

Data Structure

Fall 2021

<http://smartlead.hallym.ac.kr>

Instructor: Jin Kim
010-6267-8189(033-248-2318)
jinkim@hallym.ac.kr

Office Hours:

별을 내 가슴에 (Starry Night)



실습 관련

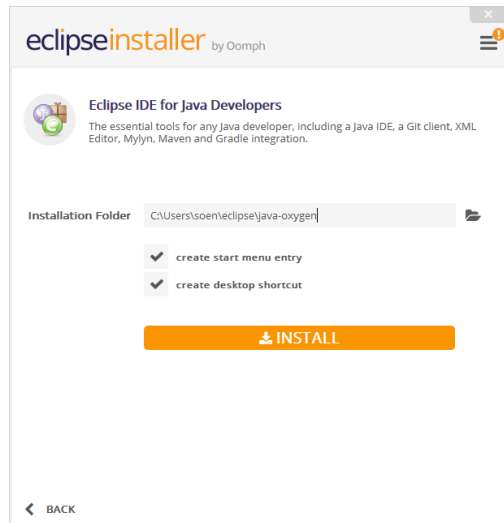
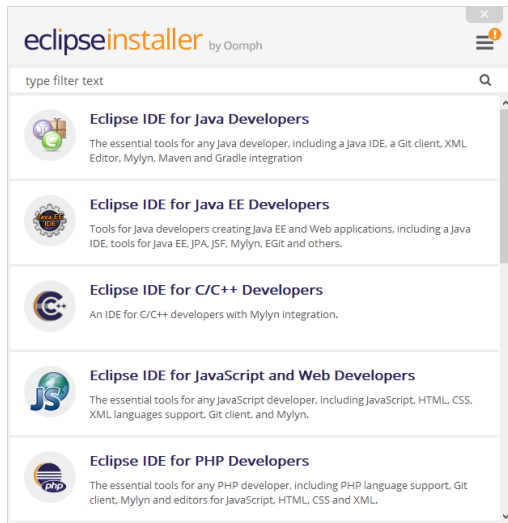
- ◆ 한 클래스는 한 파일이 일반적이지만, 자료구조 실습의 경우 **한 파일에 모든 클래스를 적도록 약속하자.**
- ◆ 파일을 업로드할 경우 ***.java 파일만을 업로드하라. 다른 파일들은 필요없다.**

Eclipse Editor

Eclipse 사용 요령 및 단축 키

이클립스

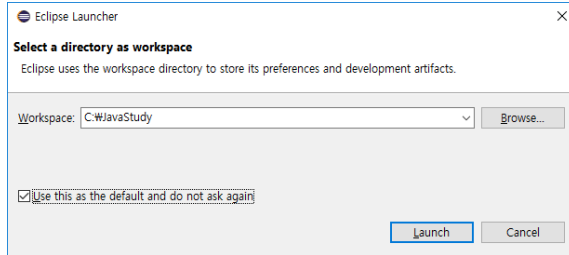
- 전통적인 자바 통합 개발 환경이다.
 - <http://www.eclipse.org>
- 최신 버전은 **SimRel(4.9.0)**이되 18년 12월 버전이 발표되었다.
- 설치 프로그램을 다운로드받아 실행하고 **Java Developers** 버전을 설치한다.



- 다운로드 설치가 번거로운 환경에서는 패키지를 미리 받아 압축만 풀어 주면 된다.

이클립스

- 최초 실행시 워크스페이스를 물어 오는데 실습 폴더로 지정한다.



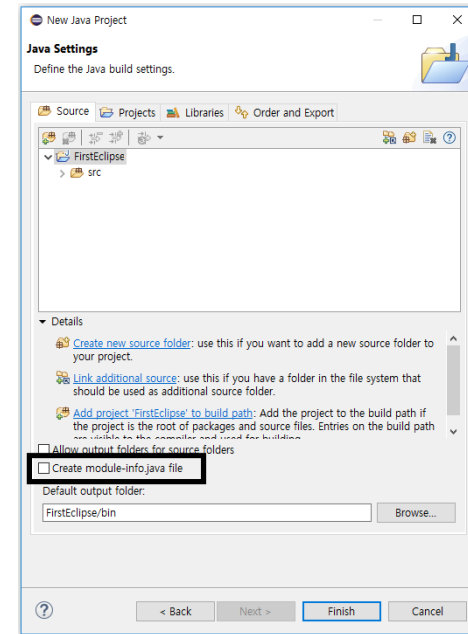
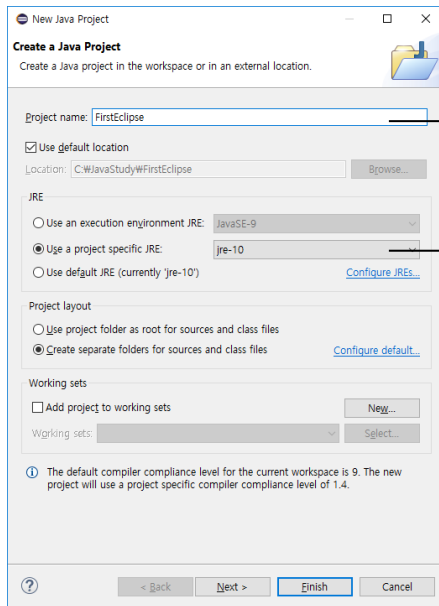
- 한글 사용을 위해 **Windows/Preferences/General/Workspace**에서 문자셋 옵션을 **UTF-8**로 변경한다. Default는 MS949로 되어있다.
- **Java/Compiler** 페이지의 레벨은 10 이상으로 맞춘다. 8 이상이면 무관하다.
- 글꼴이나 단축키는 취향에 따라 조정한다.
- **General/Appearance/Colors and Fonts** 페이지에서 Text font 항목의 폰트를 변경한다.
- 이클립스도 따로 배워야 할 정도로 부피가 크다.

배경화면 변경

- 이클립스 상단의 **Help-Eclipse Marketplace**를 선택한다.
- 검색창에 **color theme**을 검색한다. 원하는 **theme**를 선택하여 설치한다.
- 이클립스를 재시작한다.
- 재시작이 완료되면 **Window-Preferences**를 선택한다.
- **General - Appearance - Color Theme** 를 선택한 후 원하는 **theme**를 선택하여 적용한다.

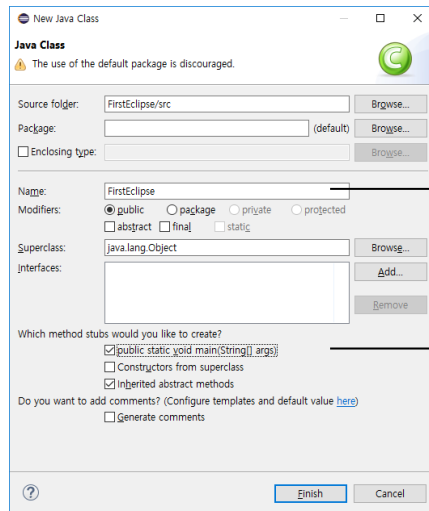
FirstEclipse

- **File/New/Java Project** 항목을 선택하고 프로젝트명을 **FirstEclipse**로 선택한다.
- **Next** 버튼을 눌러 **Create module-info.java file** 옵션을 해제한다. 빈 프로젝트가 생성된다.



FirstEclipse

- **File/New/Class** 를 선택하고 소스 파일명은 **FirstEclipse**로 입력한다.
- **public static void main(String[] args)** 옵션을 선택한다.



① 소스 파일 이름 입력

② main 옵션 선택

FirstEclipse

- 새로 만든 소스 파일에 코드를 입력한다.
 - `class FirstEclipse {`
 - `public static void main(String[] args) {`
 - `System.out.println("Java Example Program");`
 - `}`
 - `}`
- **Ctrl+S**를 눌러 저장하면 컴파일된다.
- **Ctrl+F11**을 눌러 실행하면 아래쪽 **Console**창에 실행 결과가 나타난다.
- 에러 발생시 **Problems**창에 발생 위치와 원인이 표시된다. 수정 후 다시 컴파일한다.
- 실행전 항상 저장 옵션을 선택해 두면 수정 후 바로 컴파일해 볼 수 있다.

Eclipse 화면

■ **Perspective(퍼스펙티브)-화면구성을 의미**

- 퍼스펙티브는 특정한 자원을 가지고 특정한 작업을 수행하기 위한 기능의 집합
- Java perspective는 자바 소스 파일을 편집할 때 필요한 뷰들을 제공
- 디버그 퍼스펙티브(Debug Perspective)는 자바 프로그램을 디버깅할 때 필요한 뷰들을 제공

Eclipse Perspective

Menu bar

Tool bar

Perspective
and
Fast View
bar

Resource
Navigator
view

Message
area

Text
editor

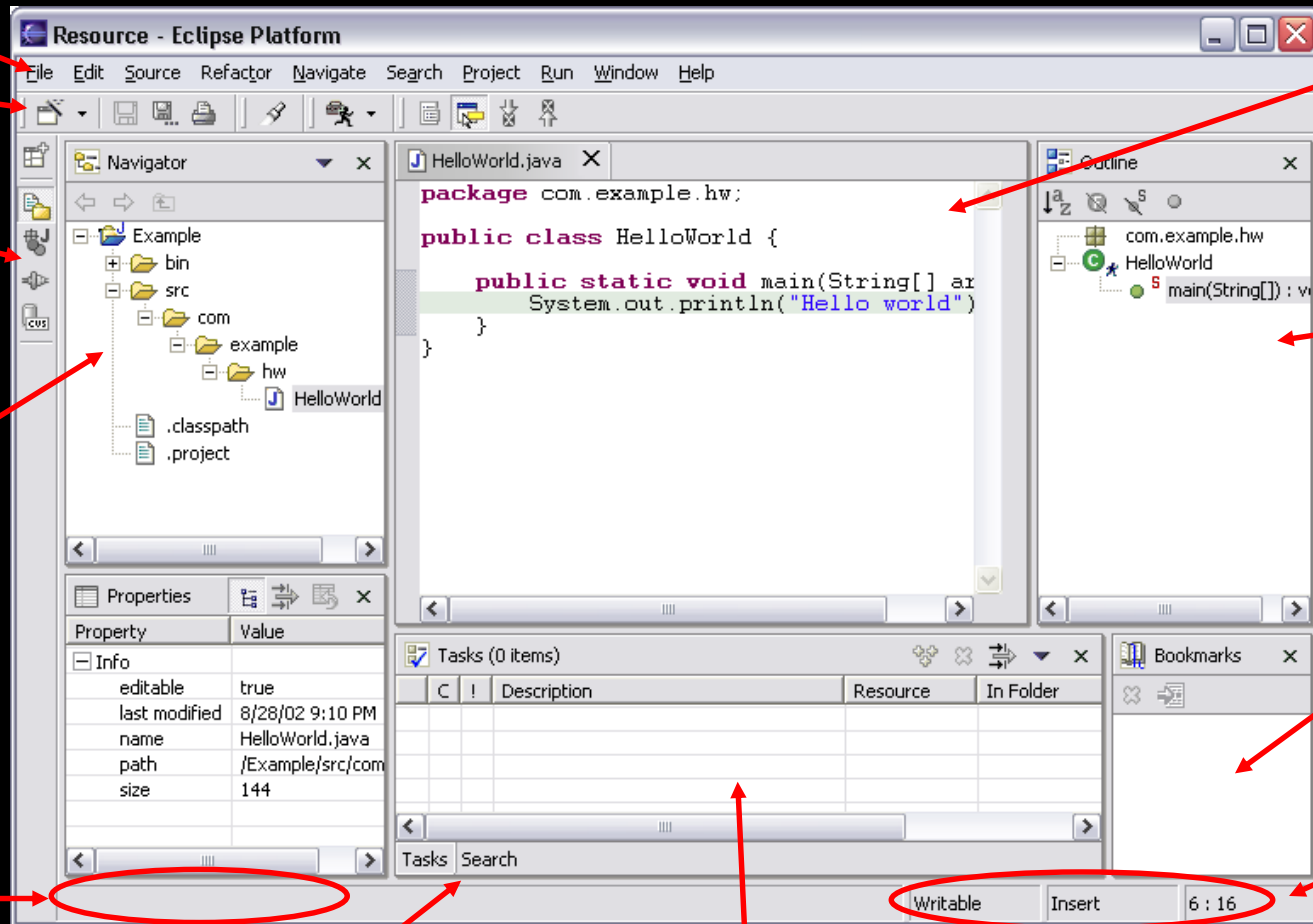
Outline
view

Bookmarks
view

Editor
Status
area

Stacked
views

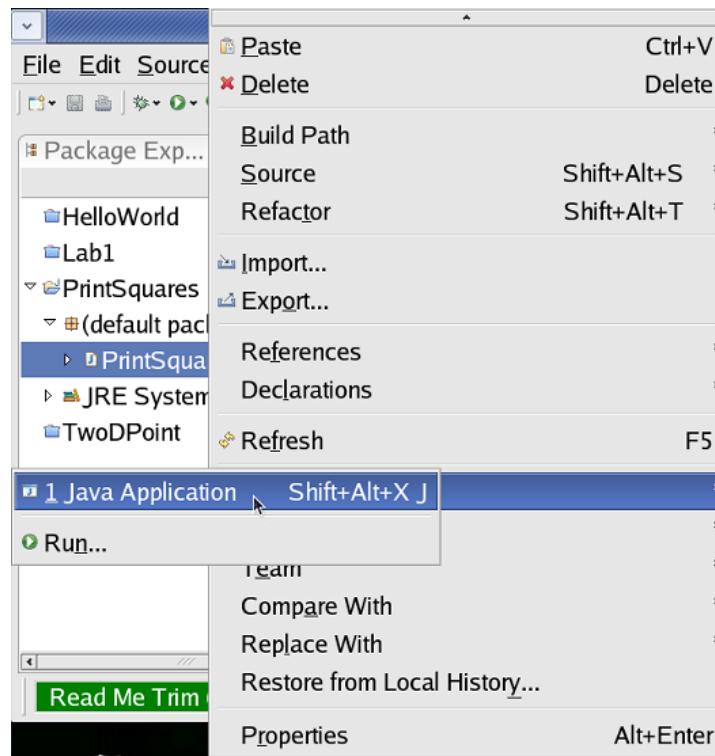
Tasks
view



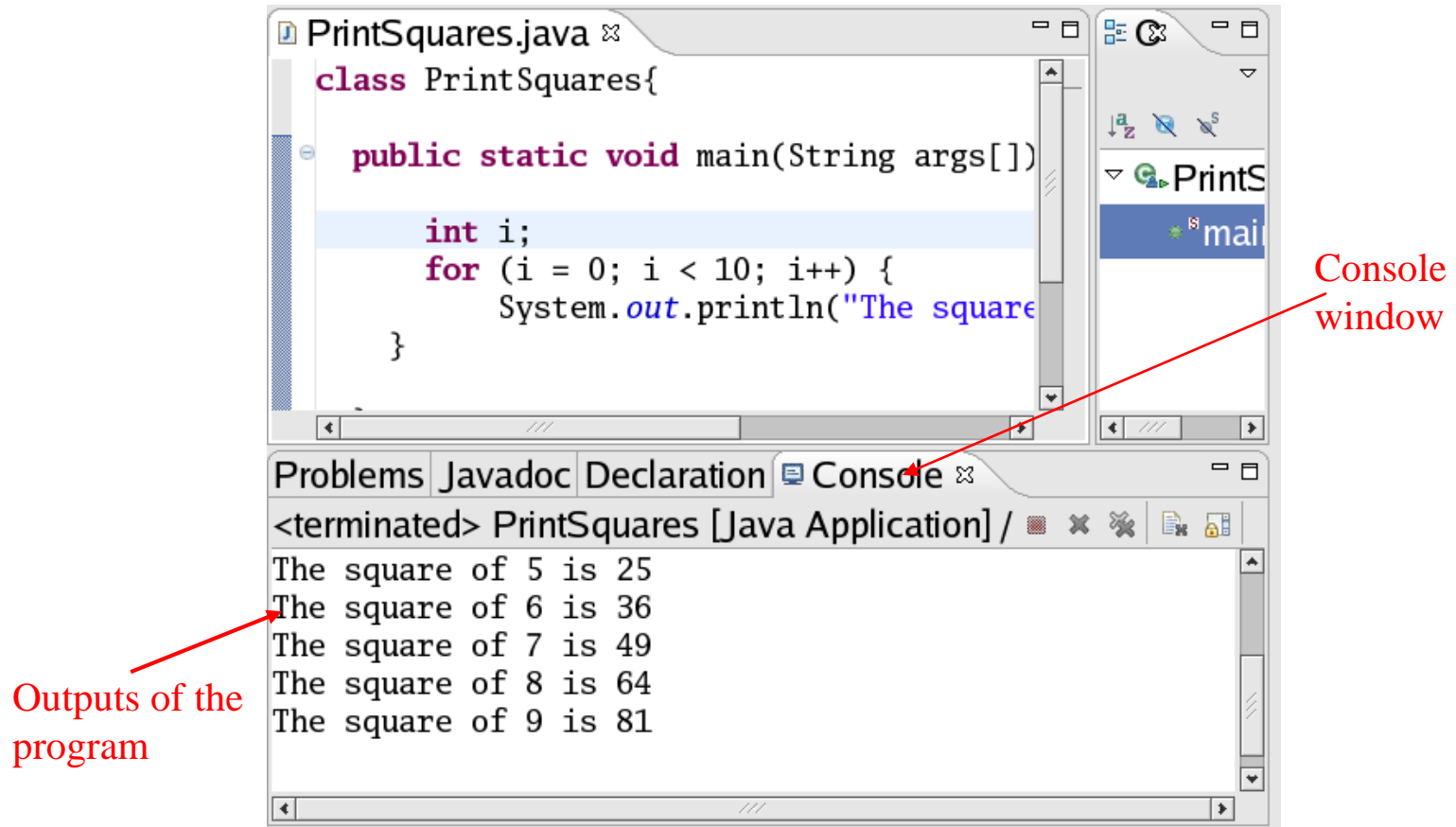
Run the program(프로그램실행)

오른쪽 단추->run as -> java application선택

1. Right click *PrintSquare.java* in *Workspace* window and select *Run As> Java Application*.

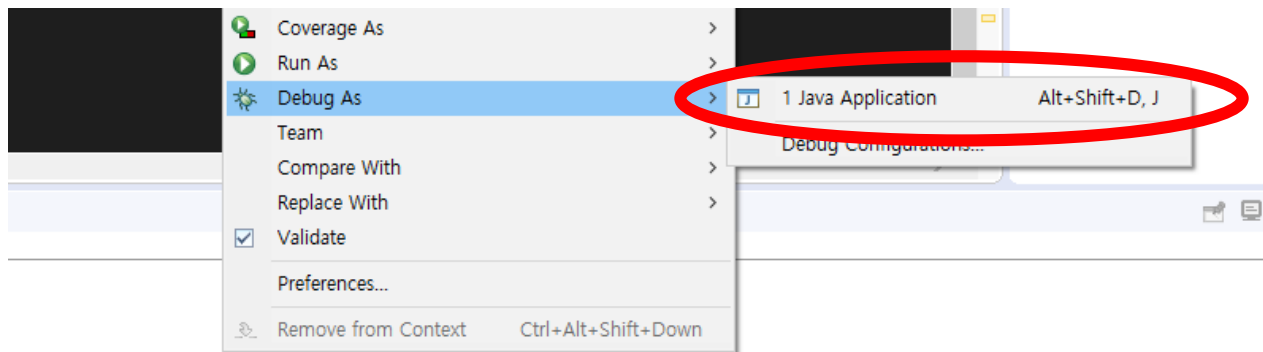


2. *Console* window shows the outputs of the program.



실행

- ◆ Ctrl + F11 : 바로 전에 실행했던 클래스 실행
- ◆ 혹은 오른쪽 단추-> Run as선택->Java application선택
- ◆ 가장 많은 자바 에러는 클래스이름과 파일이름이 일치하지 않을 때 발생. 이때 java application이 옵션으로 나타나지 않는다.

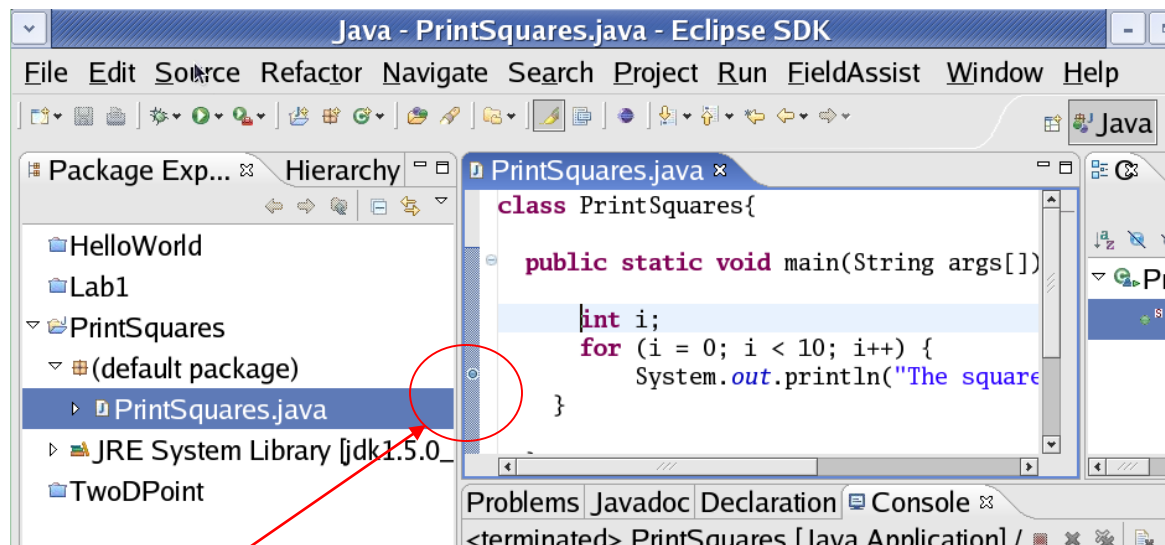


한글이 읽혀지지 않을때

- ◆ 한글은 UTF-8형식으로 읽을 수 있다.
- ◆ UTF-8으로 저장되지 않은 파일은 메모장에서 읽어들이고 후 UTF-8형식으로 다시 저장한다.
- ◆ 터미널에서 문제 : MinGW를 사용하여 C, C++컴파일후 실행할 때 한글이 정상출력안되면 폰트(글꼴)문제. 작업창에서
- ◆ >chcp 65001을 입력한다 혹은 C:\Windows\system32\cmd.exe의 속성을 찾아 글꼴을 레스터 글꼴로 바꾸면 된다

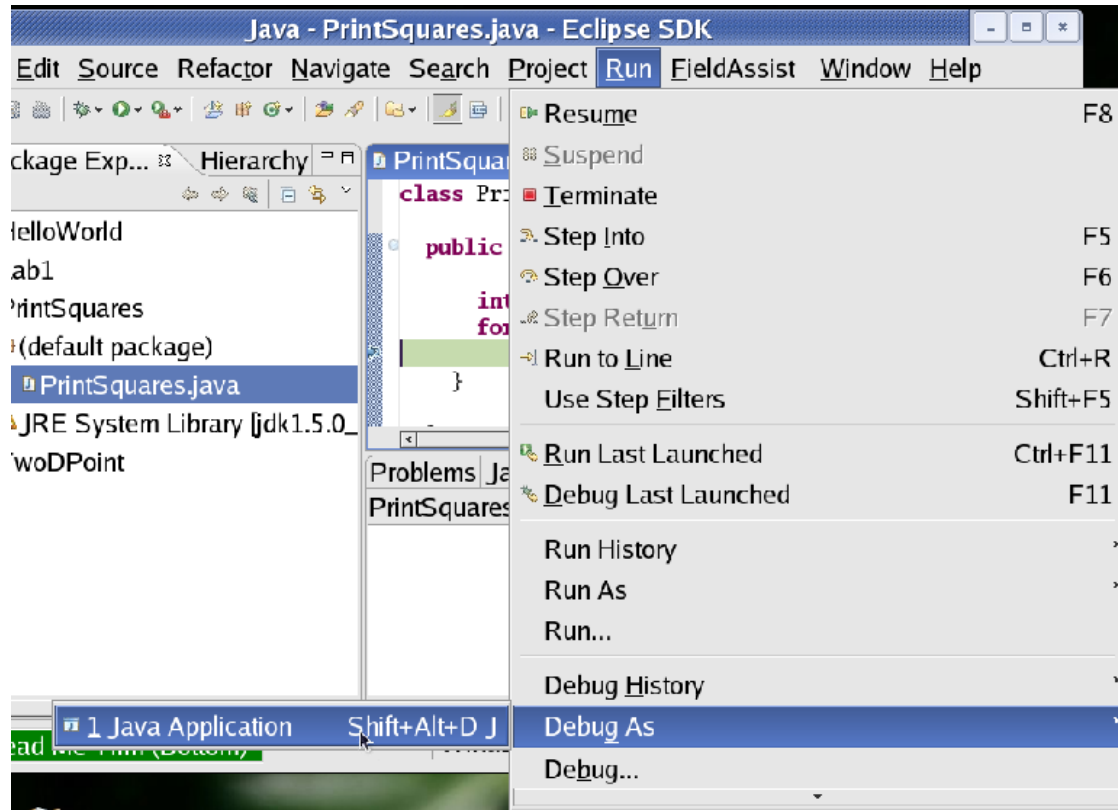
Debug a Program

1. Add breakpoints: double-click the gray bar on the left of *Edit* window. A blue dot indicates a breakpoint. To remove a breakpoint, double click the breakpoint.

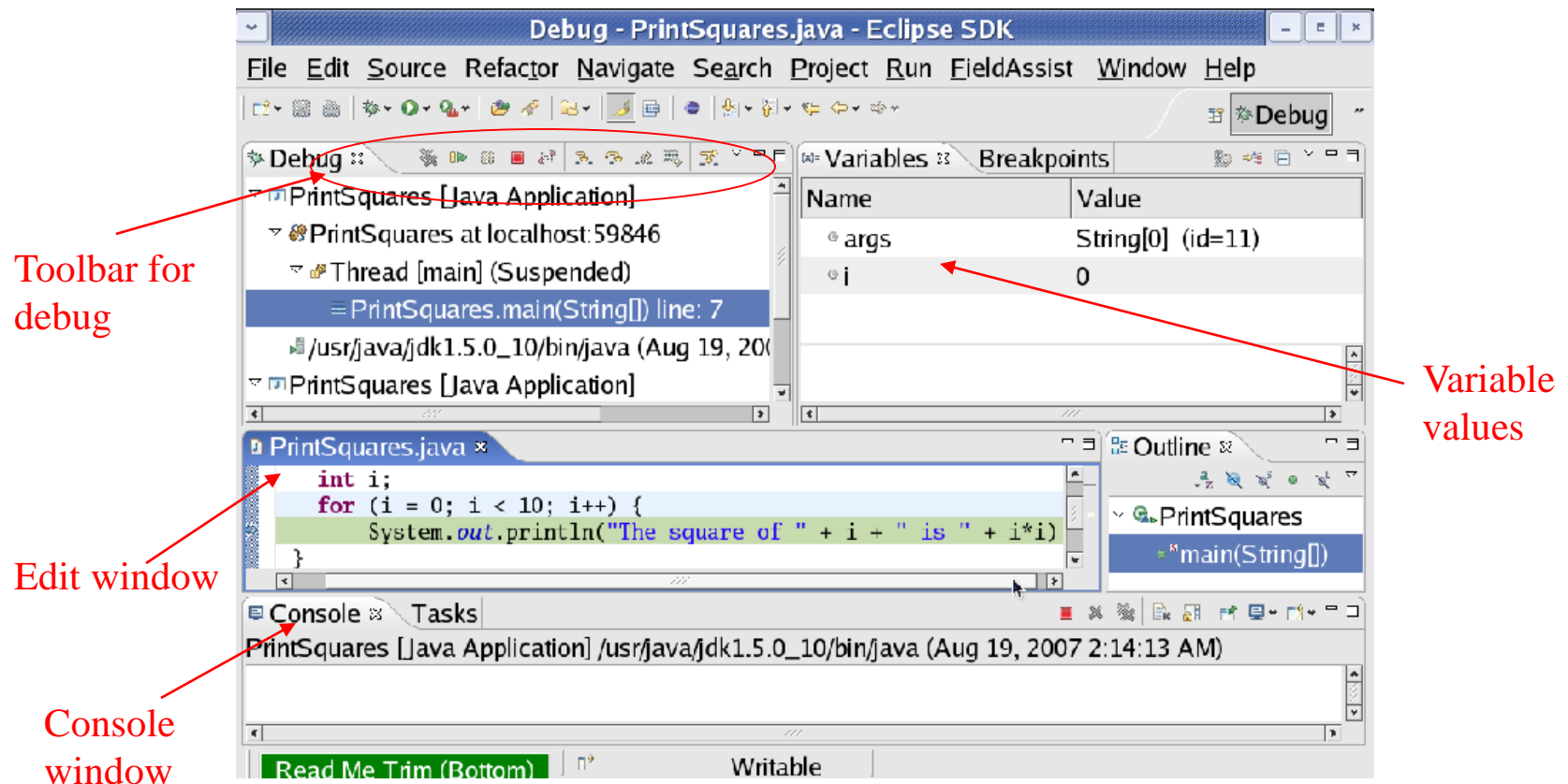


A break point

2. Select ***Run->Debug as...->Java Application*** to start the debugger.



3. Click **Yes** button in *Confirm Perspective Switch* window to switch Eclipse from *Java Perspective* to *Debug Perspective*.



4. Play with the debug commands and watch the change of variable values in *Variable* window and the outputs in *Console* window.

Resume resume the execution of a paused program.

Suspend temporarily pause the execution of a program.

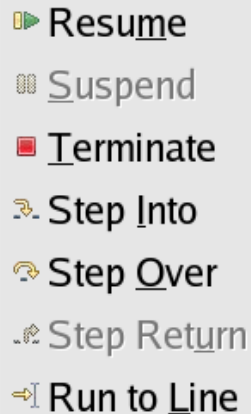
Terminate end the current debug session.

Step Into execute a single statement or step into a method.

Step Over execute a single statement. If the statement contains a call to a method, the entire method is executed without stepping into the method.

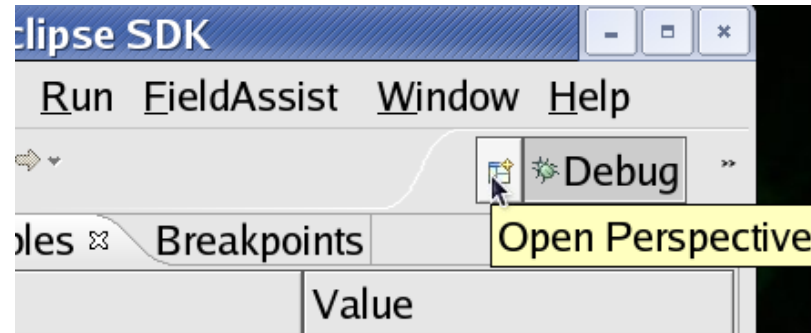
Step Return execute all the statements in the current method and returns to its caller.

Run to Line runs the program, starting from the current execution point, and pauses at a breakpoint.

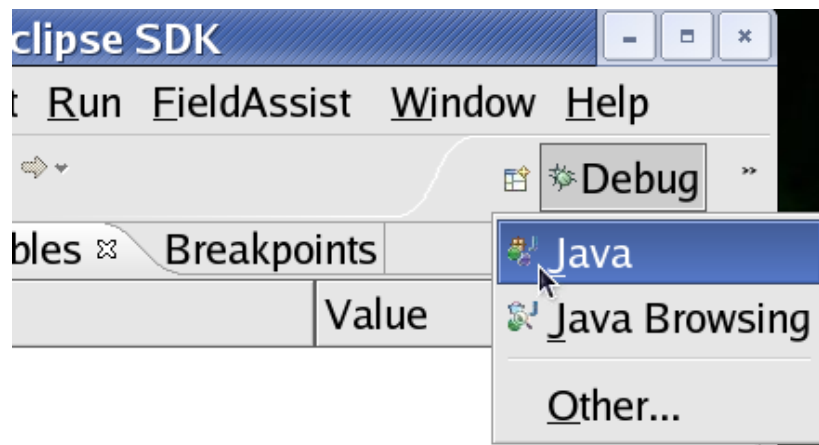


5. Switch Eclipse from *Debug Perspective* back to *Java Perspective*.

- Click *Open Perspective* button.



- Then click *Java*.

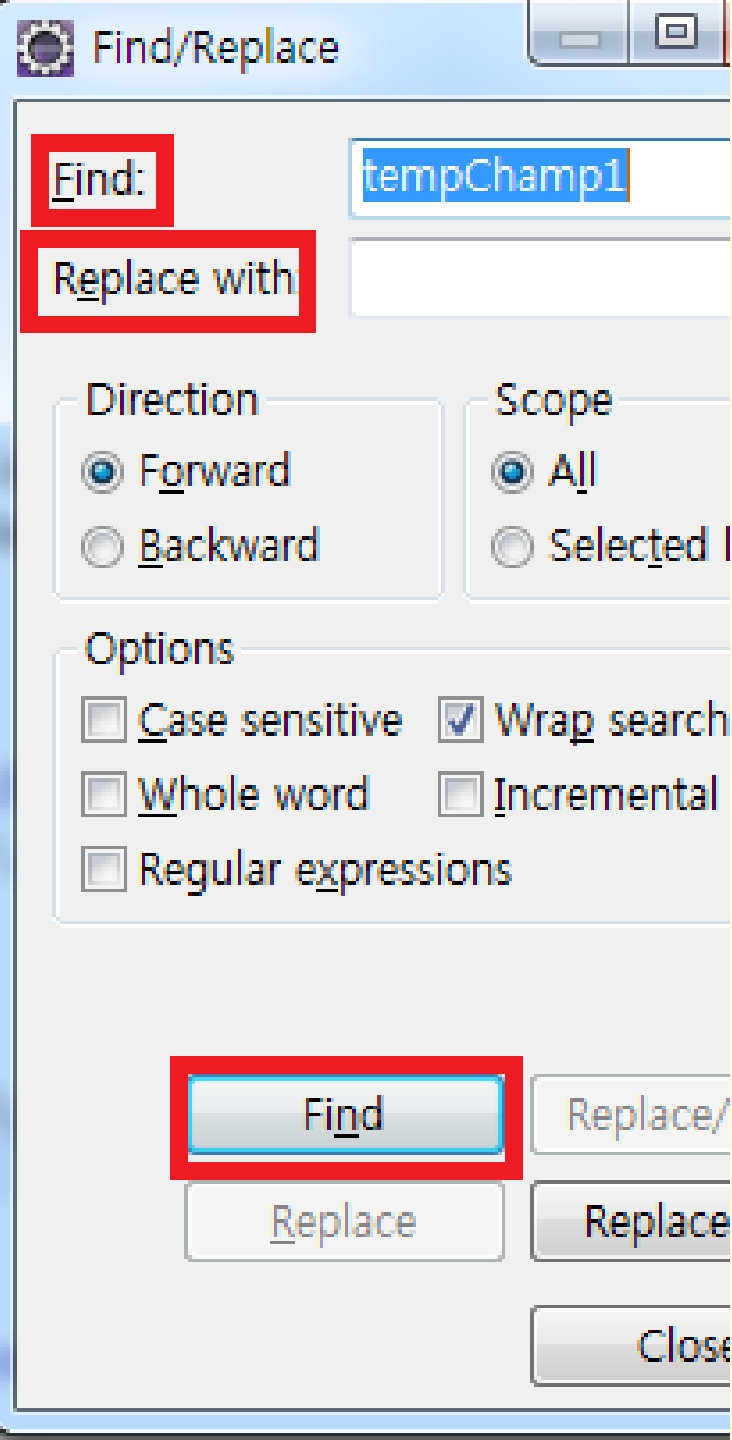


소스 네비게이션

- ◆ Ctrl + 마우스커서(혹은 F3) : 클래스나 메소드 혹은 멤버를 상세하게 검색하고자 할 때
- ◆ Alt + ->, Alt + <- : 이후, 이전
- ◆ Ctrl + o : 해당 소스의 메소드 리스트를 확인하려 할 때
- ◆ F4 : 클래스명을 선택하고 누르면 해당 클래스의 Hierarchy 를 볼 수 있다.

문자열 찾기

1. Ctrl + k : 찾고자 하는 문자열을 블록으로 설정한 후 키를 계속 누른다.
2. Ctrl + Shift + k : 역으로 찾고자 하는 문자열을 찾아감.
3. Ctrl + j : 입력하면서 찾을 수 있음.
4. Ctrl + Shift + j : 입력하면서 거꾸로 찾아갈 수 있음.
5. Ctrl + f : 기본적으로 찾기
6. Ctrl + shift + g : 커서를 메소드 위에 놓고 누르면 해당 메소드를 사용하는 곳을 찾아준다. 마우스 오른쪽-Reference - 에도 있음.



문자열 바꾸기

- 1. Ctrl + k : 찾고자 하는 문자열을 블록으로 설정한 후 키를 누른다.
- 2. 오른쪽 마우스를 클릭하고 Refact를 선택
- 혹은
 - Ct기+F 사용하여 문자열을 검색하고
찾아바꾸기(replace)하면 한번에 문자열 치환

원본 편집1

1. Ctrl + Space : 입력 보조장치(Content Assistance) 강제 호출 => 입력하는 도중엔 언제든지 강제 호출 가능하다.
2. F2 : 컴파일 에러의 빨간줄에 커서를 가져다가 이 키를 누르면 에러의 원인에 대한 힌트를 제공한다.
3. Ctrl + l : 원하는 소스 라인으로 이동
로컬 히스토리 기능을 이용하면 이전에 편집했던 내용으로 변환이 가능하다.
4. Ctrl + Shift + Space : 메소드의 가로안에 커서를 놓고 이 키를 누르면 파라미터 타입 힌트를 볼 수 있다.
5. 한줄 삭제 CTRL + D
6. 파일 닫기 : CTRL+W
7. 들여쓰기 자동 수정. (3.0 NEW) : CTRL+I
8. 블록 주석(/*..*/) 추가.(3.0 NEW): CTRL+SHIFT+/
8.1 Ctrl + / 해주면 여러줄이 한꺼번에 주석처리됨. 주석 해제하려면 반대로 하면 됨.
9. 위(아래)줄과 바꾸기 : ALT+UP(DOWN)
10. 블록 선택하기. : ALT+SHIFT+방향키

원본 편집2

- 11. 메소드의 파라미터 목록 보기. : CTRL+SHIFT+SPACE
- 12. 자동으로 import 하기 : CTRL+SHIFT+O
- 13. 열린 파일 모두 닫기 : CTRL + SHIFT + F4
- 14. 블록 주석 제거 : CTRL+SHIFT+
- 15. 전체화면 토글 : CTRL+M
- 16. 한줄(블럭) 복사 : Ctrl + Alt + 위(아래)
- 17. 다음 annotation(에러, 워닝, 북마크 가능)으로 점프 : Ctrl + , or .
- 18. 쿼 픽스 : Ctrl + 1
- 19. 메소드 정의부로 이동 : F3
- 20. 하이어라키 팝업 창 띄우기(인터페이스 구현 클래스간 이동시 편리) : Ctrl + T
- 21. 메소드나 필드 이동하기 CTRL + O
- 22. ULTRAEDIT나 EDITPLUS 의 CTRL+TAB 과 같은 기능. : CTRL+F6

원본 편집3

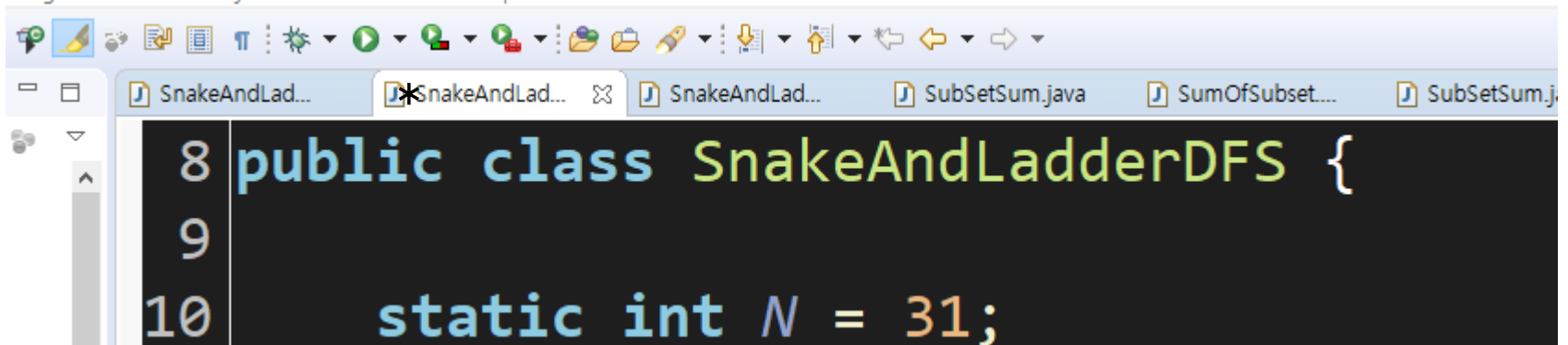
Source -> format 을 선택하면 형식에 맞추어 준다.

클래스 이름 변경 : 클래스 이름위에 오른쪽 마우스 선택후 -> Refact 선택

폰트 변경 : Window-> Preferences->Appearance-> Colors and Fonts->Java->Java Editor Text Font

혹은 Ct rl + (+ / -) 키

파일 탭의 *파일이름 : 파일이 저장되어 있지 않은 상태.



템플릿 사용

1. `sysout` 입력한 후 `Ctrl + Space` 하면 `System.out.println();` 으로 바뀐다.
2. `try` 입력한 후 `Ctrl + Space` 하면 `try-catch` 문이 완성된다.
3. `for` 입력한 후 `Ctrl + Space` 하면 여러가지 `for` 문을 완성할 수 있다.
4. 템플릿을 수정하거나 추가하려면 환경설정/자바/편집기/템플리트 에서 할 수 있다.

주석 처리

Ctrl + / 해주면 여러줄이 한꺼번에 주석처리됨. 주석 해제하려면 반대로 하면 됨.

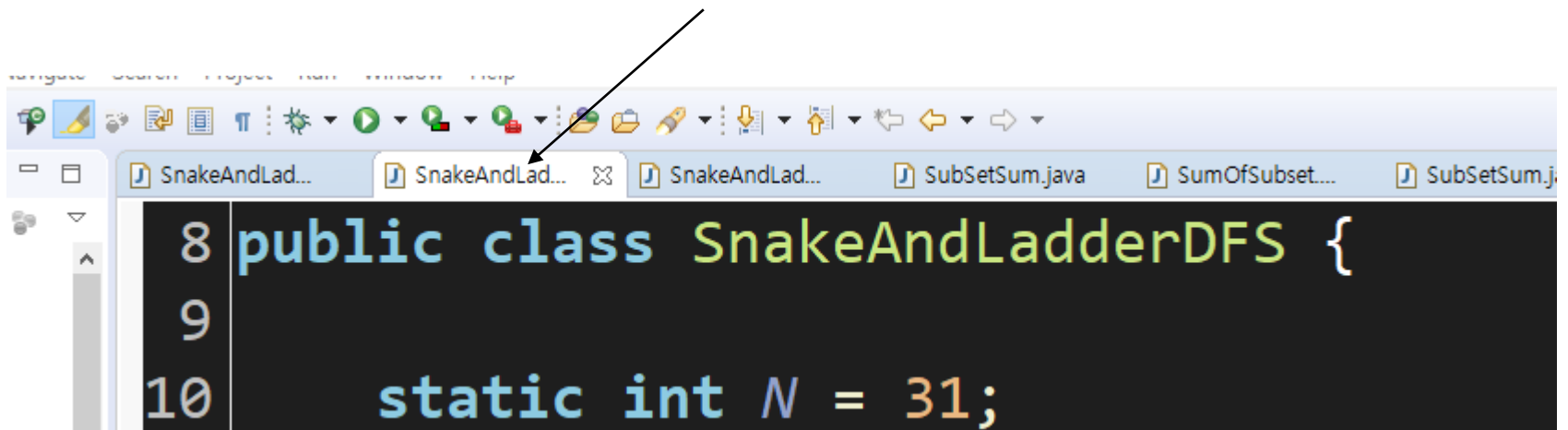
이클립스 단축키

Eclipse의 모든 단축키는 **Window >> Preferences >> Workbench >> Keys** 메뉴에서 확인 및 변경이 가능하다. 그 동안 다른 툴에서 자신의 손에 익숙한 단축키들이 있다면 이 메뉴에서 단축키들을 변경하는 것이 가능하다.

화면분할

- ◆ 파일이름이 표시된 탭을 드래그하면, 여러 파일을 한 화면에서 볼 수 있다. 즉 화면분할.

탭에 마우스 왼쪽 단추를 누른 상태에서 이리저리 drag해보라



이클립스 레이아웃 초기화

화면의 구성이 복잡해져 초기화하려면

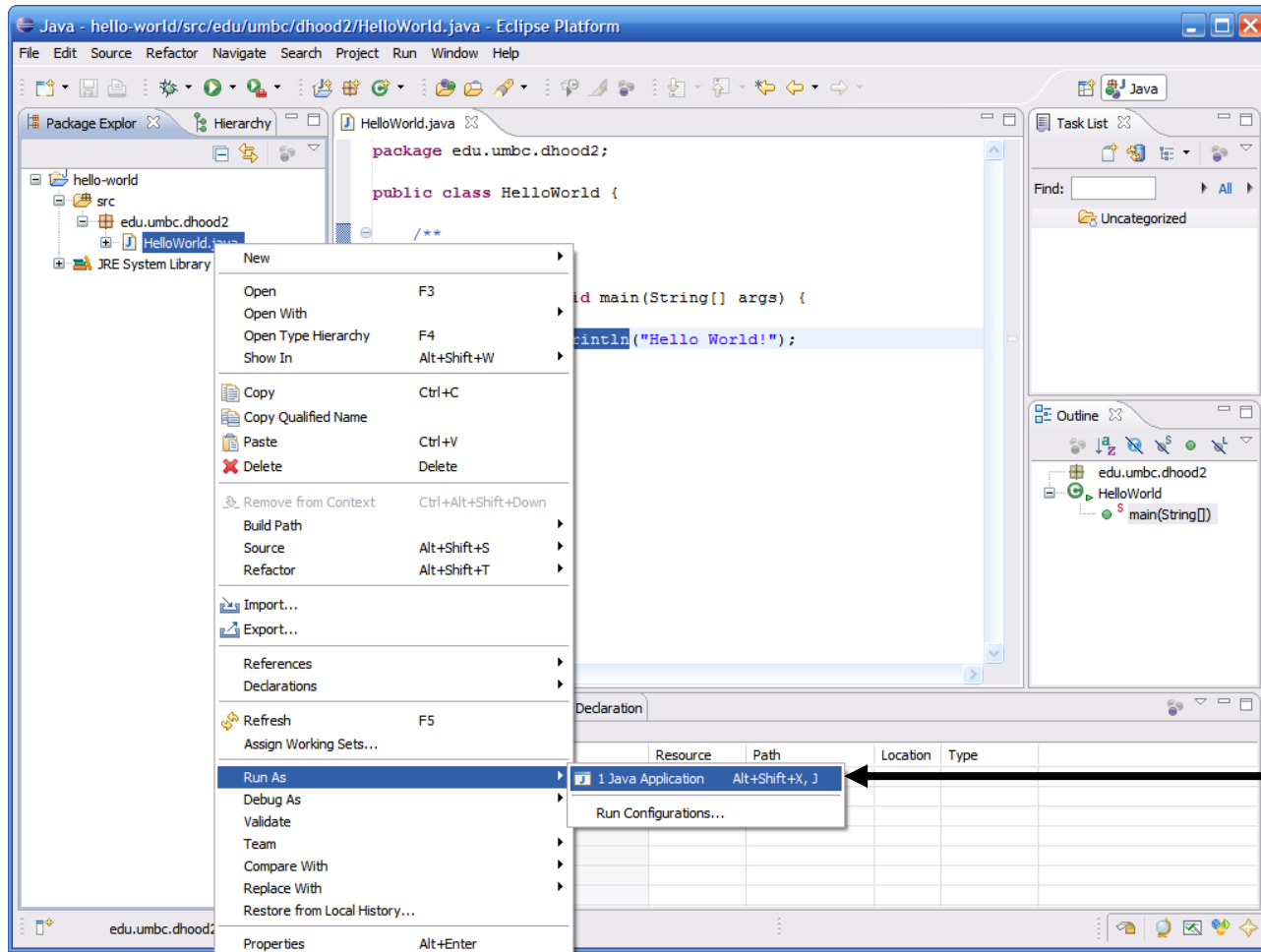
Eclipse의 화면을 초기화하려면 **Window >> Perspective >> Reset Perspective**

이클립스 테마변경 방법

Eclipse의 화면을 초기화하려면 **Window >> Preferences >> Appearance >> Theme**
원하는 테마를 선택하고, Apply를 선택, OK 클릭

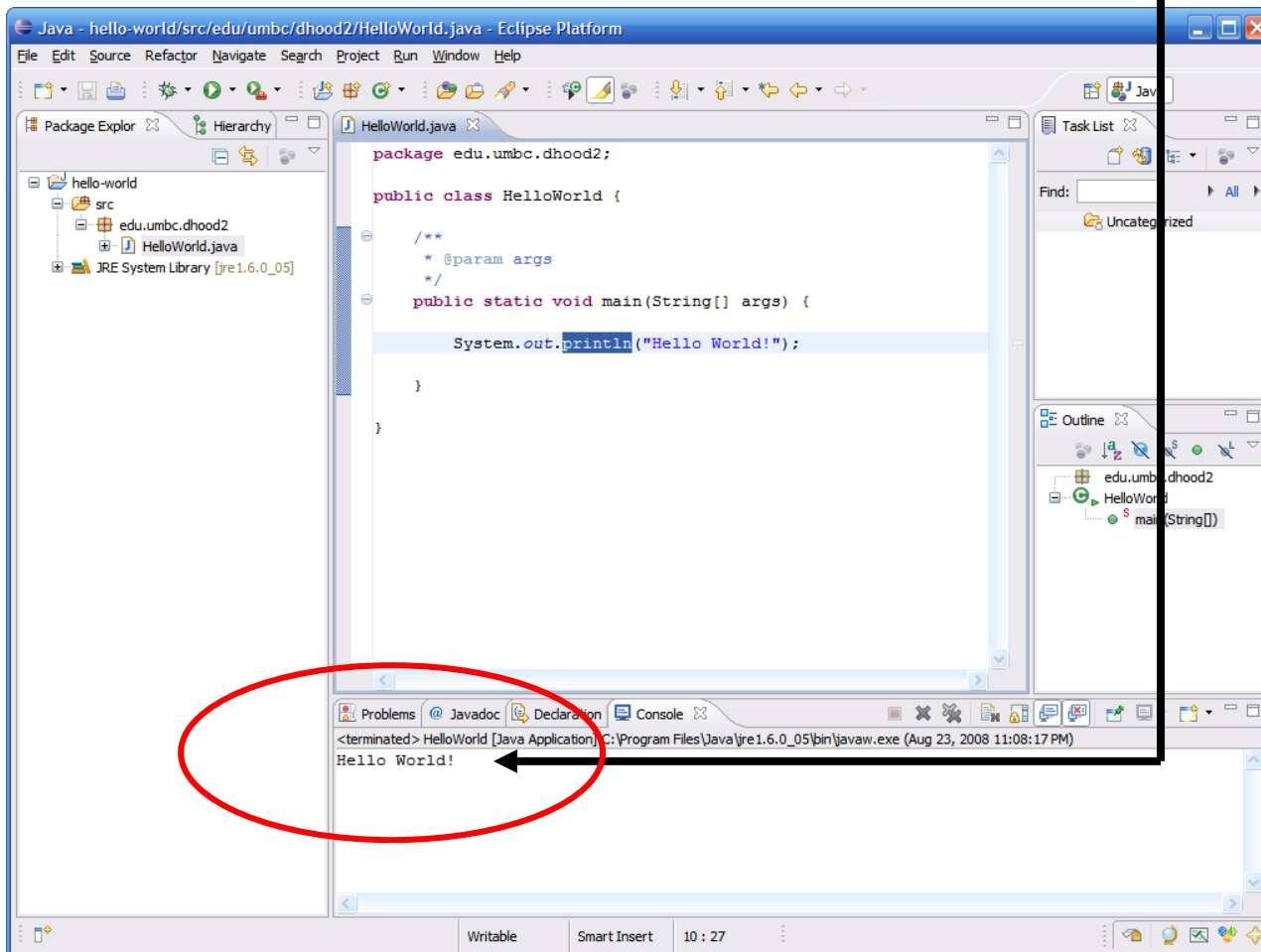
Running Code(실행)

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



Running Code (continued)

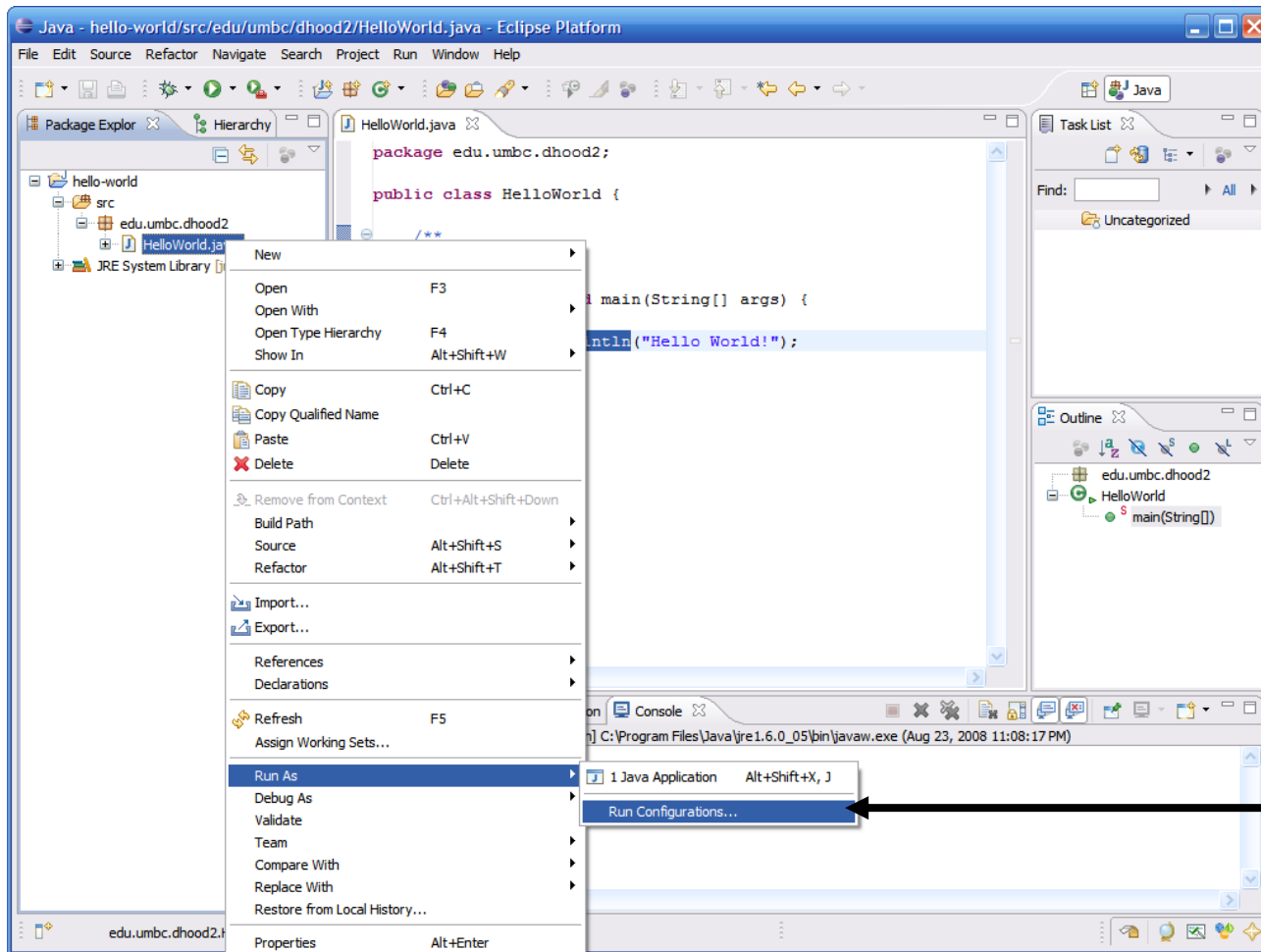
- ◆ The output of running the code can be seen in the Console tab in the bottom pane



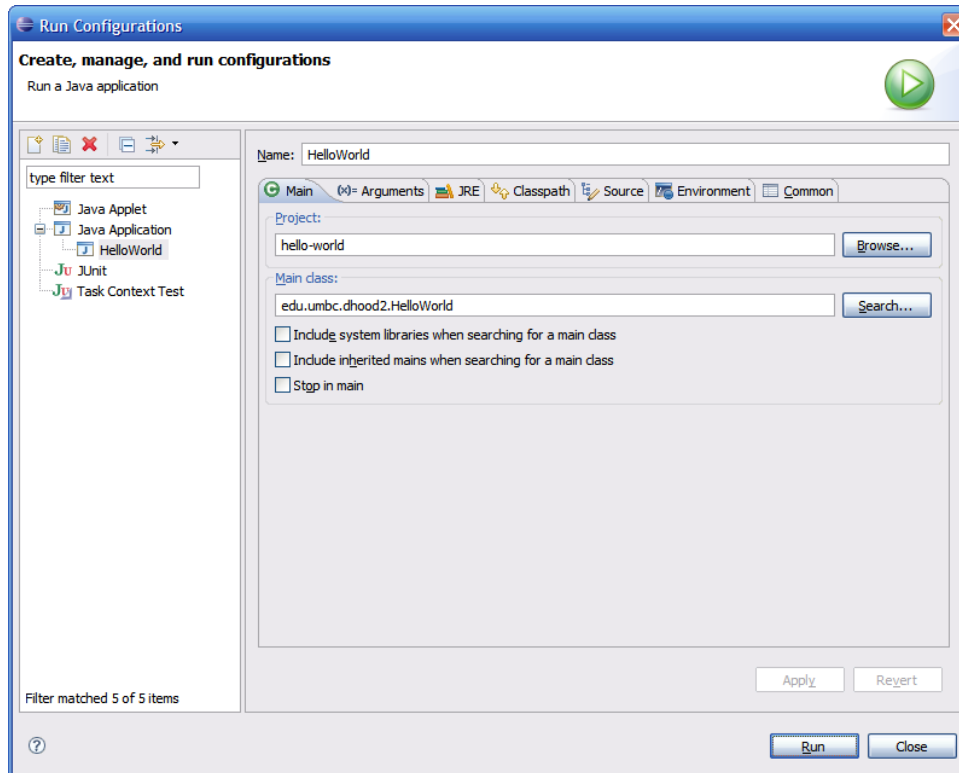
Run Configuration

여러 옵션 조정 가능

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



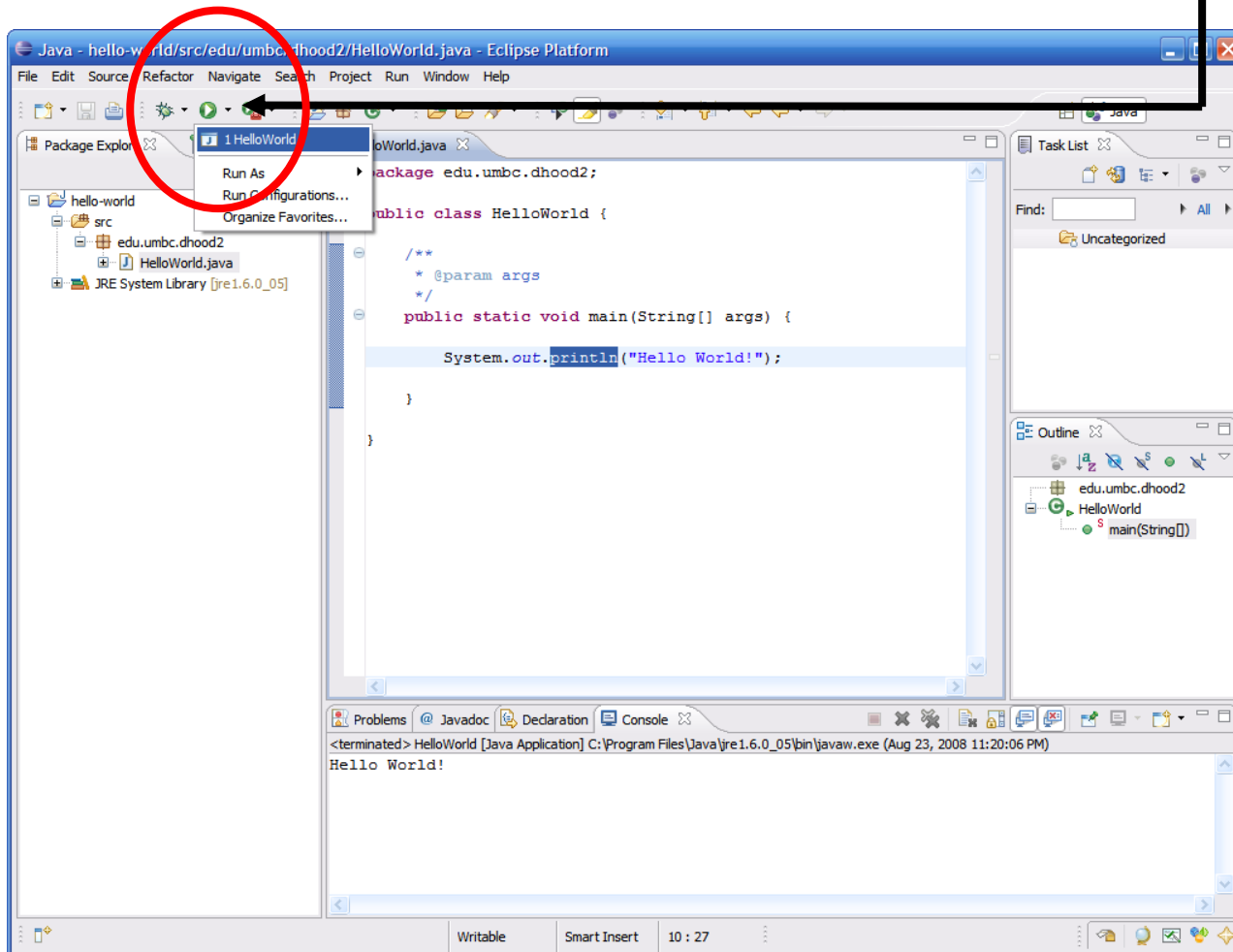
Run Configuration (continued)



- ◆ Here you can change/add any of the following:
 - ◆ JVM arguments
 - ◆ Command line arguments
 - ◆ Classpath settings
 - ◆ Environment variables
 - ◆ Which JVM to use

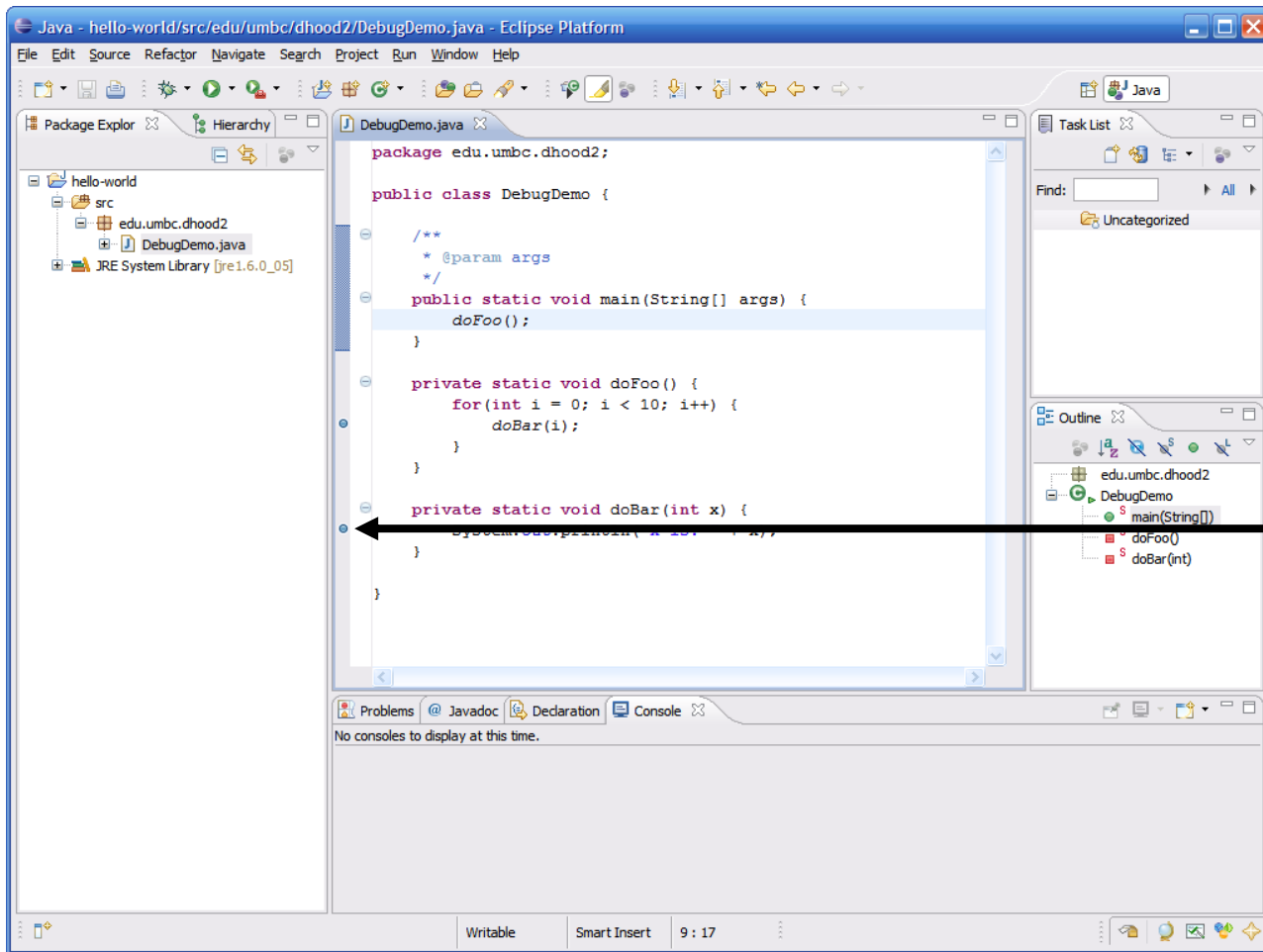
Re-Running Code(다시실행)

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



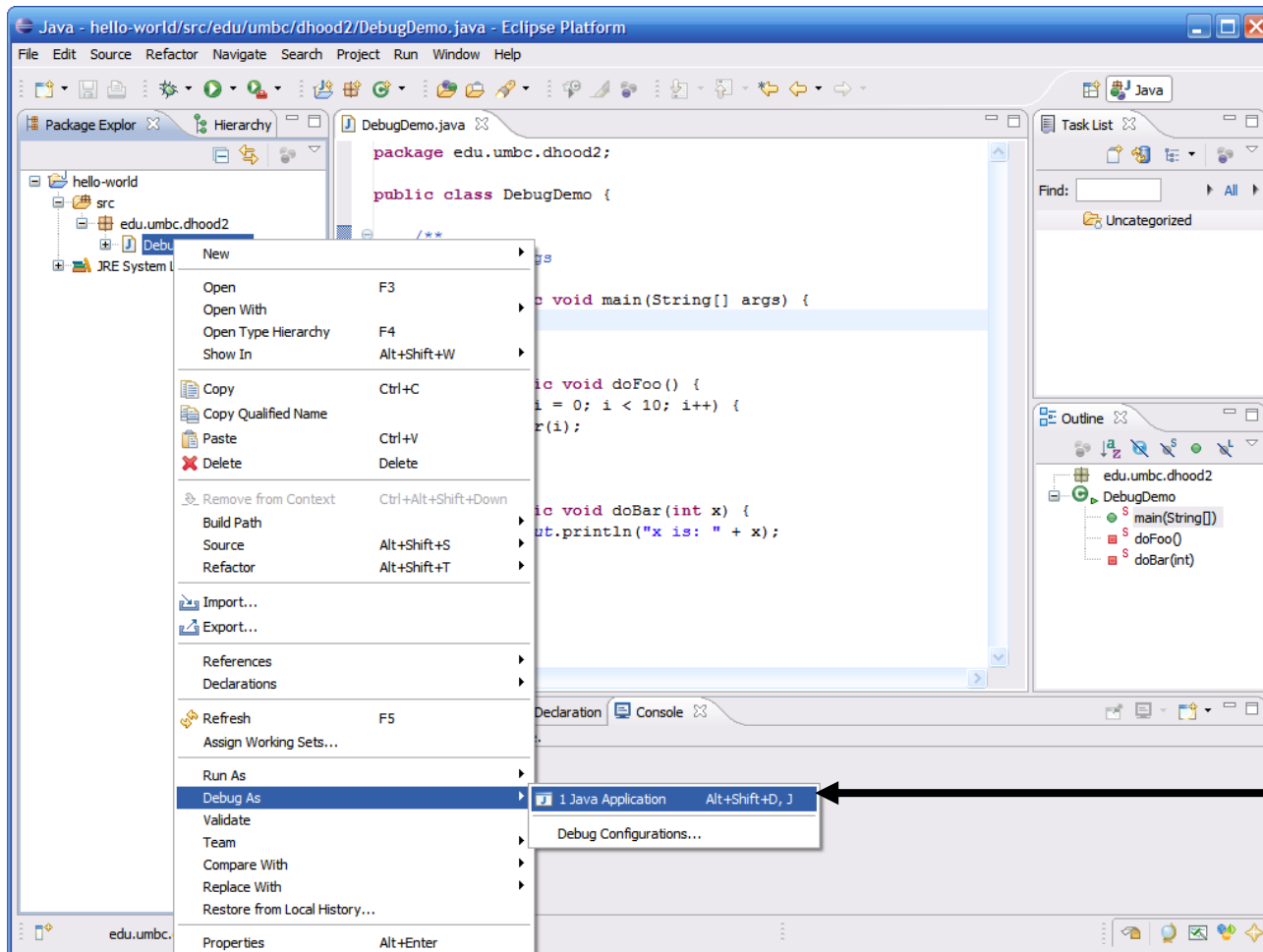
Debugging Code(디버그)

- ◆ 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



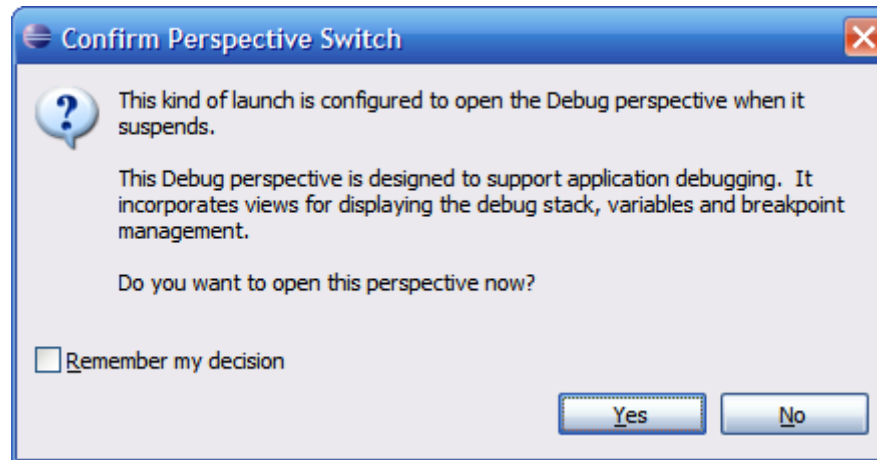
Debugging Code (continued)

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



Debugging Code (Continued)

- ◆ 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)

Innocent face



brave husky





감사합니다.

