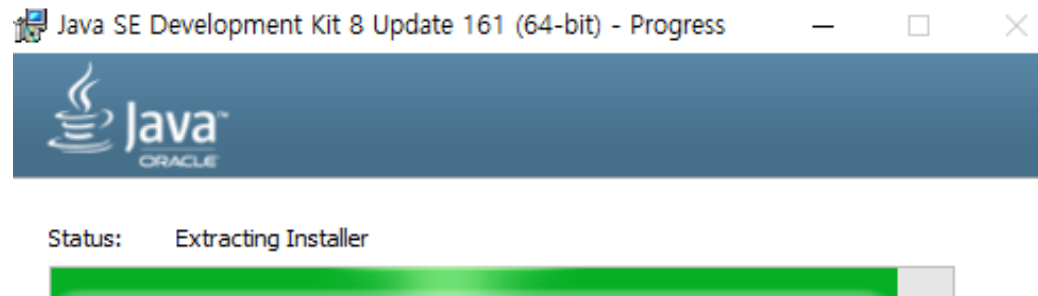
- Java SE
- Java EE and Glassfish
- Java ME
- Java Card
- NetBeans IDE
- Java Mission Control

Java Resources

- Java APIs
- Technical Articles
- Demos and Videos
- Forums
- Java Magazine
- Developer Training
- Tutorials
- Java.com

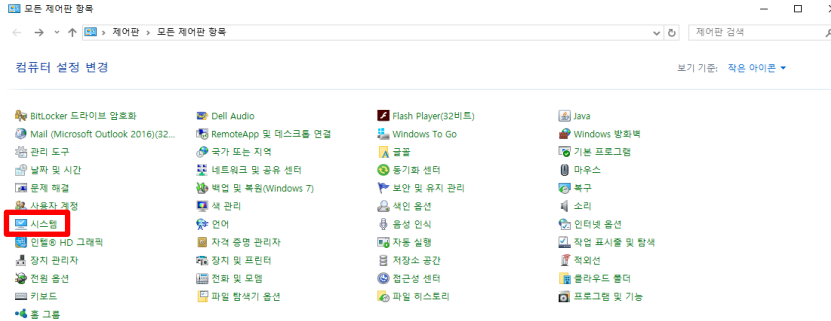
● JDK 설치 (2)

- JDK 설치



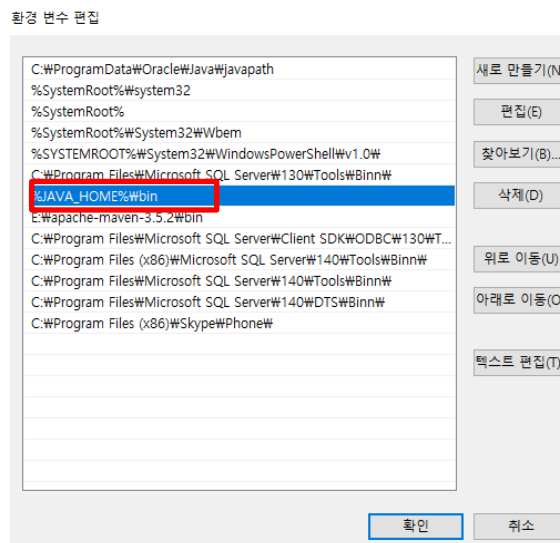
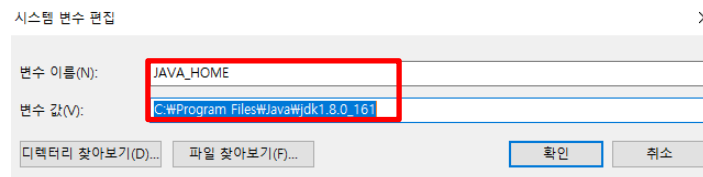
● JDK 설치 (3)

● 환경변수 설정



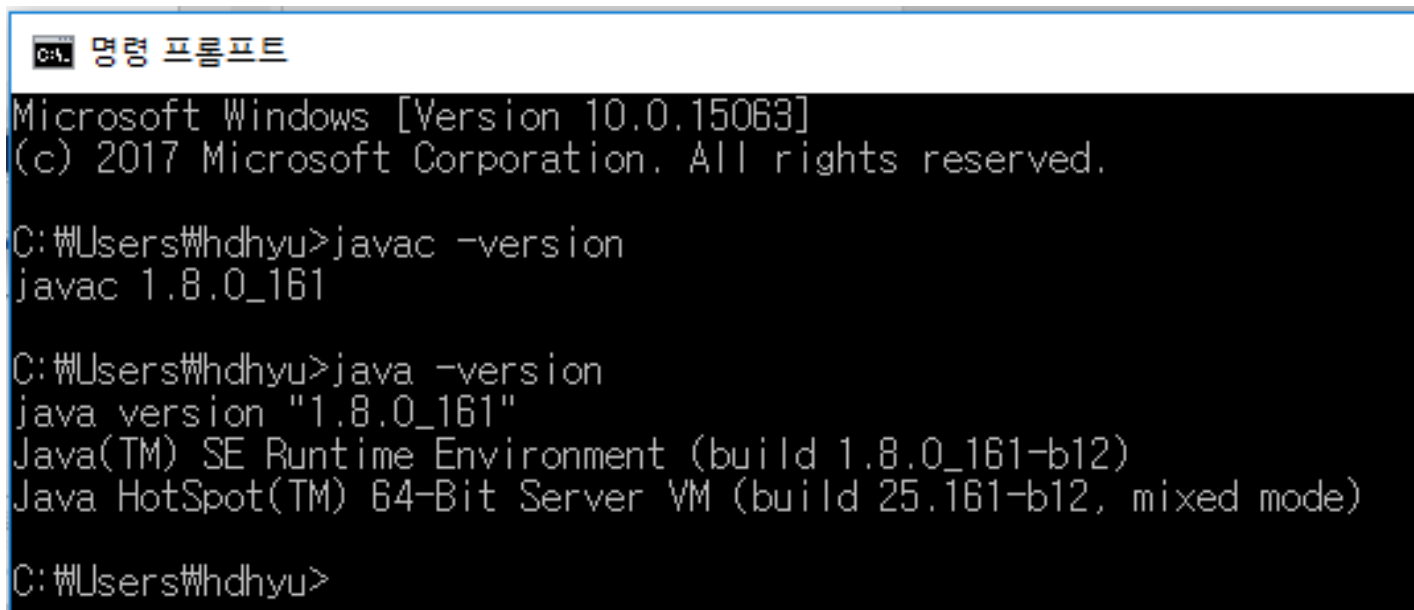
● JDK 설치 (4)

- 고급-> 환경 변수-> 시스템 변수, 새로 만들기-> JAVA_HOME, C:\Program Files\Java\jdk1.8.0_161-> 시스템 변수, Path 편집-> %JAVA_HOME%\bin



● JDK 설치 (5)

- Window + r -> cmd, enter -> 명령 프롬프트 접속 -> javac -version, java version 입력 -> 아래와 같이 출력된다면 JDK 설치 완료



```
C:\> 명령 프롬프트

Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\hdhyu>javac -version
javac 1.8.0_161

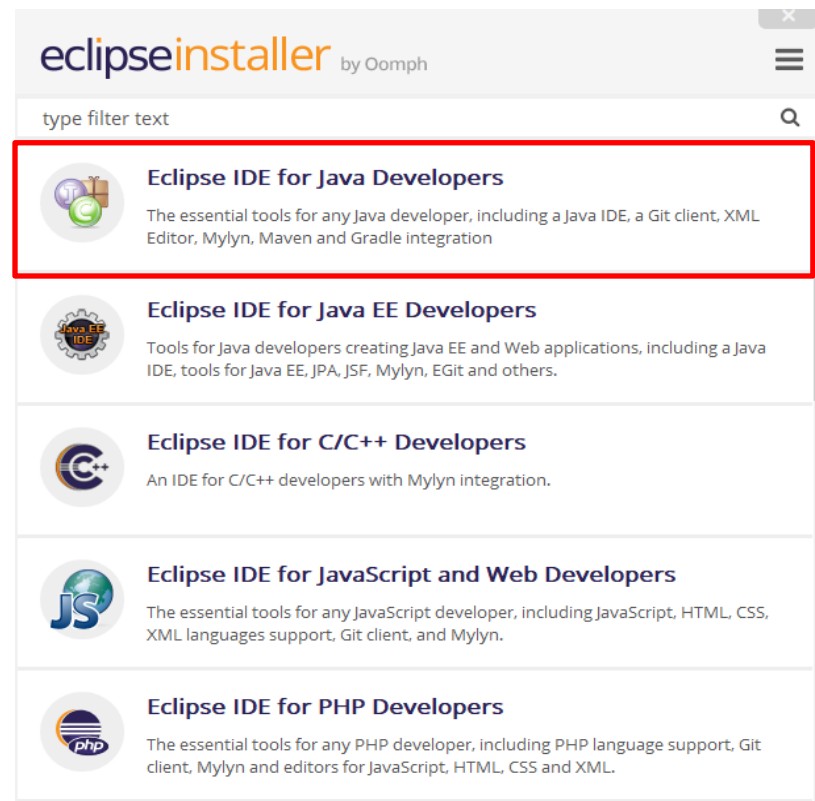
C:\Users\hdhyu>java -version
java version "1.8.0_161"
Java(TM) SE Runtime Environment (build 1.8.0_161-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.161-b12, mixed mode)

C:\Users\hdhyu>
```

CSLAB

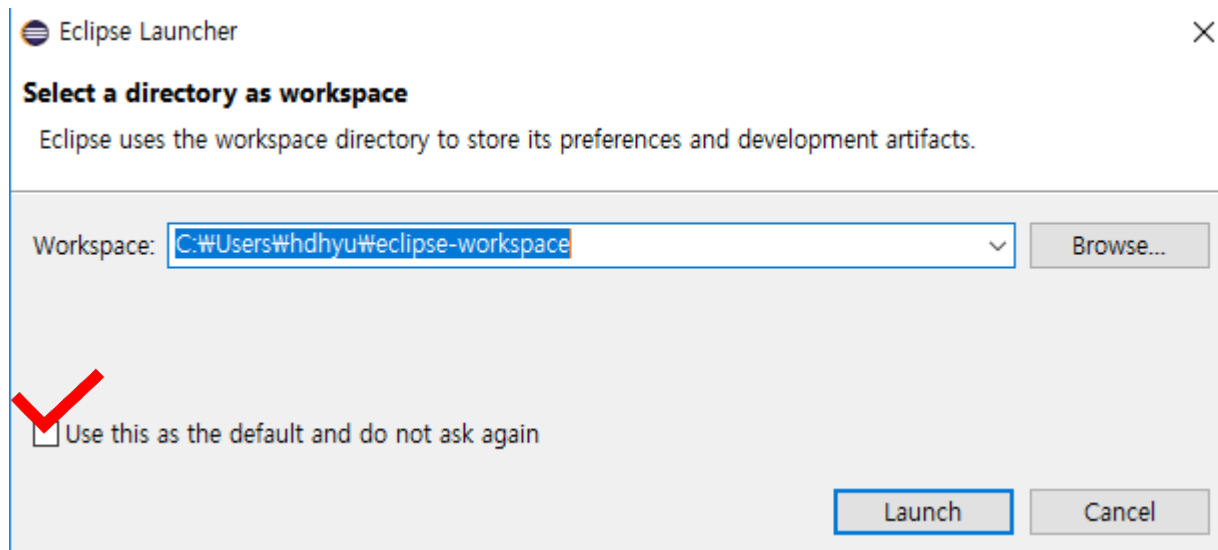
● Eclipse (1)

- 통합 개발 환경 Eclipse (Oxygen 2) 다운로드:
<https://www.eclipse.org/downloads/download.php?file=/oomph/pp/oxygen/R2/eclipse-inst-win64.exe>



● Eclipse (2)

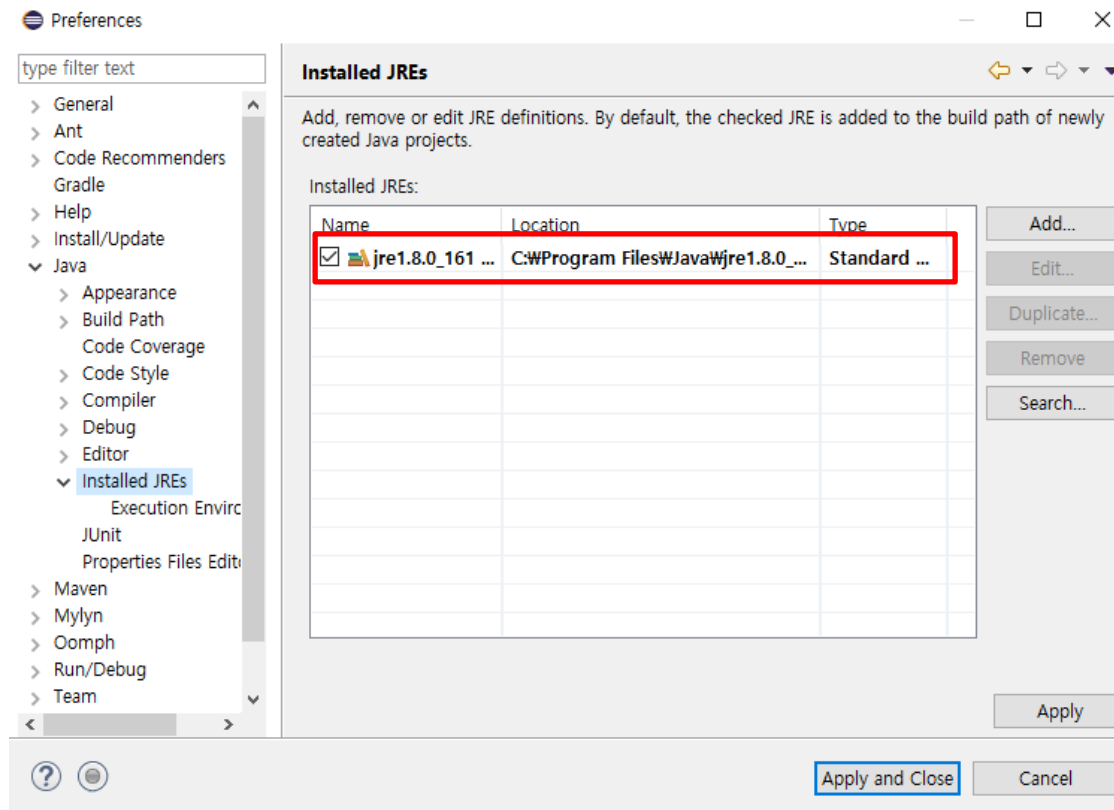
- 클래스가 저장되는 기본 workspace 설정



CSLAB

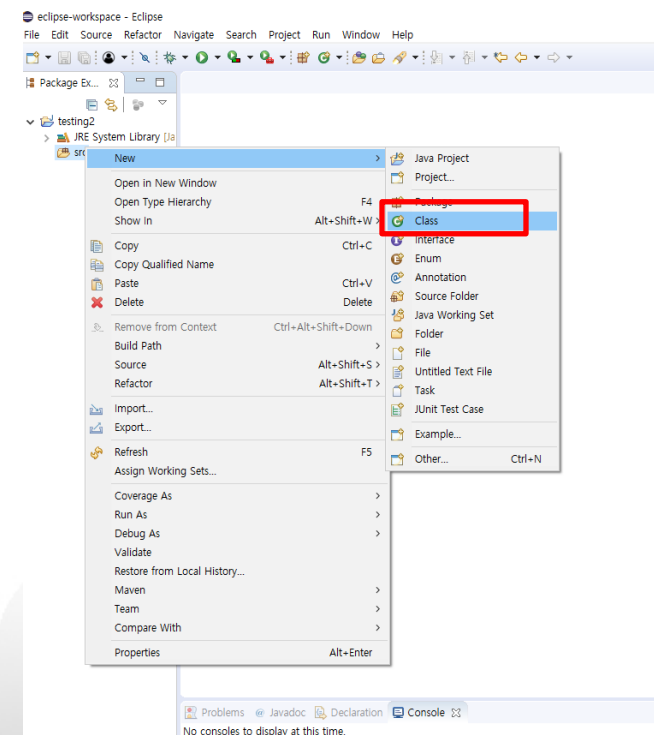
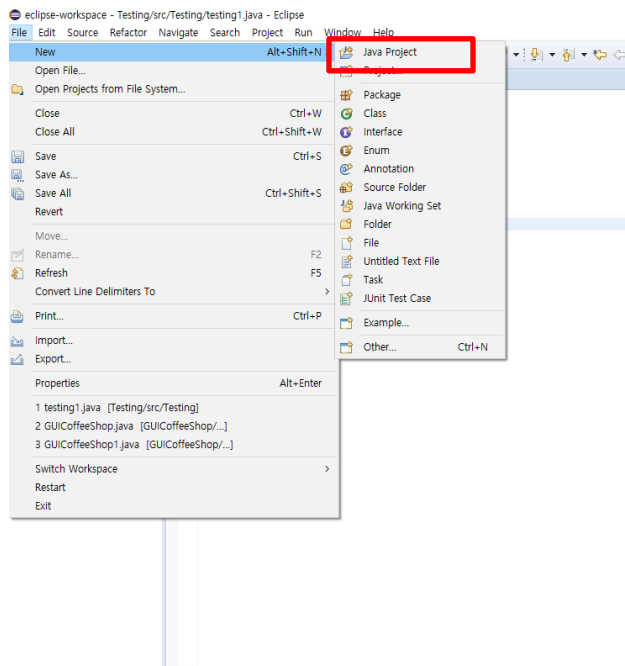
● Eclipse (3)

- Windows -> Preferences -> Java -> Installed JREs
JRE (Java Runtime Environment, JDK에 포함됨) 설정 확인

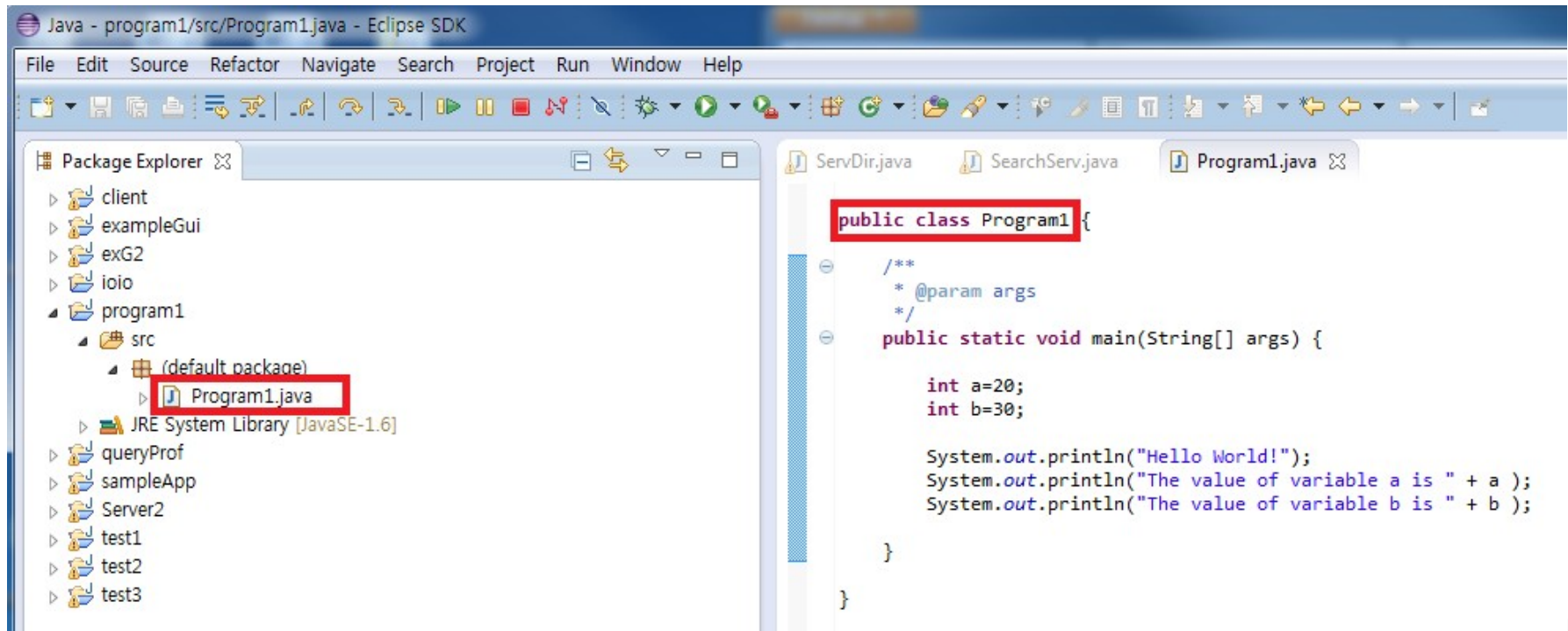


Running Eclipse

- File-> New-> Java Project-> 자바 프로젝트 생성
src 오른쪽 마우스 버튼-> New-> Class-> 패키지 및 클래스 생성
(public static void main(String[] args) 체크 시 메인 메소드 자동 생성)

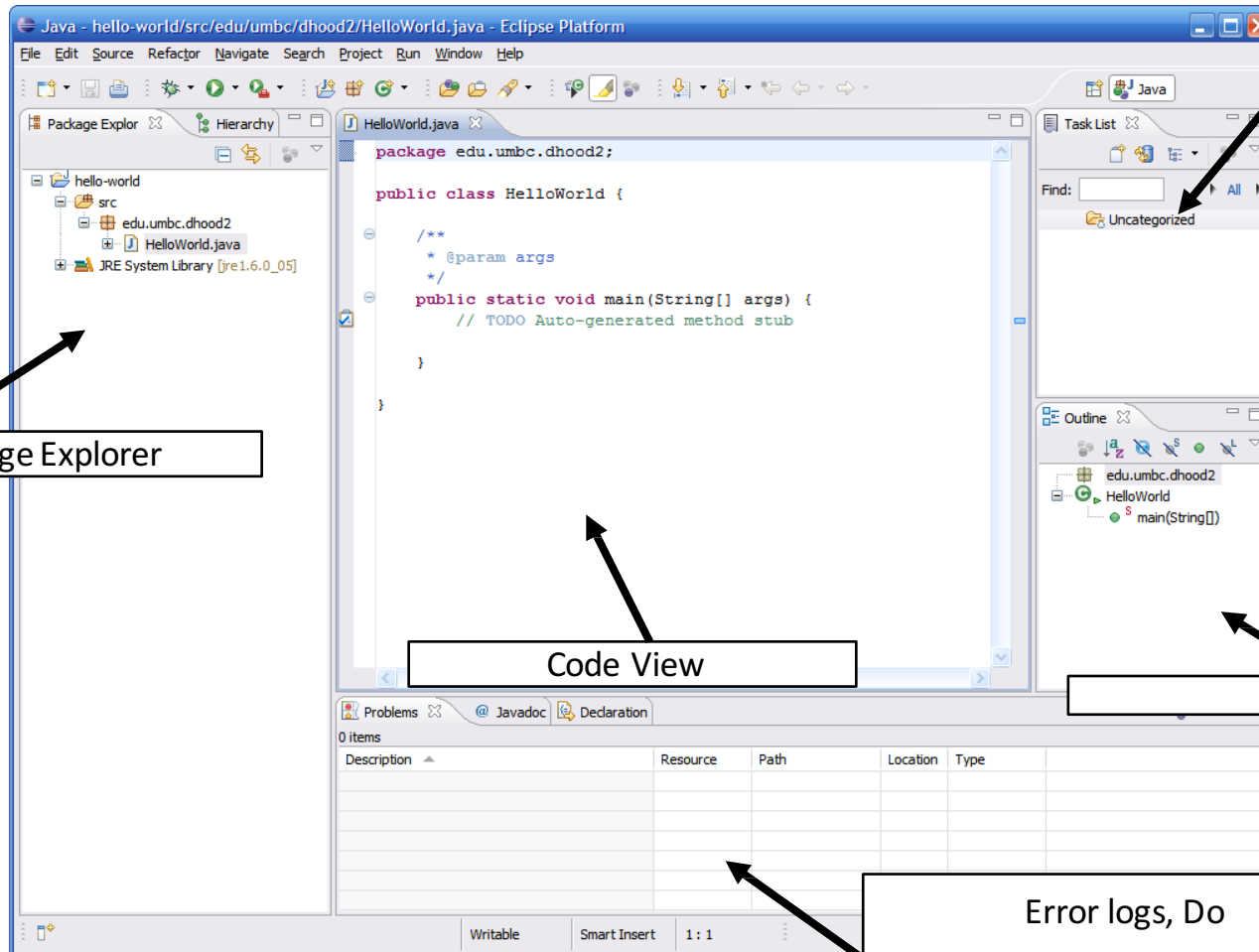


● Running Eclipse (Contd)



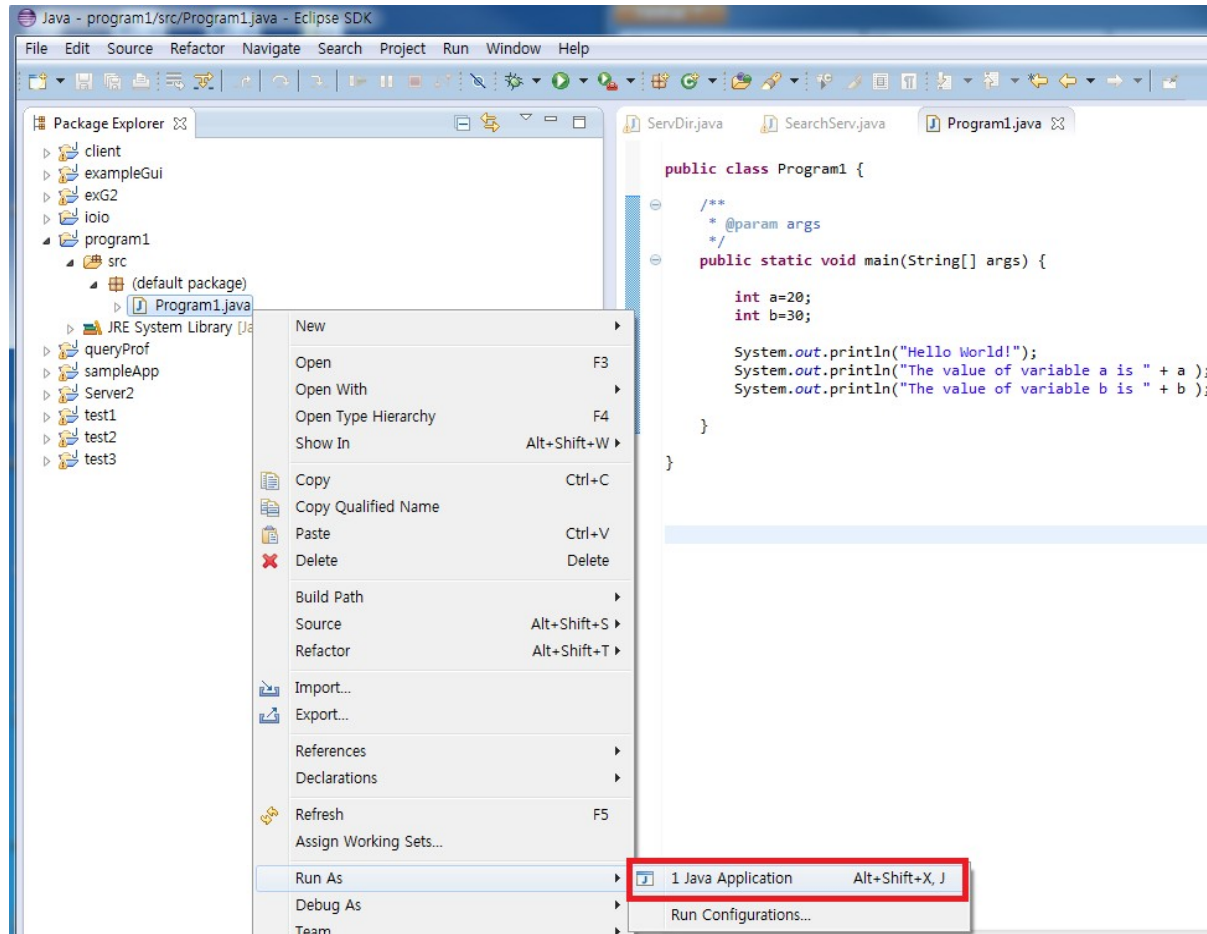
CSLAB

Running Eclipse (Contd)



Error logs, Do
cumentations,
console,properties

Running Eclipse (Contd)

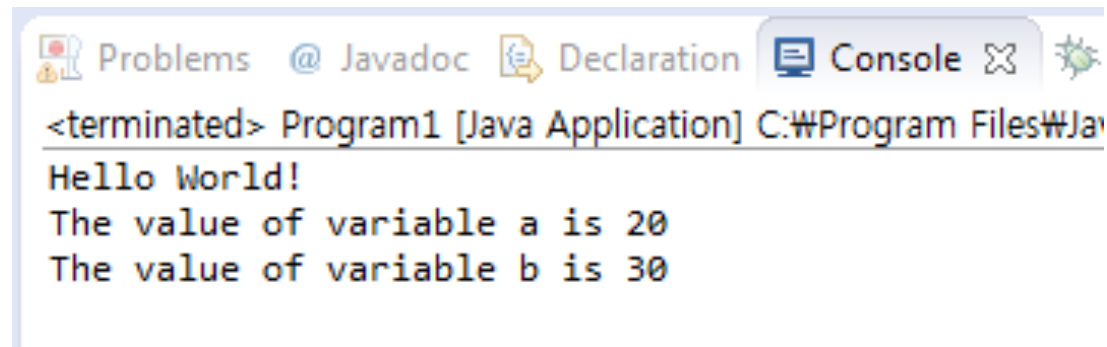


● Running Eclipse (Contd)

- Console Output(System.out.println)

```
int a=20;  
int b=30;
```

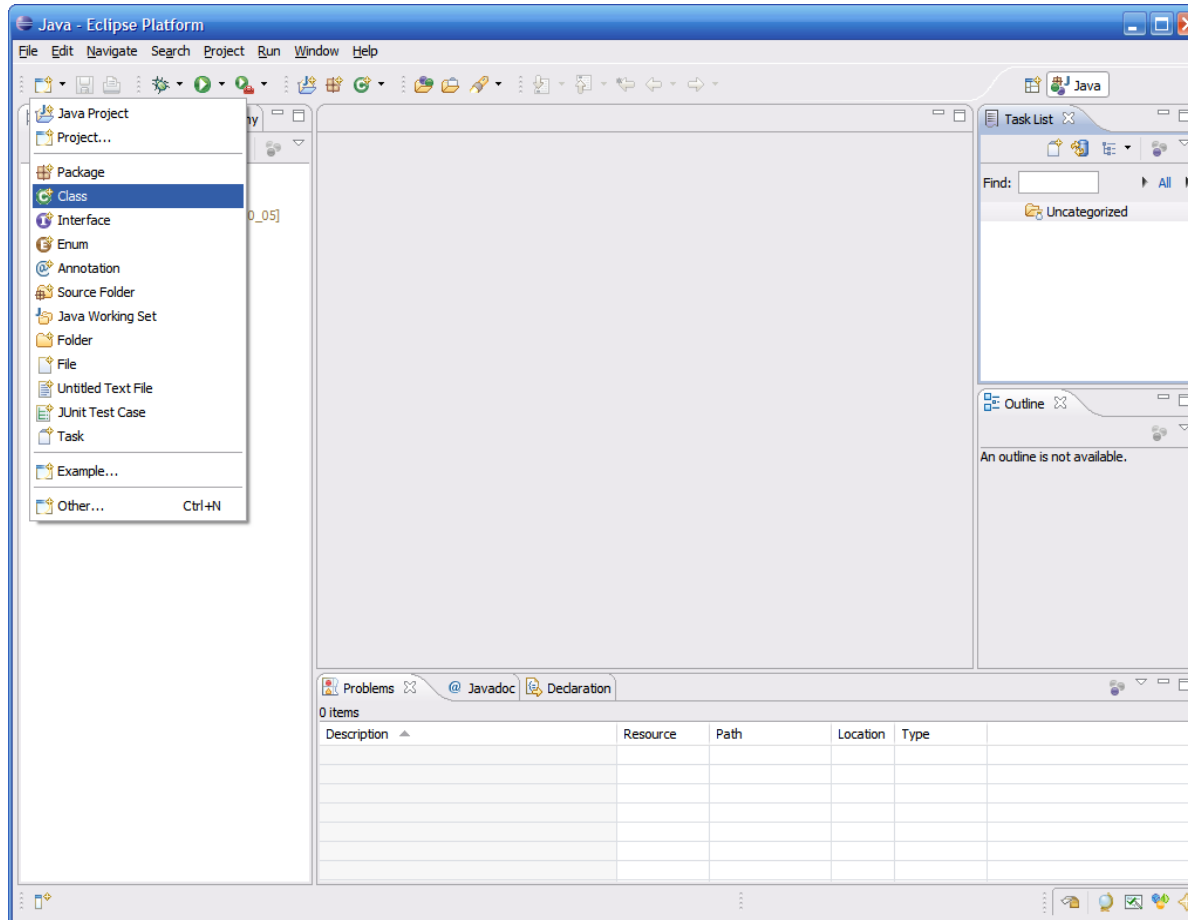
```
System.out.println("Hello World!");  
System.out.println("The value of variable a is " + a );  
System.out.println("The value of variable b is " + b );
```



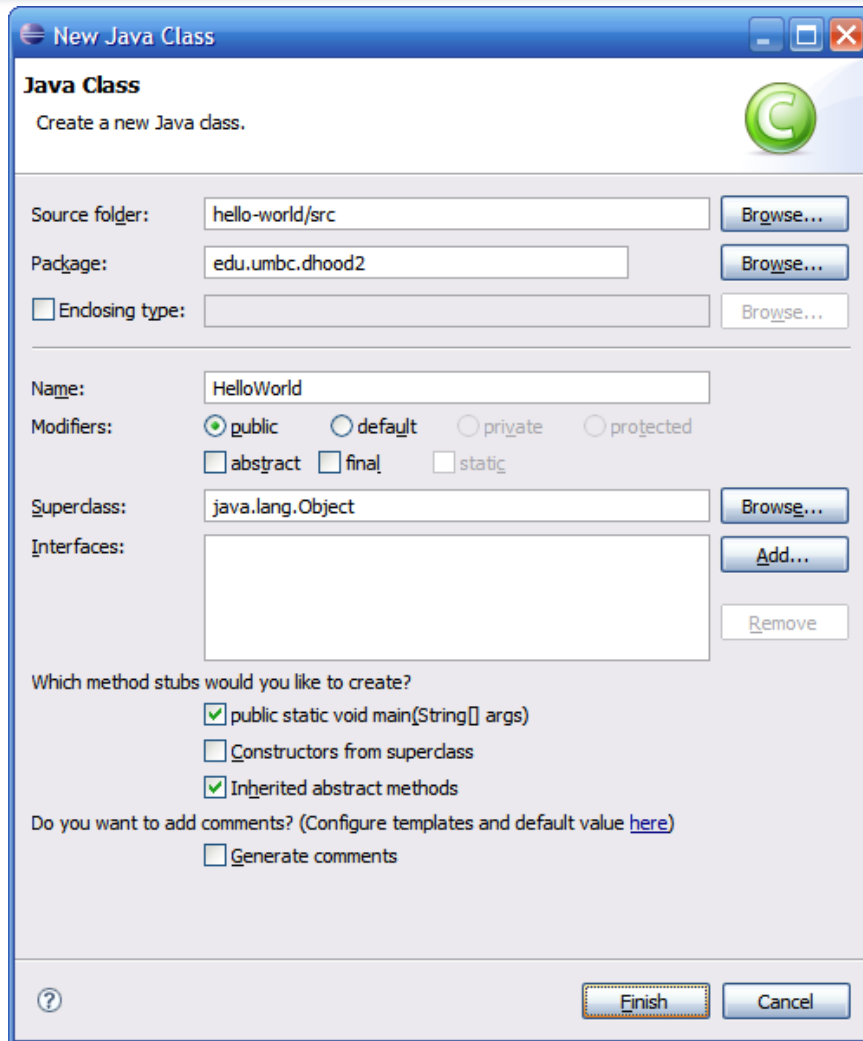
CSLAB

● Creating Class in eclipse

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



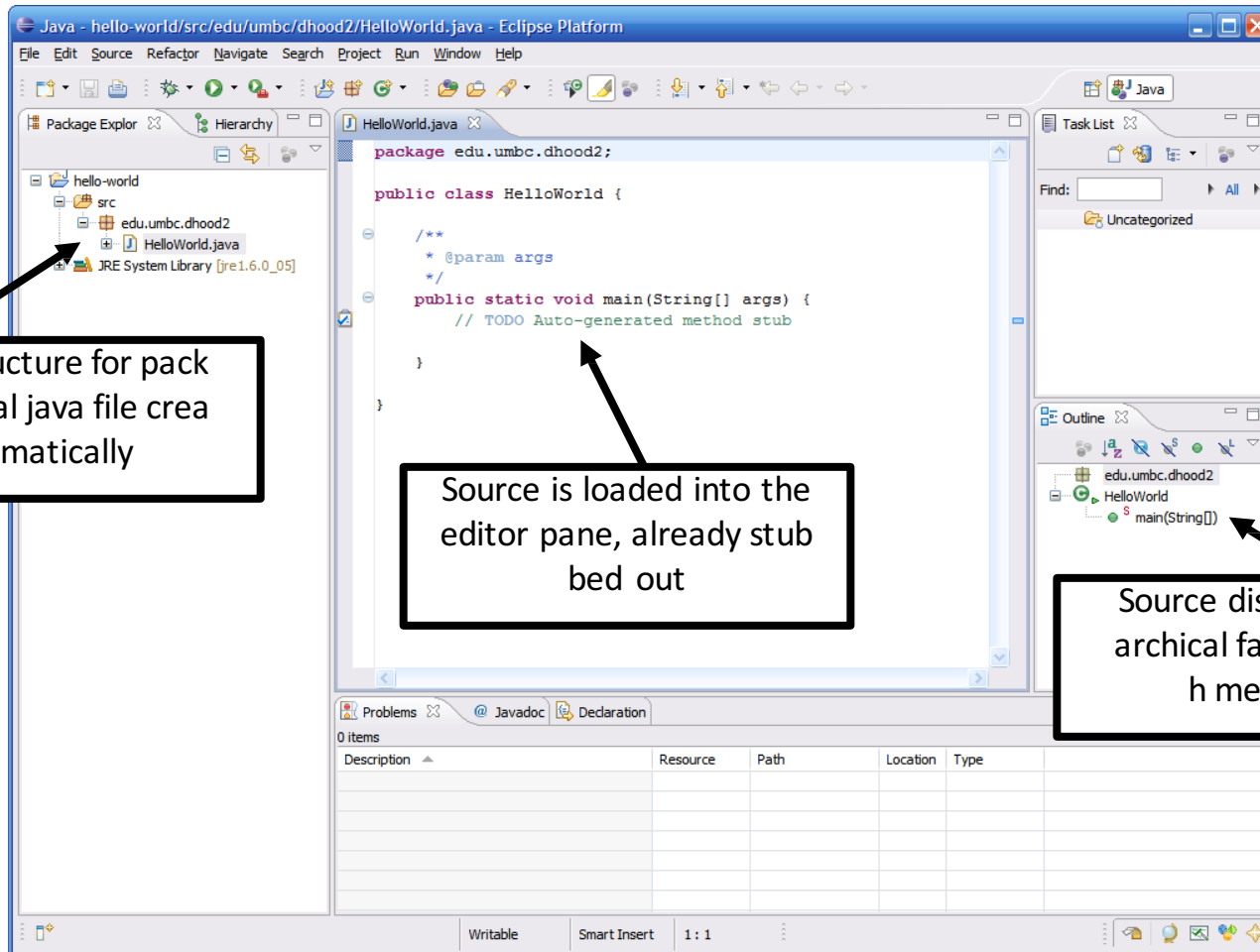
● Creating Class in eclipse (Contd)



- 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
 - Other cool features
- Fill in necessary information then click Finish to continue

• The Created Class

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



● 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

CSLAB

● Example Compilation Error

- This code contains a typo in the println statement...

The screenshot shows the Eclipse IDE with the following components and annotations:

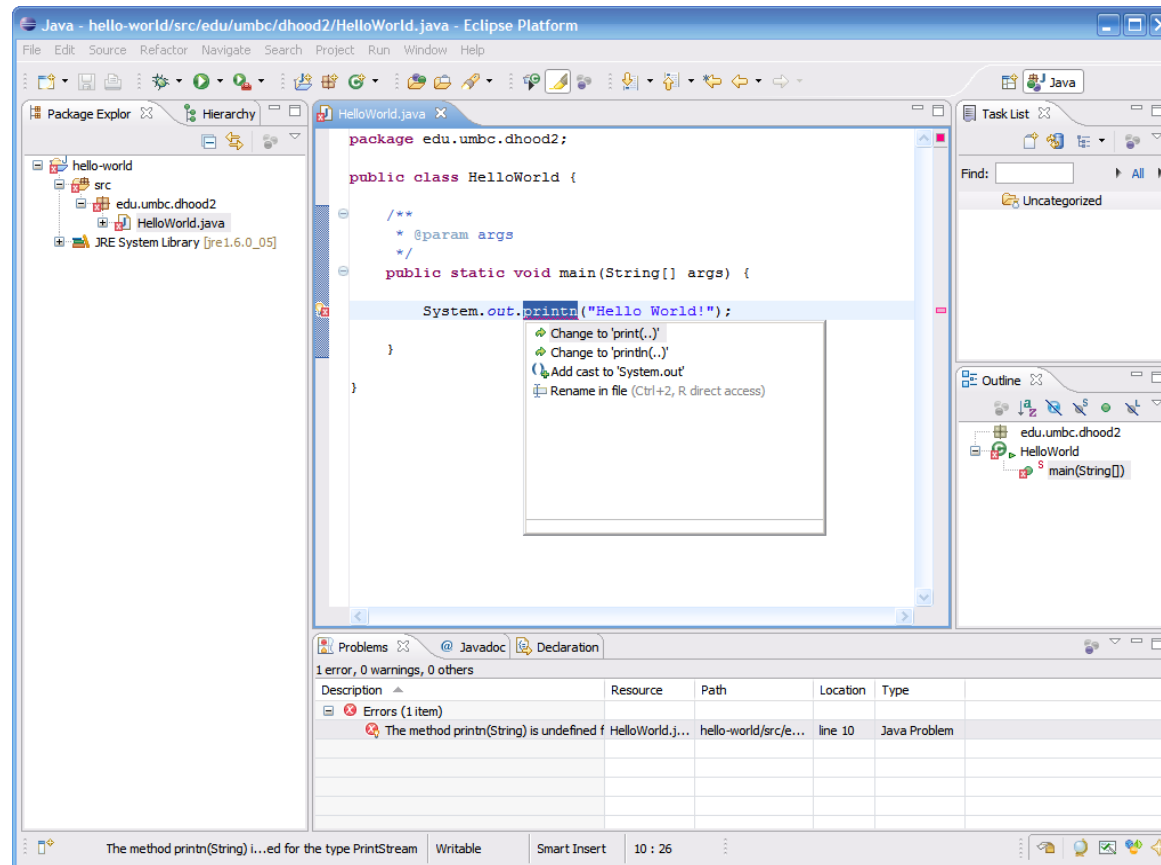
- Package Explorer:** Packages/Classes with errors are marked with a red X. (Annotation: "Packages/Classes with errors are marked with a red X")
- Code Editor:** The method `System.out.println("Hello World!");` is underlined with a red squiggly line. (Annotation: "Error underlined with red squiggly line (just like spelling errors in many word processors)")
- Problems Tab:** A tabular representation of all errors across all files of all open projects. (Annotation: "The Problems tab will contain a tabular representation of all errors across all files of all open projects")
- Task List:** Methods with errors are marked with a red X. (Annotation: "Methods with errors are marked with a red X")
- Outline:** Position in file is marked with a red line – 1 click allows you to jump to line with error. (Annotation: "Position in file is marked with a red line – 1 click allows you to jump to line with error")

Problems Tab Table:

Description	Resource	Path	Location	Type
1 error, 0 warnings, 0 others				
Errors (1 item)				
The method println(String) is undefined for HelloWorld	hello-world\src\edu\umbc\dhod2\HelloWorld.java	hello-world\src\edu\umbc\dhod2\HelloWorld.java	line 10	Java Problem

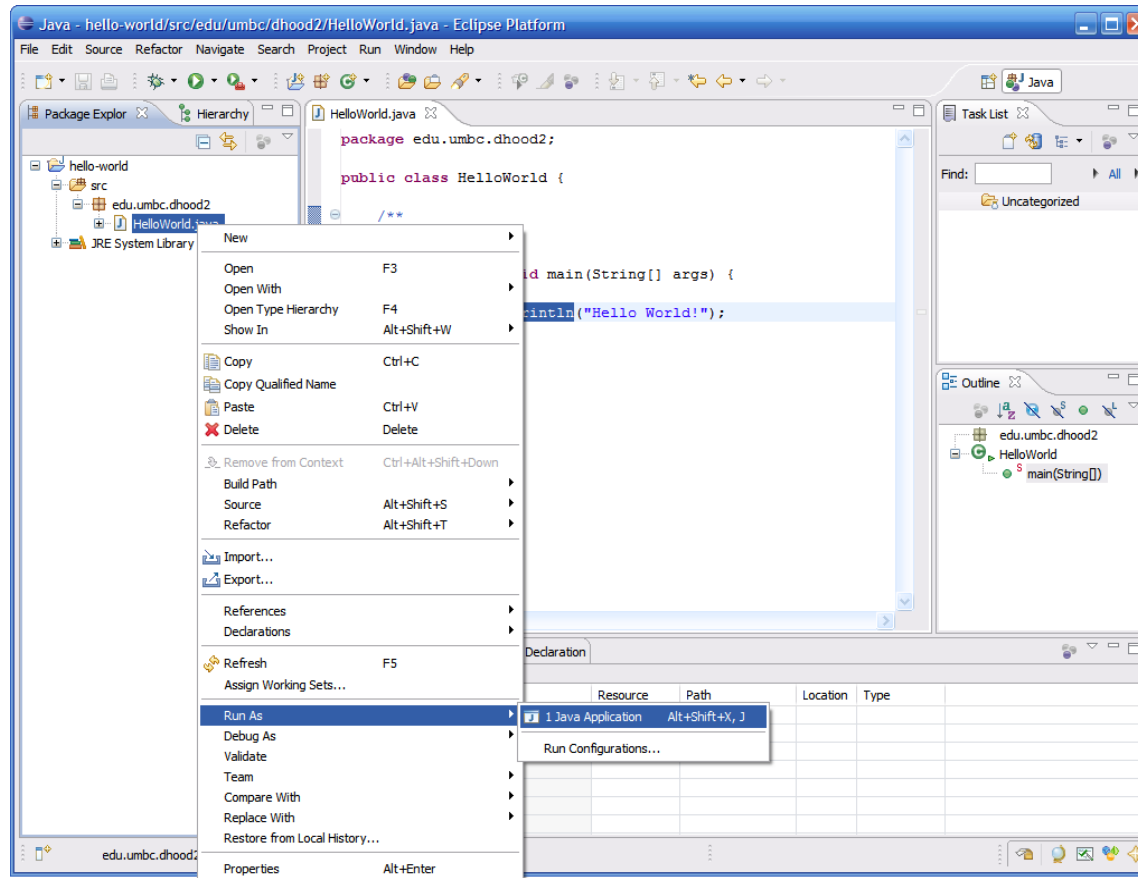
● Example Compilation Error (Contd)

- When clicking on the light bulb, Eclipse suggests changing `println` to either `print` or `println`



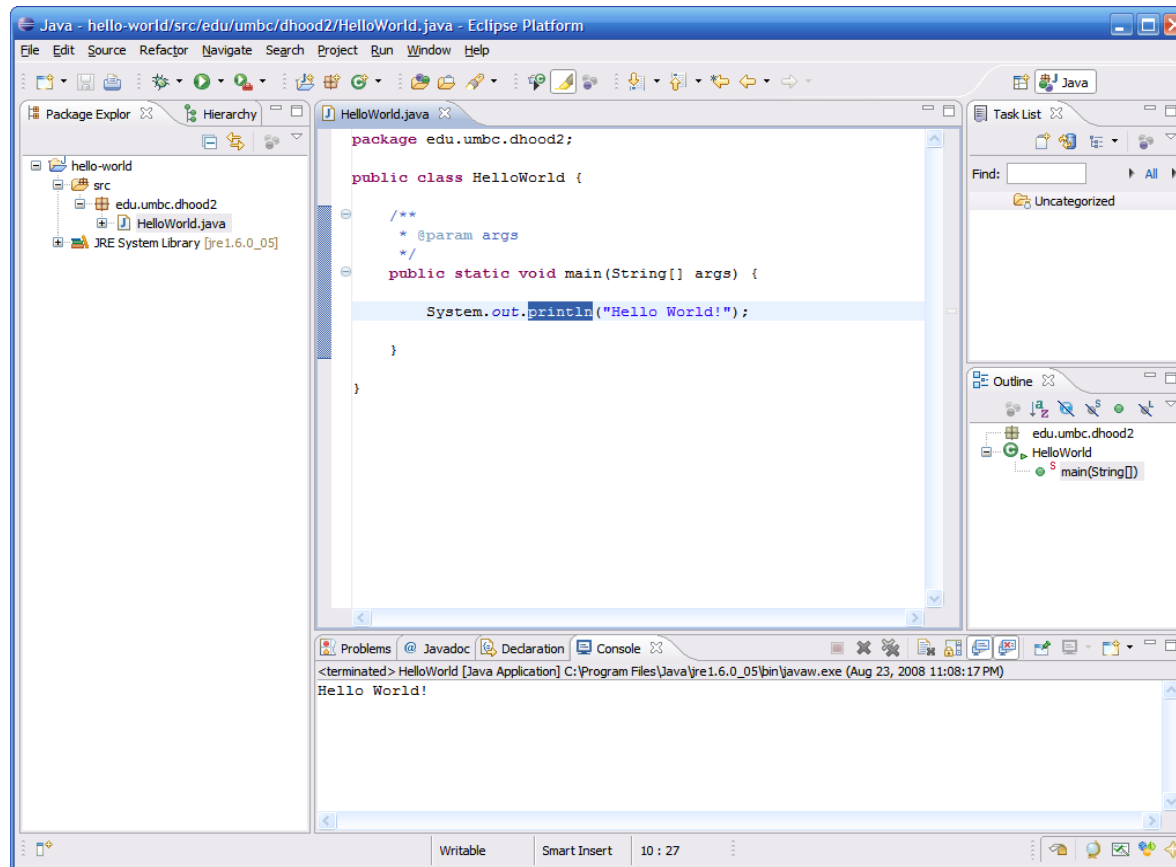
● Running Code

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



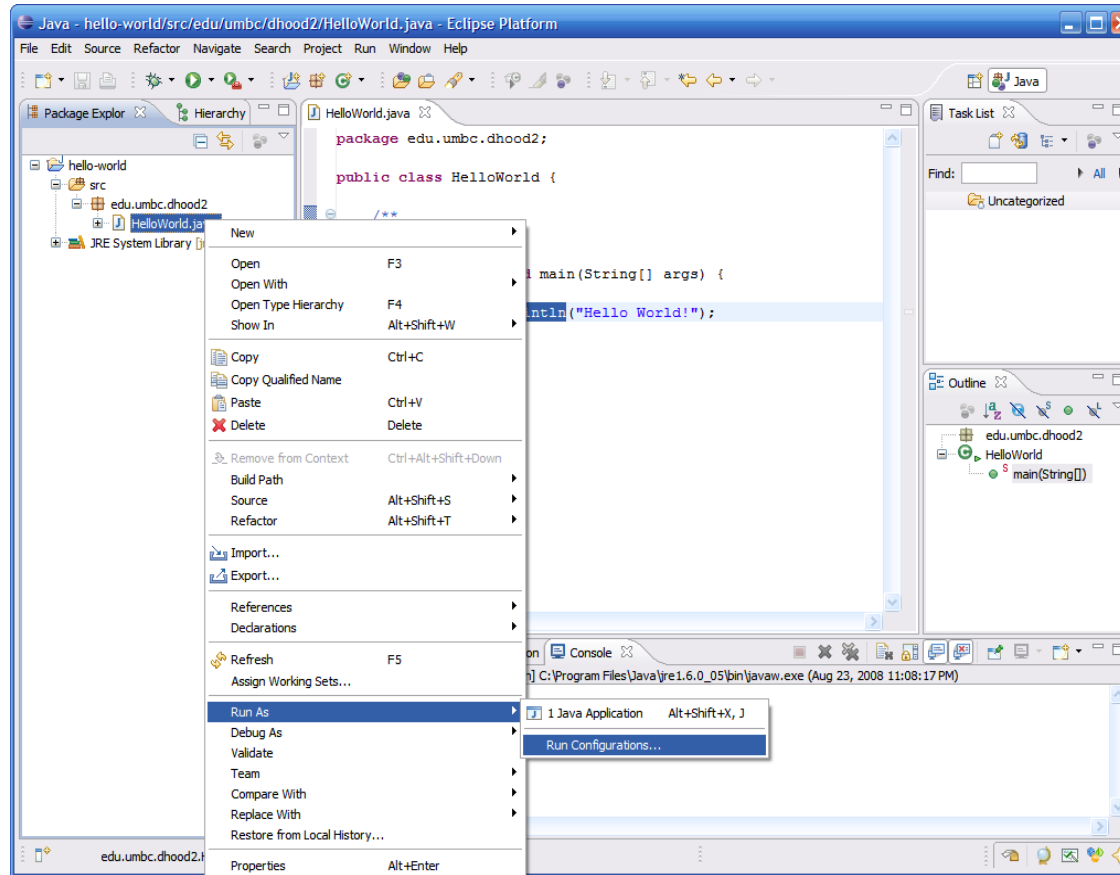
● Running Code (Contd)

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



● Run Configuration

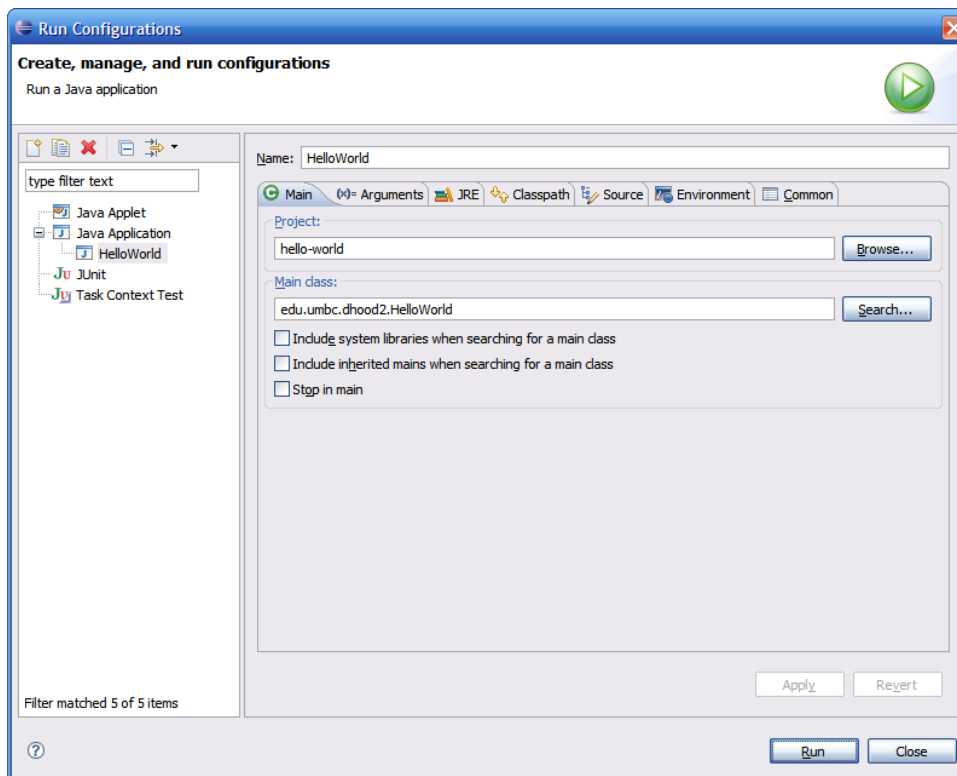
- Advanced options for executing a program can be found by right clicking the class then clicking Run As -> Run...



• Run Configuration (Contd)

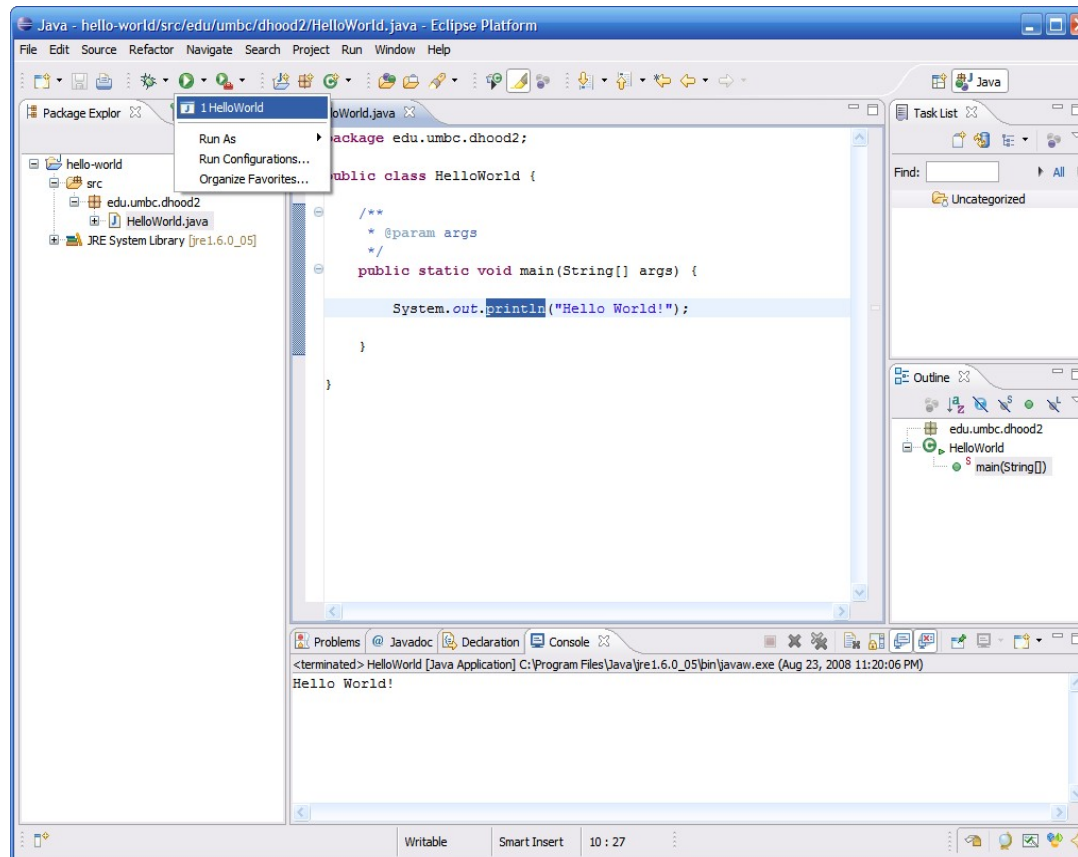
- Here you can change/add any of the following:

- JVM arguments
- Command line arguments
- Classpath settings
- Environment variables
- Which JVM to use



● 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



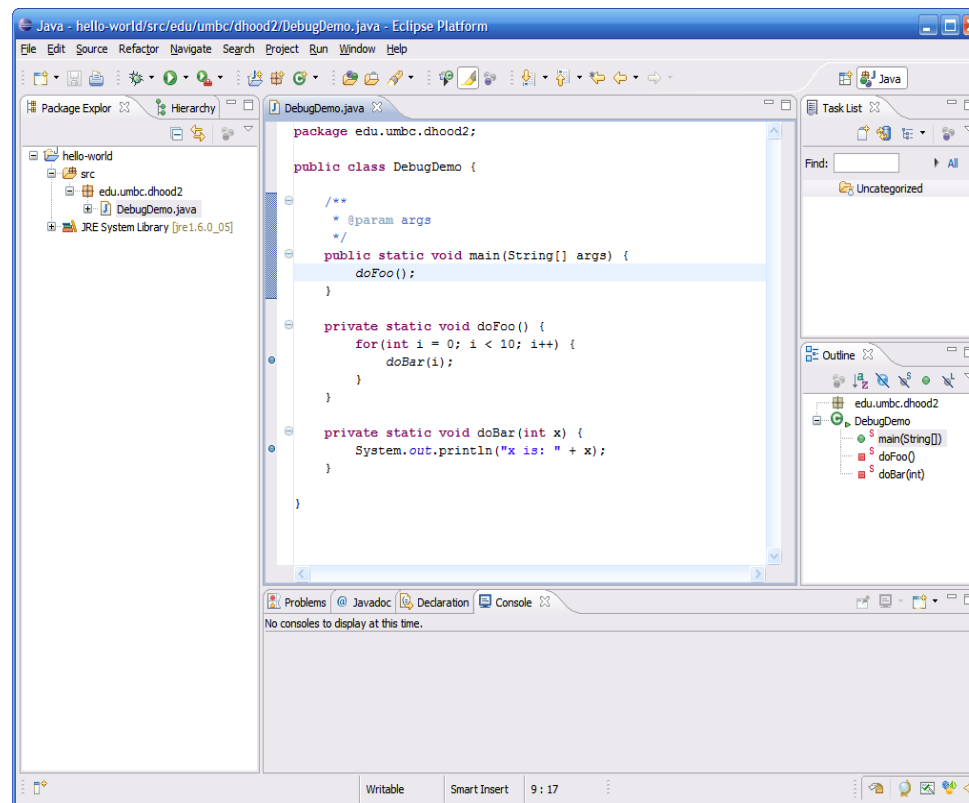
● Self-Test (1)

- 명령 프롬프트 창에 다음과 같은 명령어를 입력하여 결과를 출력할 것
 - 명령어: `javac -version`
 - 출력: `javac 1.8.0_161`
 - 명령어: `java -version`
 - 출력: `java version "1.8.0_161"`
- 프로그램 수행을 통해 이클립스 콘솔 창에 `Hello World!` 를 띄울 것

CSLAB

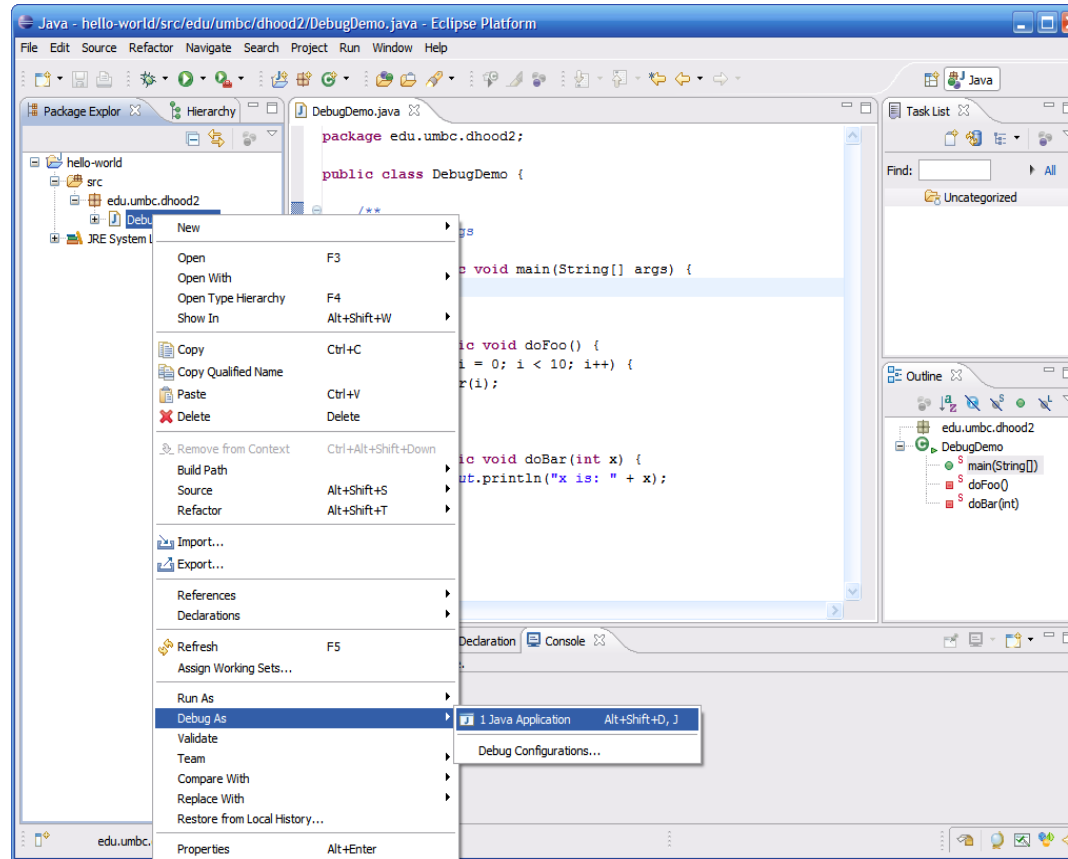
● Debugging Code

- Eclipse comes with a pretty good built-in debugger
- You can set break points in your code by double clicking in the left hand margin – break points are represented by these blue bubbles



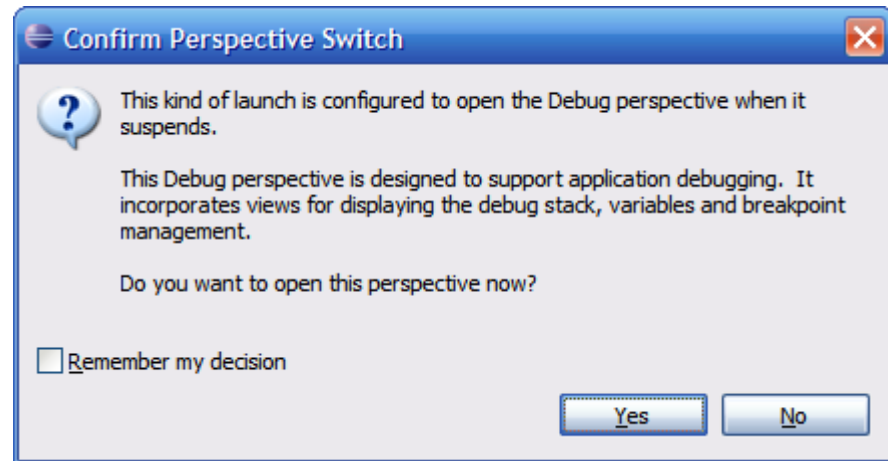
● Debugging Code (Contd)

- An easy way to enter debug mode is to right click on the class and select Debug As à Java Application



● Debugging Code (Contd)

- The first time you try to debug code you will be presented with the following dialog



- Eclipse is asking if you want to switch to a perspective that is more suited for debugging, click Yes
- Eclipse has many perspectives based on what you are doing (by default we get the Java perspective)

CSLAB

● Debug Perspective

Note new Debug perspective – click Java to return to normal

These buttons allow you to step through the code
Step Into: F5
Step Over: F6

List of breakpoints

Variables in scope are listed here along with their current values (by right clicking you can change values of variables as you program is running)

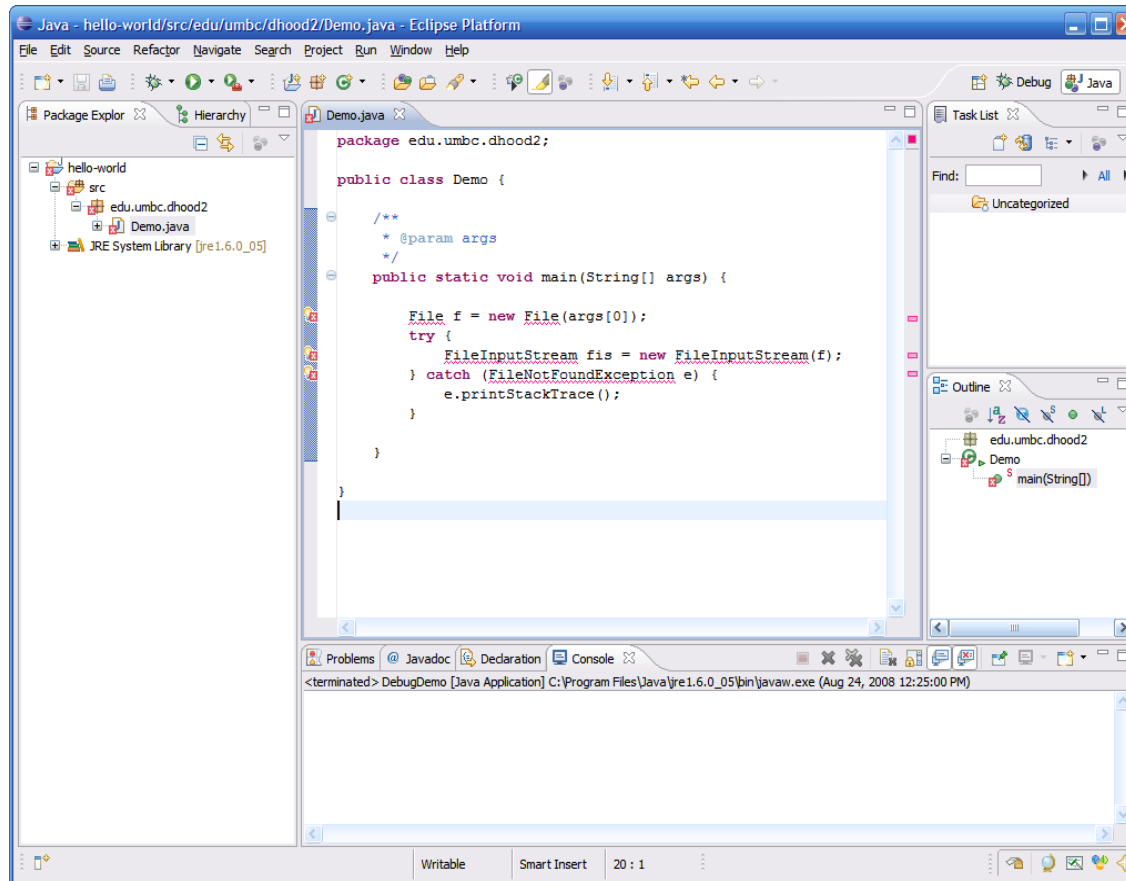
This pane shows the current line of code we broke on

Current high level location (class and method)

Output console, just like in normal run mode

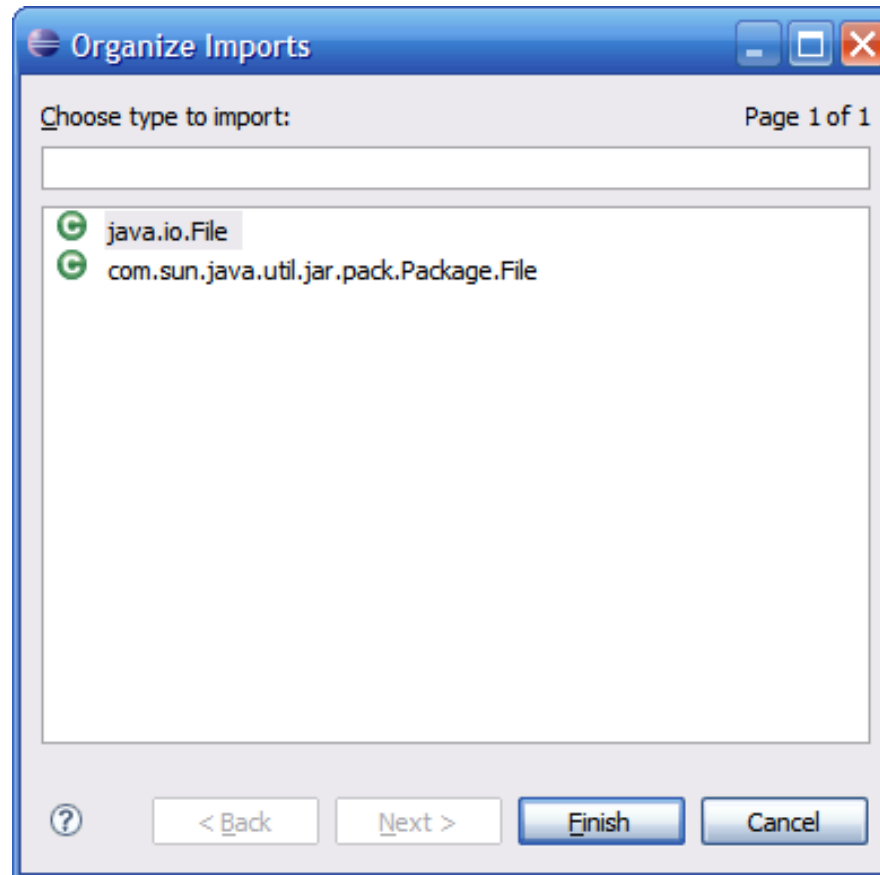
● Import Organization

- Eclipse can automatically include import statements for any classes you are using, just press **Control + Shift + o** (letter o)



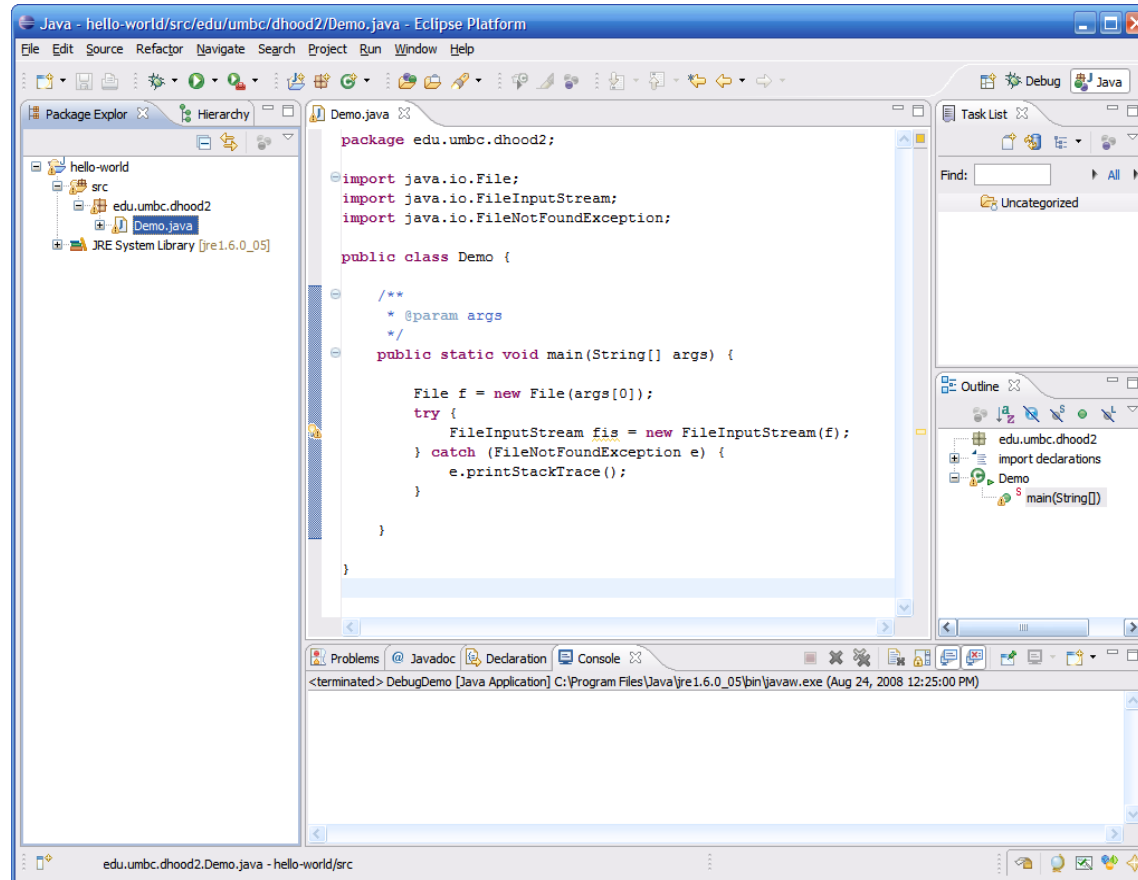
● Import Organization (Contd)

- If the class is ambiguous (more than one in the API) then it will ask you to select the correct one



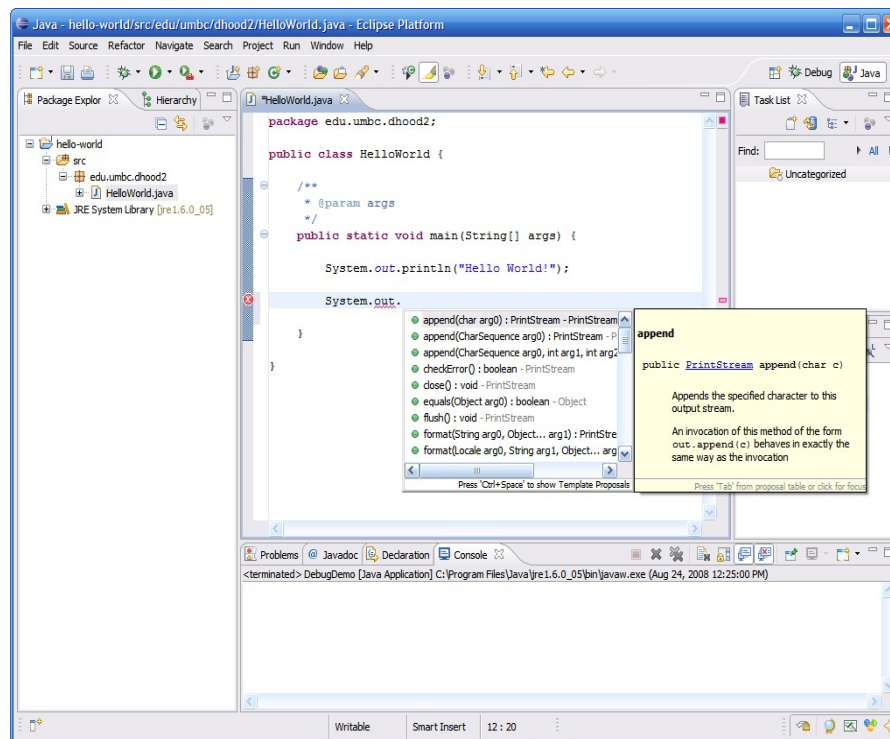
LAB

- **Import Organization (Contd)**
 - **Import statements automatically included and organized**
 - You can organize imports to clean them up at any time



Context Assist

- If you are typing and press a “.” character and pause a second, Eclipse will show you a list of all available methods for the class
 - Prevents having to browse javadocs to see what methods are available
 - Get context assist at any time by pressing Control + Space



LAB

● Self-Test(2)

- 아래 코드를 작성 후 디버깅을 통해 변수 i 값의 변화를 확인할 것

```
public class DebugProj {  
    public static void main(String[] args) {  
        for (int i=0; i<10; i++)  
            ● System.out.println(i);  
    }  
}
```

해당 line에 break point

- 콘솔 창에 0 ~ 9 까지의 값과 debugging perspective에 i 값이 9가 된 화면을 띄어 놓을 것

CSLAB