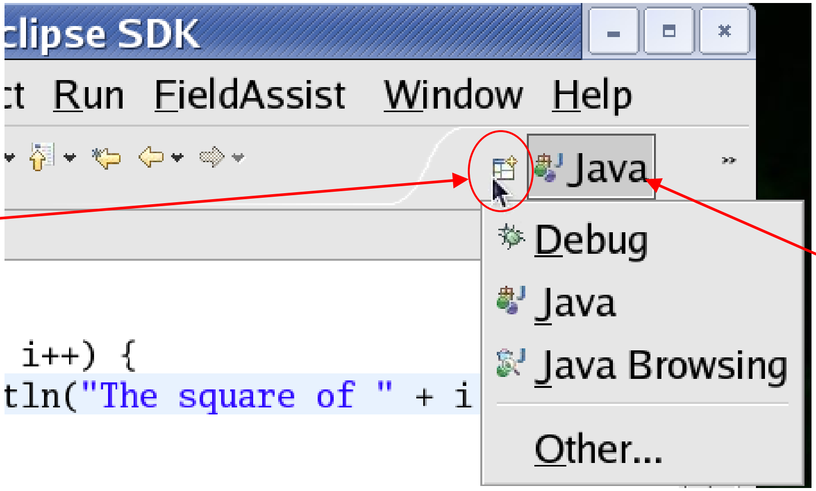
Introduction to Eclipse

Start Eclipse

Initialize Eclipse

- Choose a workspace (a directory used by Eclipse to store your programs)
  - When you first start Eclipse, Eclipse will ask you to specify the workspace to use.
  - usb 메모리 등, 자신의 작업 폴더를 지정

- Choose a perspective (the layout of Eclipse user interface).
  - Open ***Java perspective*** (an interface for editing java source code): click ***Open Perspective*** button > click ***Java***.



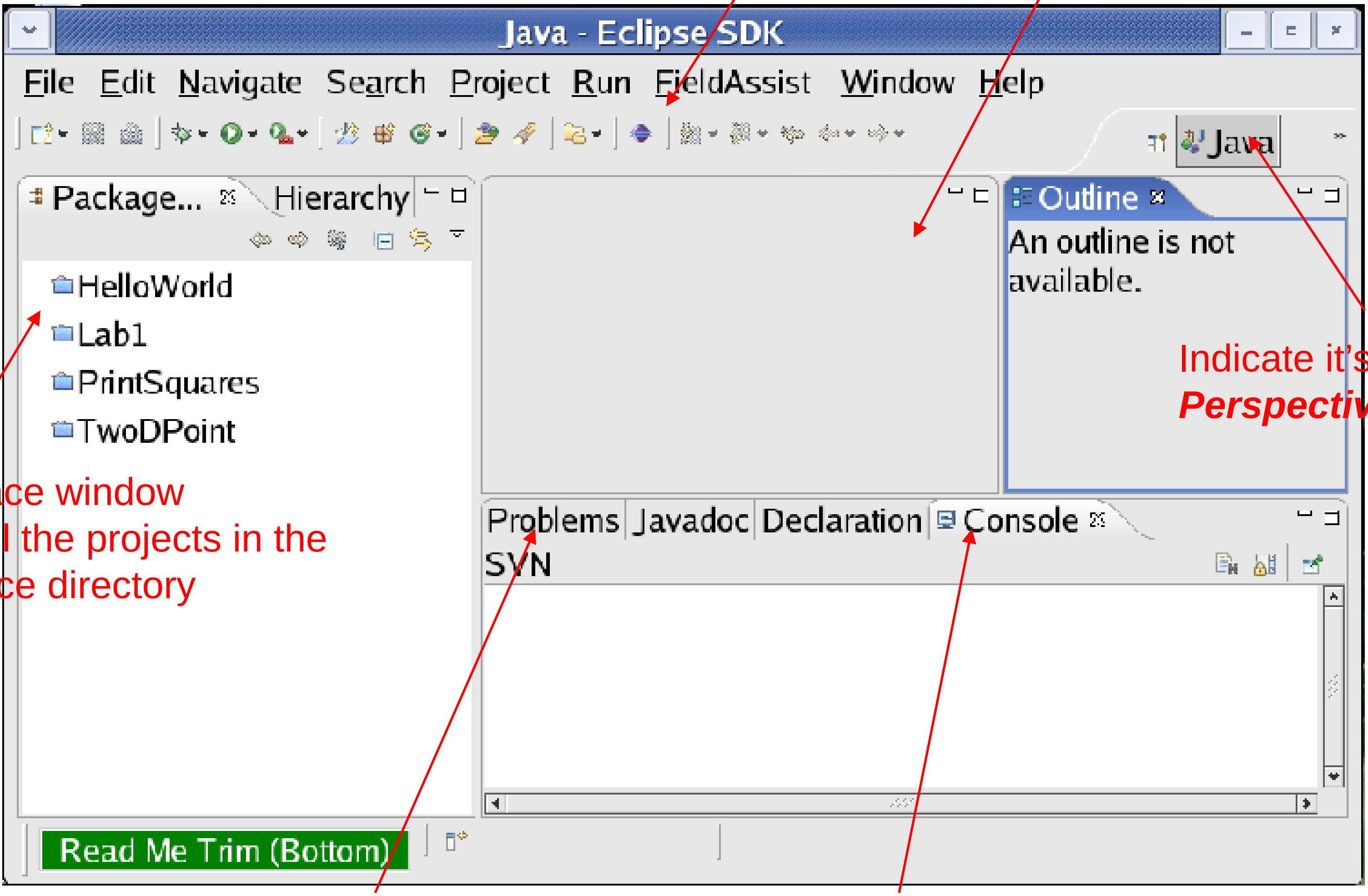
```
 i++) {
tln("The square of " + i
```

Open Perspectives button

Pick ***Java perspective***

  - ***Debug Perspective*** (an interface for debugging the program).

Overview of Eclipse Java Perspective



An outline is not available.

Toolbar

Edit window

Indicate it's ***Java Perspective***

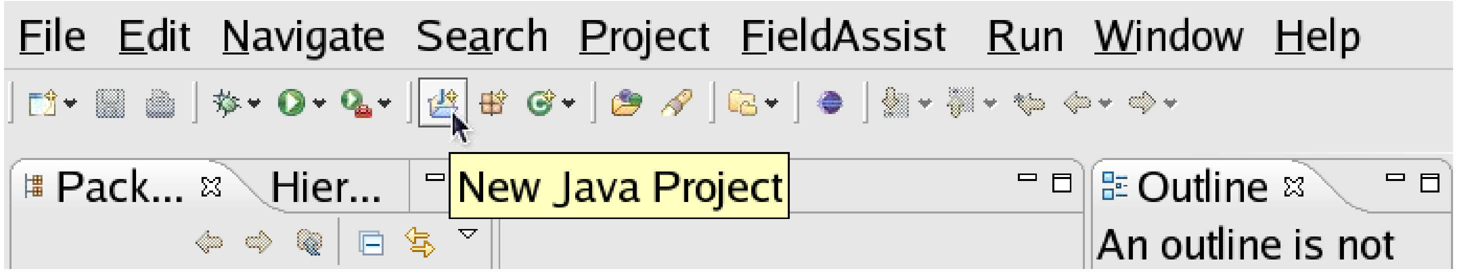
Workspace window shows all the projects in the workspace directory

Compilation outputs

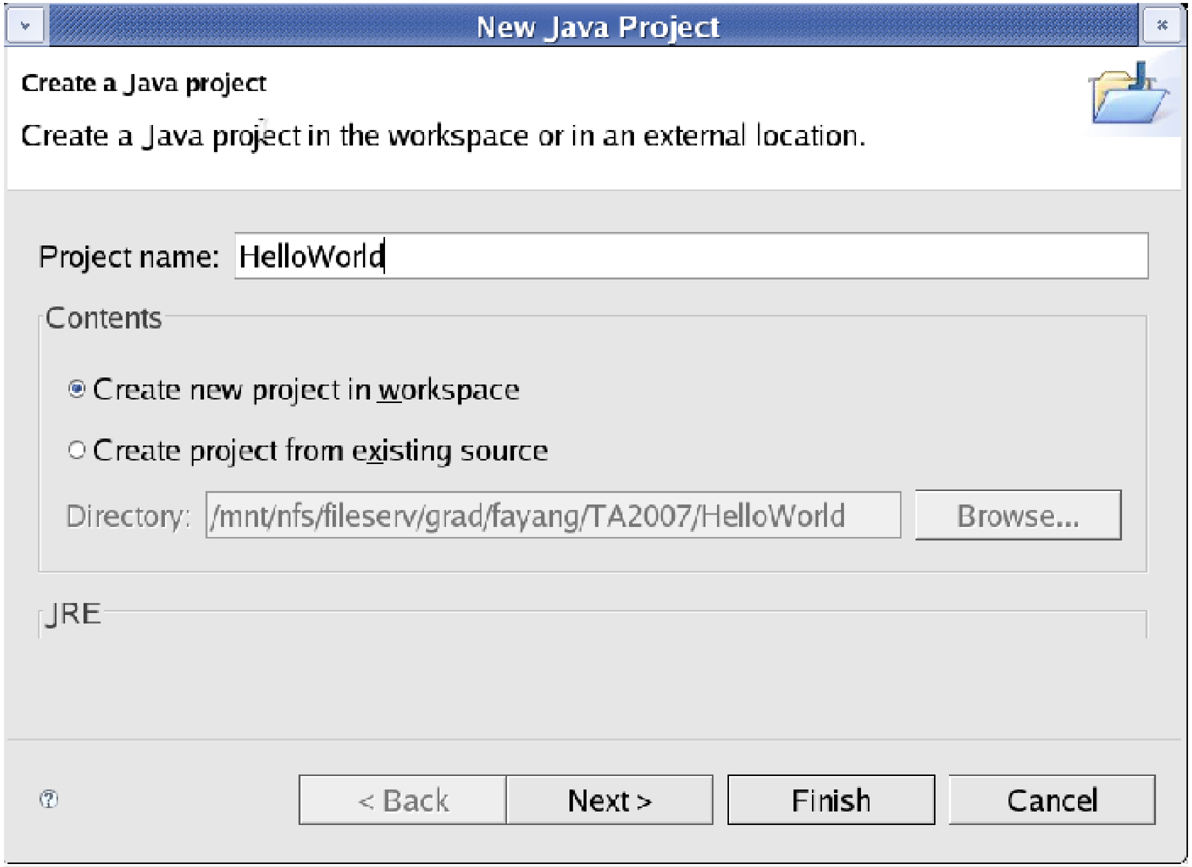
Console outputs

Create A New Java Application Example: create a HelloWorld java application

1. Create a new project named ***HelloWorld***.
   - First click ***New Java Project*** button.



- Then in ***New Java Project*** window input the project name as ***HelloWorld*** and click ***Finish*** button.



New Java Project

Create a Java project

Create a Java project in the workspace or in an external location.

Project name: HelloWorld

Contents

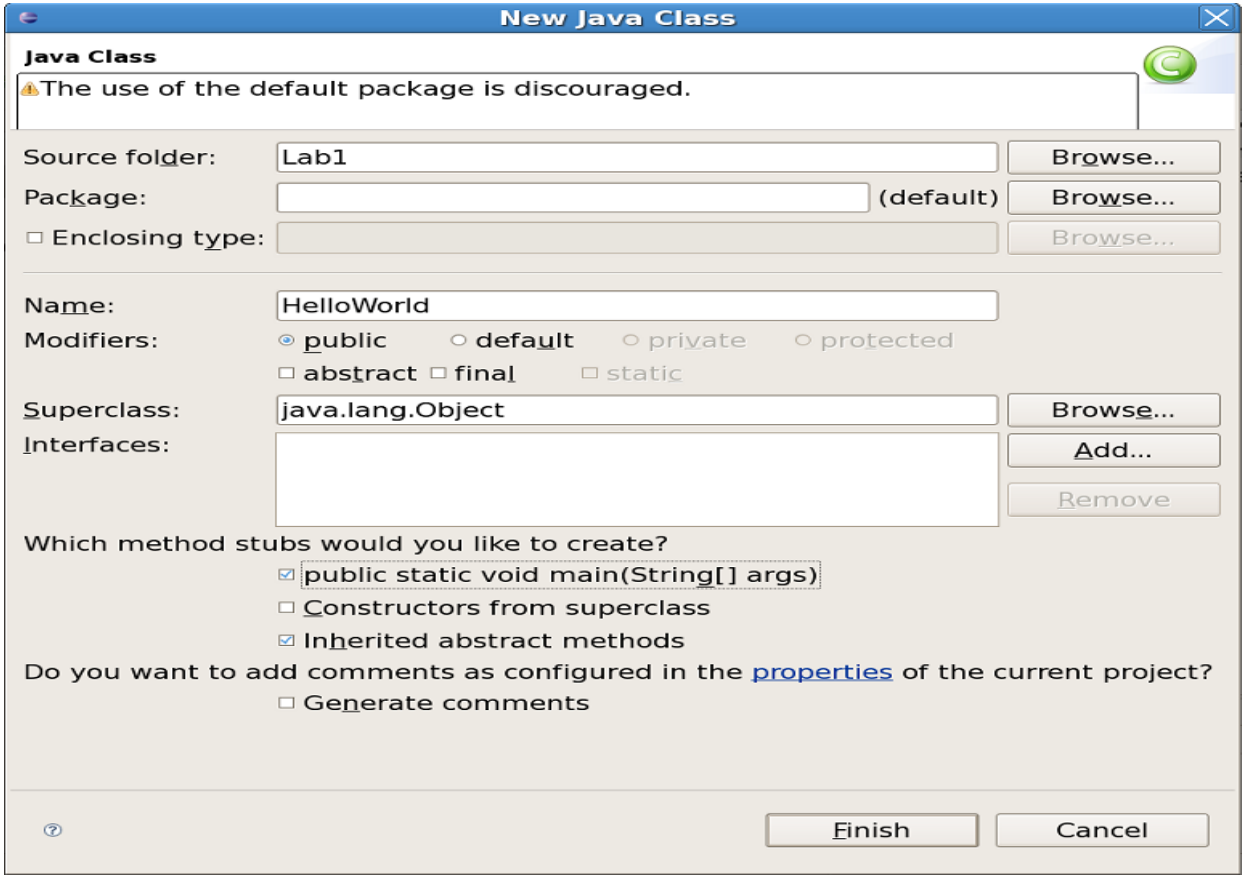
● Create new project in workspace

○ Create project from existing source

Directory: /mnt/nfs/fileserv/grad/fayang/TA2007/HelloWorld

2. Click ***New Java Class*** button to create a Java class.

– In ***New Java Class*** window, input ***HelloWorld*** as the name and check the box "public static void main (String[] args)" if you want a main method.



New Java Class

Java Class

The use of the default package is discouraged.

Source folder: Lab1

Package: (default)

□ Enclosing type:

Name: HelloWorld

Modifiers: ● public ○ default ○ private ○ protected

□ abstract □ final □ static

Superclass: java.lang.Object

Interfaces:

Which method stubs would you like to create?

☑ public static void main(String[] args)

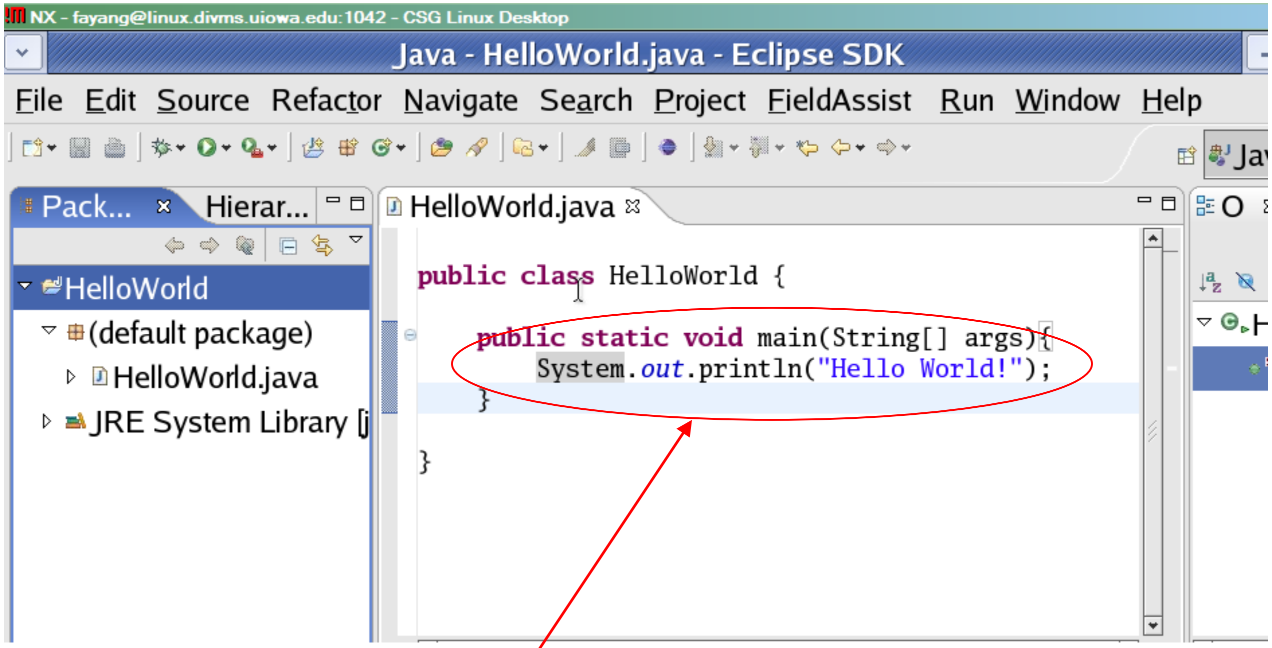
□ Constructors from superclass

☑ Inherited abstract methods

Do you want to add comments as configured in the properties of the current project?

□ Generate comments

4. Modify ***HelloWorld.java*** source code as follows:

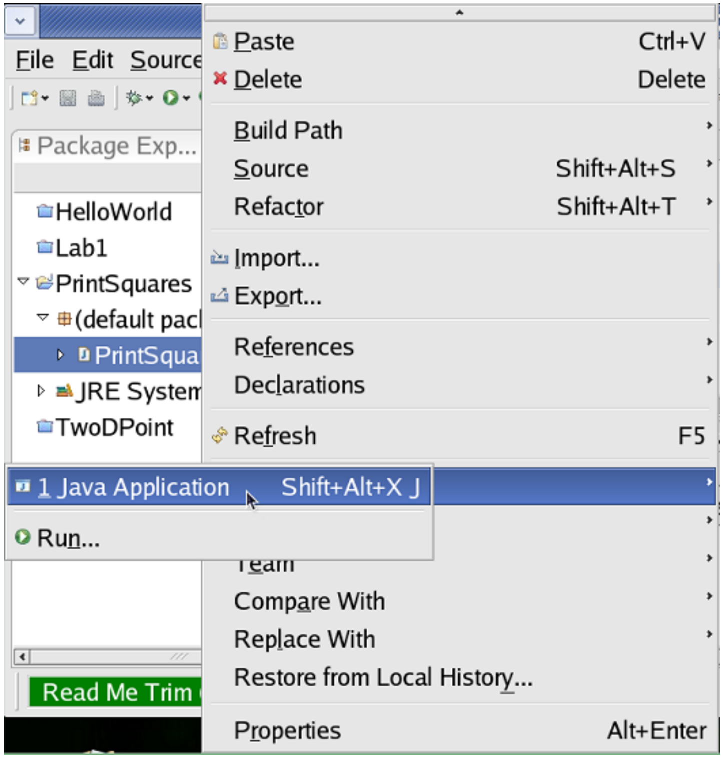

NX - fayang@linux.divms.uiowa.edu:1042 - CSG Linux Desktop
Java - HelloWorld.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project FieldAssist Run Window Help
Pack...
Hierar...
HelloWorld
(default package)
HelloWorld.java
JRE System Library
HelloWorld.java

```
public class HelloWorld {

    public static void main(String[] args){
        System.out.println("Hello World!");
    }

}
```

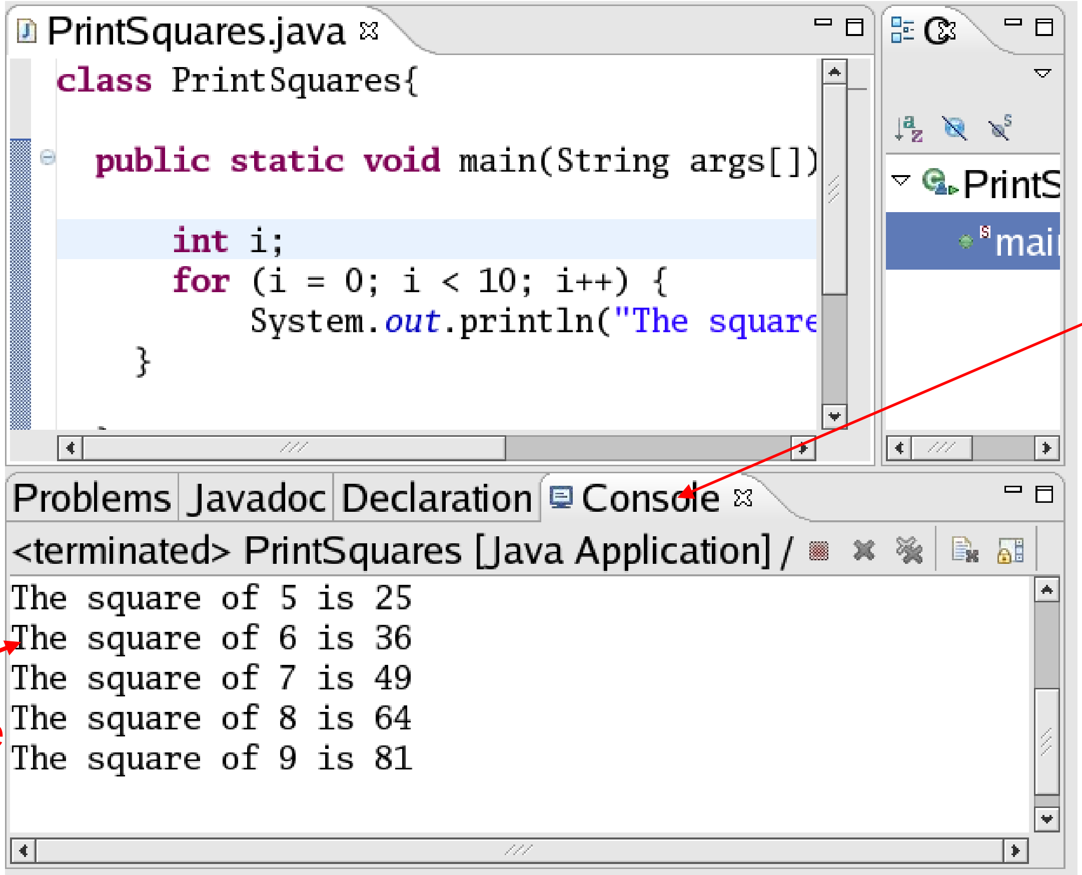
Add ***System.out.println("Hello World");***
inside ***Main*** method.

Run the program

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



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



```
class PrintSquares{

  public static void main(String args[])

      int i;
      for (i = 0; i < 10; i++) {
          System.out.println("The square
    }
```

<terminated> PrintSquares [Java Application] /

```
The square of 5 is 25
The square of 6 is 36
The square of 7 is 49
The square of 8 is 64
The square of 9 is 81
```

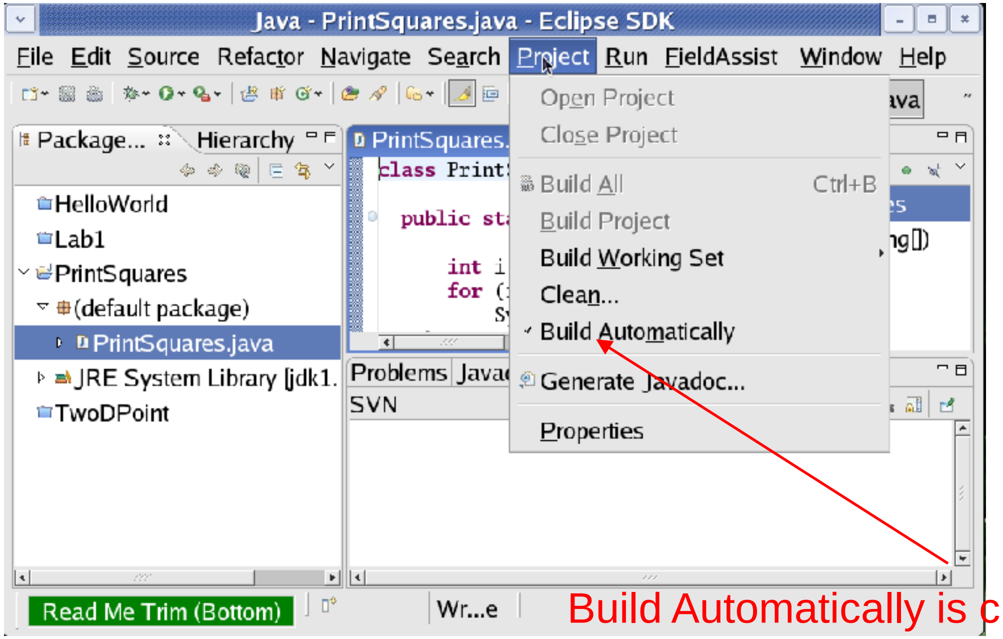
Console window

Outputs of the program

이하는 관심 있는 사람만 참조

Compile the program

- If ***Build Automatically*** is checked, the program will be automatically compiled whenever you save the program.

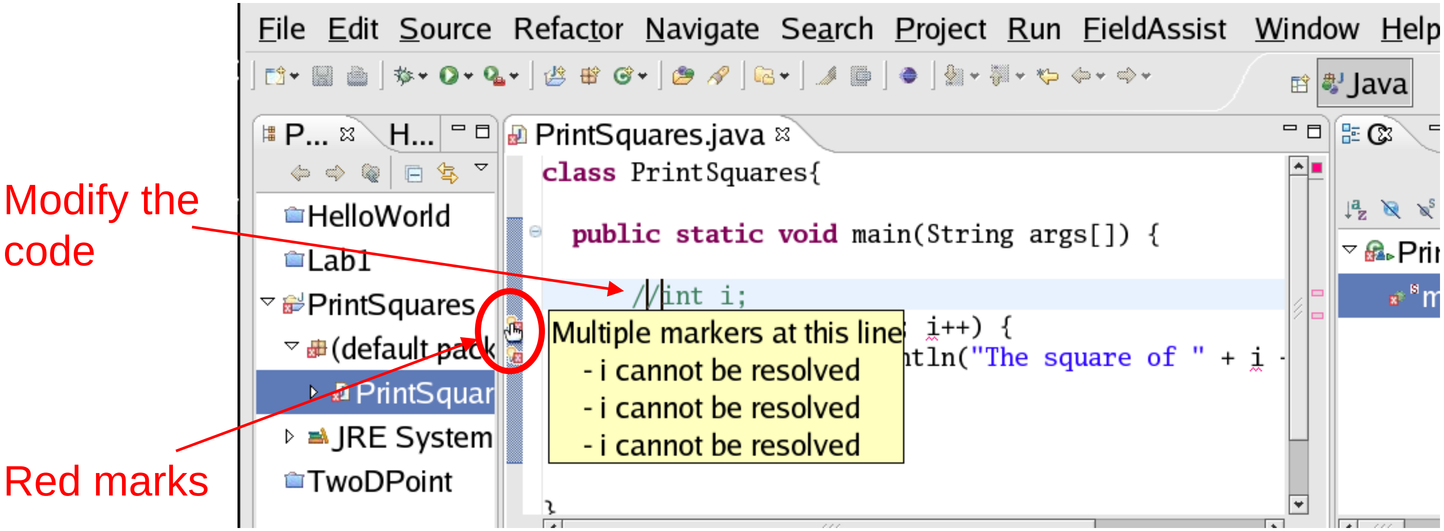


Build Automatically is checked

1. Modify “PrintSquares.java” source code as follows:

int i; —Change to→ //int i;

2. The red marks on the left side of *Edit* window indicate that there are errors in ***PrintSquares.java***. Move the cursor over a red mark to see the error message.



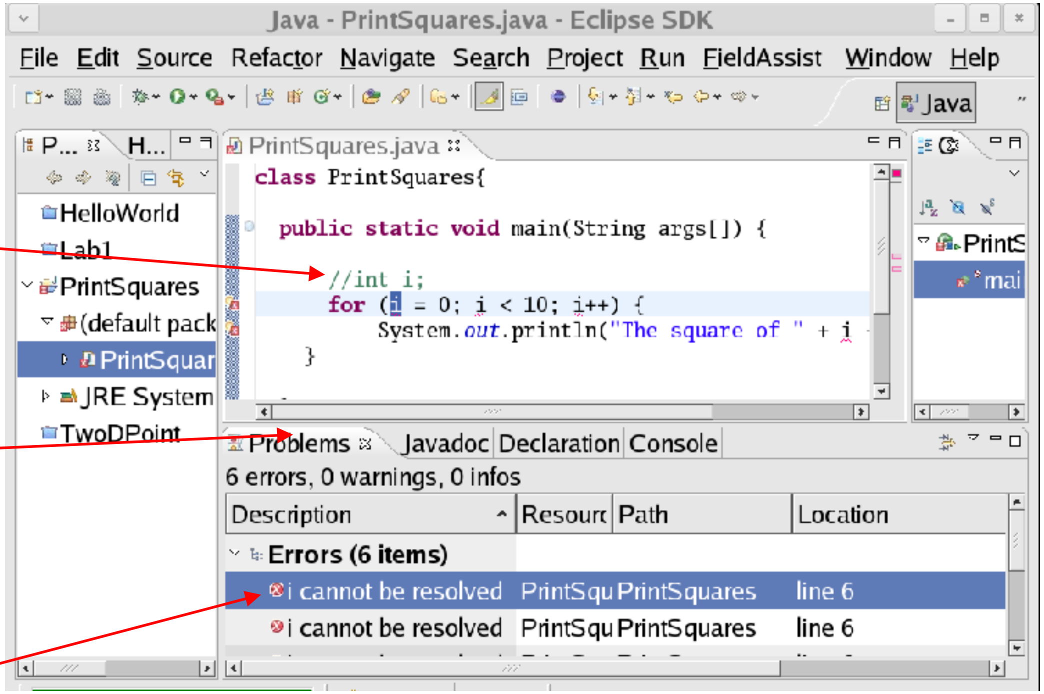
Modify the code

Red marks

```
class PrintSquares{

    public static void main(String args[]) {

        //int i;
                                          i++) {
                                    tln("The square of " + i

}
```

Multiple markers at this line
- i cannot be resolved
- i cannot be resolved
- i cannot be resolved

3. Click ***Save*** button on the ***toolbar*** to compile the program. ***Problems*** window shows the errors in the source code. Double click an error message and the cursor in ***Edit*** window will automatically move to the line in the source code where the error appears.



```
class PrintSquares{

  public static void main(String args[]) {

     //int i;
     for (i = 0; i < 10; i++) {
         System.out.println("The square of " + i
     }
```

6 errors, 0 warnings, 0 infos

Description	Resourc	Path	Location
Errors (6 items)			
i cannot be resolved	PrintSqu	PrintSquares	line 6
i cannot be resolved	PrintSqu	PrintSquares	line 6

Cursor move to here

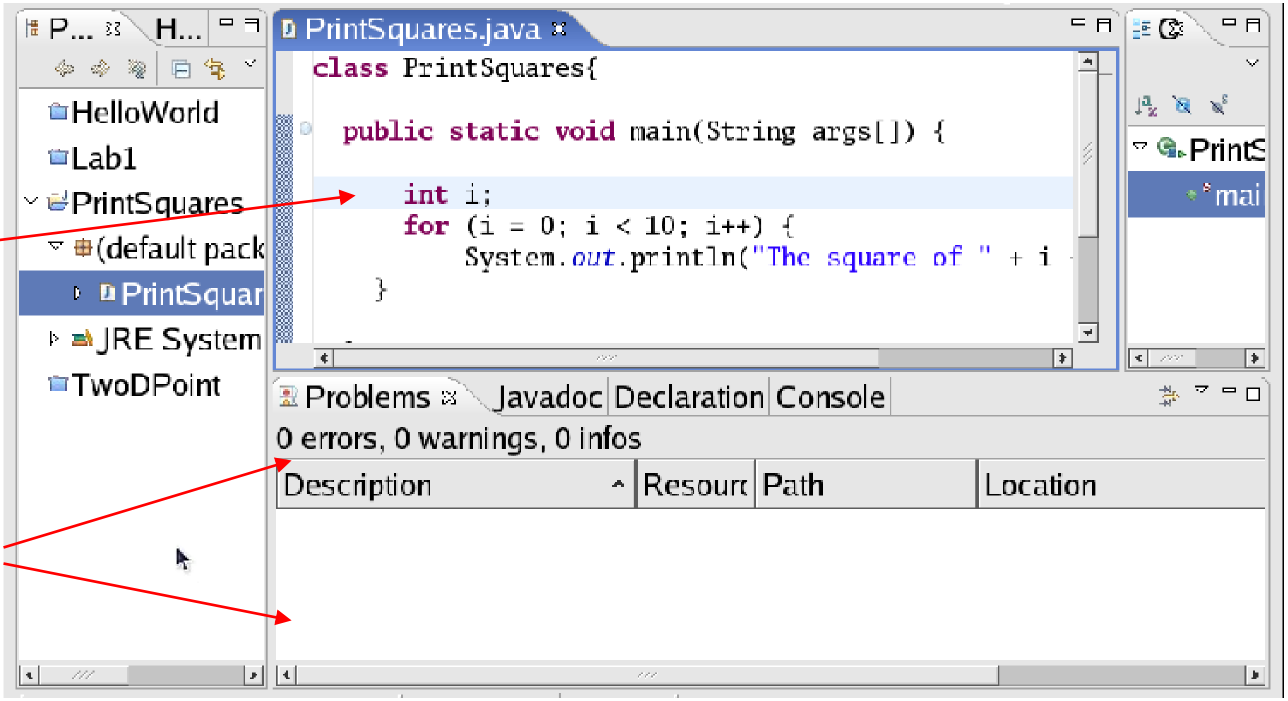
Problems window

Double click the error message

4. Correct ***PrintSquares.java*** source code as follows:

//int i; Change back to → int i;

5. Click ***Save*** button to compile the code again.



```
class PrintSquares{

  public static void main(String args[]) {

      int i;
      for (i = 0; i < 10; i++) {
          System.out.println("The square of " + i
    }
```

0 errors, 0 warnings, 0 infos

Description	Resourc	Path	Location

Error has been corrected

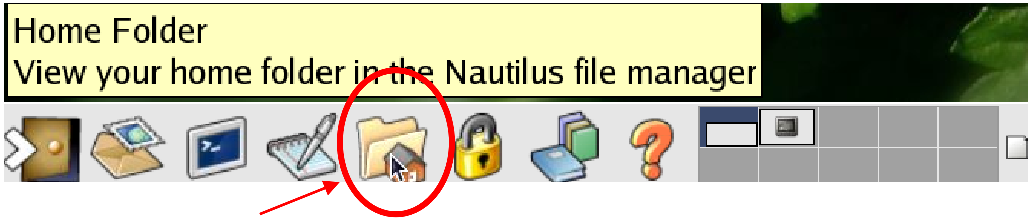
No errors

Load an Existing Java Program

1. In Eclipse create a project named ***PrintSquares***.
    - Click ***New Java Project*** button.
    - Type ***PrintSquares*** as the project name and then click ***Finish*** button.

Load an Existing Java Program

1. Open ***Home Folder*** and find the Workspace directory you use for Eclipse.



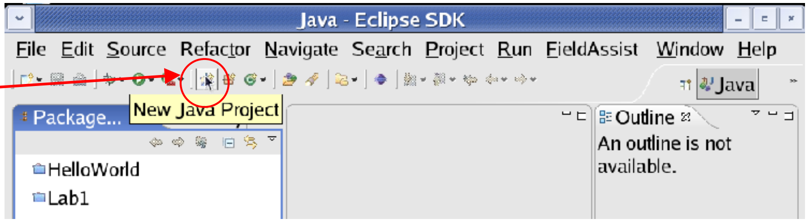
Home Folder
View your home folder in the Nautilus file manager

2. Create a folder named ***PrintSquares*** (or any other name you prefer) under the workspace directory.

3. Download **PrintSquares.java** from Lab1 Document to ***PrintSquares*** folder you just created.

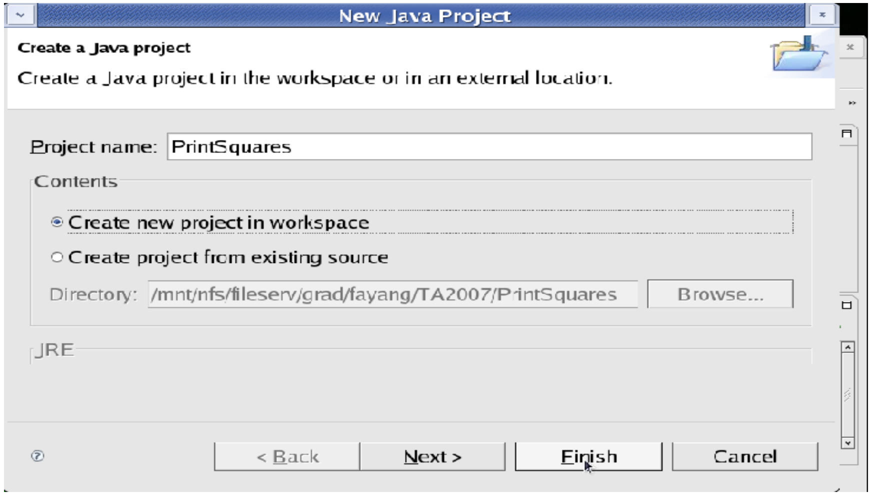
4. In Eclipse create a project named ***PrintSquares***.

- Click ***New Java Project*** button.


Java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run FieldAssist Window Help
Java
New Java Project
Package...
HelloWorld
Lab1
Outline
An outline is not available.

New Java Project button

- Type ***PrintSquares*** as the project name and then click ***Finish*** button.



Create a Java project

Create a Java project in the workspace or in an external location.

Project name: PrintSquares

Contents

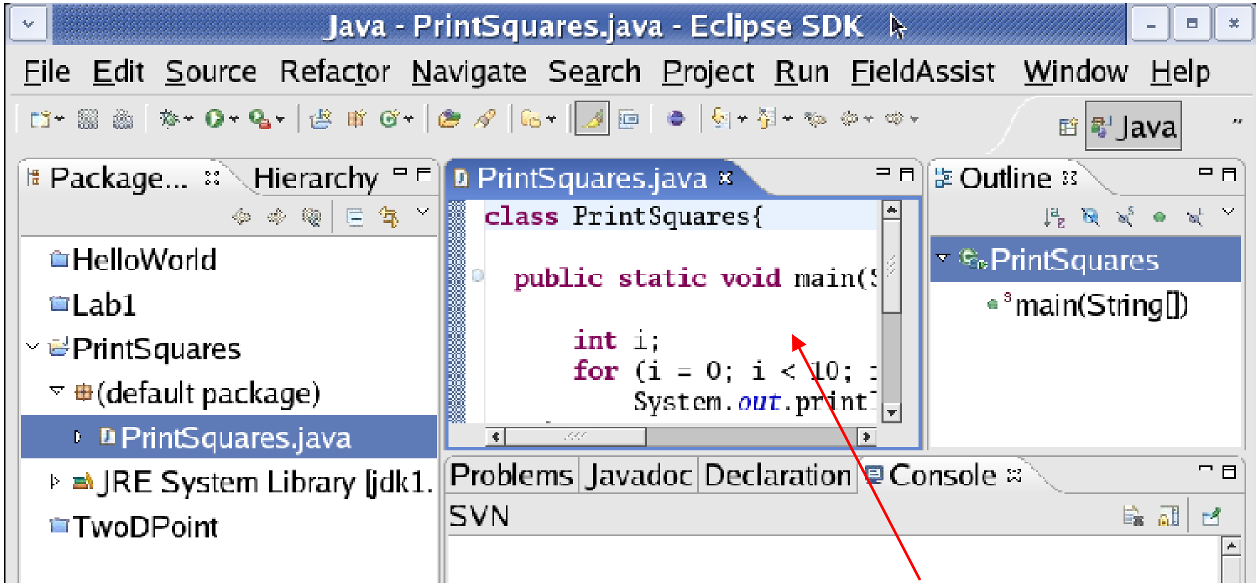
● Create new project in workspace

○ Create project from existing source

Directory: /mnt/nfs/fileserv/grad/fayang/TA2007/PrintSquares

JRE

5. In Workspace window double click ***PrintSquares***, then ***(default package)***, and then ***PrintSqaures.java***. The source code of ***PrintSquares.java*** is shown in **Edit** window.

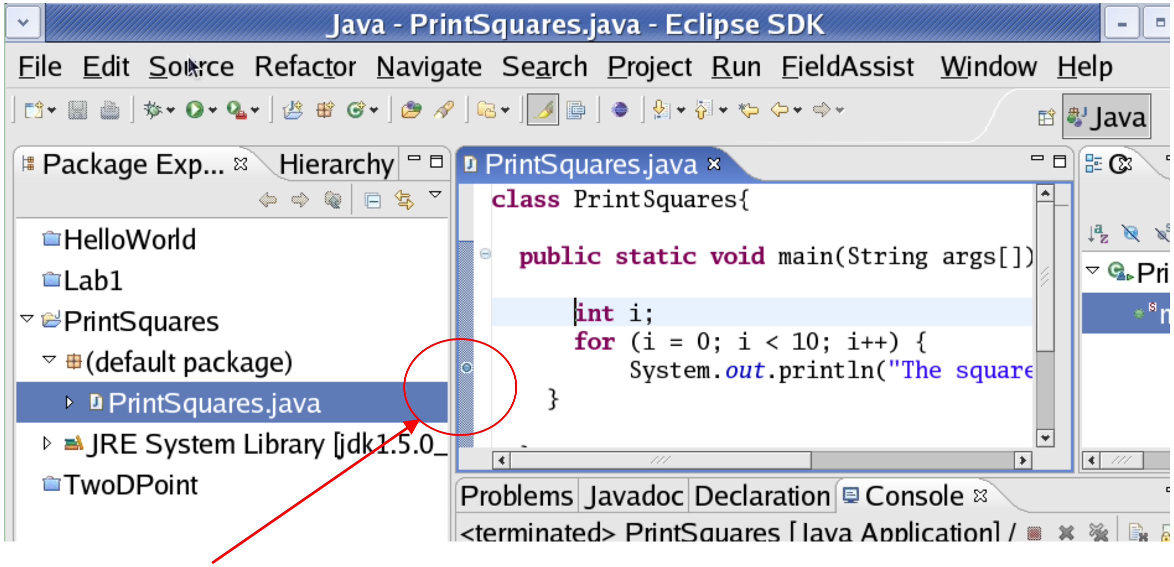


```
class PrintSquares{

  public static void main(S

     int i;
     for (i = 0; i < 10; i
        System.out.print
```

Edit window

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 break point, double click the breakpoint.

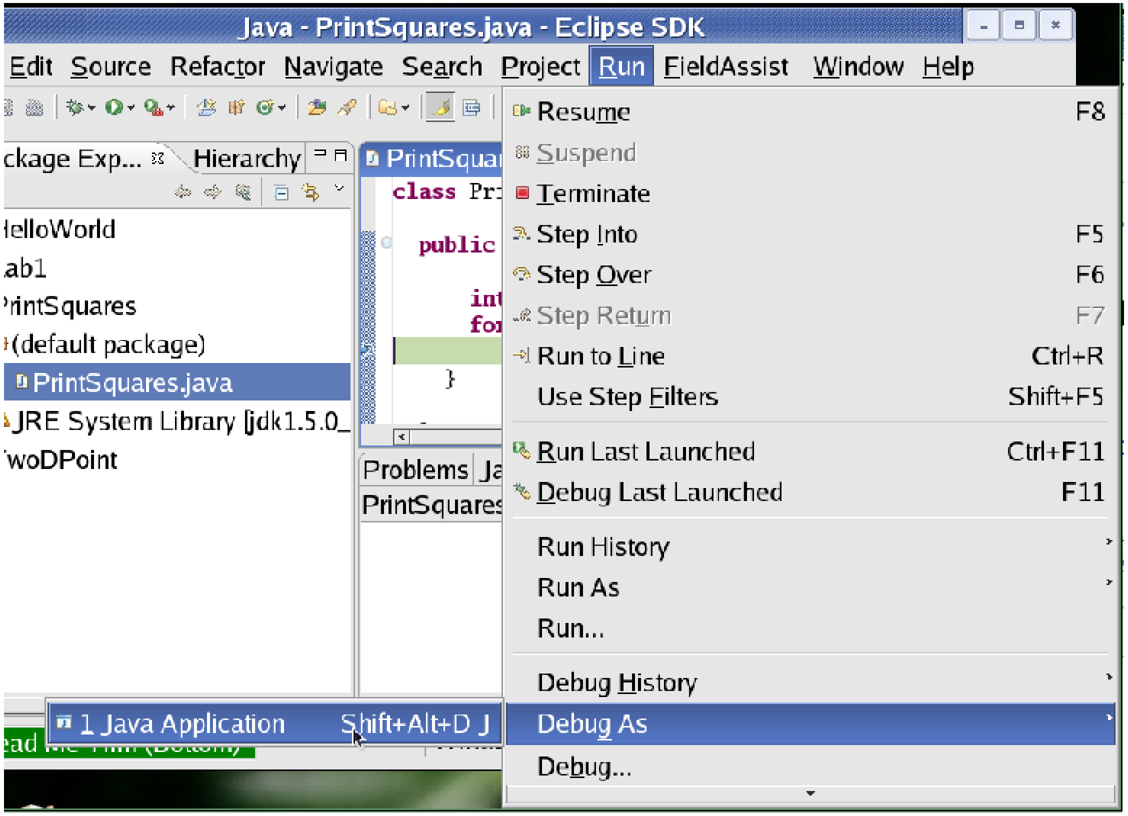


```
class PrintSquares{

  public static void main(String args[])

      int i;
      for (i = 0; i < 10; i++) {
          System.out.println("The square
    }
```

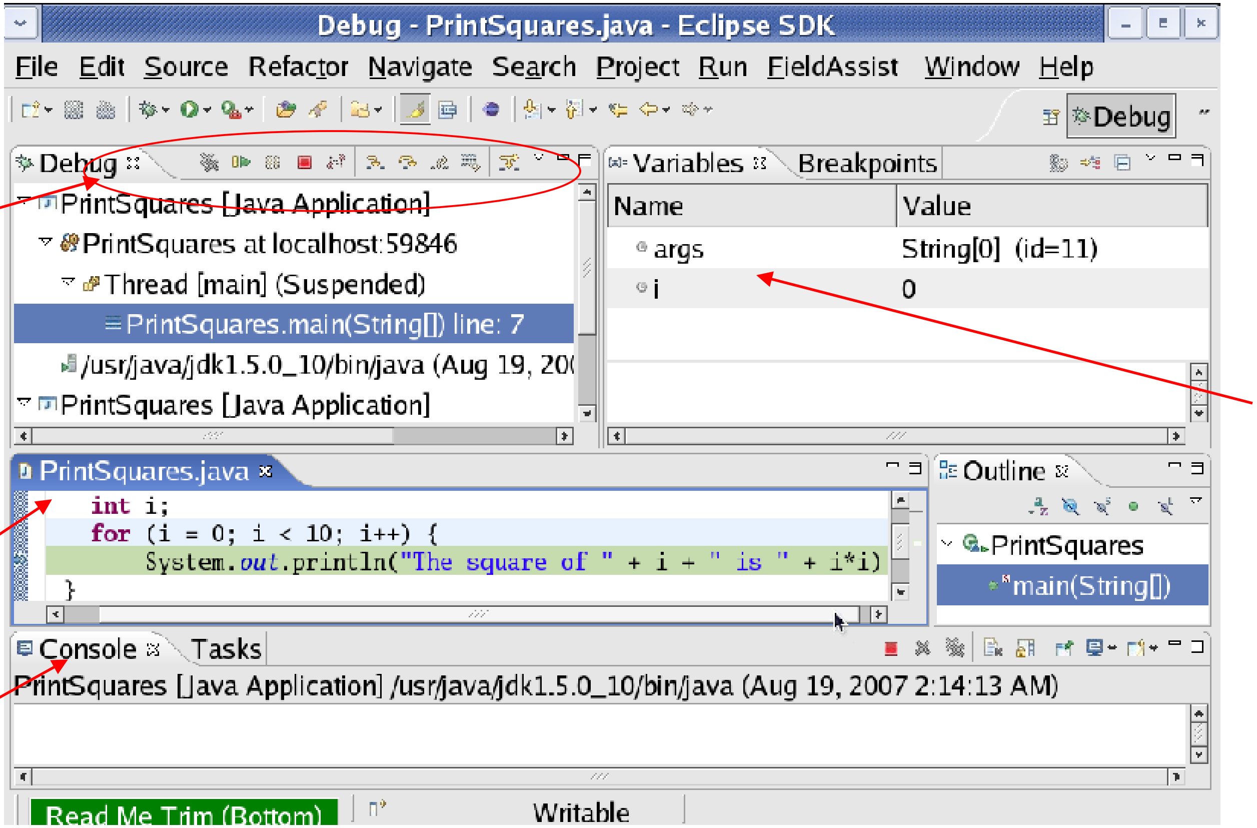
<terminated> PrintSquares [Java Application] /

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***.


Debug - PrintSquares.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run FieldAssist Window Help
Debug
Debug
Variables
Breakpoints
Outline
PrintSquares
main(String[])
Console
Tasks
Read Me Trim (Bottom)
Writable

PrintSquares [Java Application]
PrintSquares at localhost:59846
Thread [main] (Suspended)
PrintSquares.main(String[]) line: 7
/usr/java/jdk1.5.0_10/bin/java (Aug 19, 20(
PrintSquares [Java Application]

Name	Value
args	String[0] (id=11)
i	0

PrintSquares.java

```
        int i;
        for (i = 0; i < 10; i++) {
            System.out.println("The square of " + i + " is " + i*i)
    }
```

PrintSquares [Java Application] /usr/java/jdk1.5.0_10/bin/java (Aug 19, 2007 2:14:13 AM)

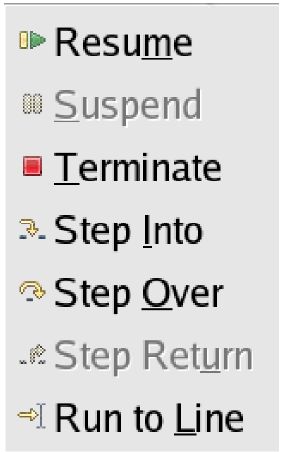
Toolbar for debug

Variable values

Edit window

Console window

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.

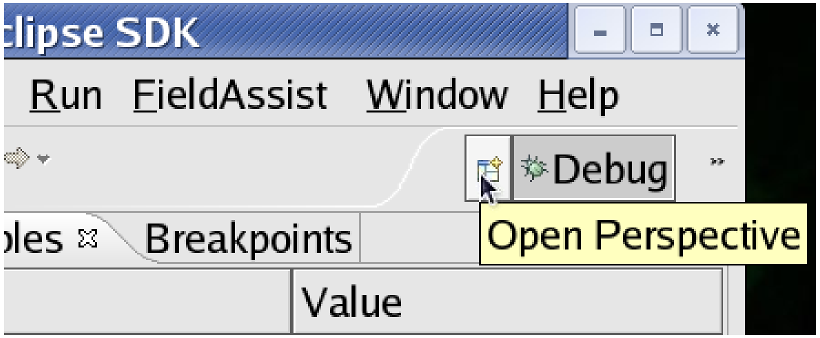
Step Over execute a single statement. If the statement contains a call to a method, the entire method is executed without stepping into the method.

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

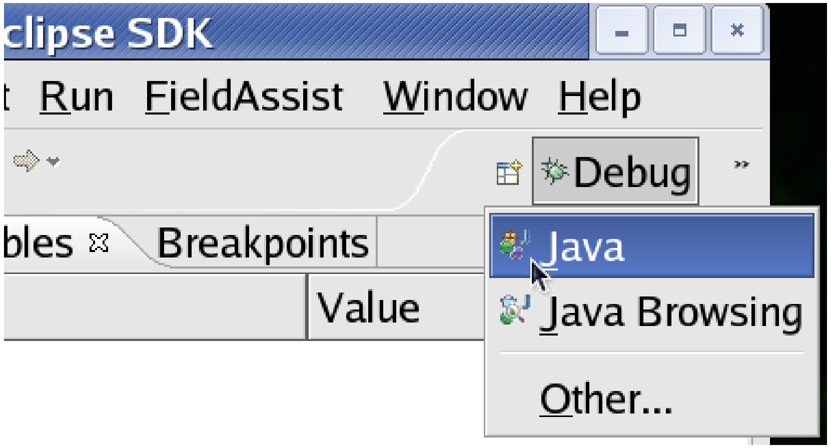
Run to Line runs the program, starting from the current execution point, and pauses at a breakpoint.

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

- Click ***Open Perspective*** button.



- Then click ***Java***.


clipse SDK
t Run FieldAssist Window Help
Debug
bles
Breakpoints
Value
Java
Java Browsing
Other...