Java Integrated Development Environments: ECLIPSE

Part1

Installation

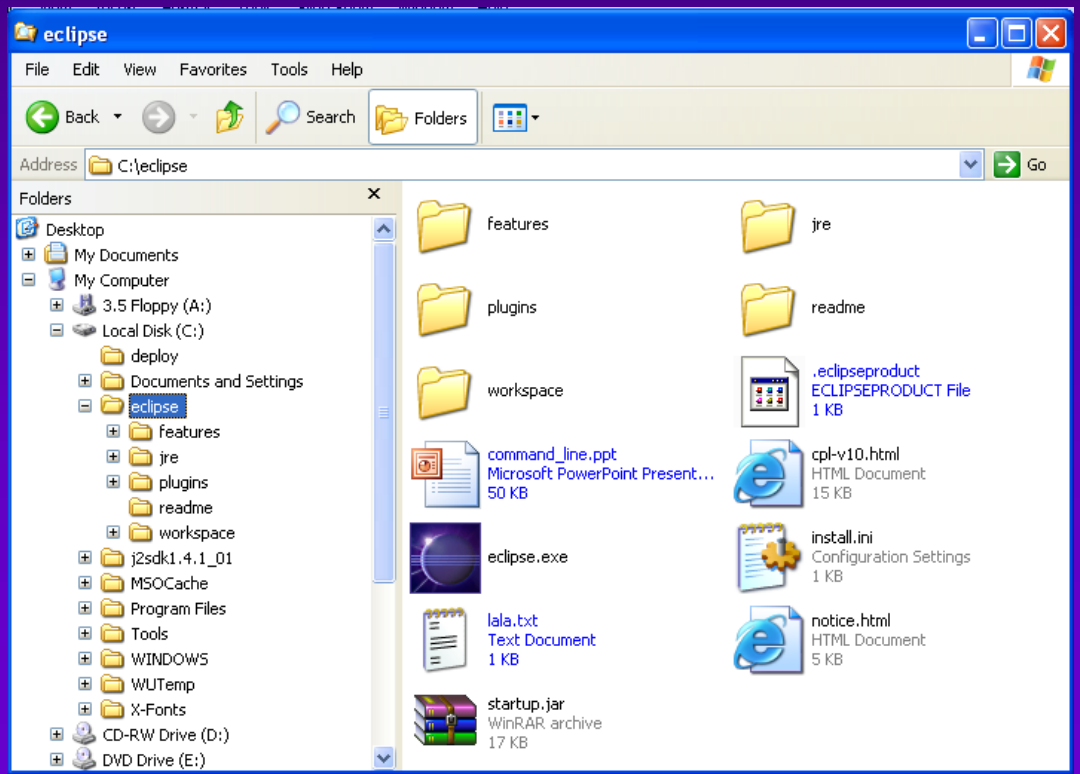
Eclipse Installation

- Detailed Instructions http://www.3plus4software.de/eclipse/installation_en.html
- Basic Ingredients:
- Windows, Linux, Solaris, QNX or Mac OS/X operating system.
- 256 MB RAM or more
- A Java 2 runtime environment (JRE) or Java 2 Software Development Kit (J2SDK). Eclipse needs version 1.3 or higher .

- Eclipse SDK version 2.1.3 (the last stable version)

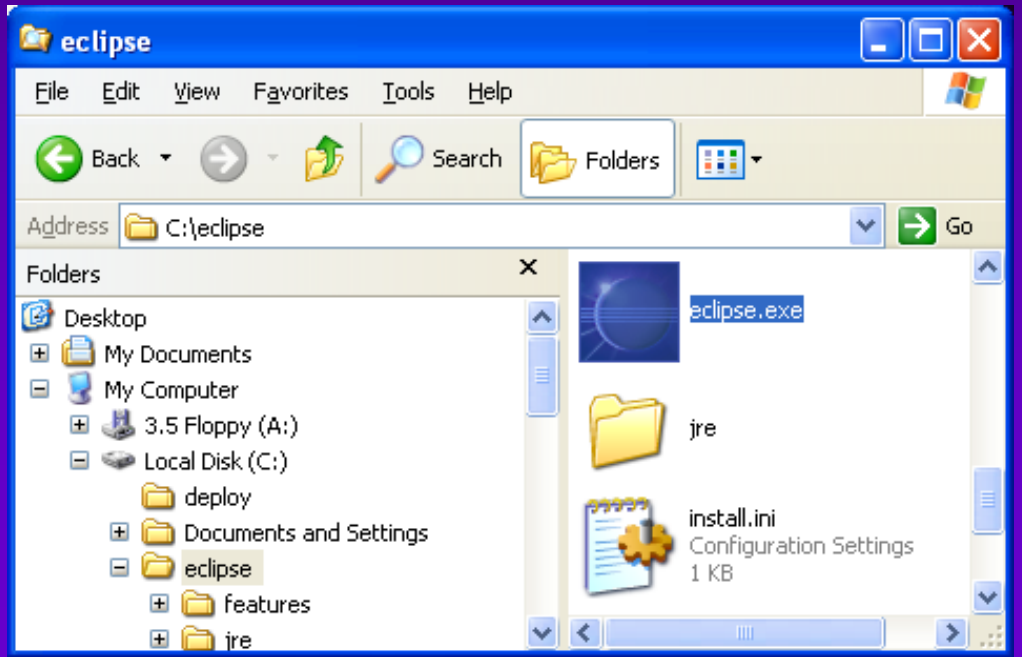
Eclipse Installation

- Extract eclipse-SDK-*-*.zip in the path that will now more be the *%eclipse_home%*. An *eclipse* directory will be created.
  - *E.g. C:\eclipse*



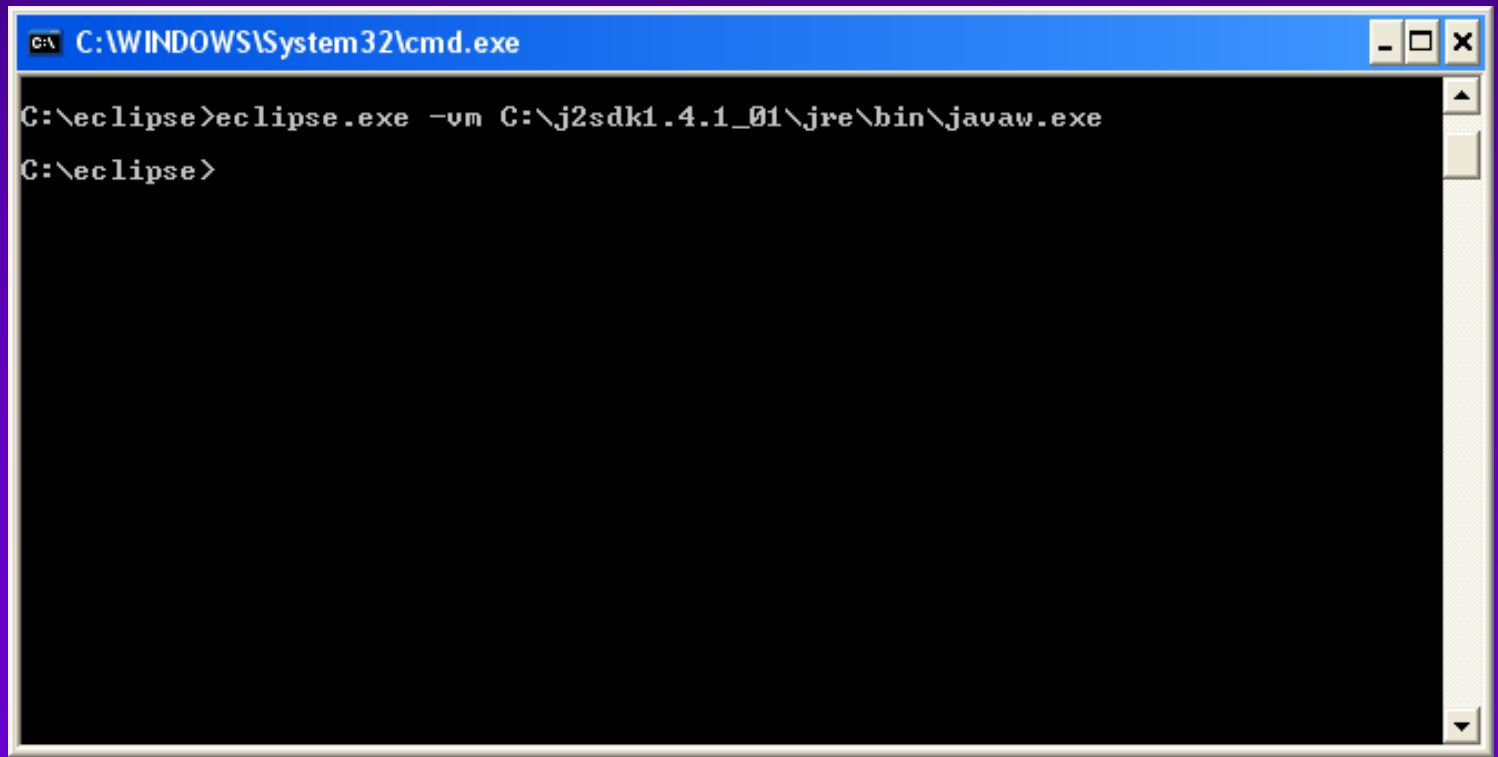
Eclipse Installation

- Eclipse is a Java program. You have to specify the Java path. There are two ways:
  - *1st:Specify the Java path in the PATH environment variable*
  - *2nd:Copy the jre folder from the JAVA_HOME directory to the ECLIPSE_HOME directory*
- Run eclipse; Click on the eclipse icon



Eclipse Installation

- Alternative. Run eclipse from a command line window specifying the Java path (-vm)
- Gives flexibility in specifying various parameters when starting eclipse



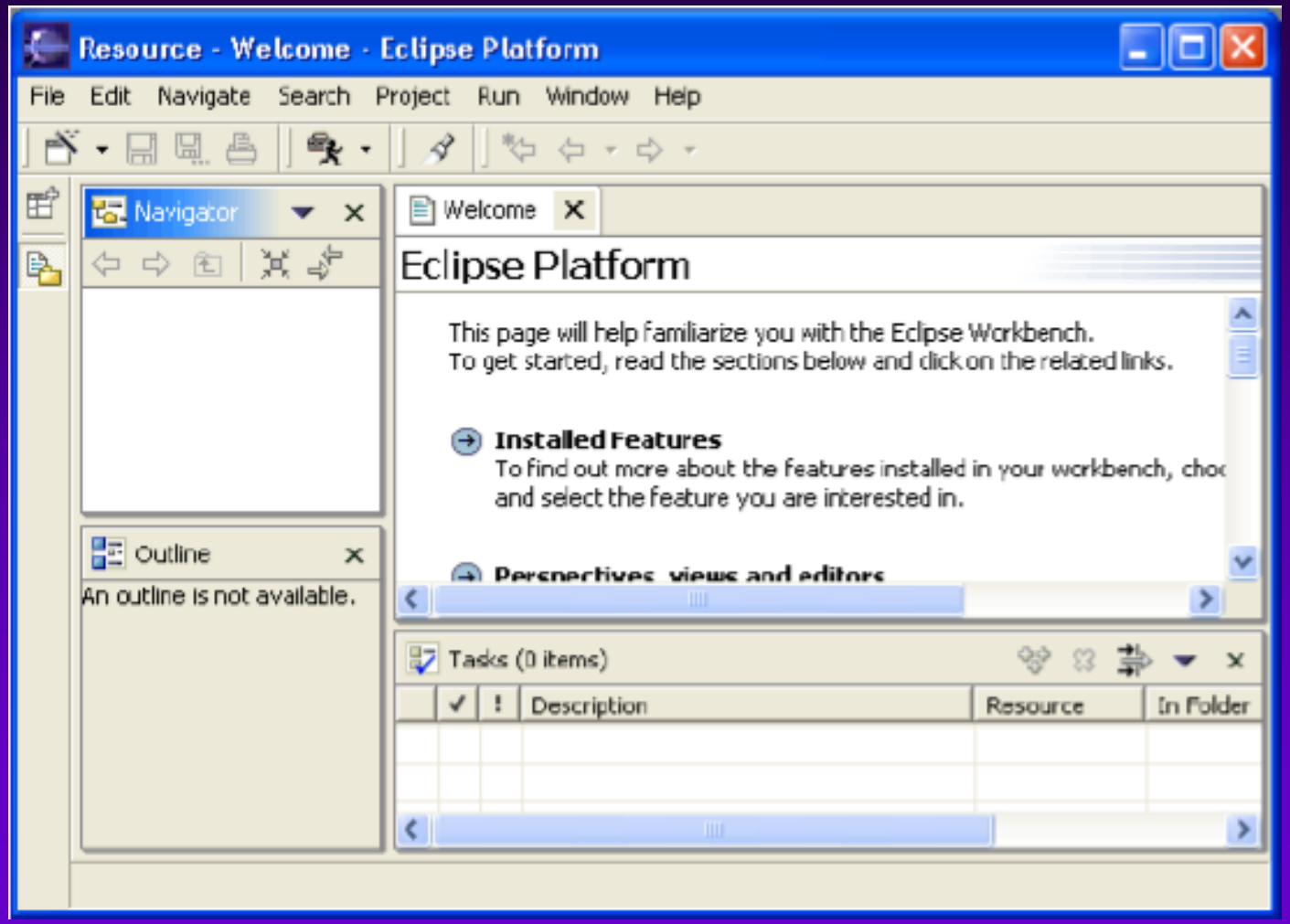
Eclipse Installation

- For more details on running eclipse visit this: http://help.eclipse.org/help21/index.jsp?topic=/org.eclipse.platform.doc.user/tasks/running_eclipse.htm
- The eclipse site http://www.eclipse.org

Part2

Workbench

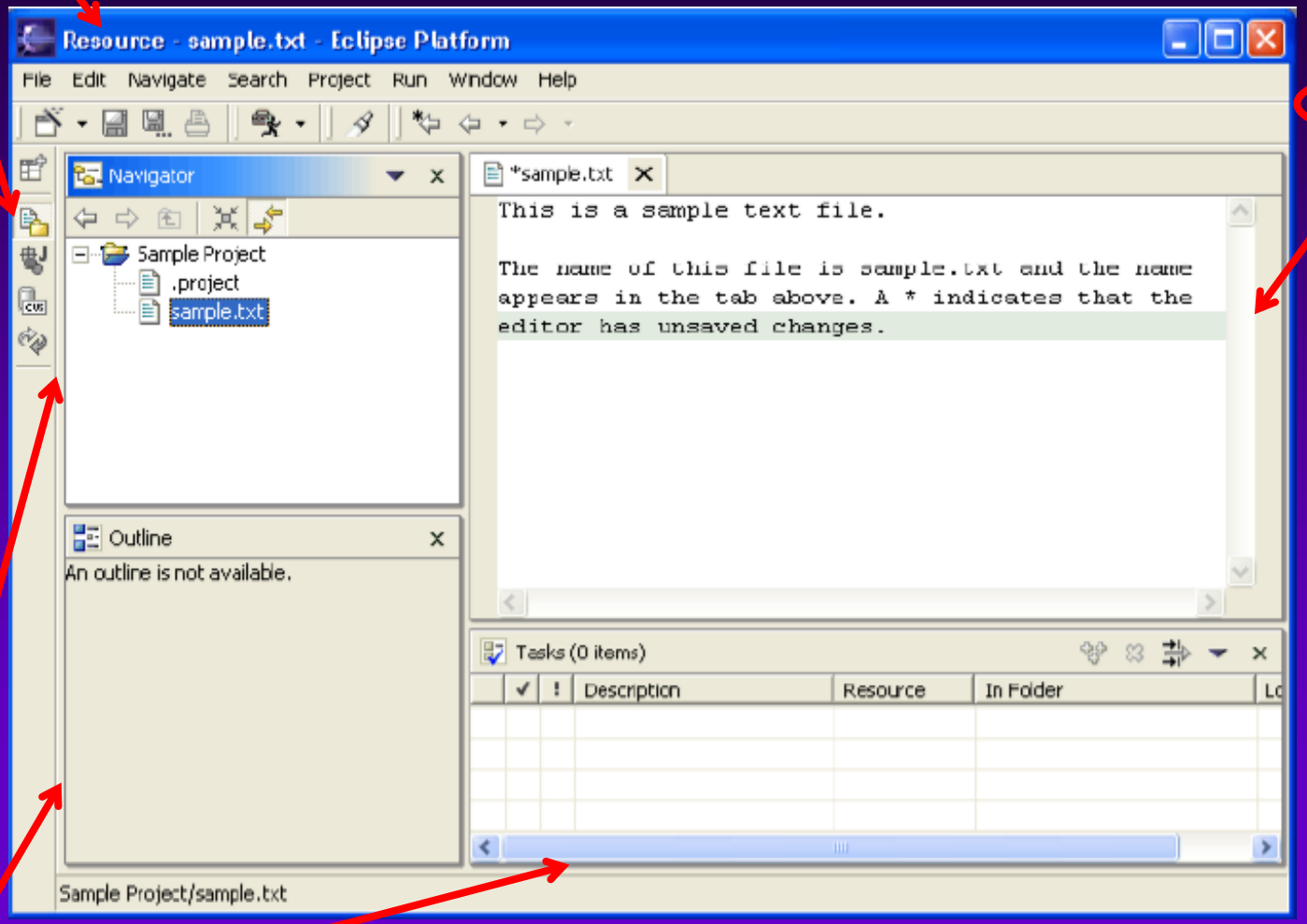
Workbench


Resource - Welcome - Eclipse Platform
File Edit Navigate Search Project Run Window Help
Navigator
Outline
An outline is not available.
Welcome
Eclipse Platform
This page will help familiarize you with the Eclipse Workbench.
To get started, read the sections below and click on the related links.
Installed Features
To find out more about the features installed in your workbench, choo
and select the feature you are interested in.
Tasks (0 items)
Description
Resource
In Folder

Workbench

- Editors
  - A visual component within the workbench
  - Used to edit or browse a resource
- Views
  - A visual component within the workbench
  - Used to navigate a hierarchy of information, open an editor or display properties of an active editor
- Perspectives
  - Groups of views an editors within the workbench

A simple example...

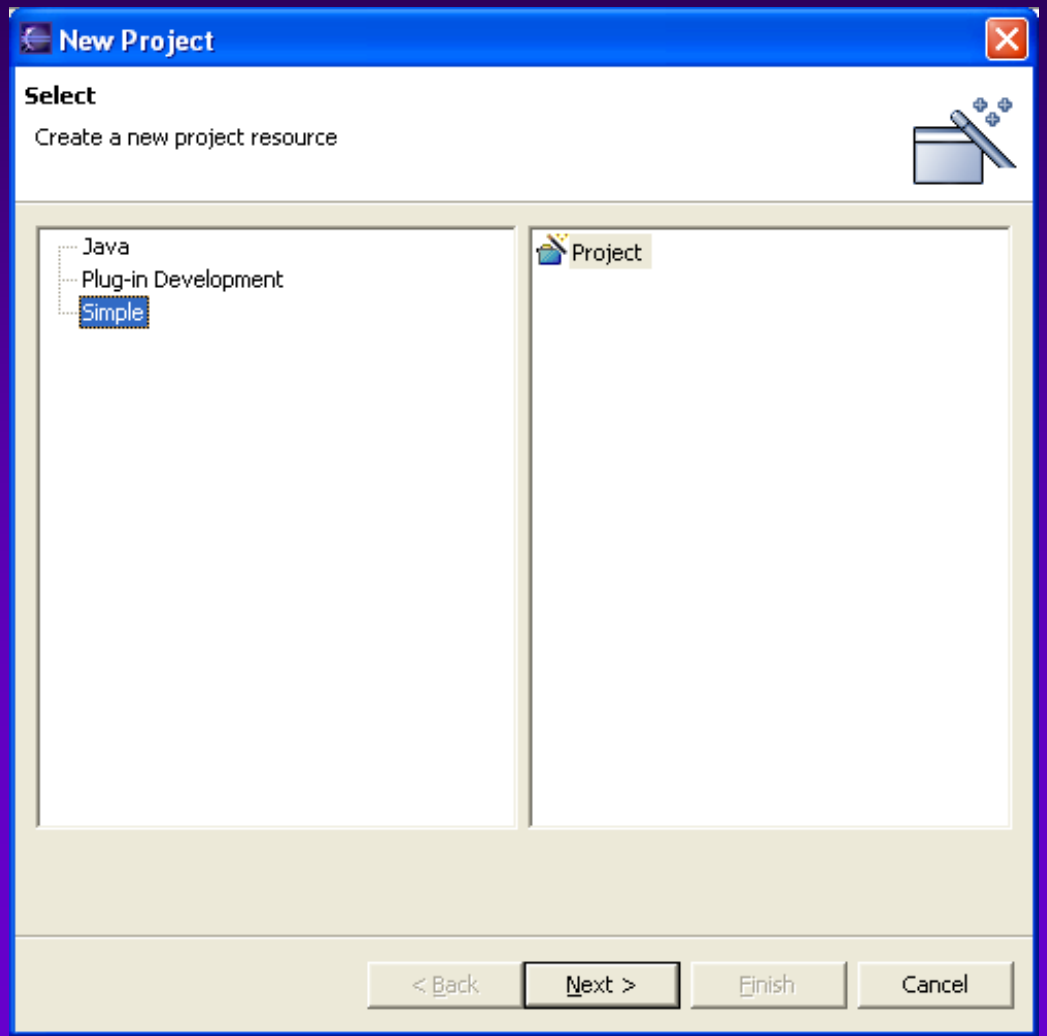


Creating resources

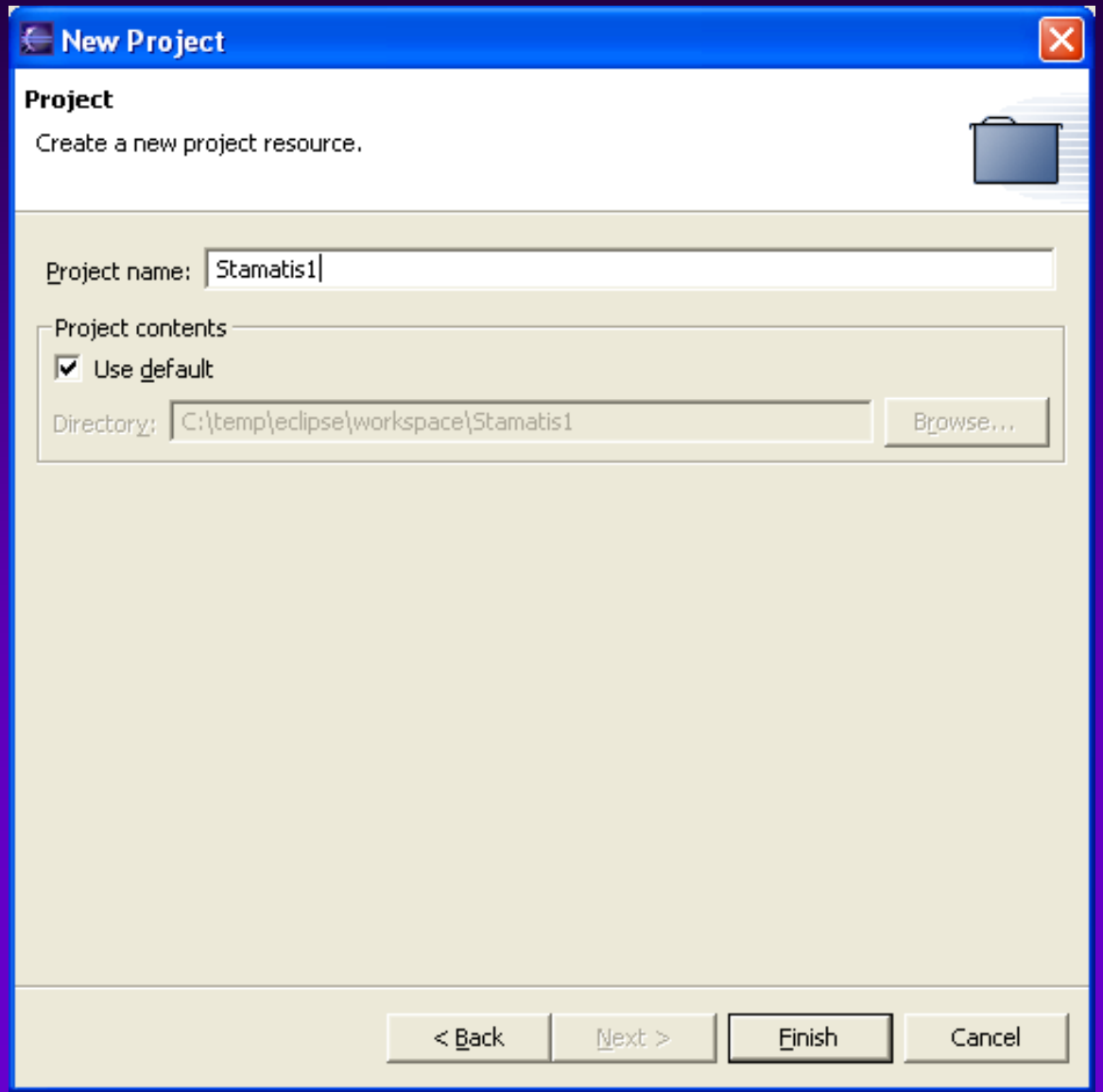
- Three alternatives
  1. File menu
  2. Navigator context menu (right-click on selected resource)
  3. New Wizard button

Creating a new project

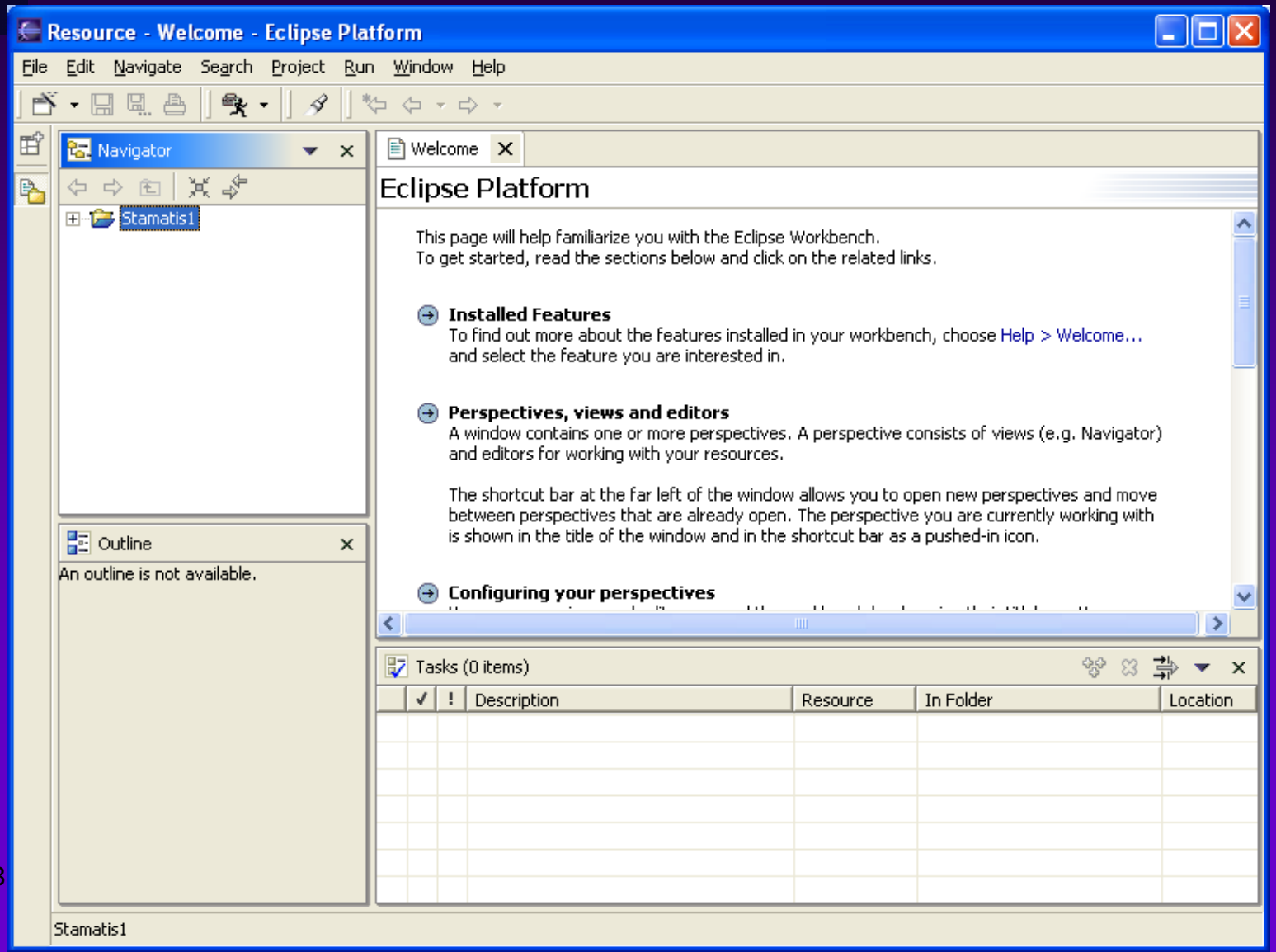
- From the menubar, select File > New > Project...



Creating a new project

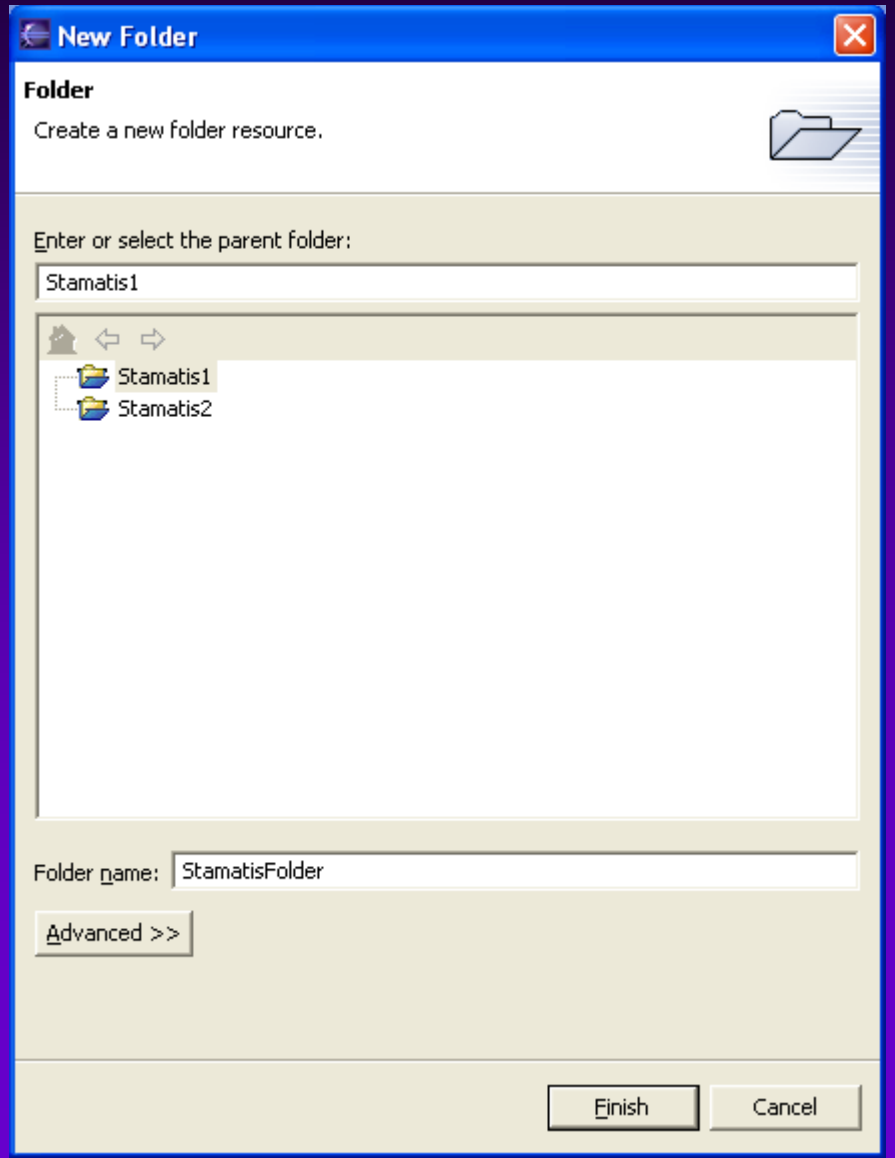


Creating a new project

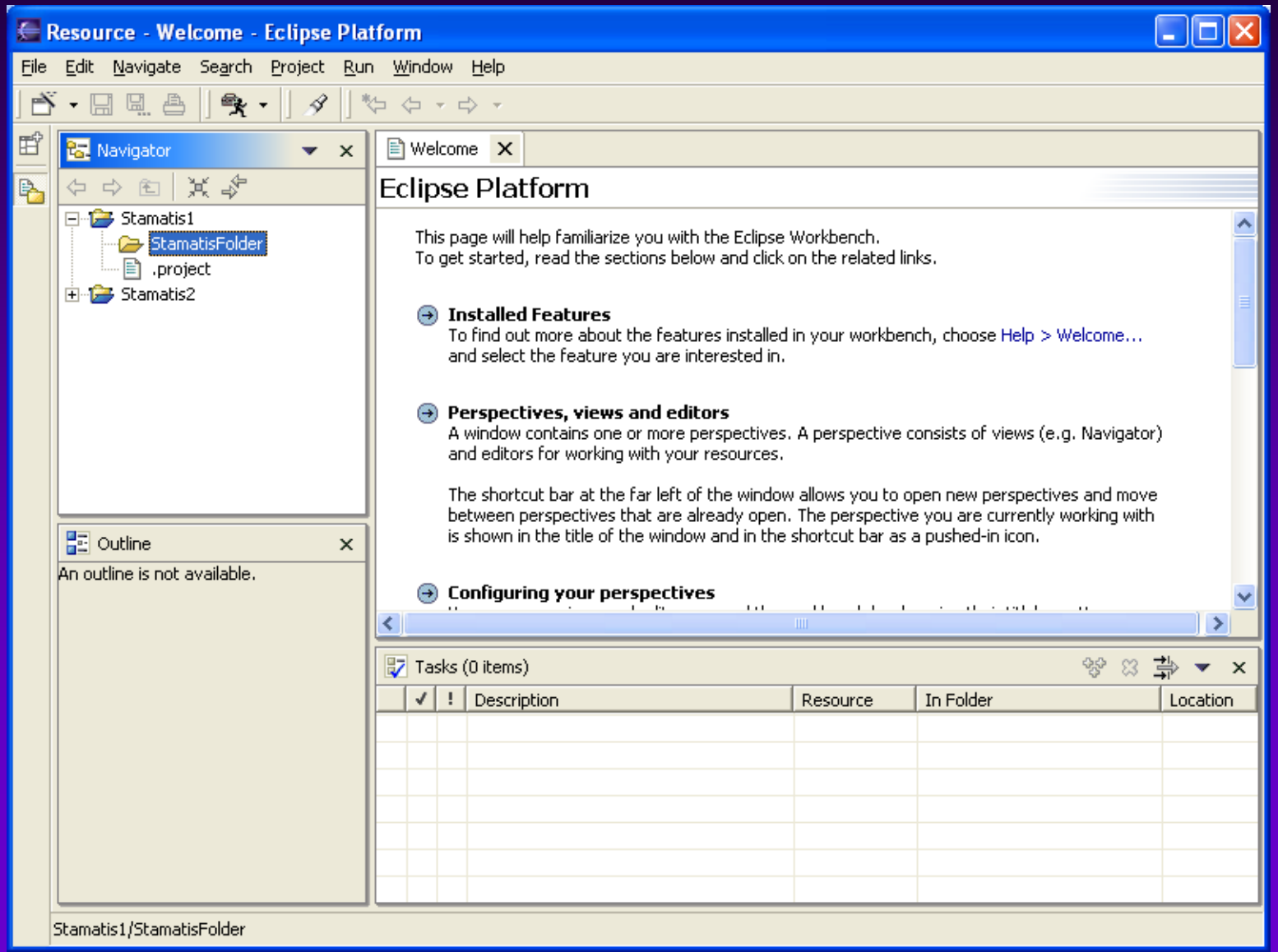


Creating a new folder

- From the popup menu of a project in the navigation view, select New > Folder

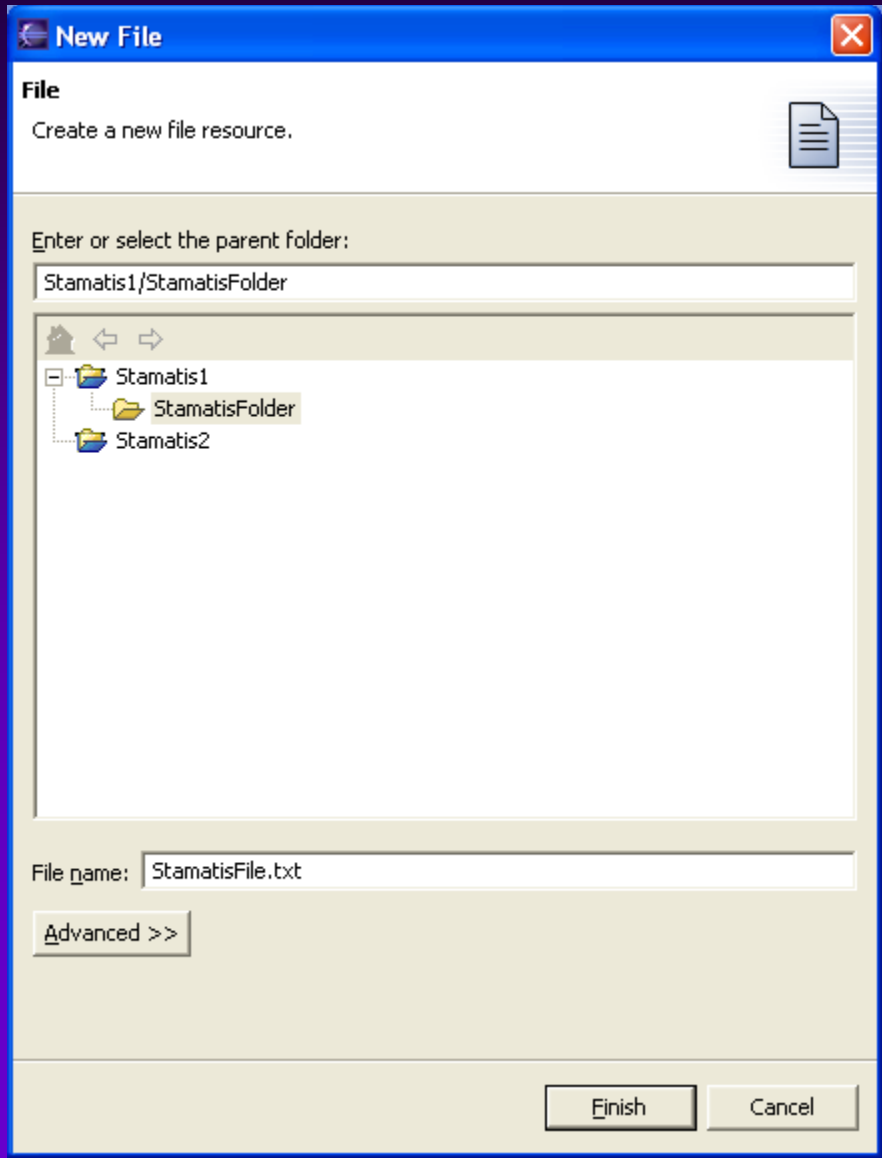


Creating a new folder

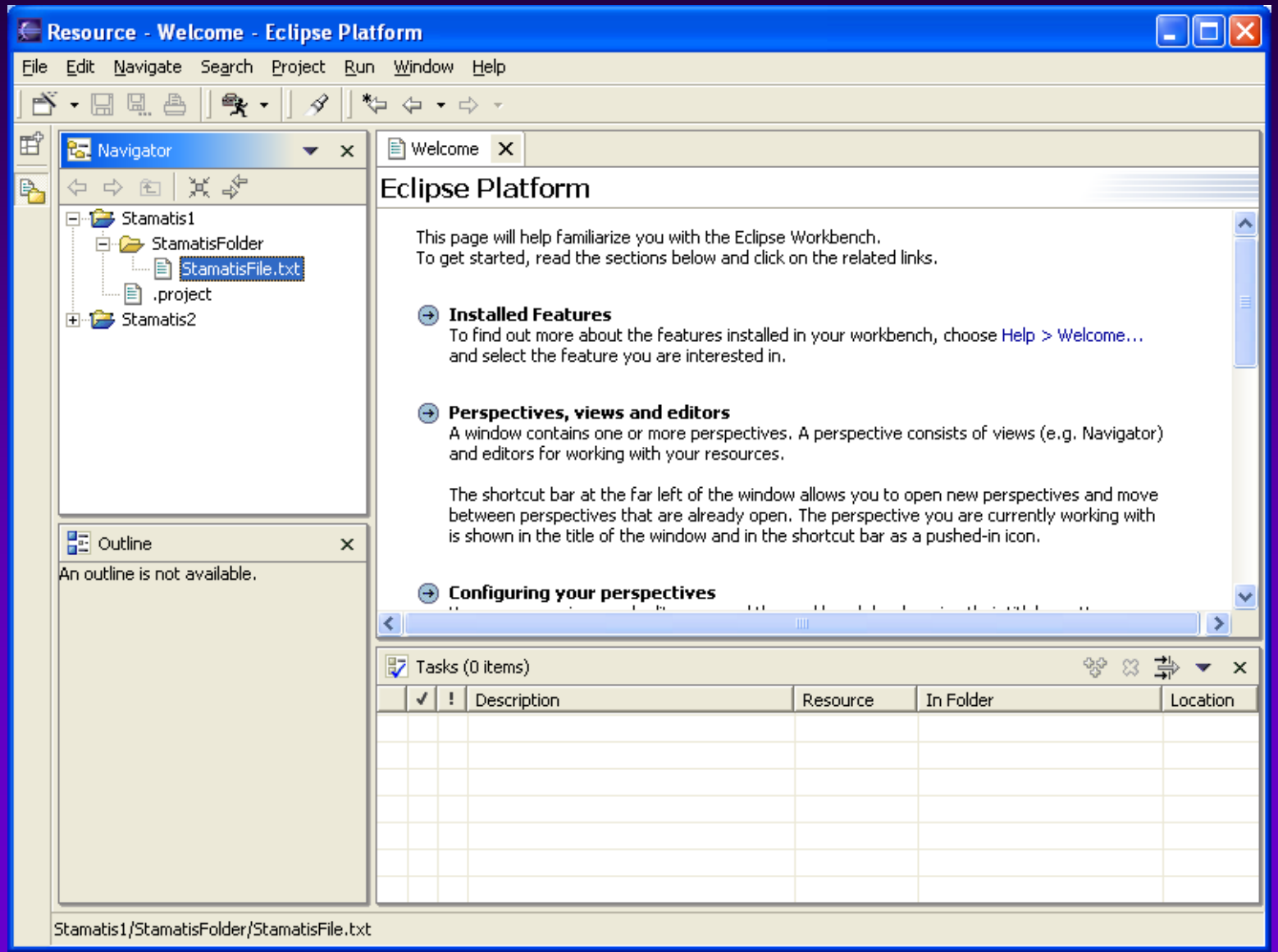


Creating a new file

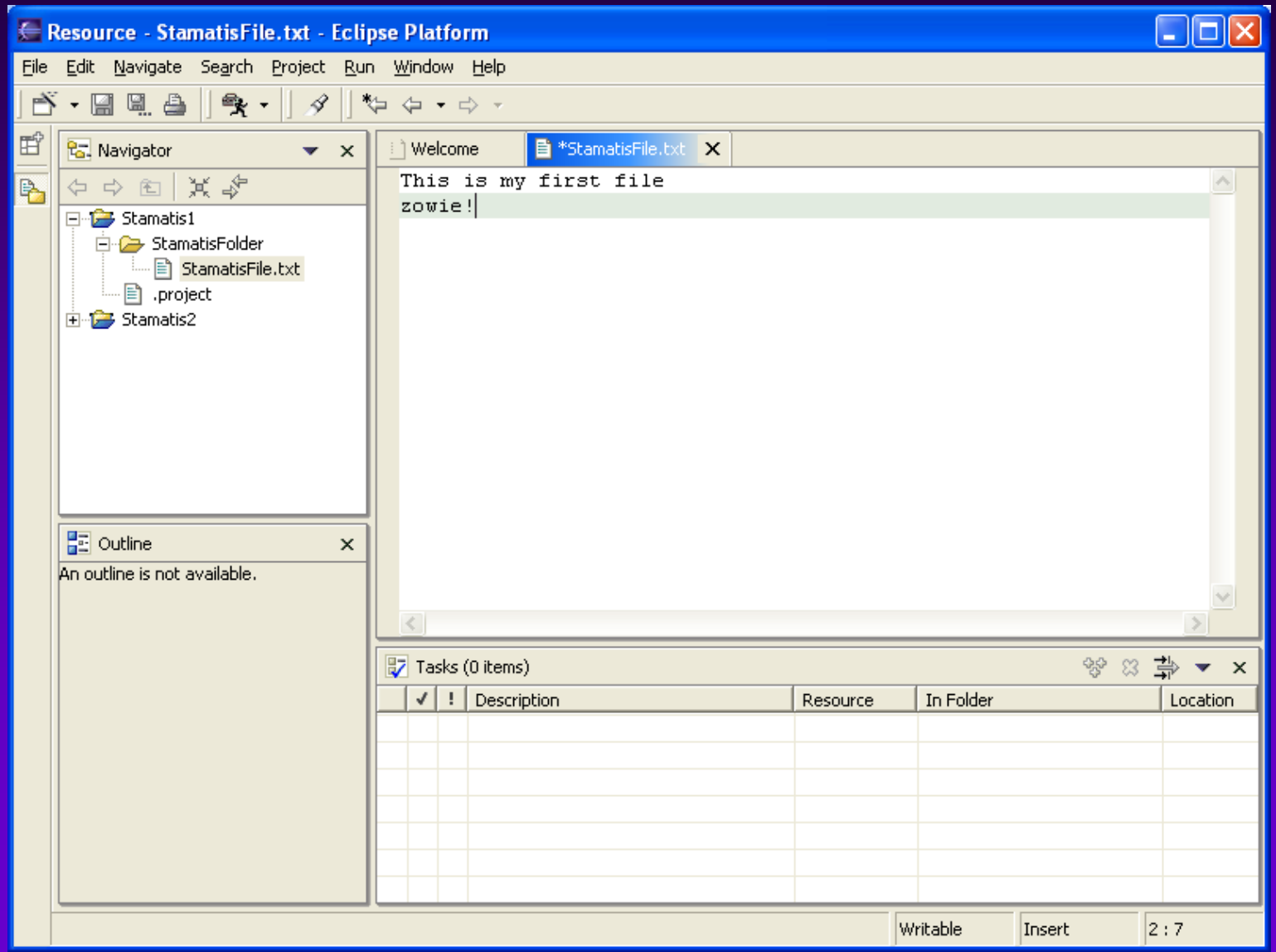
- From the toolbar use the “New” Wizard button and select “File”



Creating a new file



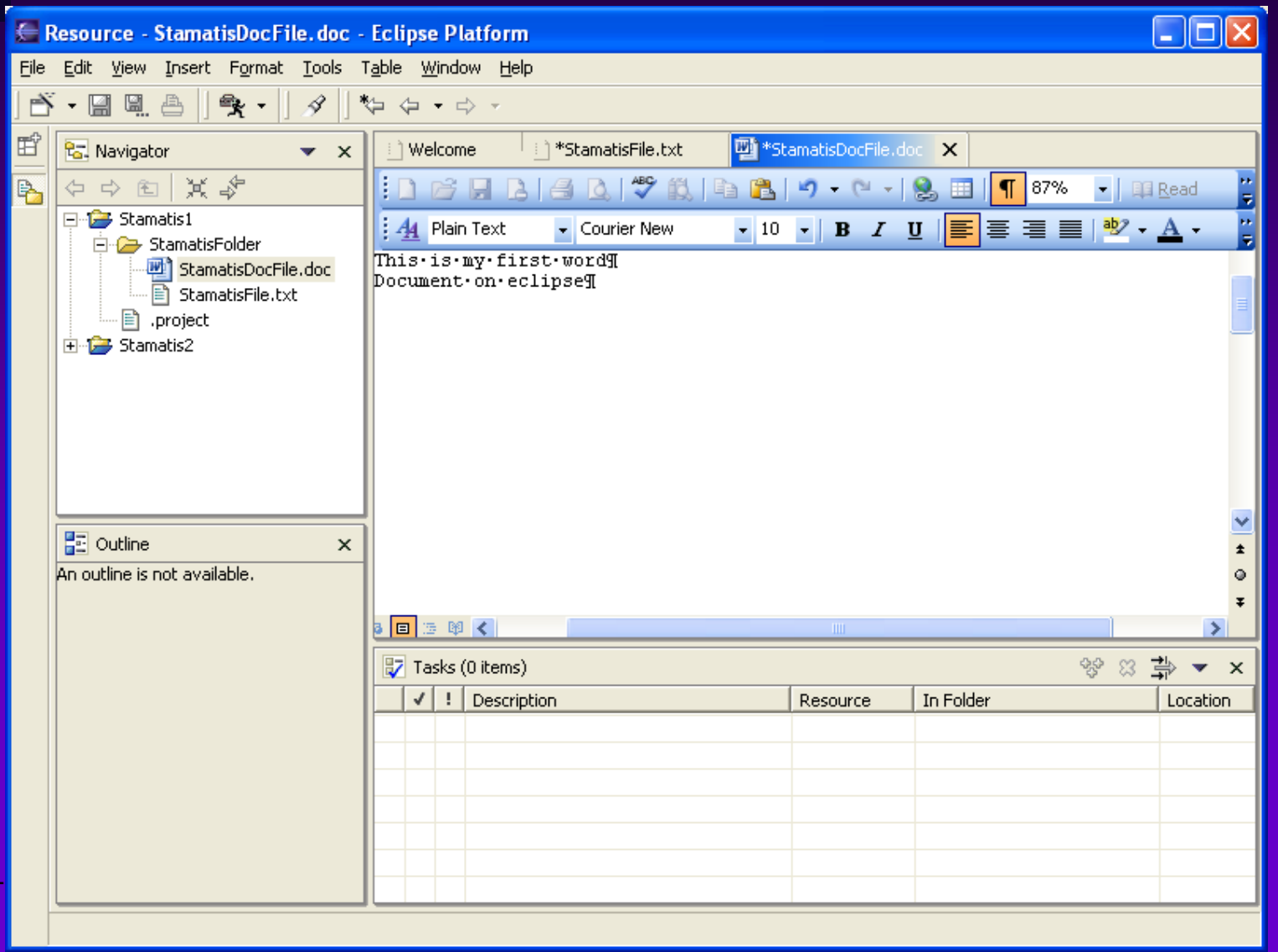
Creating a new file



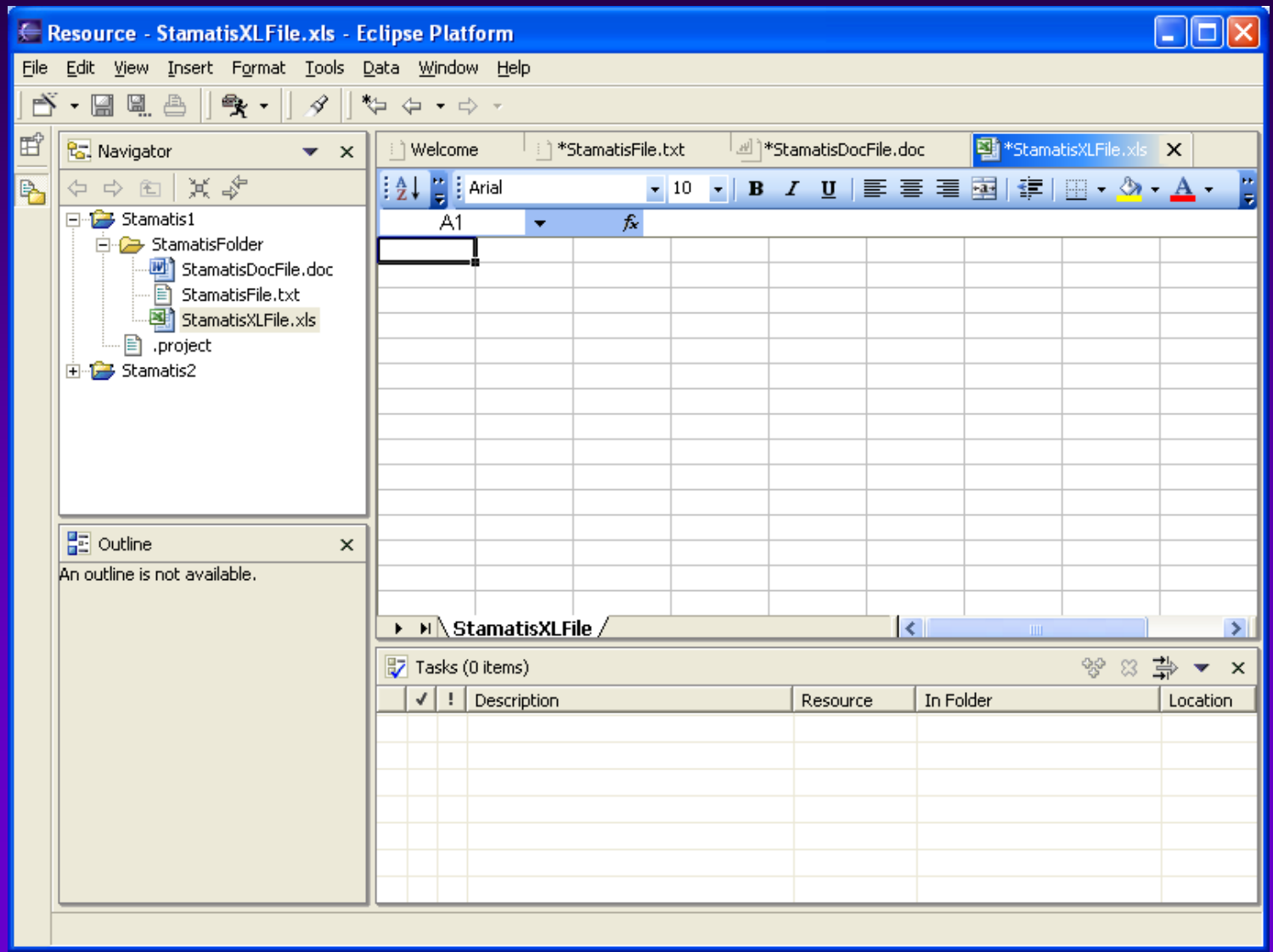
Editors

- Editors are tiled using tabs (if there is more than one file opened)
- The appropriate editor is used according to the type of file

Editors

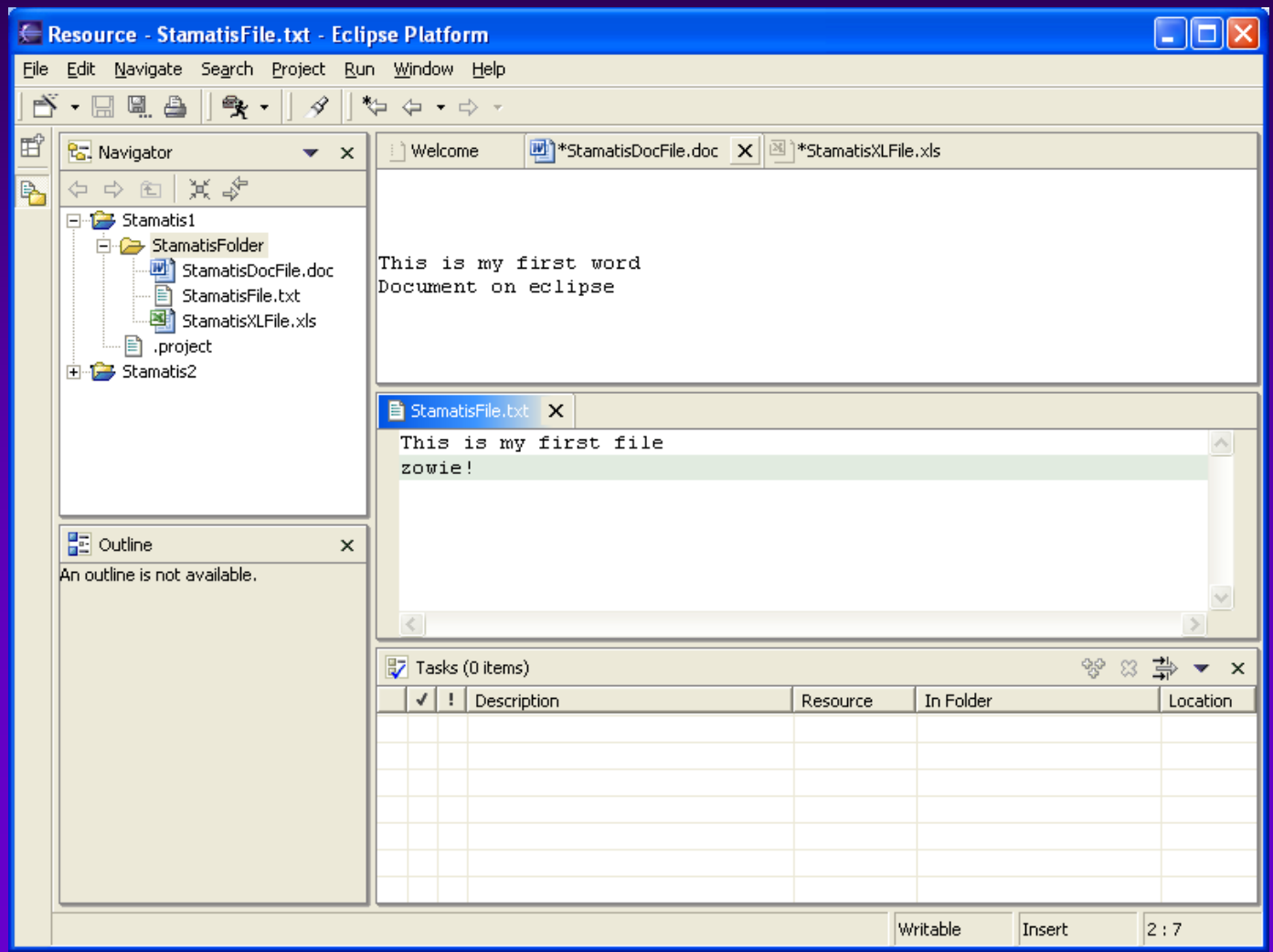

Resource - StamatisDocFile.doc - Eclipse Platform
File Edit View Insert Format Tools Table Window Help
Navigator
Stamatis1
StamatisFolder
StamatisDocFile.doc
StamatisFile.txt
.project
Stamatis2
Outline
An outline is not available.
Welcome
*StamatisFile.txt
*StamatisDocFile.doc
87%
Read
Plain Text
Courier New
10
This·is·my·first·word¶
Document·on·eclipse¶
Tasks (0 items)
Description
Resource
In Folder
Location

Editors


Resource - StamatisXLFile.xls - Eclipse Platform
File Edit View Insert Format Tools Data Window Help
Navigator
Stamatis1
StamatisFolder
StamatisDocFile.doc
StamatisFile.txt
StamatisXLFile.xls
.project
Stamatis2
Outline
An outline is not available.
Welcome
*StamatisFile.txt
*StamatisDocFile.doc
*StamatisXLFile.xls
Arial
10
A1
StamatisXLFile
Tasks (0 items)
Description
Resource
In Folder
Location

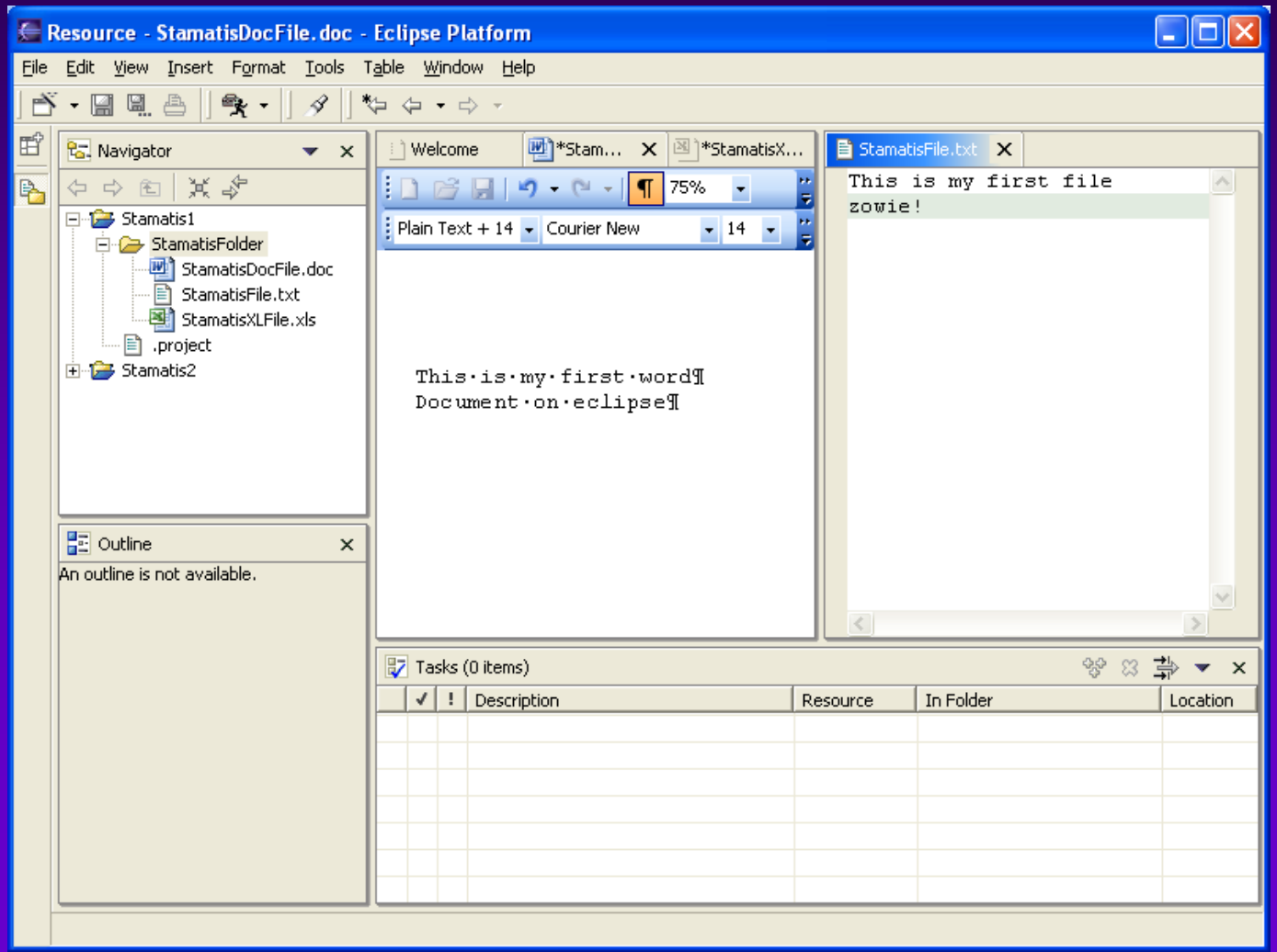
Stack editors

- Possible both side by side and one above the other



Stack editors

- Possible both side by side and one above the other

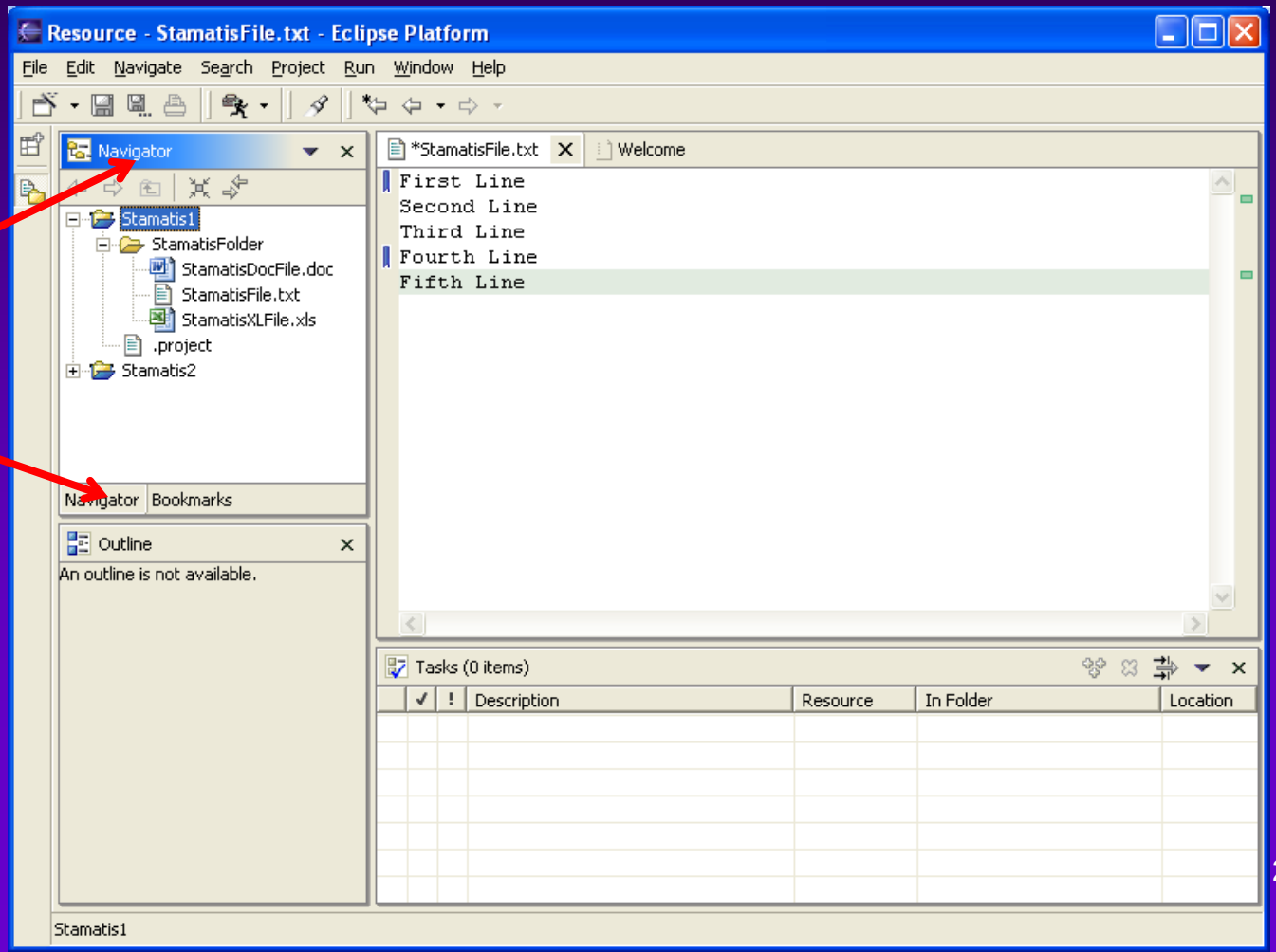


Views

- Views help you work with resources in addition to editors

Views (example)

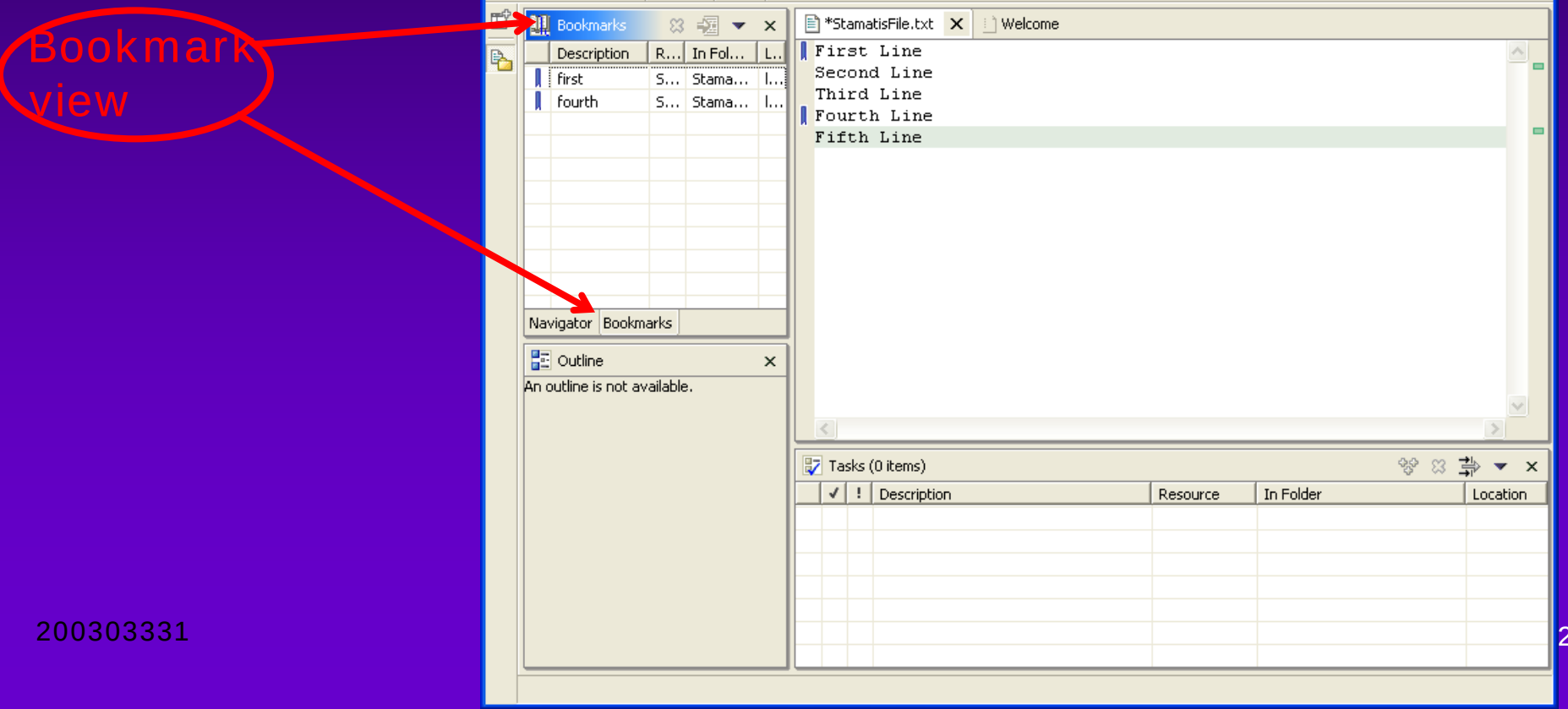
- The Navigator view displays the projects and other resources that you are working with



Navigator View

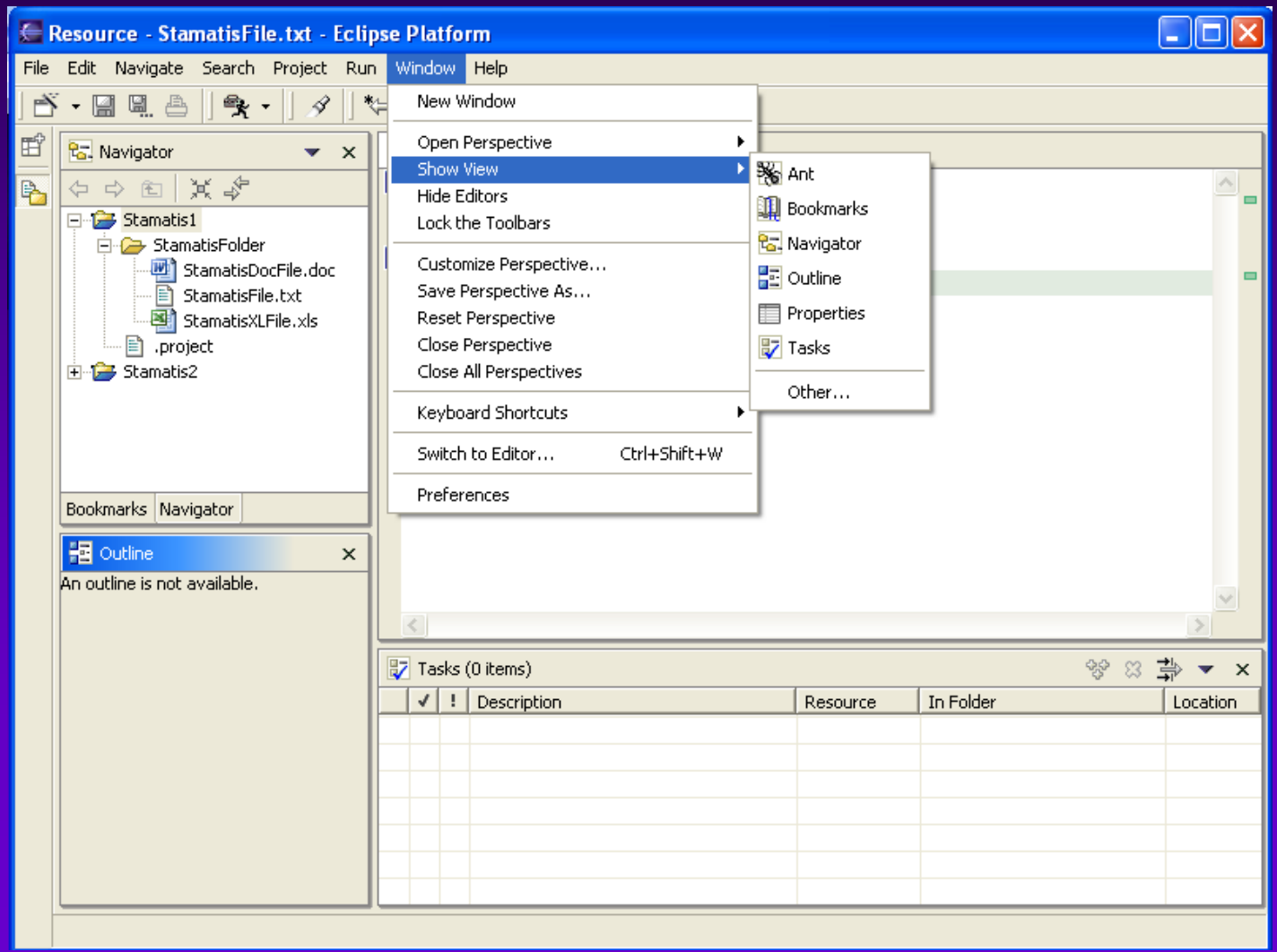
Views (example)

- The bookmark view displays all bookmarks in the workbench along with the name of the file the bookmark is associated



Views

- Display views from Window > Show View menu



Views

- Experiment as much as you want with the views
- Rearrange the Workbench using the Window > Reset Perspective menu

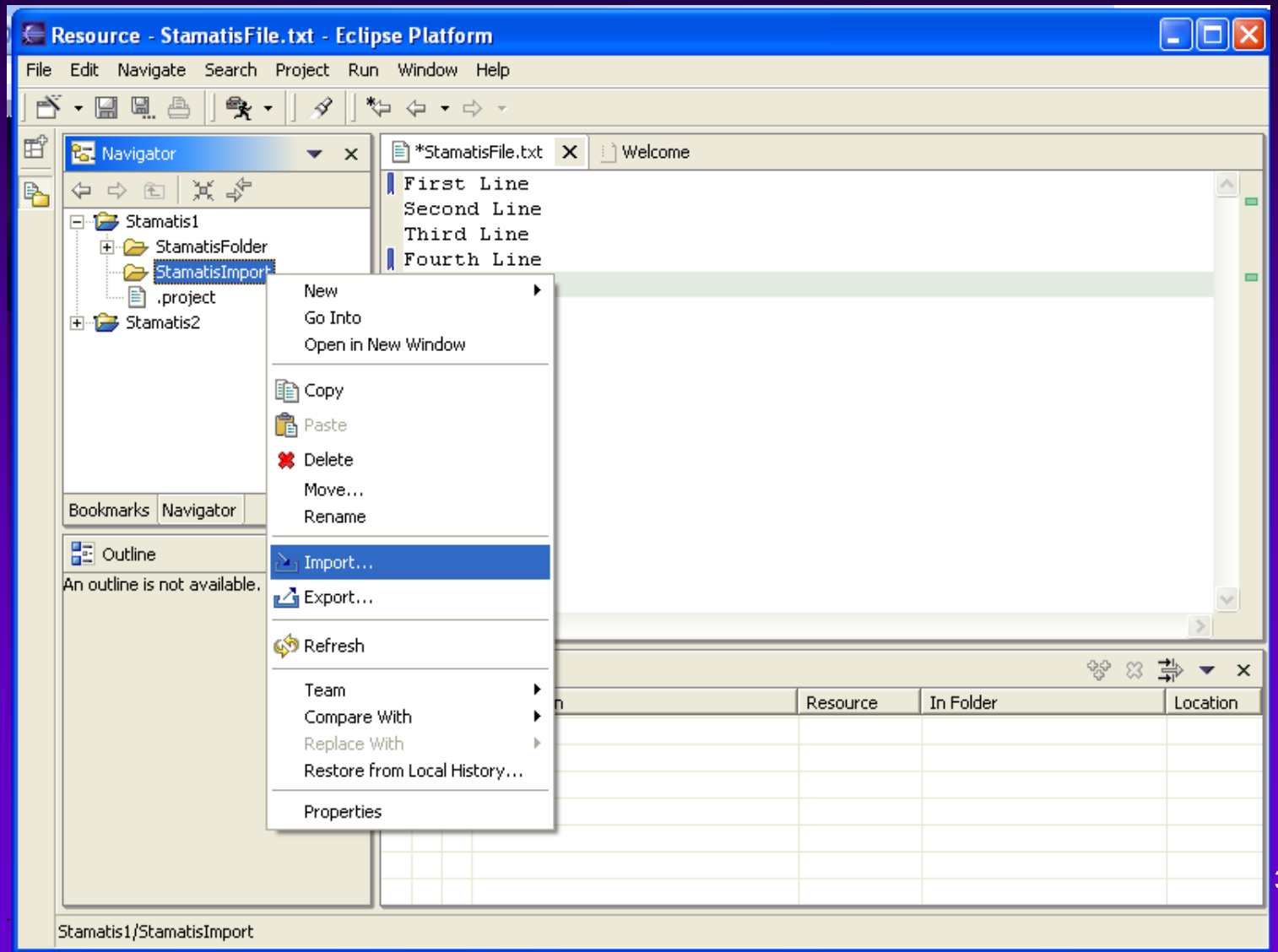
Show me the files...

- Q: Where are my files actually stored?
- A:
  1. Open your file system explorer
  2. Navigate to the location where you installed eclipse (%ECLIPSE_HOME%). You will find a sub-directory called “workspace”
  3. You will find the projects, folders and files in that sub-directory

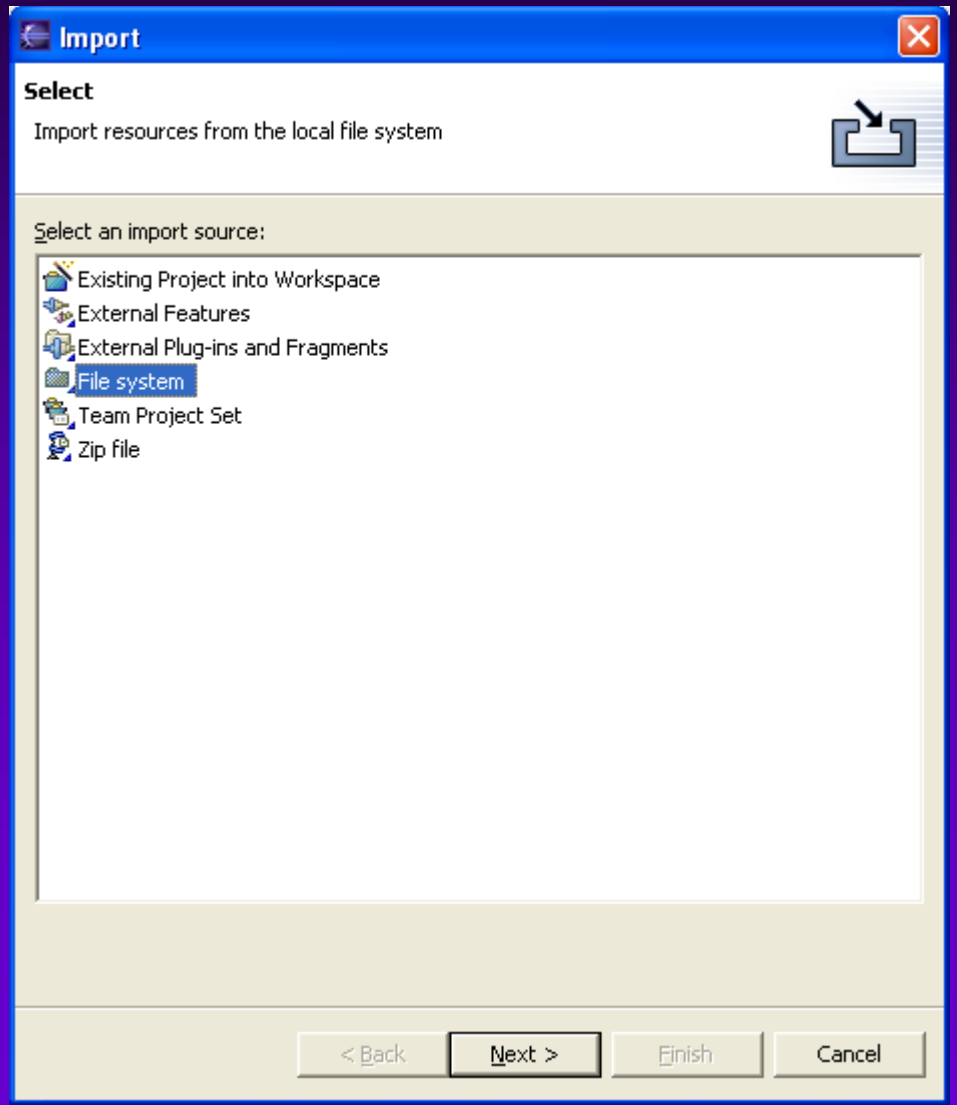
Import files

1. Drag and Drop files from the file system
2. Copy and Paste files from the file system
3. Use the import wizard either from the popup menu of the navigator or from the File menu

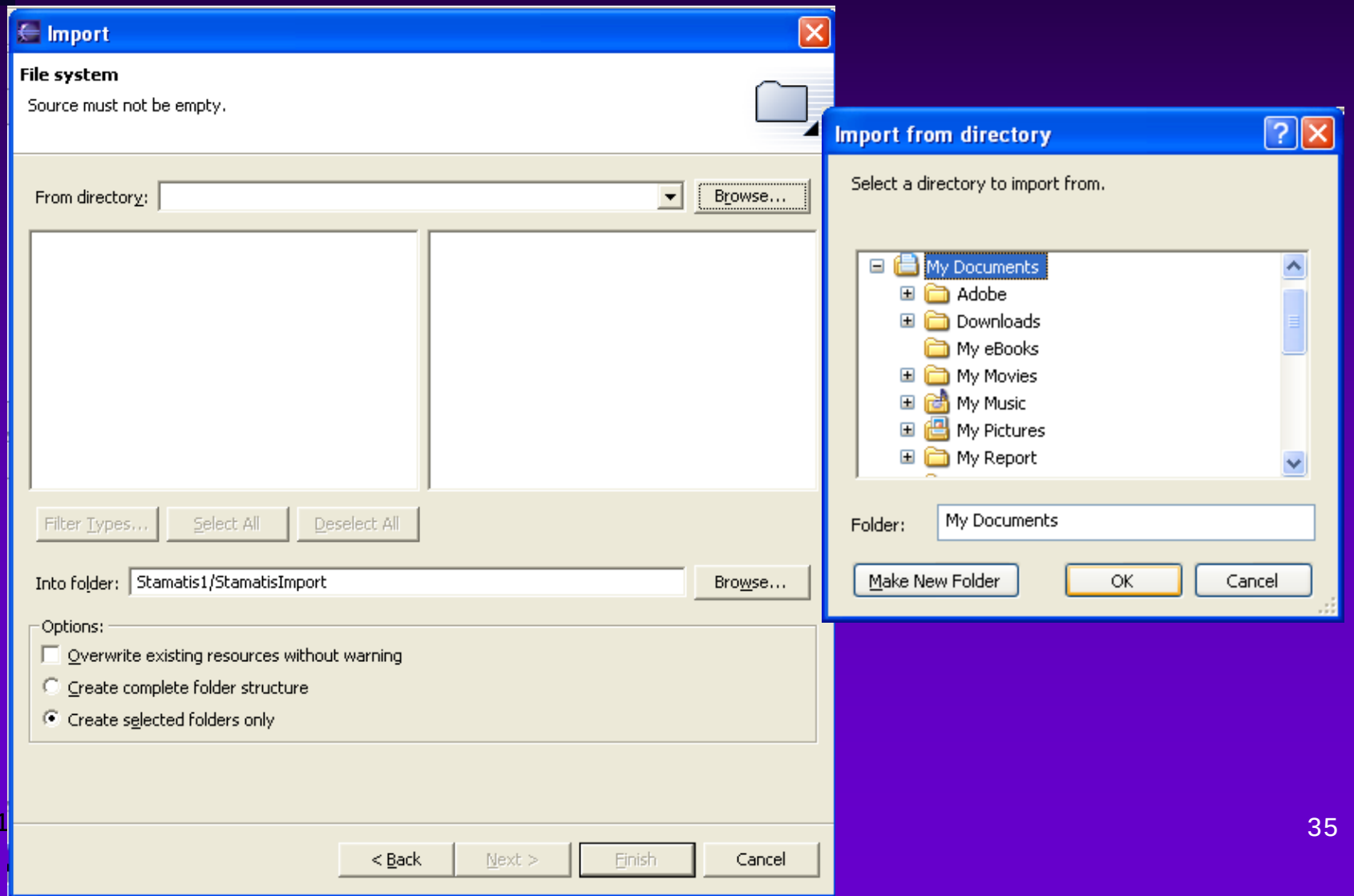
Import files



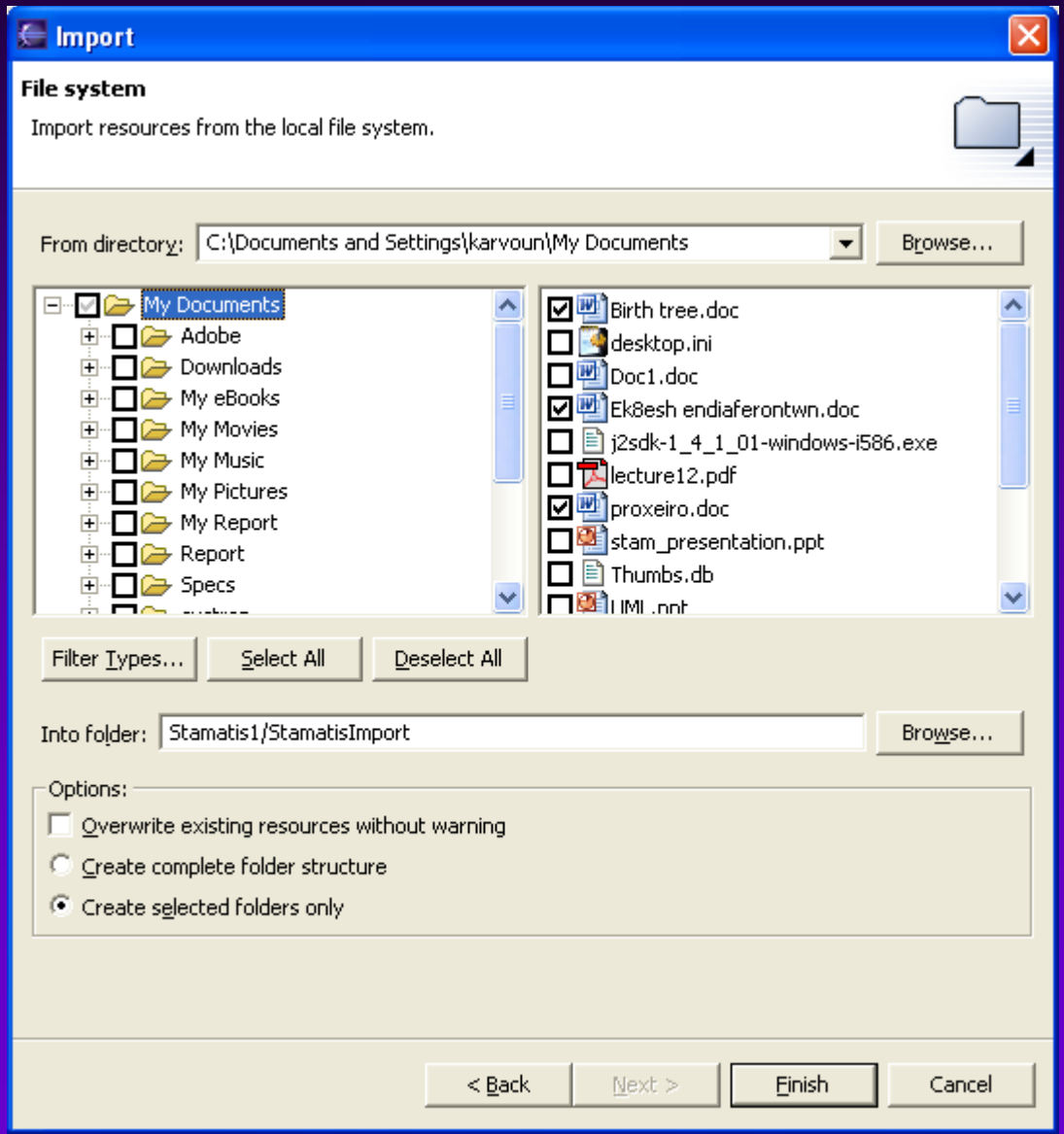
Import files



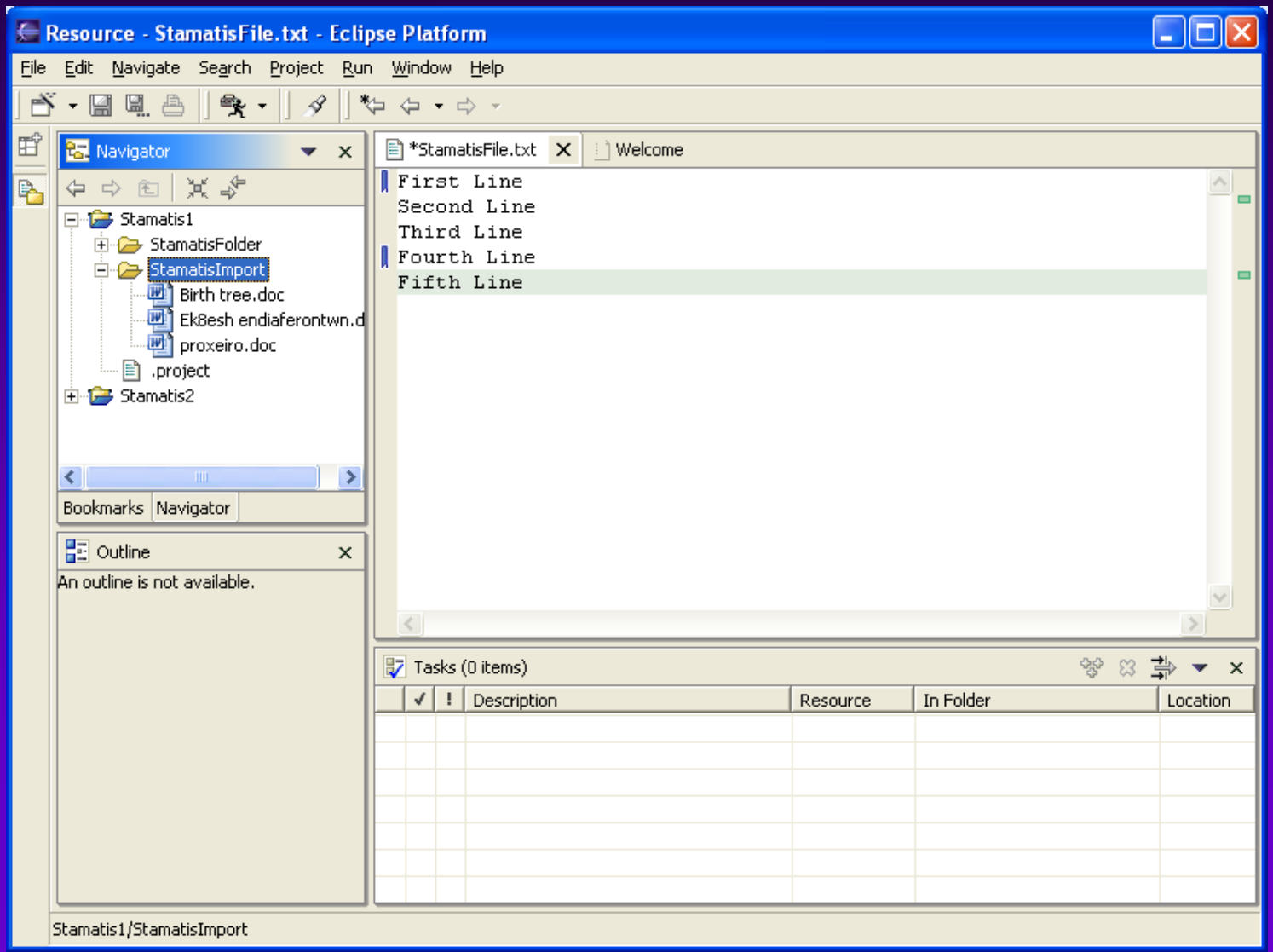
Import files



Import files



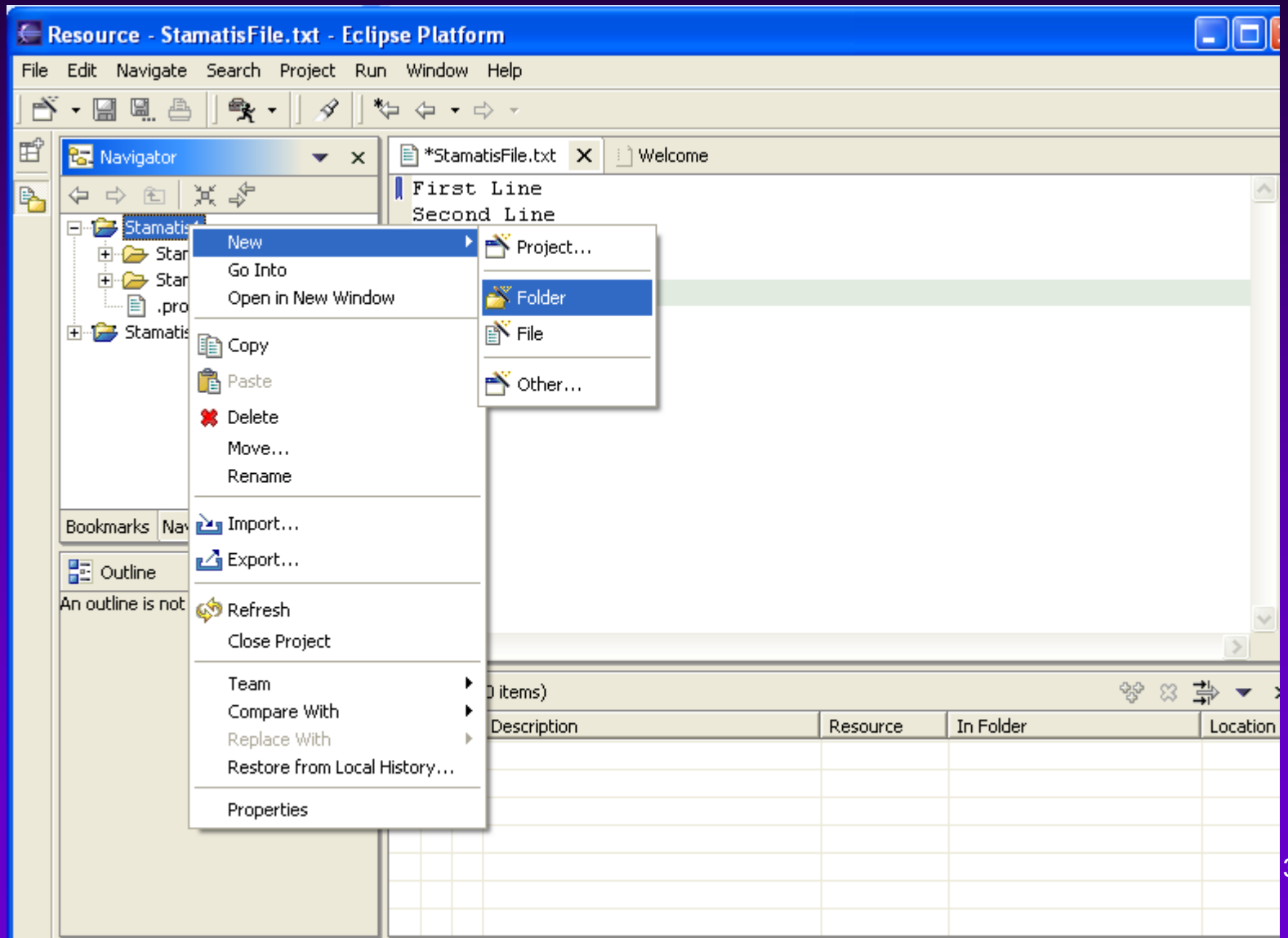
Import files



Link folder to file system

- If you do not want the contents of the folder to be copied in the “workspace” you have to “link” the folder to a position in the file system upon its creation

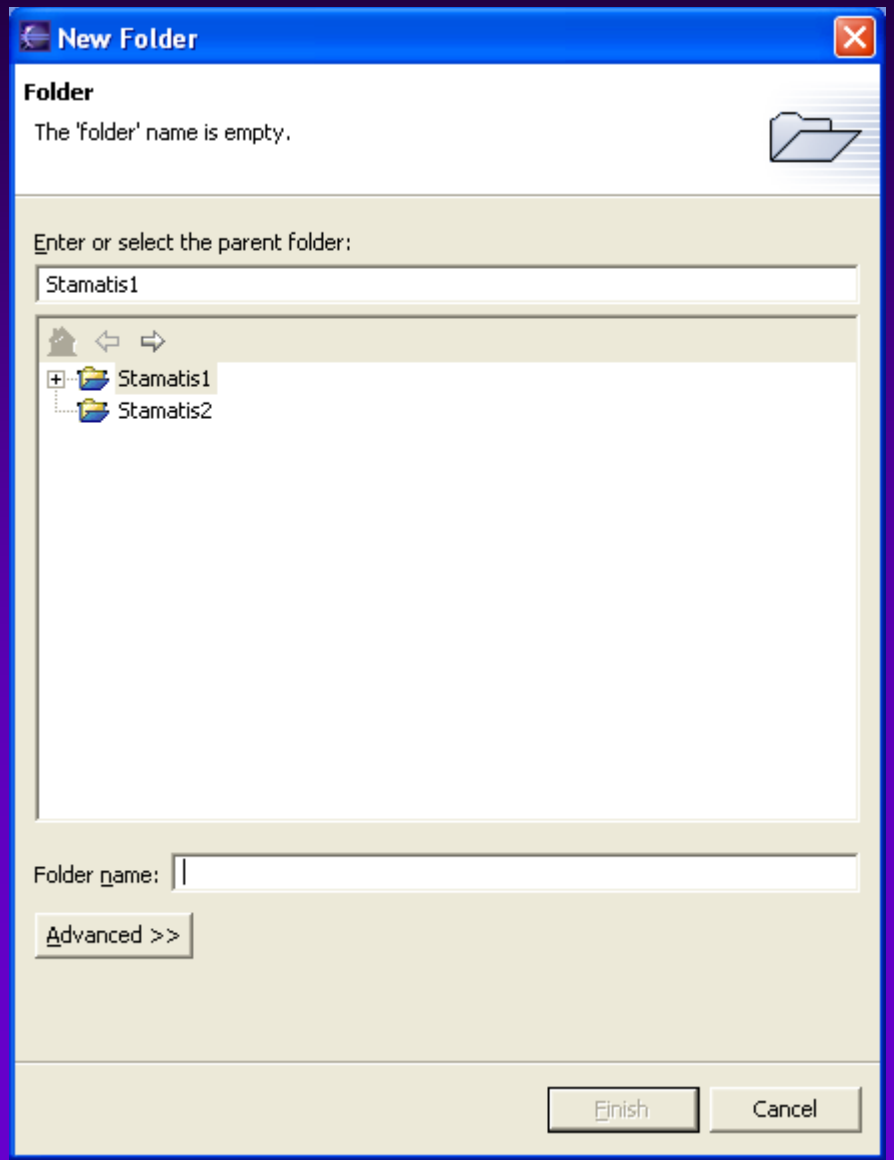
Link folder to file system



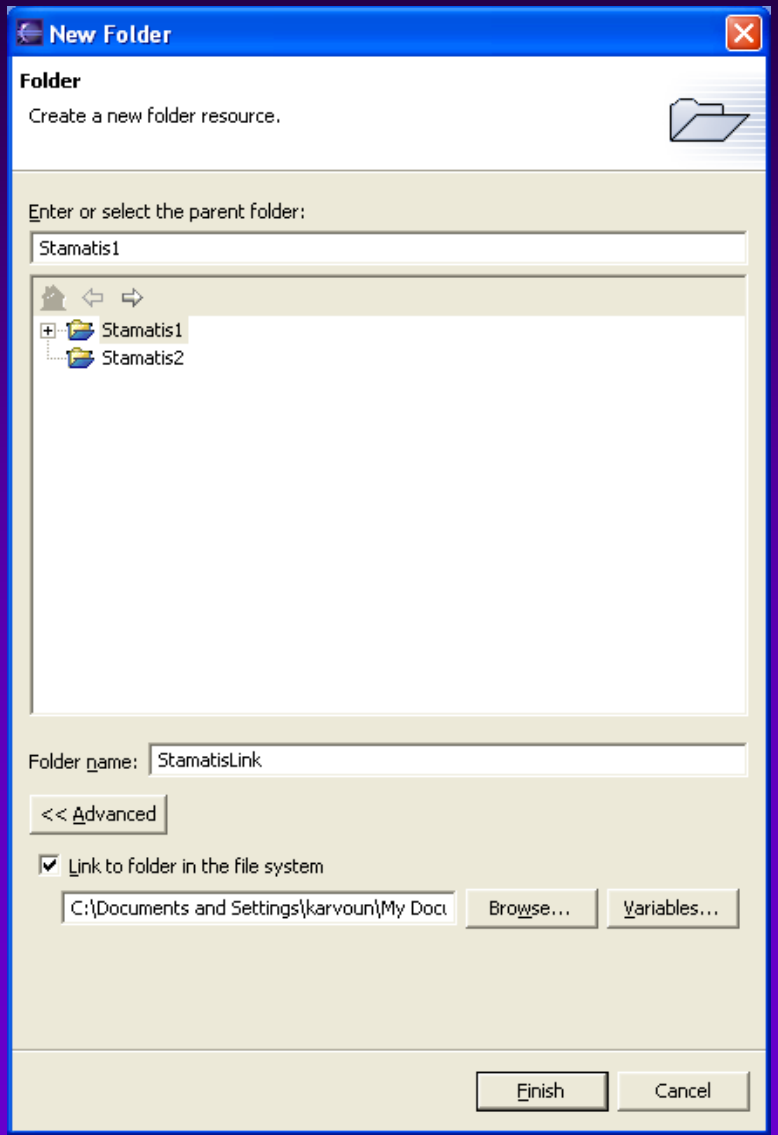
Link folder to file system

- Click “Advanced”
- Check “Link to folder in file system”
- Browse to find the desired folder

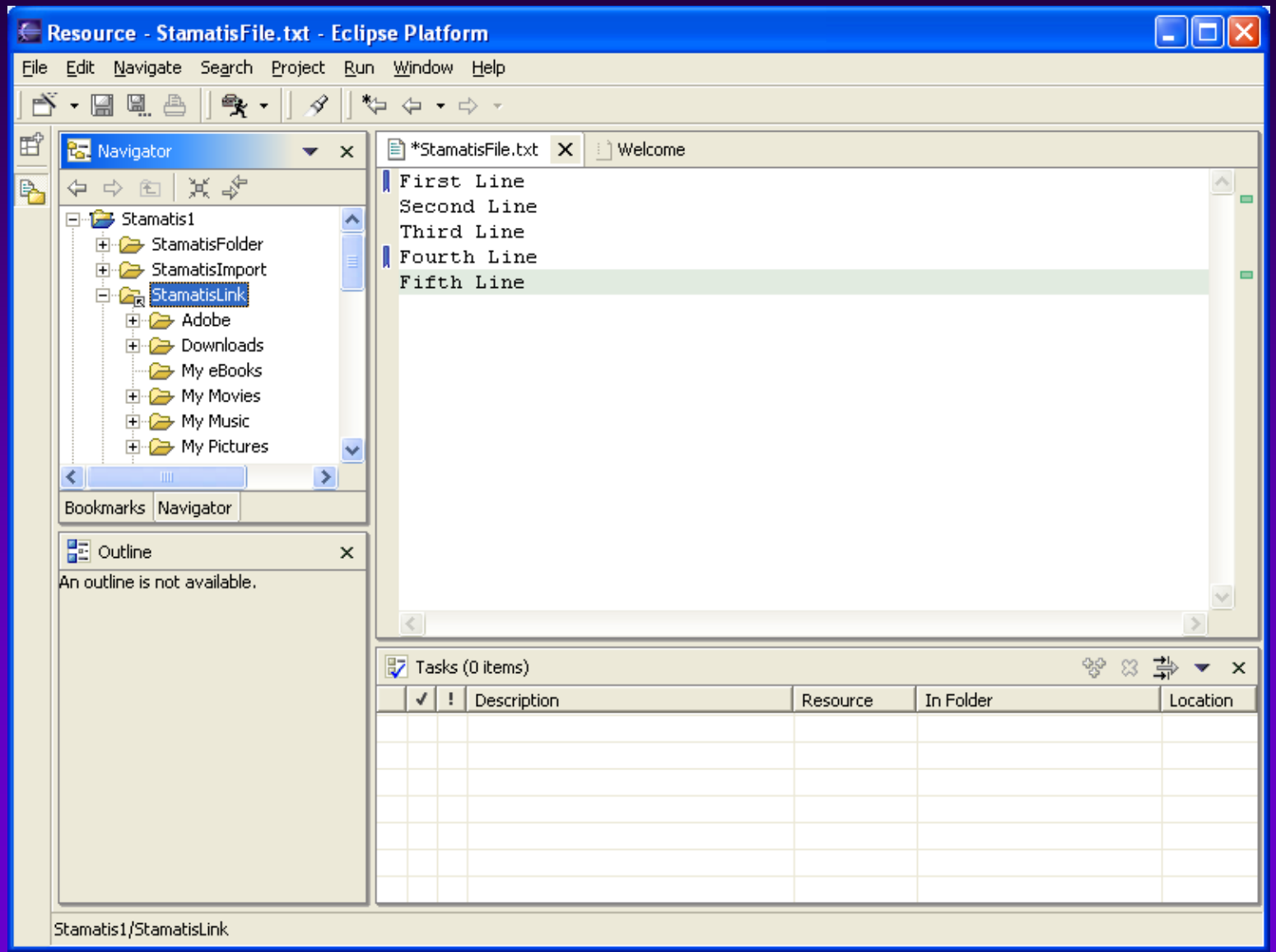
Link folder to file system



Link folder to file system



Link folder to file system



Link folder to file system

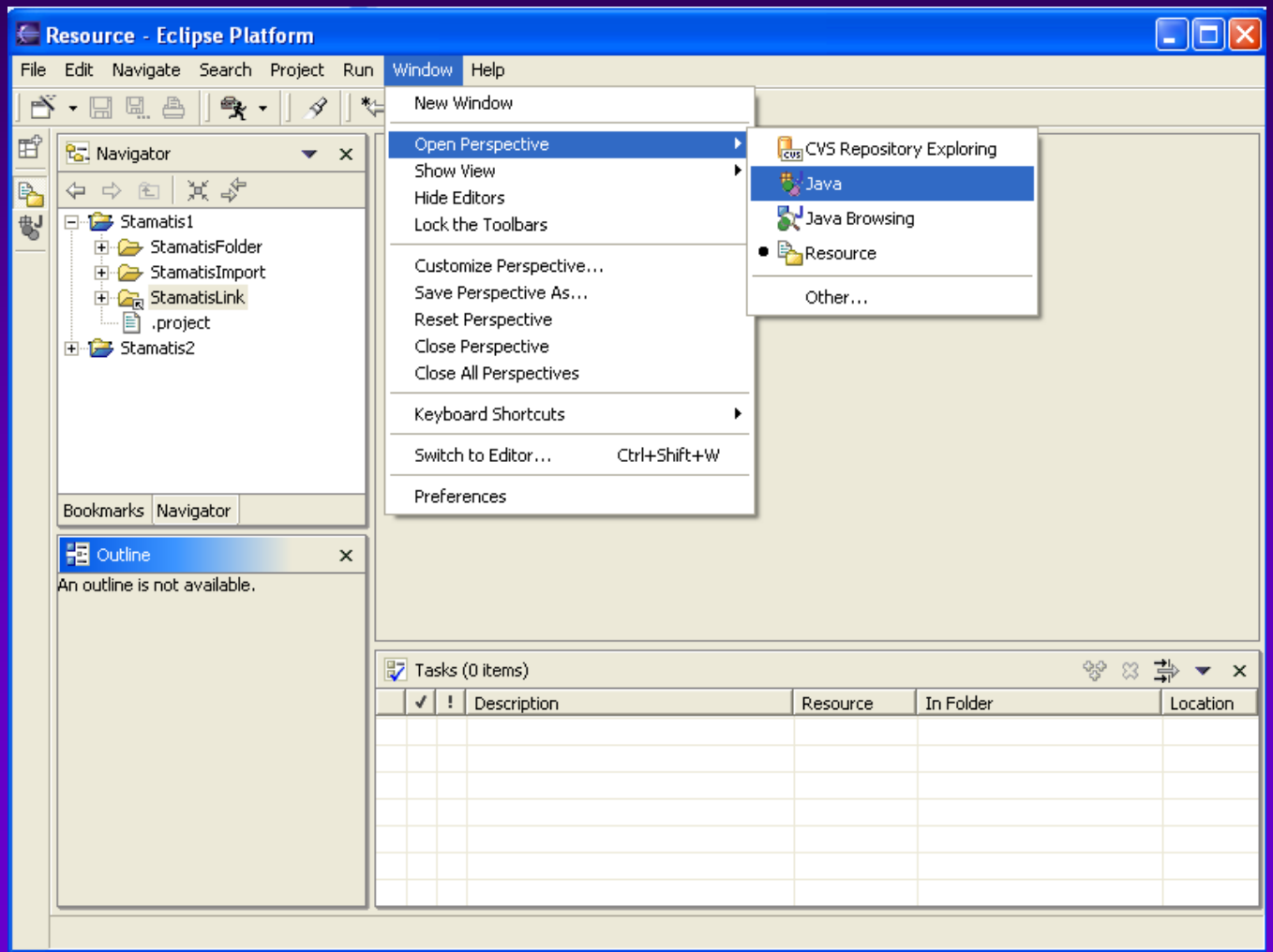
- If you sneak in the file system explorer you will see that there is no folder “StamatisLink” in the “workspace” folder. Any change will be automatically made to the resources linked in the file system.

Part3

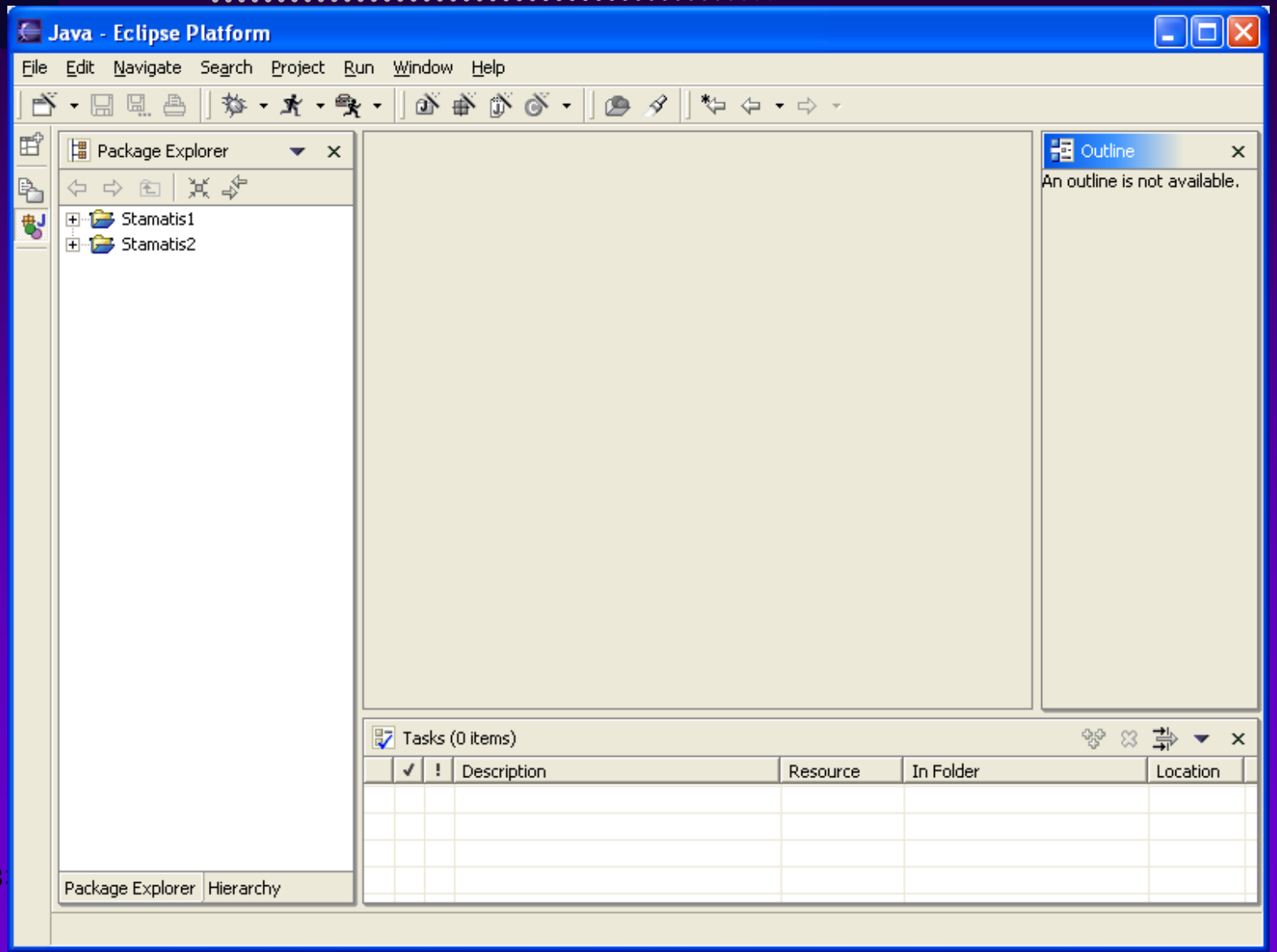
Java Development

Prepare Workbench

- Switch to Java Perspective using Window > Open Perspective > Java Menu



Prepare Workbench

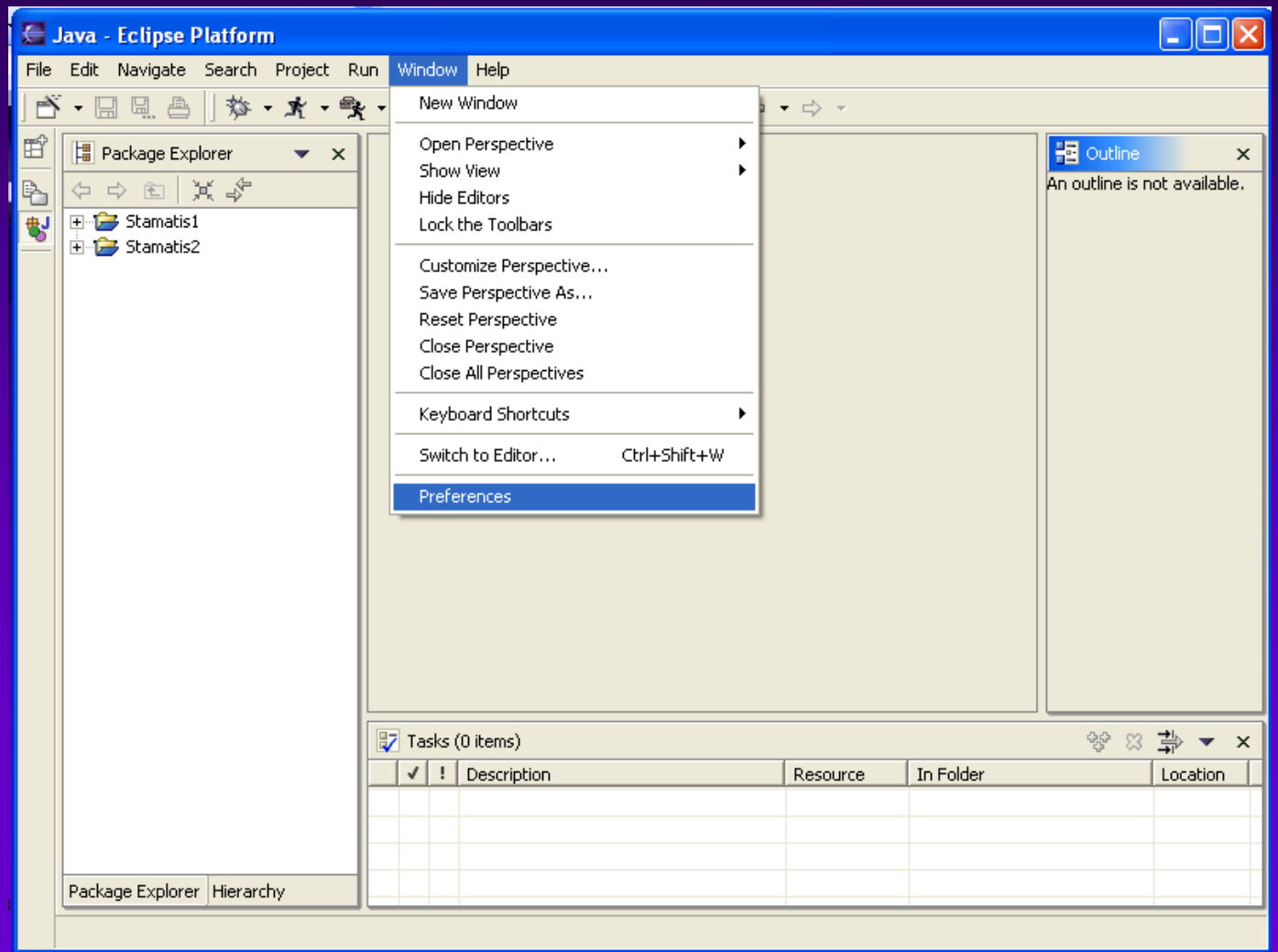


Prepare Workbench

- Java perspective has the following views
  - Package Explorer
  - Hierarchy
  - Outline
  - Search
  - Console
  - Tasks

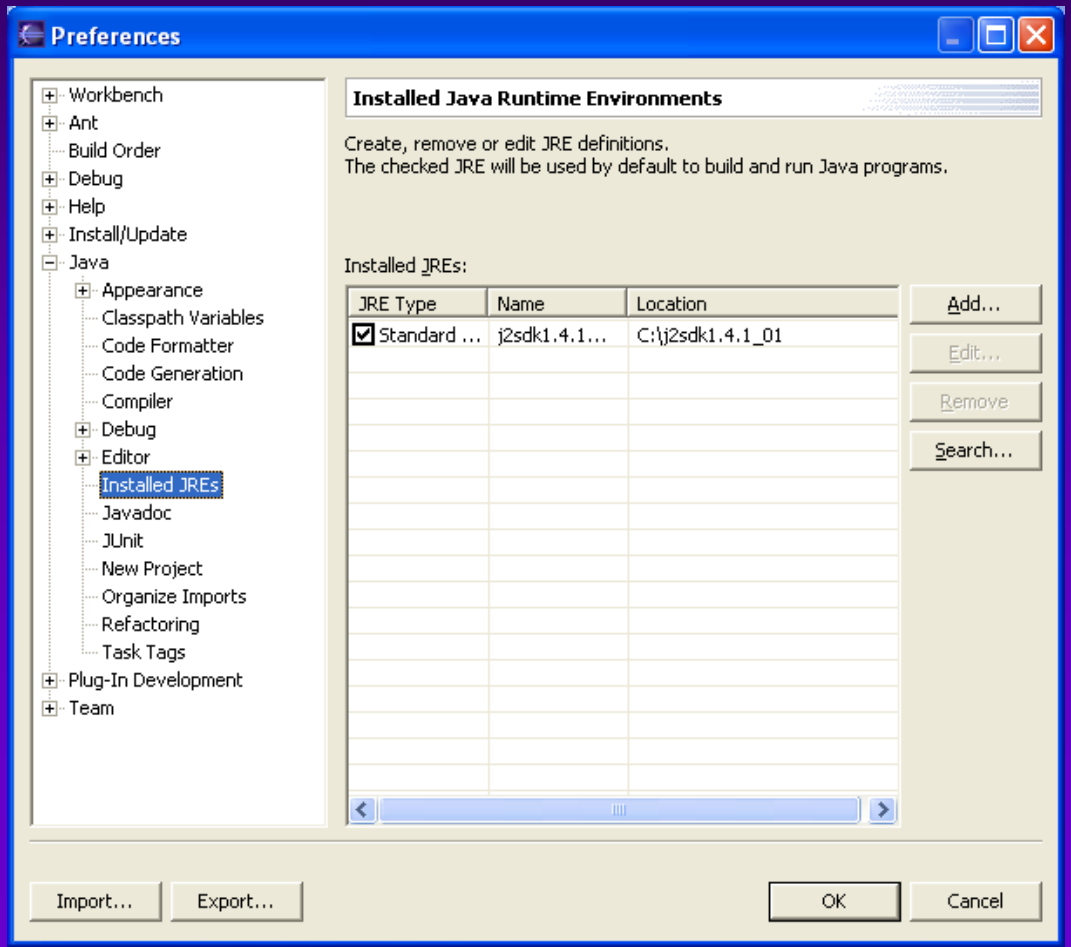
Prepare Workbench

- Verify JRE installation and classpath variables
- Select menu Window > Preferences



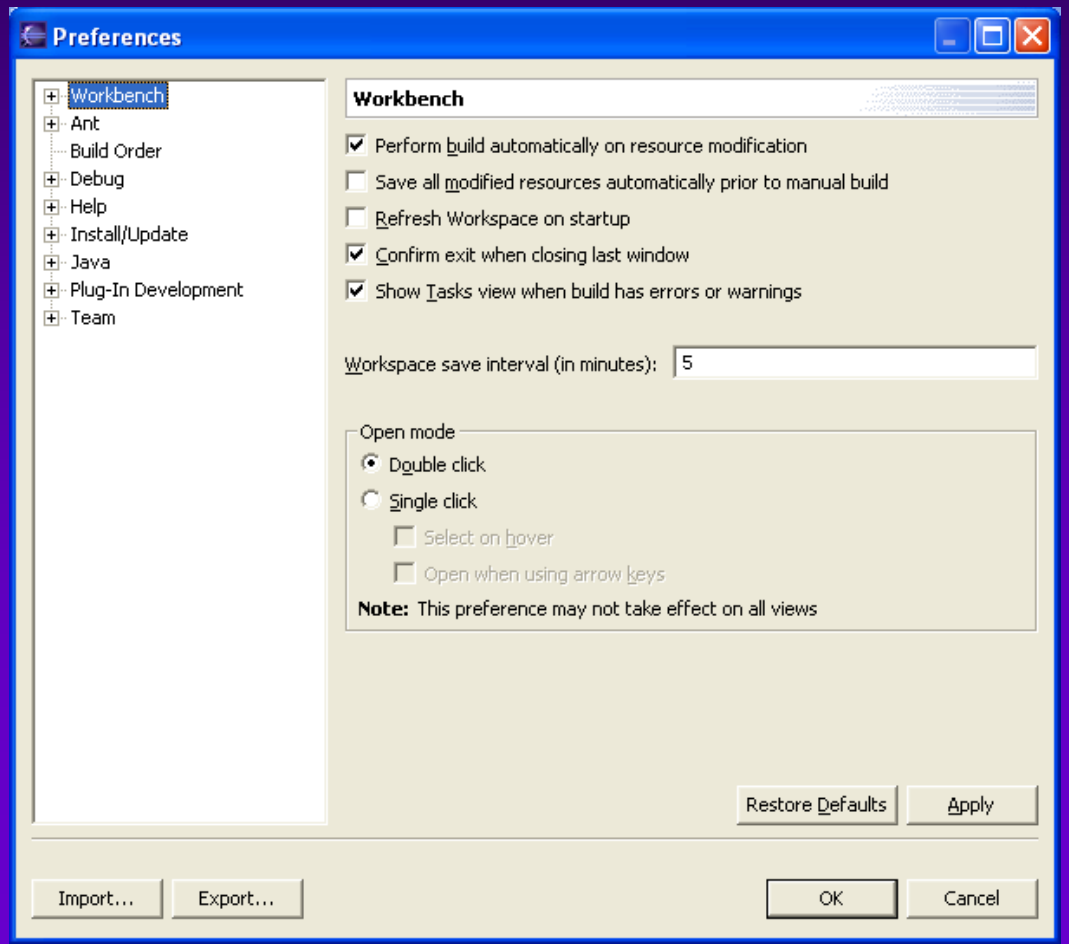
Prepare Workbench

- Select Java > Installed JREs
- Confirm that a JRE has been automatically detected.
- The use of an SDK rather than a JRE is recommended



Prepare Workbench

- Select “Workbench”
- Make sure “Perform build automatically on resource modification” is checked



Code Organization

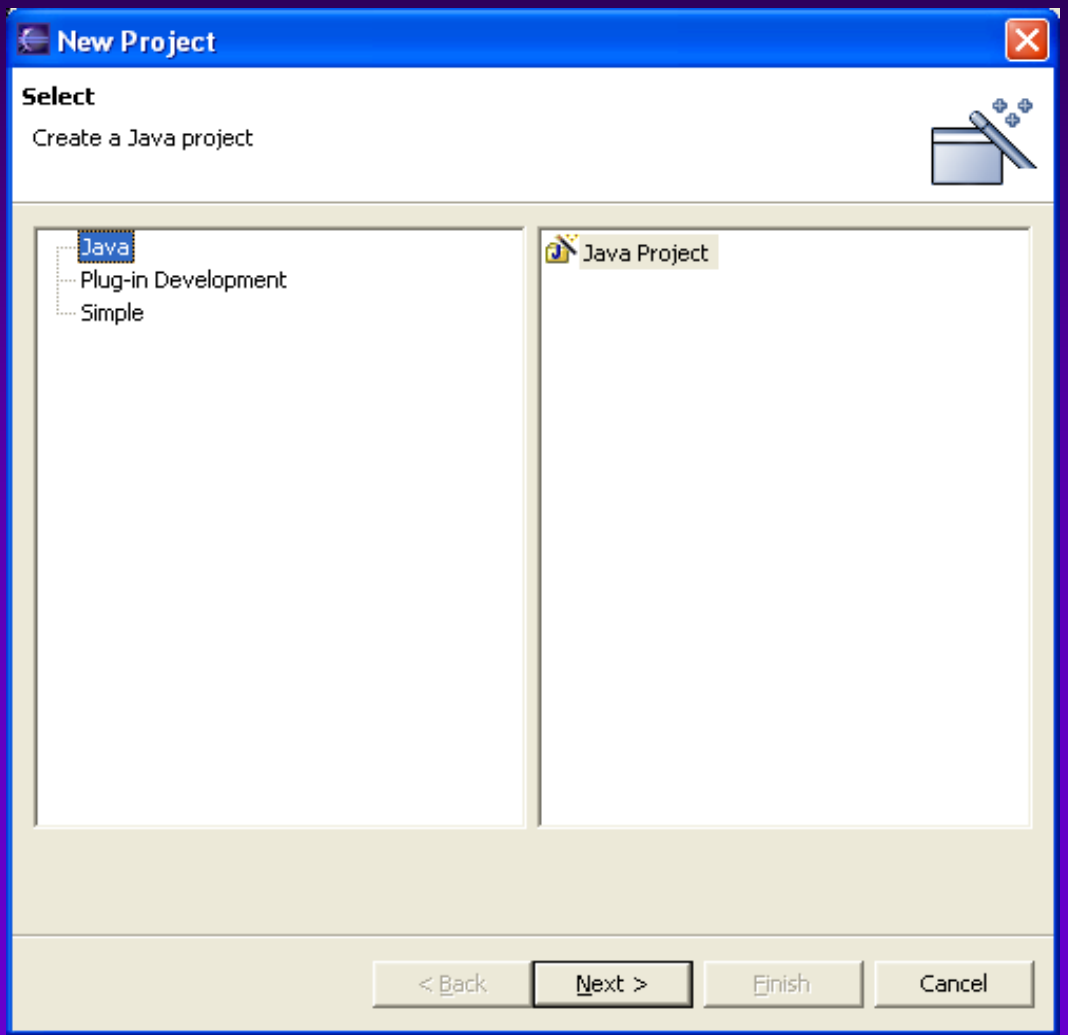
- Projects
- Packages
- Classes

Create a new Java project

- Projects are the containers of packages
- All java projects are directly stored inside a project
- Generated CLASS files are stored along with the JAVA source files
- Two ways to create a new project
  1. From the File > New > Project menu
  2. From the “New Java Project” icon in the toolbar

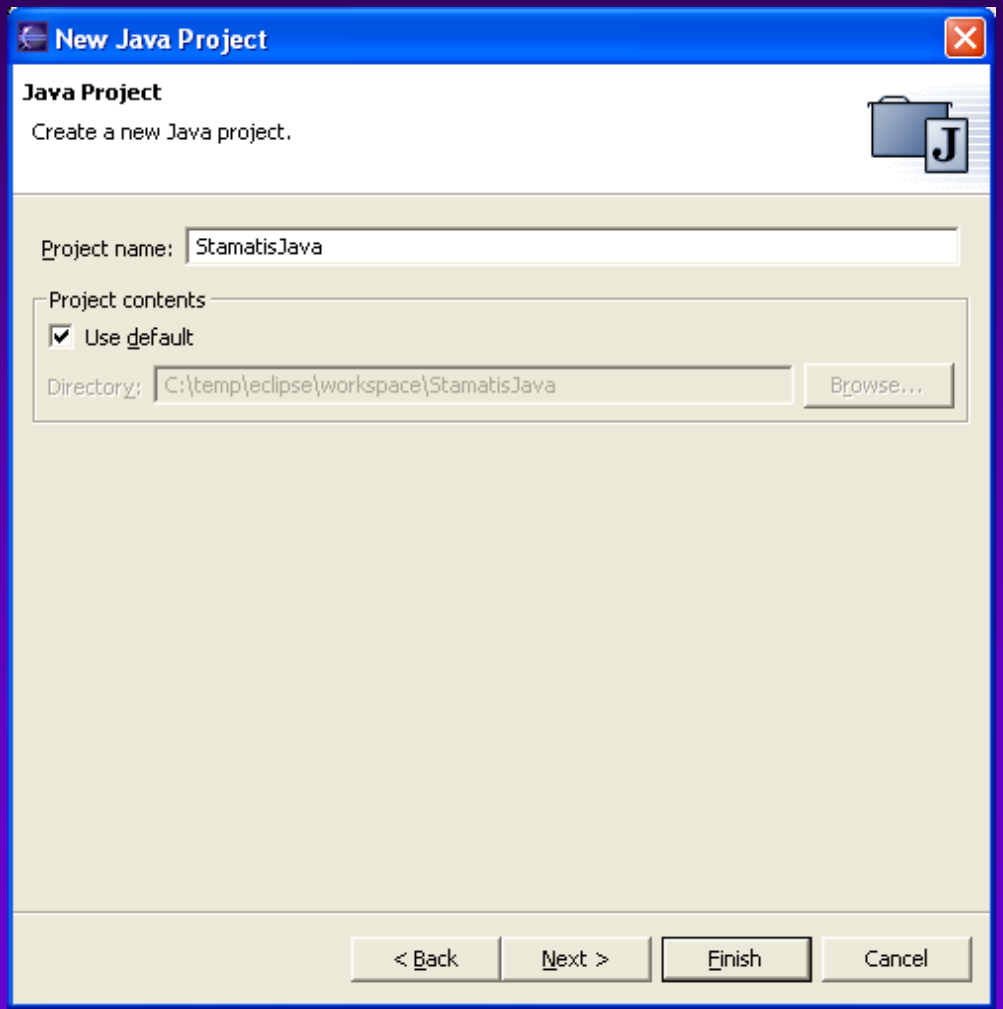
Create a new Java project

- Select “Java”, then Next



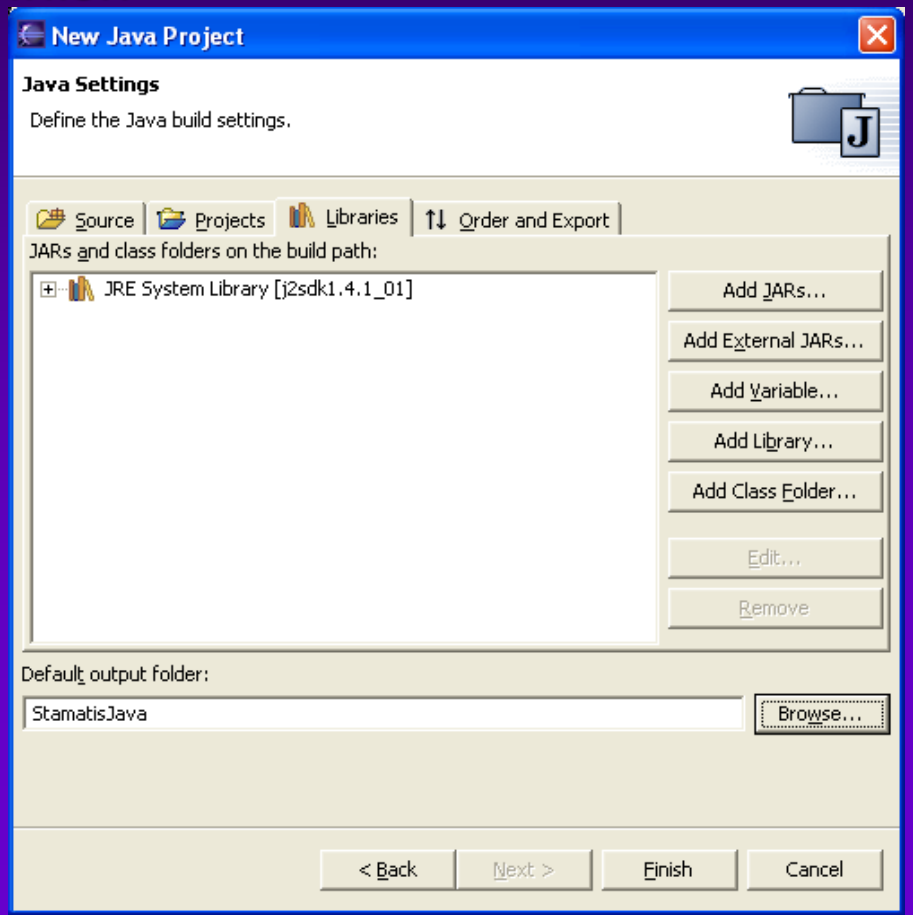
Create a new Java project

- Choose whether you will “import” content in the project. In this case the content is NOT copied in the workspace

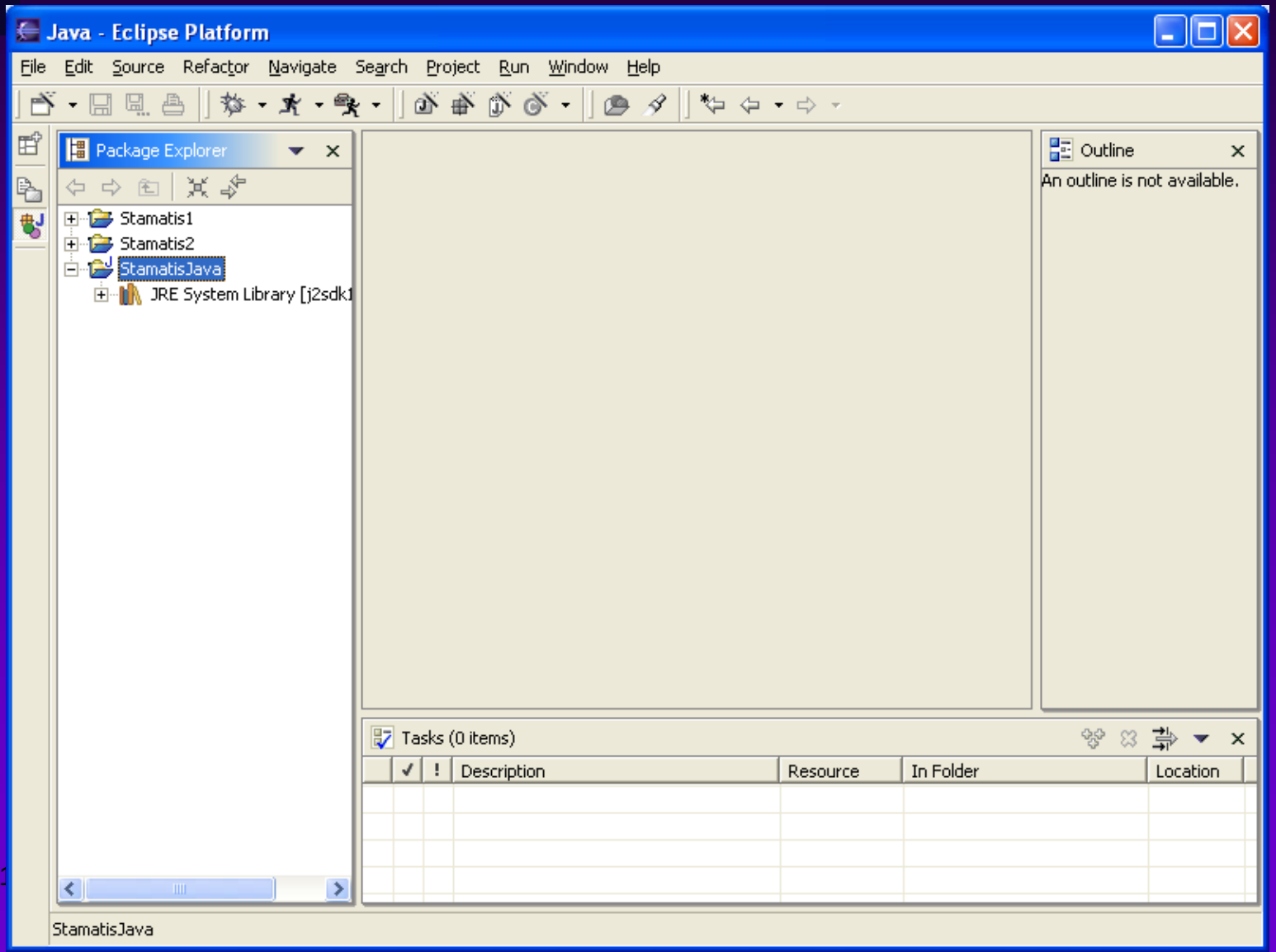


Create a new Java project

- Add any external jars necessary to run your code (if any)
- Click “Finish”



Create a new Java project

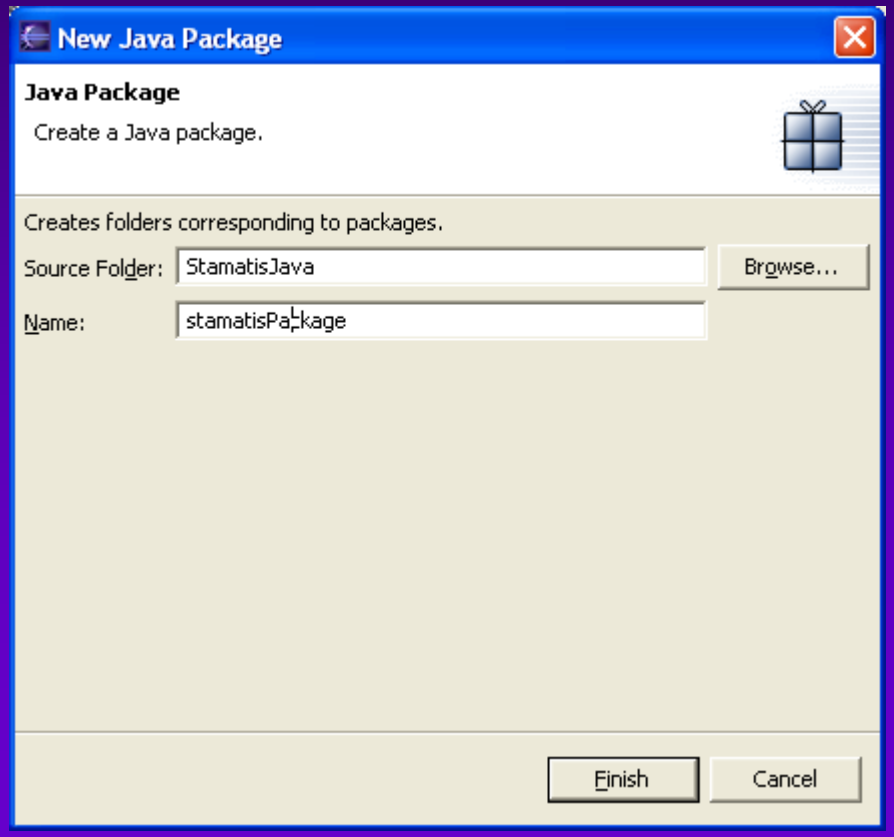


Create a new Java package

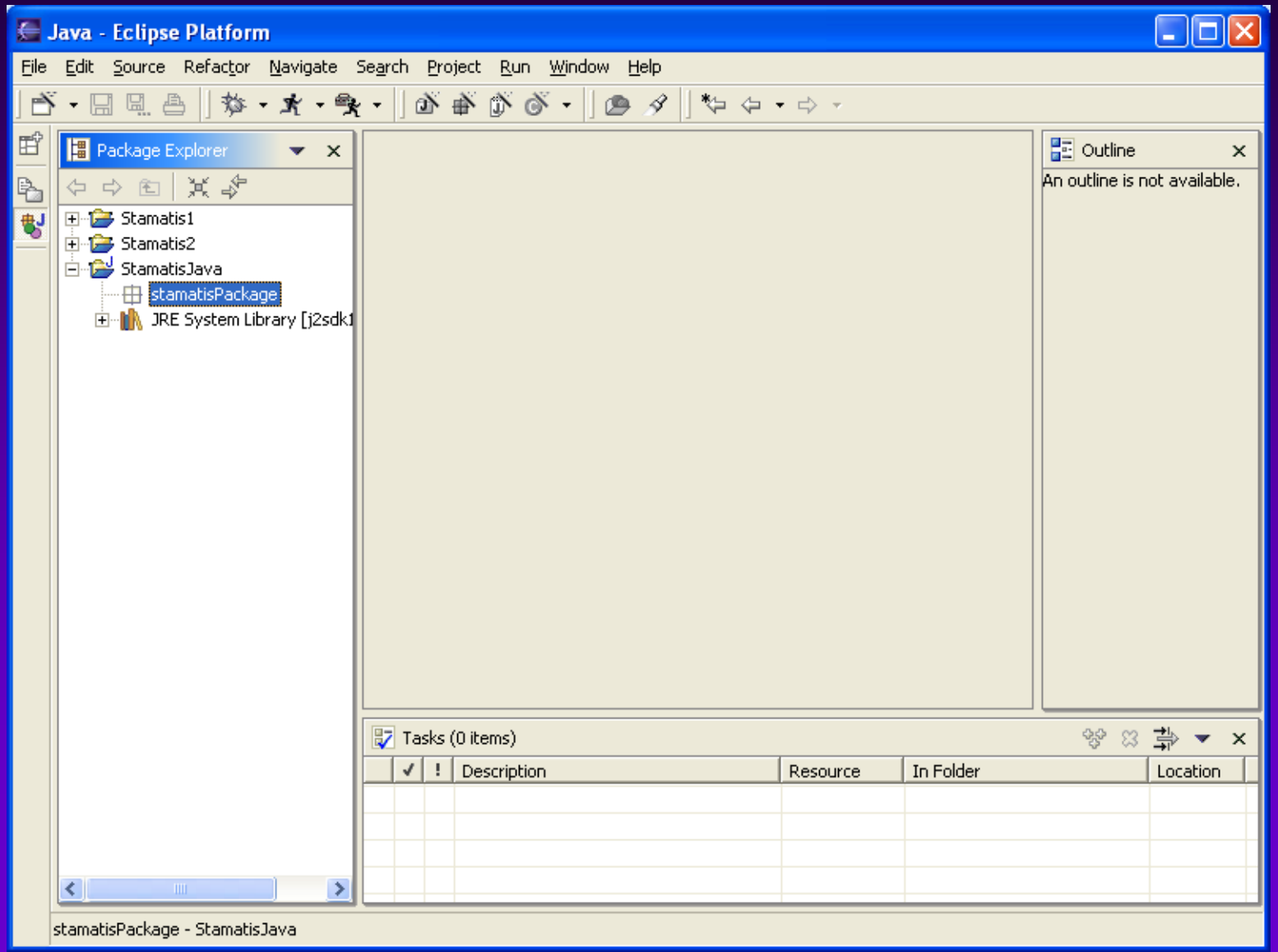
- Choose the project that will contain the package from the Package Explorer
- Three ways to create the new package
  1. From the pop-up menu (right-click on the project) select New > Package
  2. From the “File” menu select New > Package
  3. From the toolbar using the “New Package” icon

Create a new Java package

- Give a name to the package
- Click “Finish”



Create a new Java package

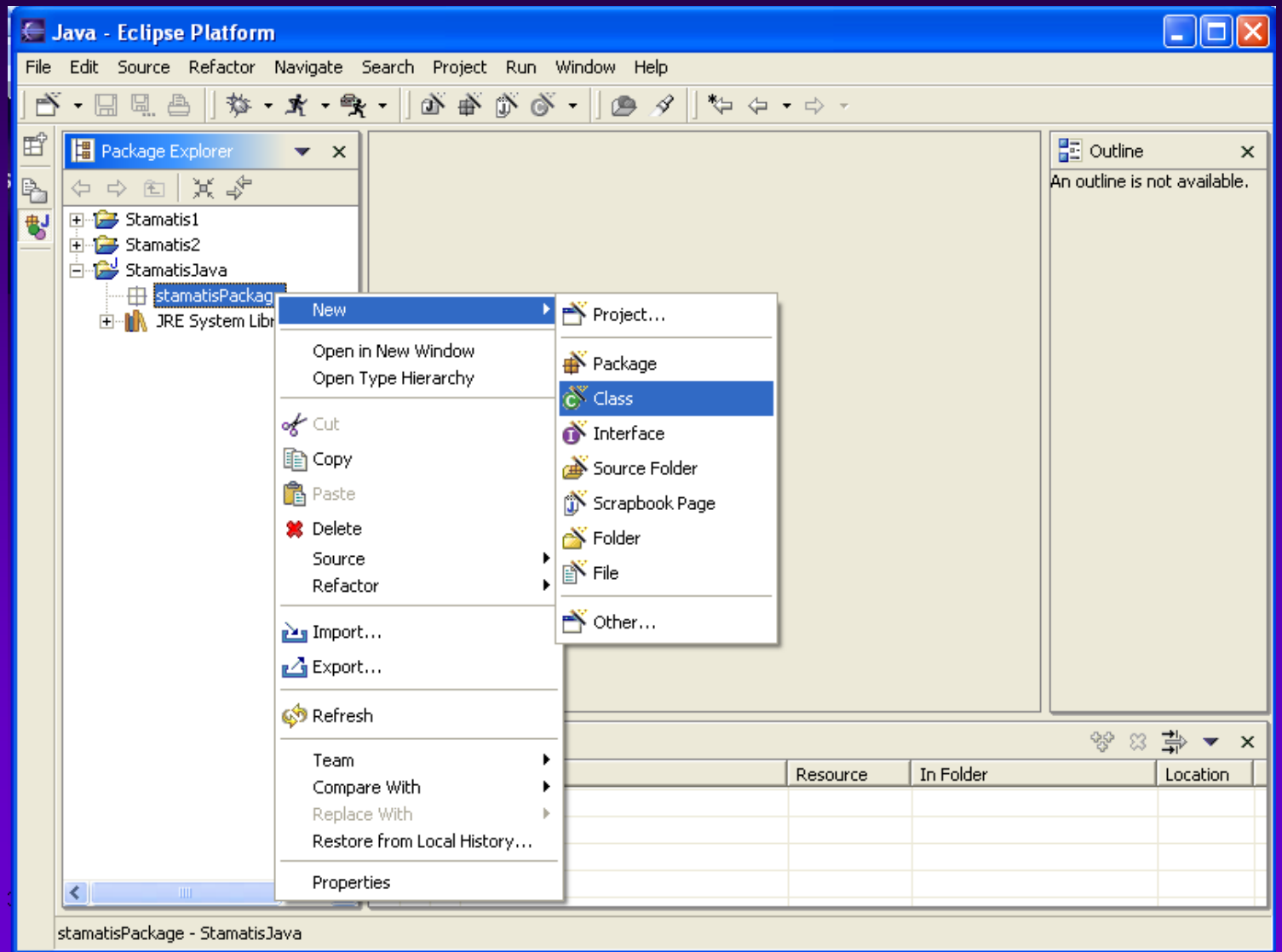


Create a new Java Class

- There are various ways to create a new Java Class
  - Select the package where the new class will reside
  1. Use “New Class” icon in the workbench toolbar
  2. From the package’s pop-up menu (right-click) choose New > Class
  3. From the toolbar select the “New” icon’s drop-down and then select “Class”
  4. From the toolbar select the “New” icon and then “Java” in the left pane and “Class” in the right pane
  5. Select File > New > Class menu

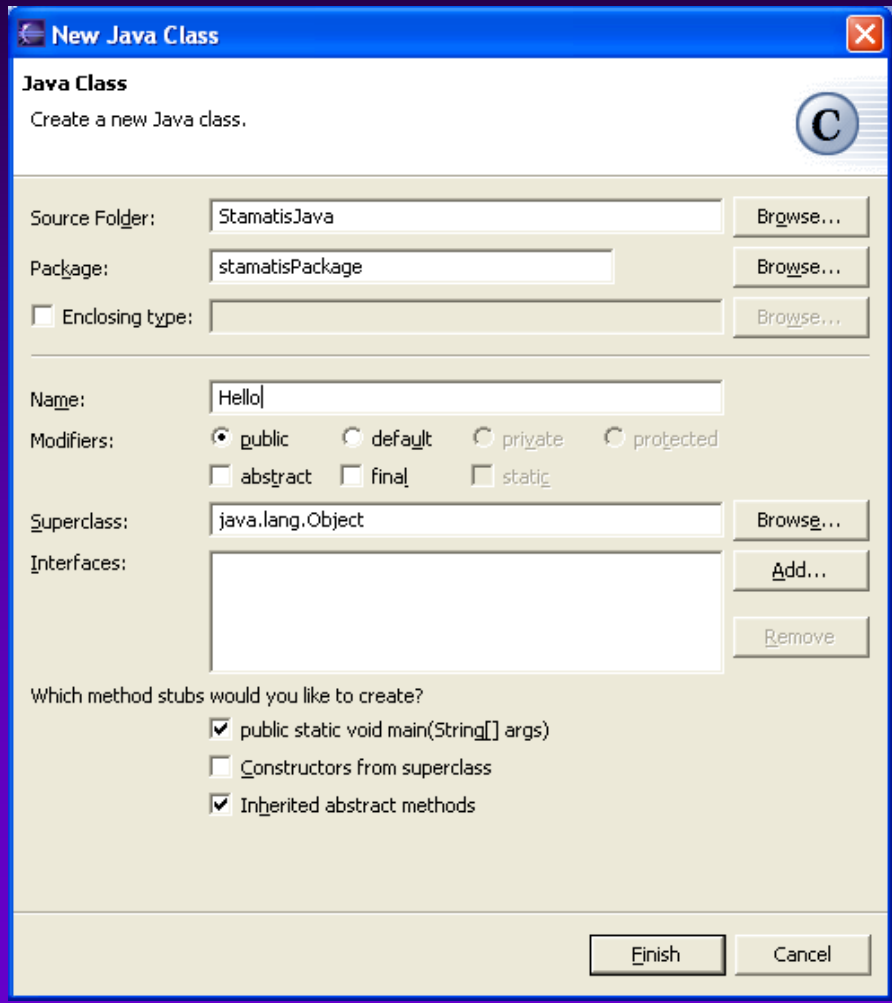
- You can always type a new class in an existing Java file from the Package Explorer

Create a new Java Class

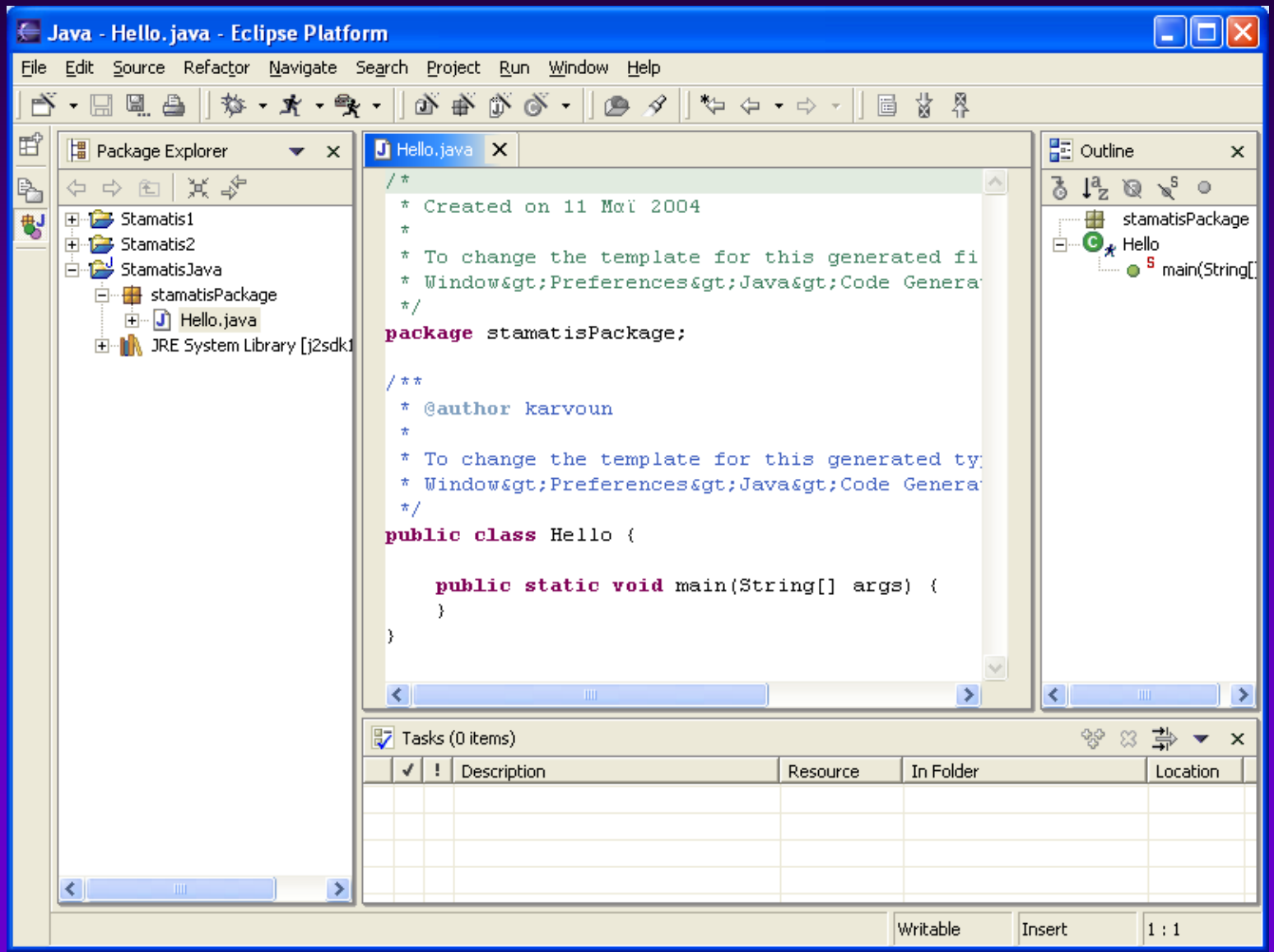


Create a new Java Class

- Select a class name
- Select possible superclass
- Select possible implementing interfaces
- e.t.c

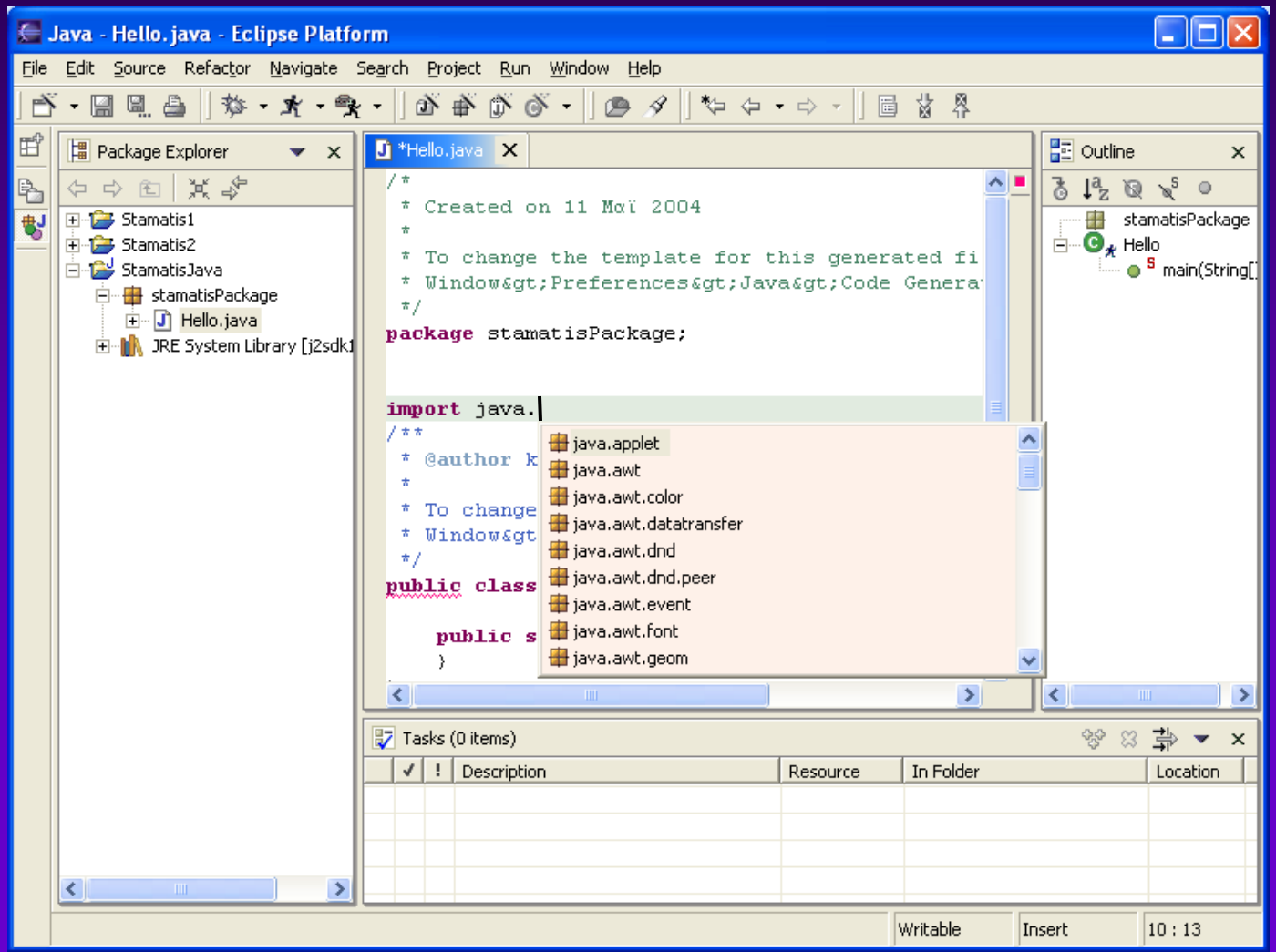


Create a new Java Class



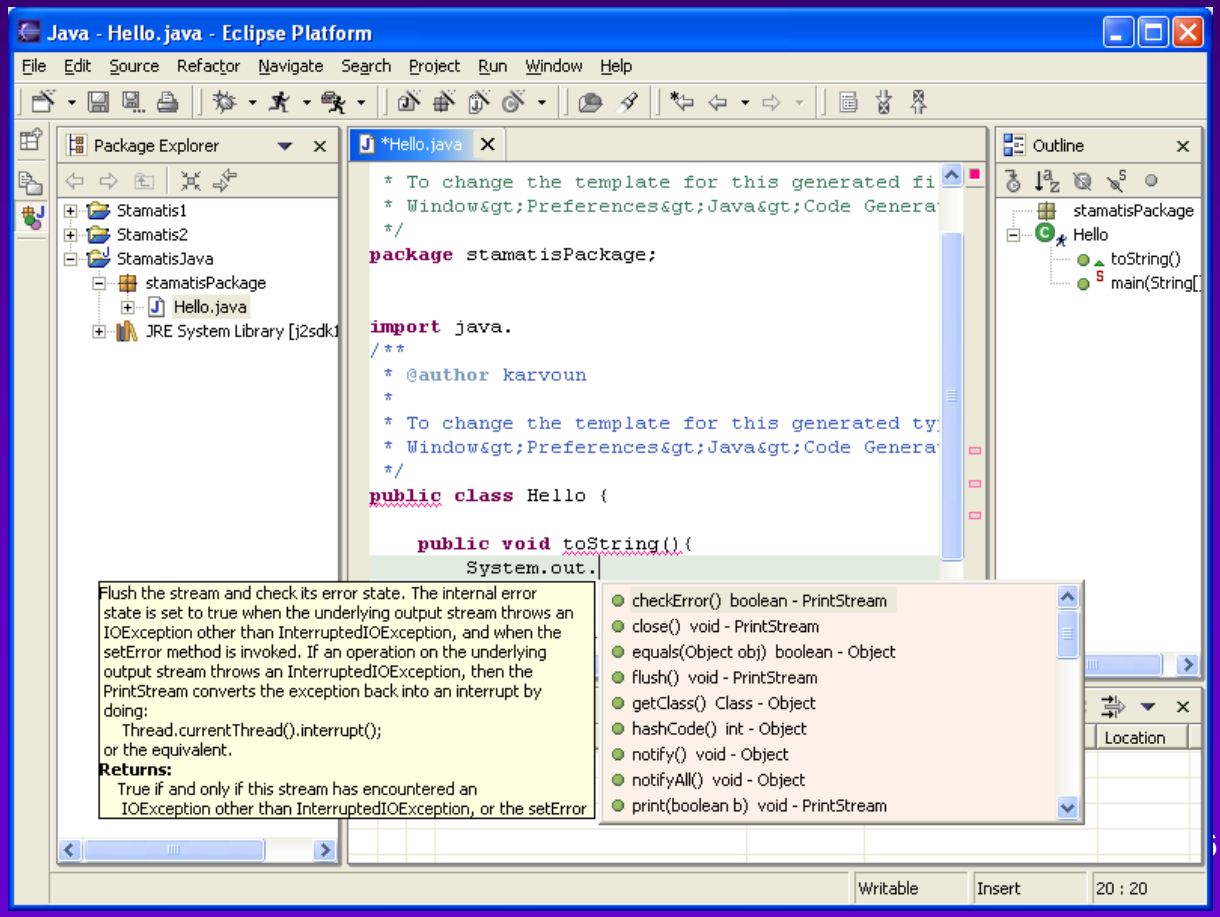
Java editor features

- Import assistance



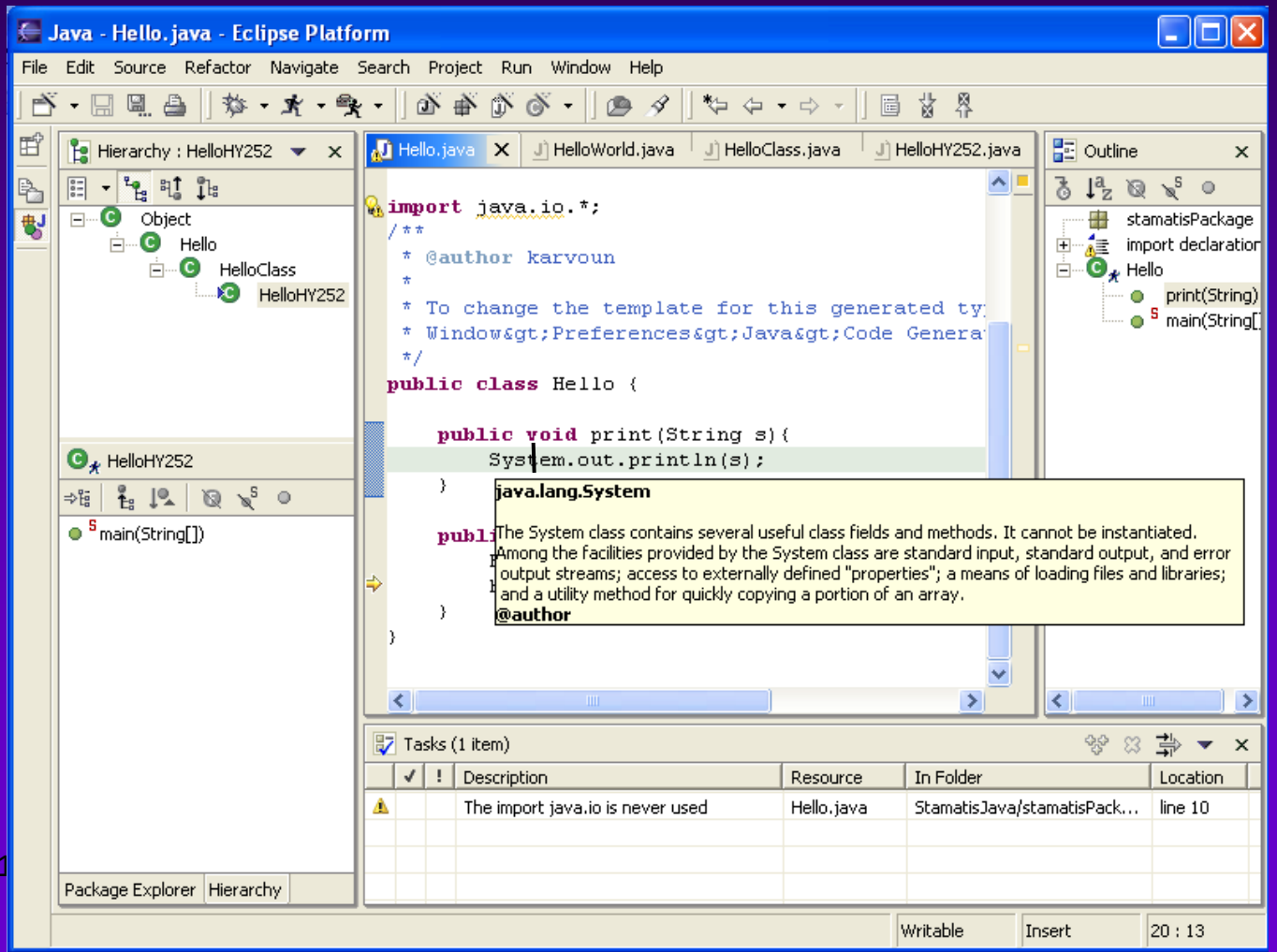
Java editor features

- Method Completion (automatically or using Ctrl+Space)



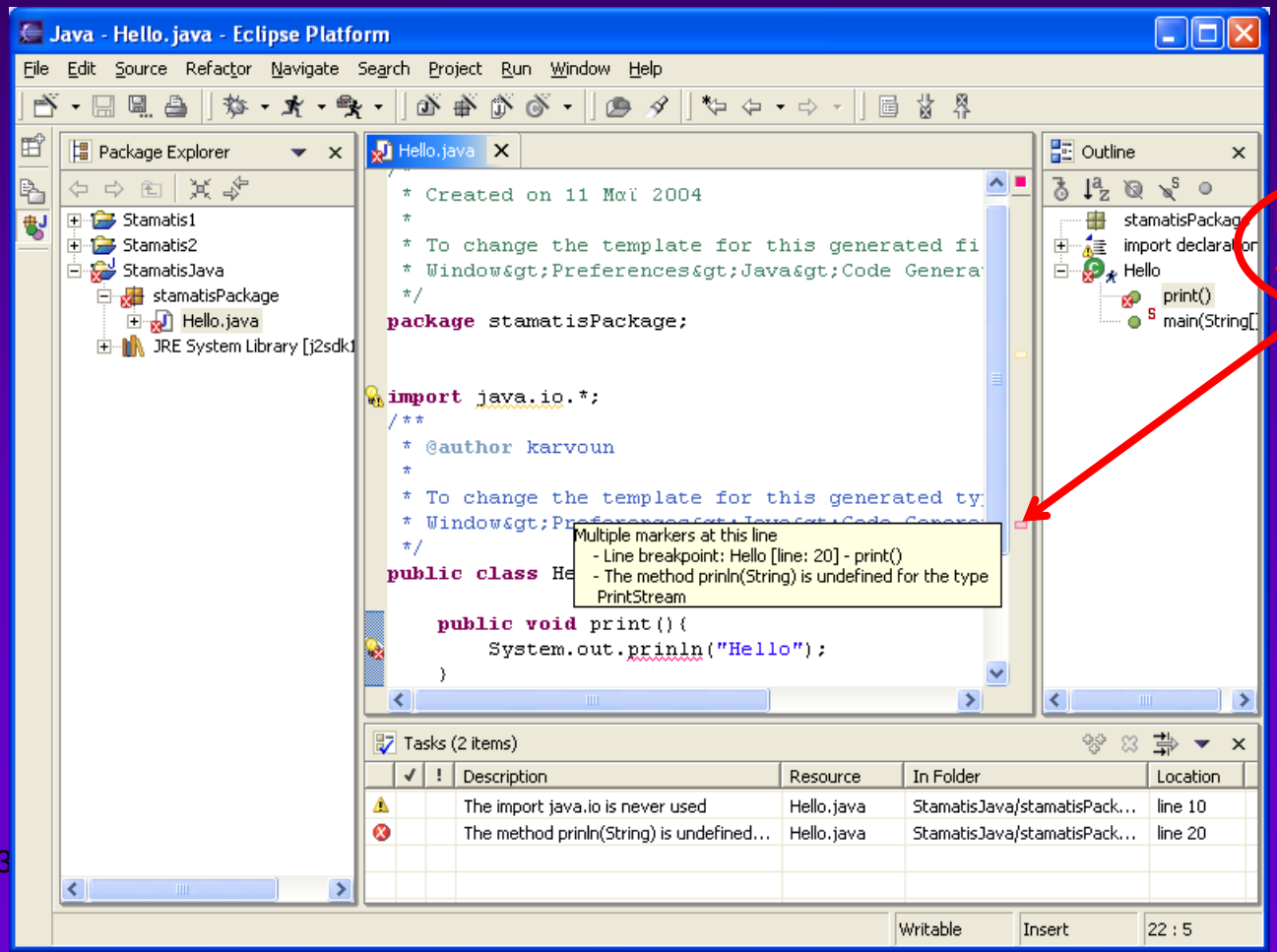
Java editor features

- Hovering over identifier shows Javadoc spec



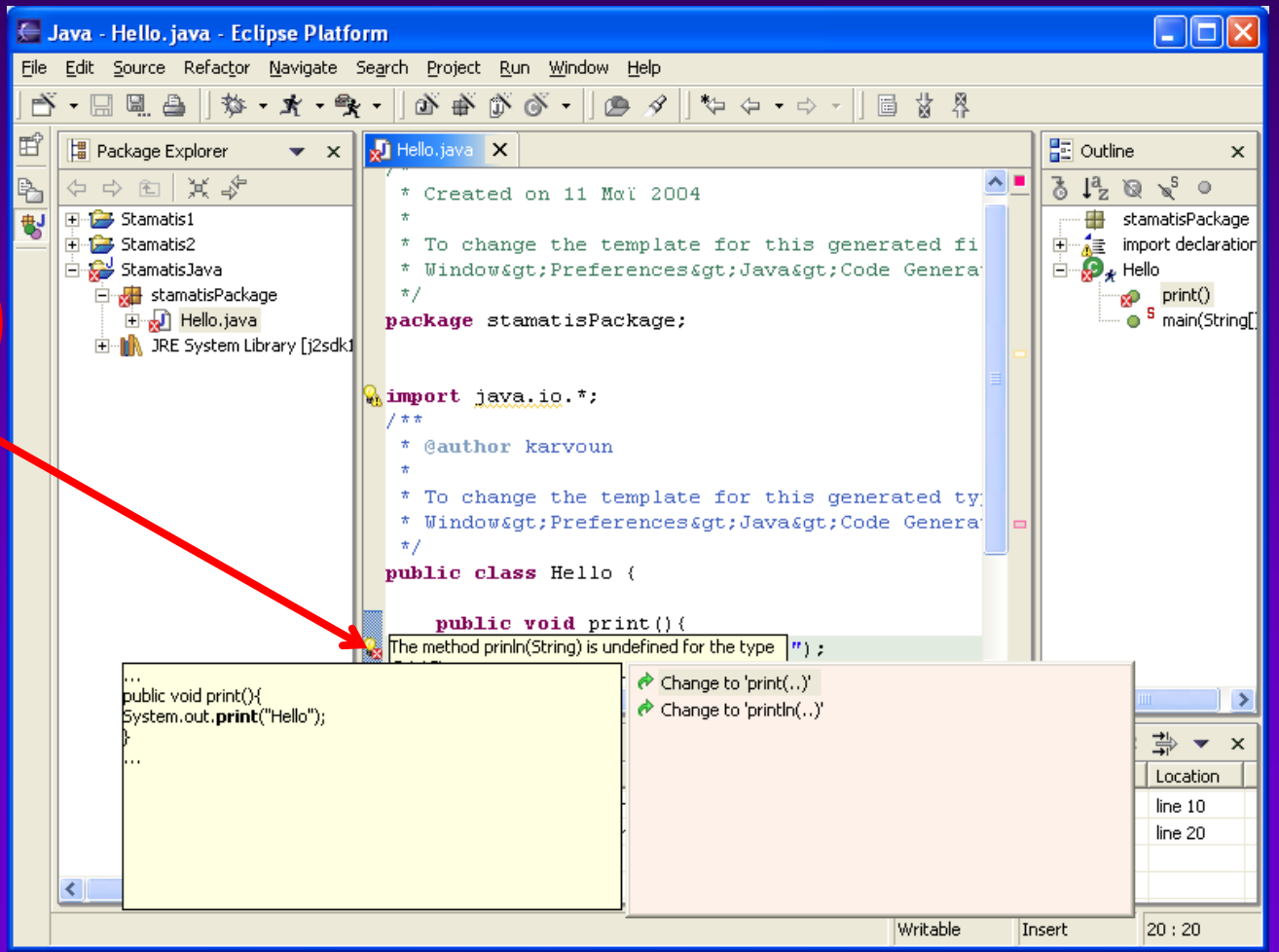
Java editor features

- On-the-fly spell check



Java editor features

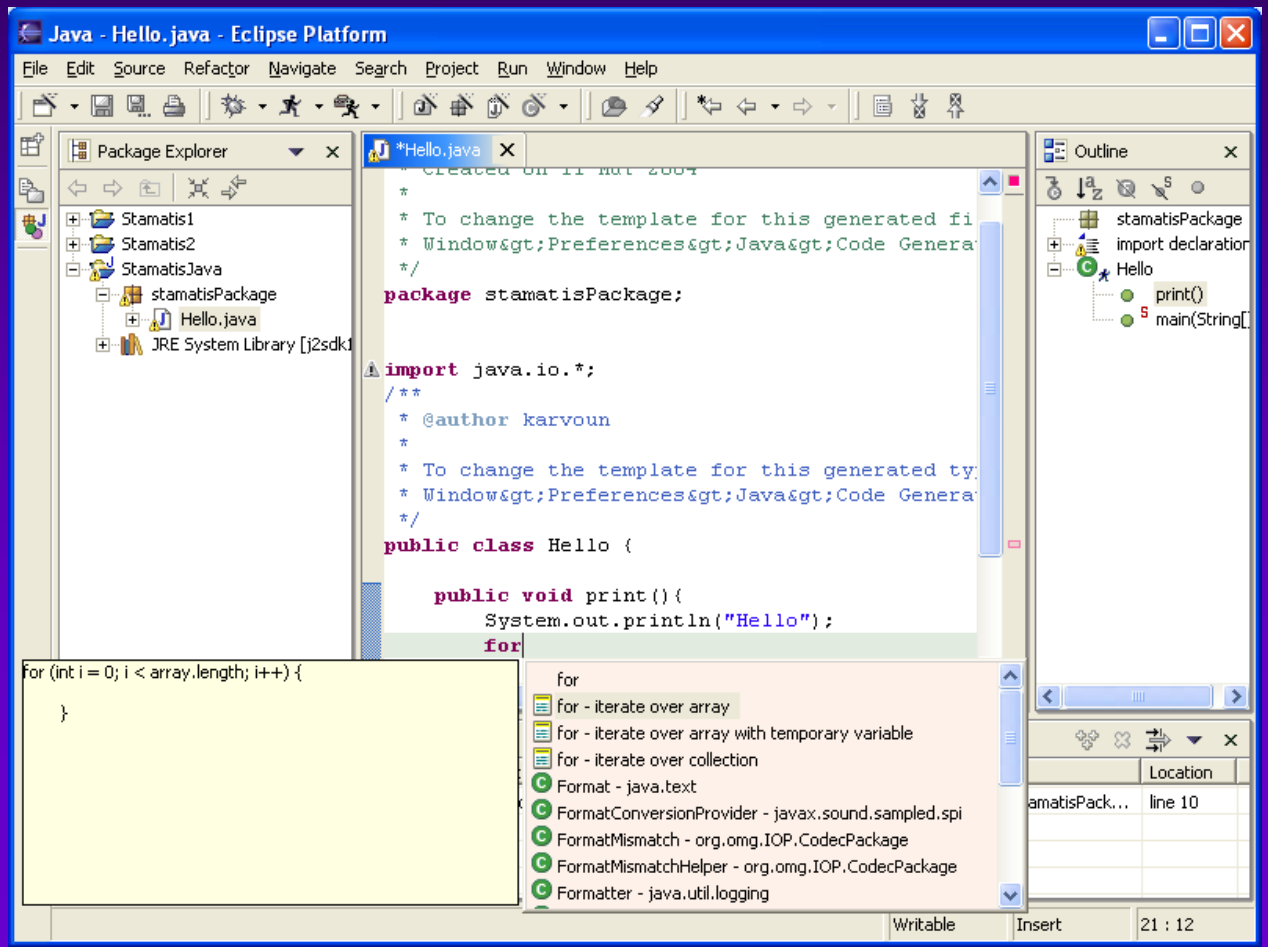
- Quick Fix



Double click here to get suggestions

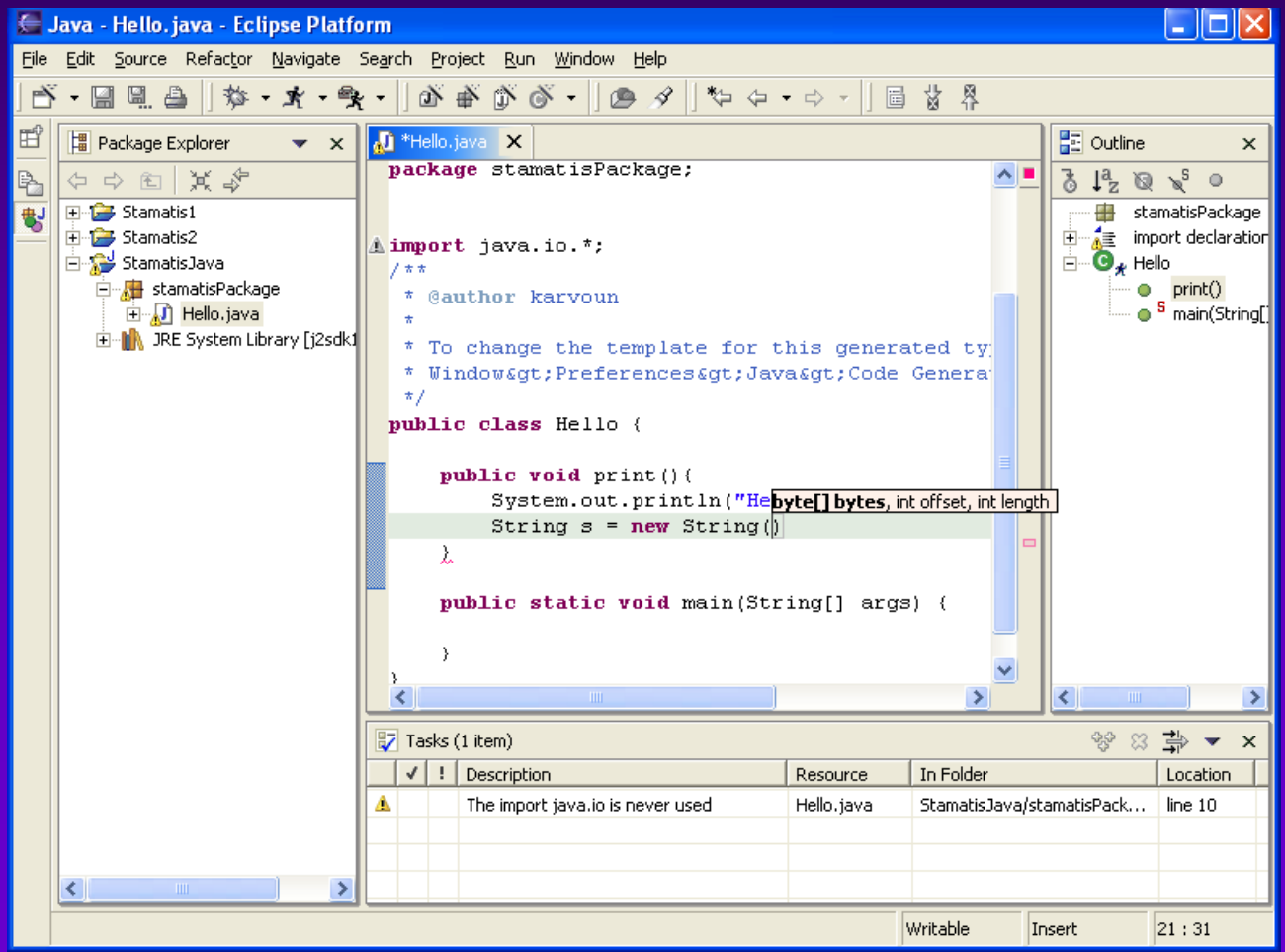
Java editor features

- Code templates (Ctrl+Space)



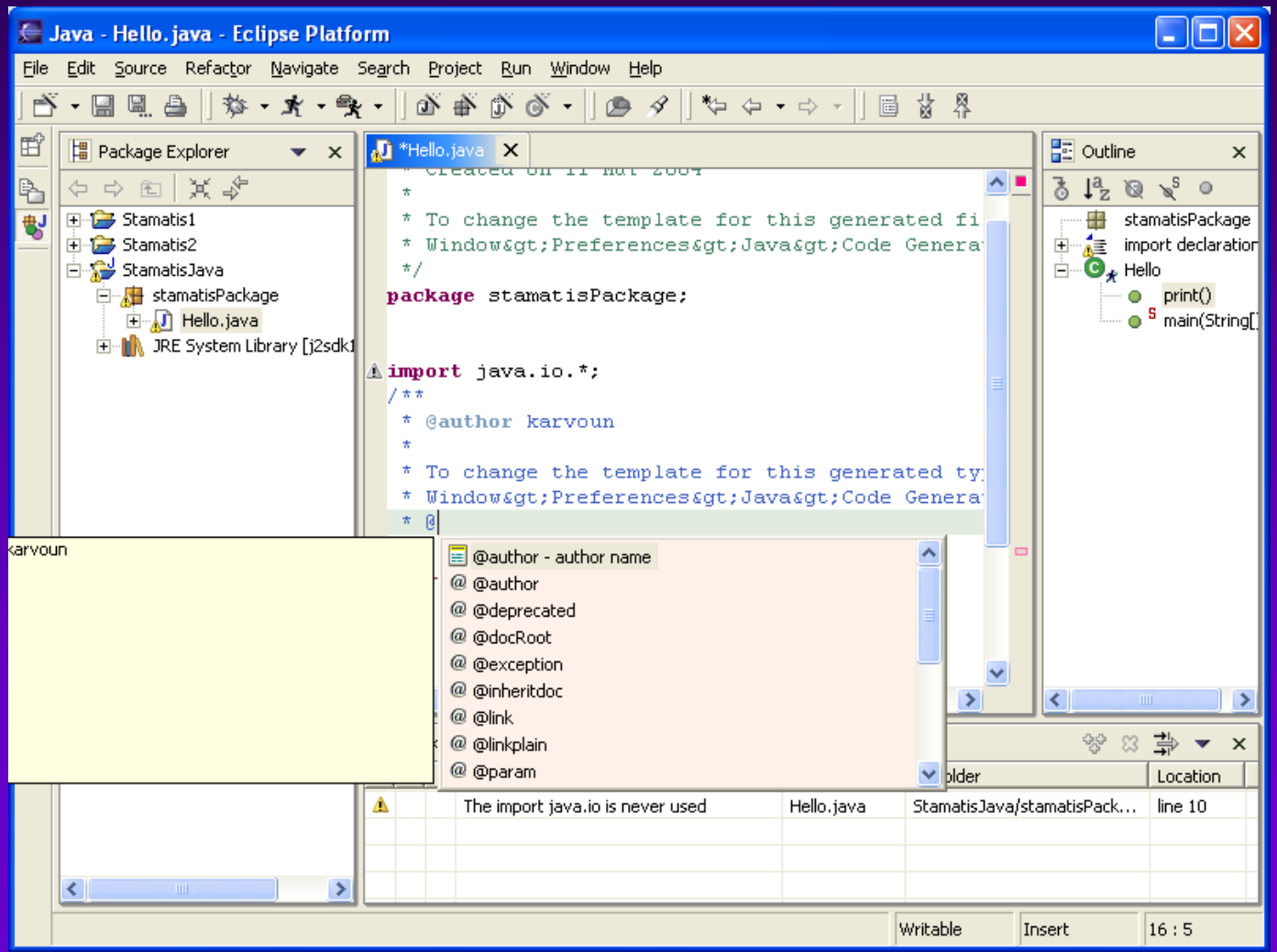
Java editor features

- Argument hints

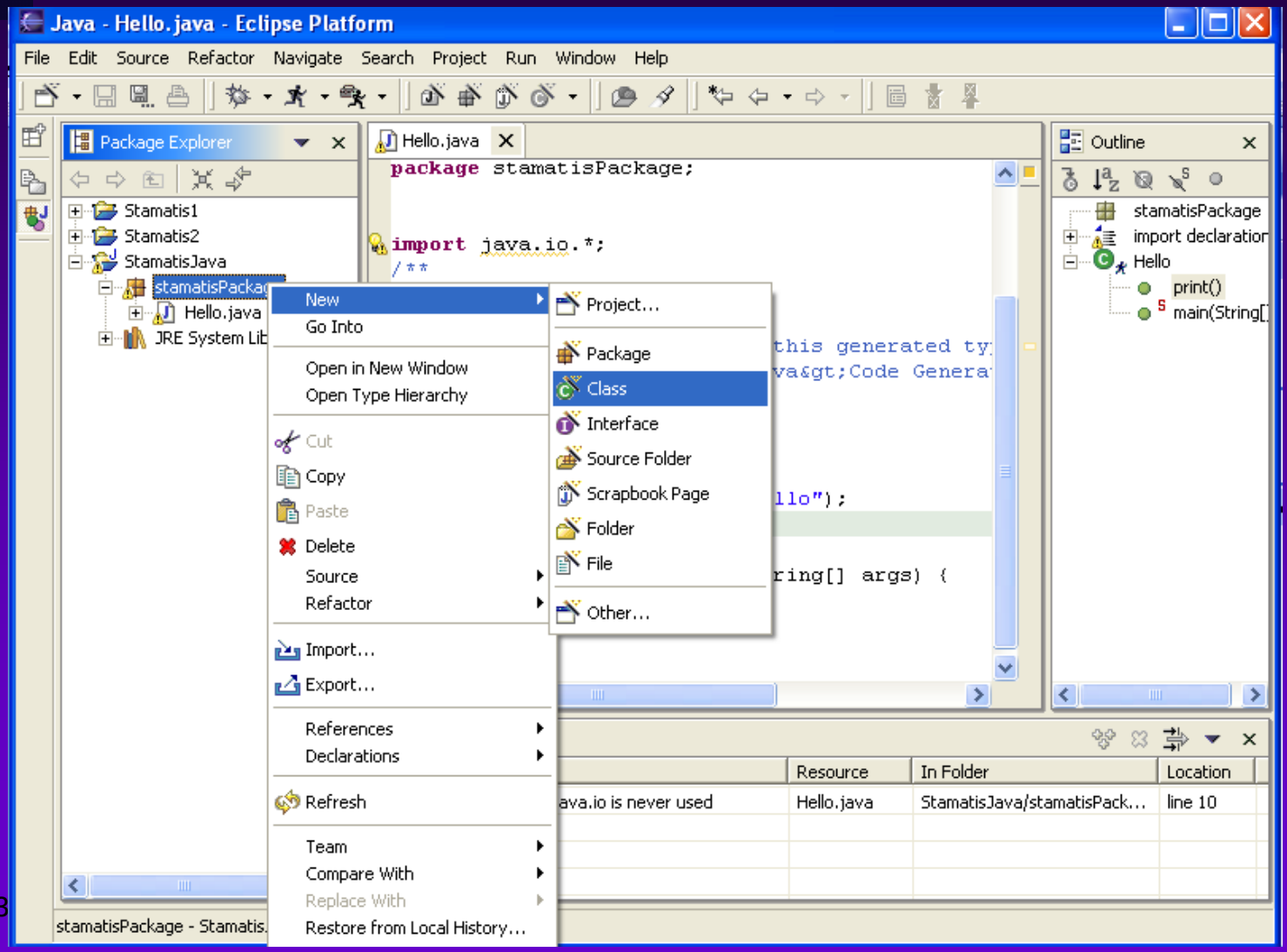


Java editor features

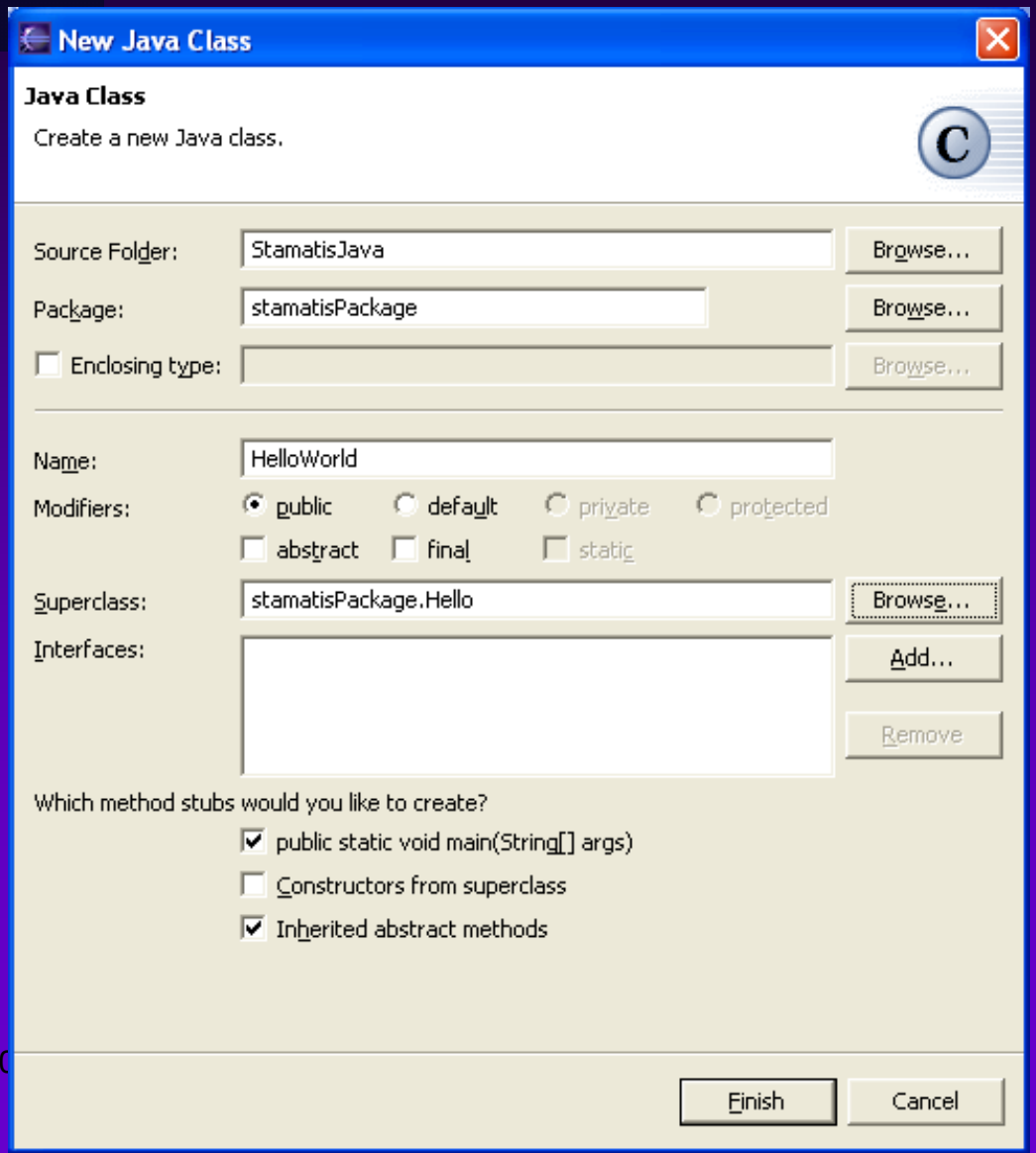
- JavaDoc Assist

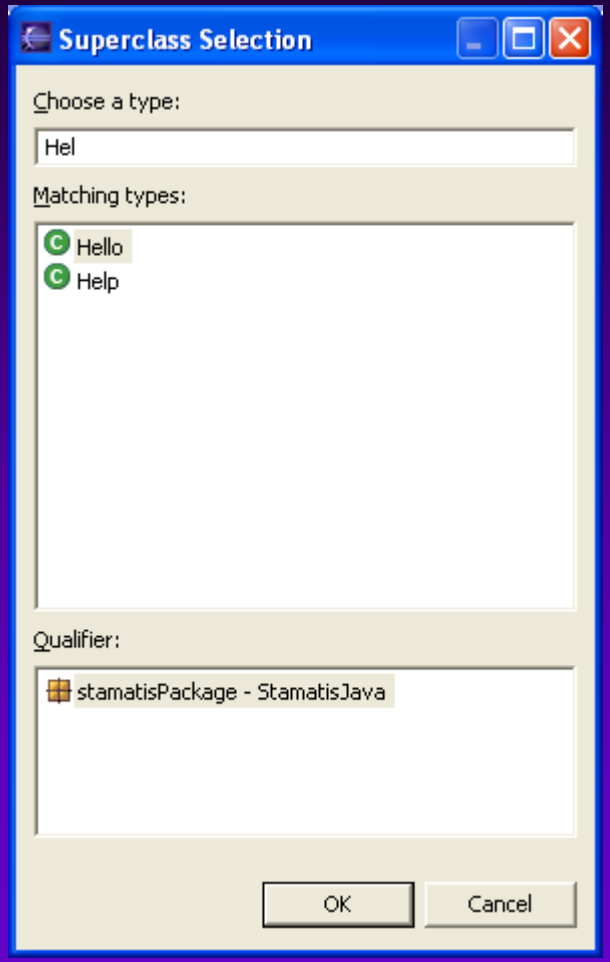


Class Hierarchies

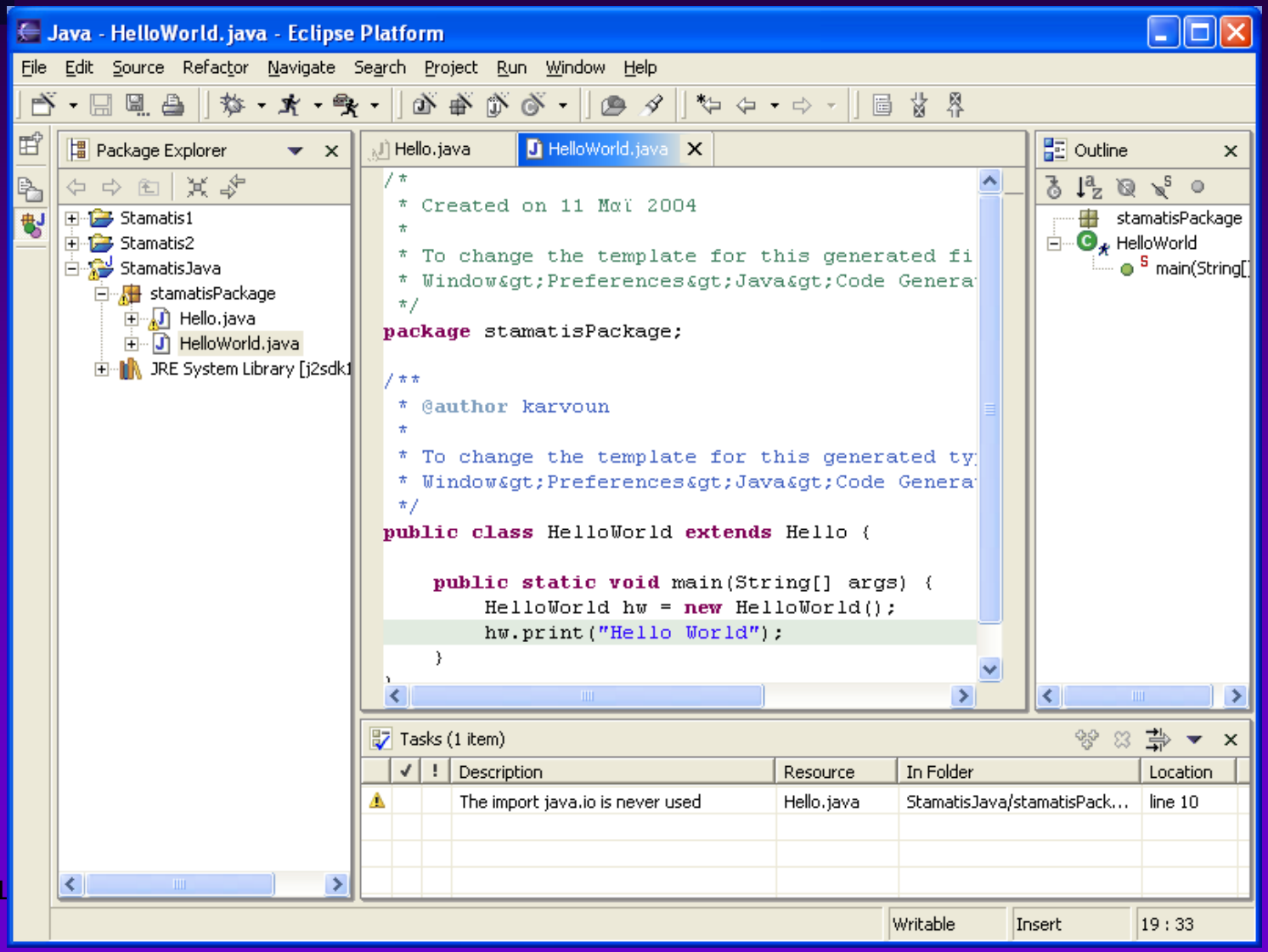


Class Hierarchies

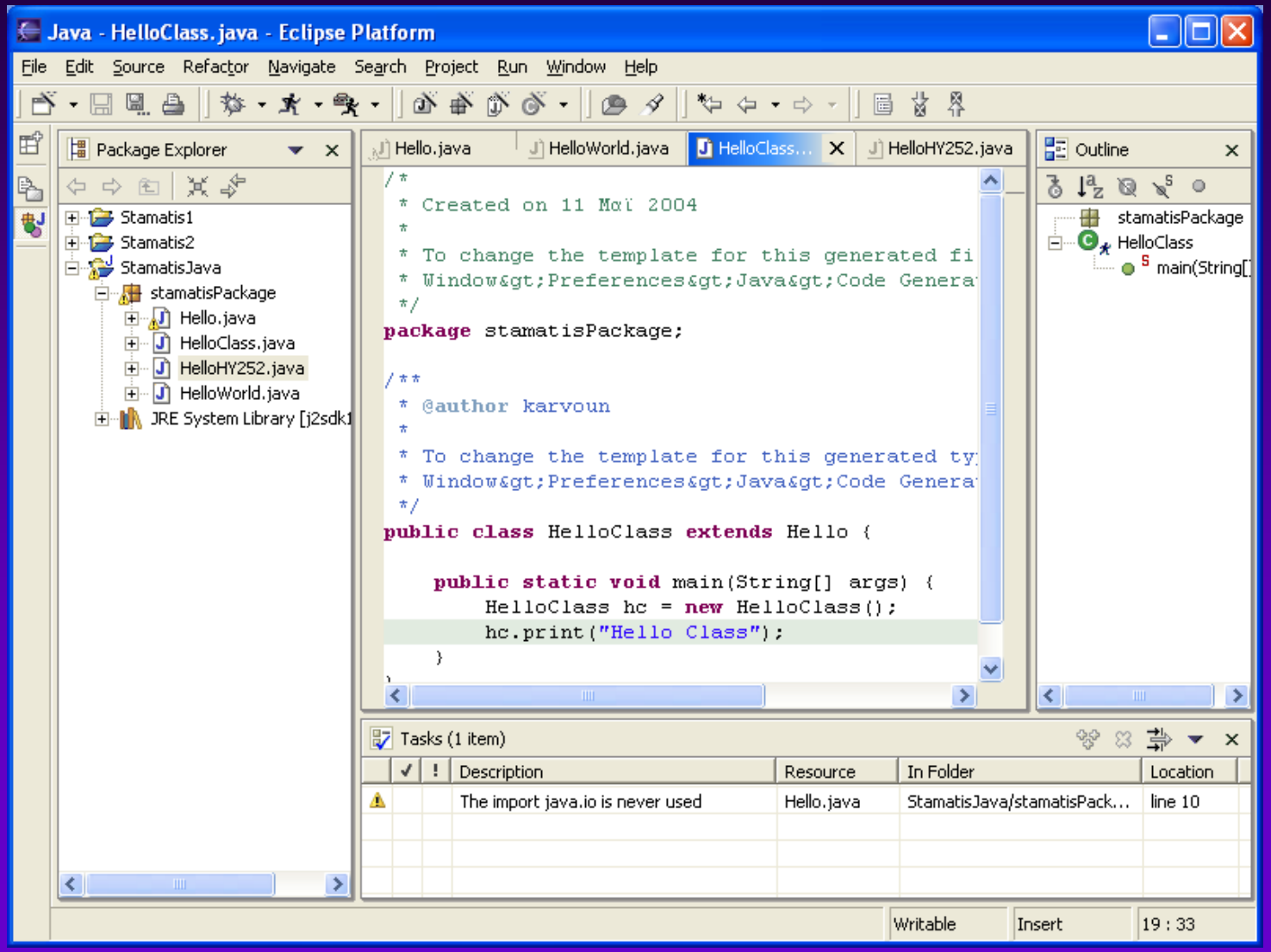




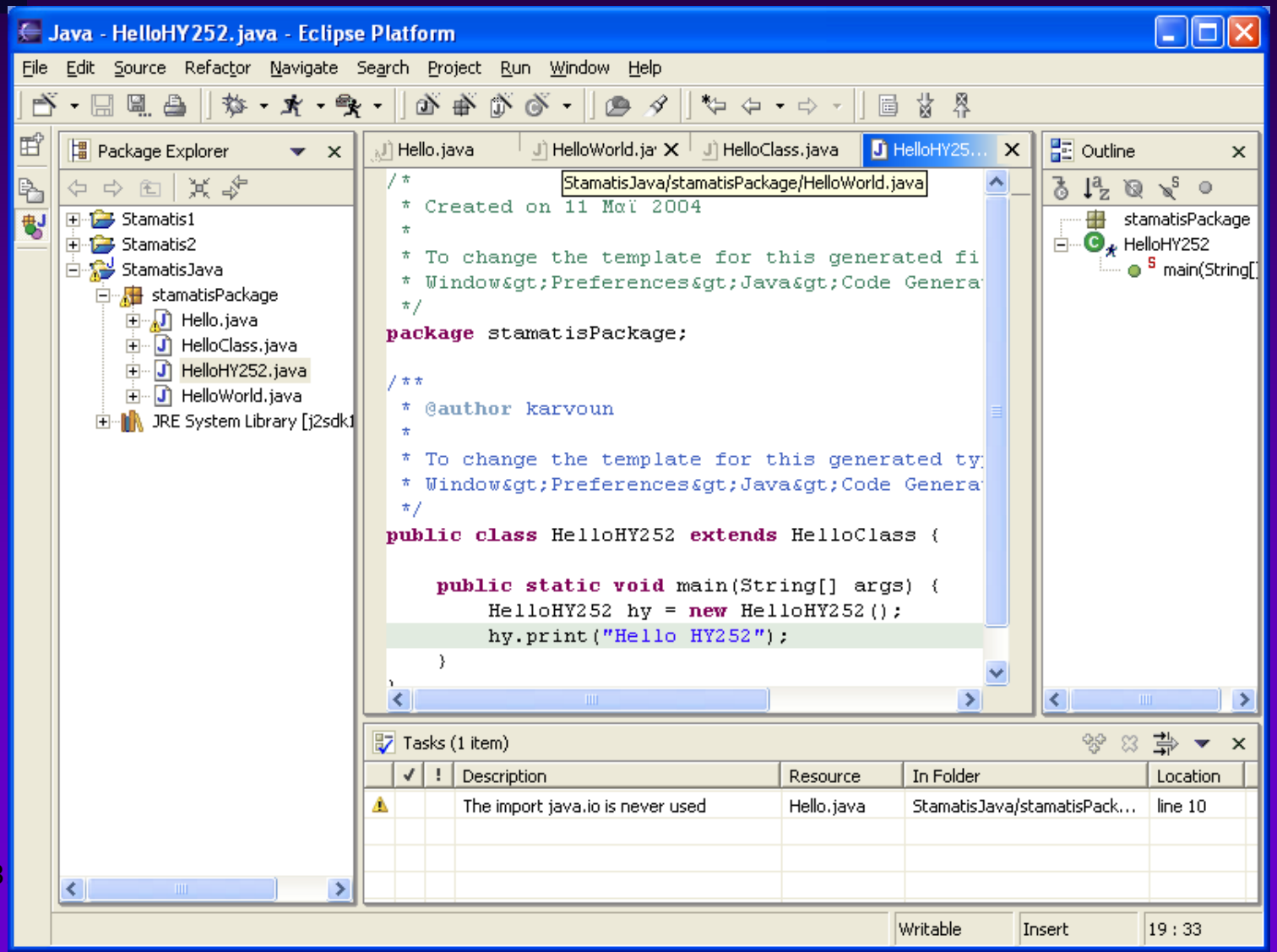
Class Hierarchies



Class Hierarchies

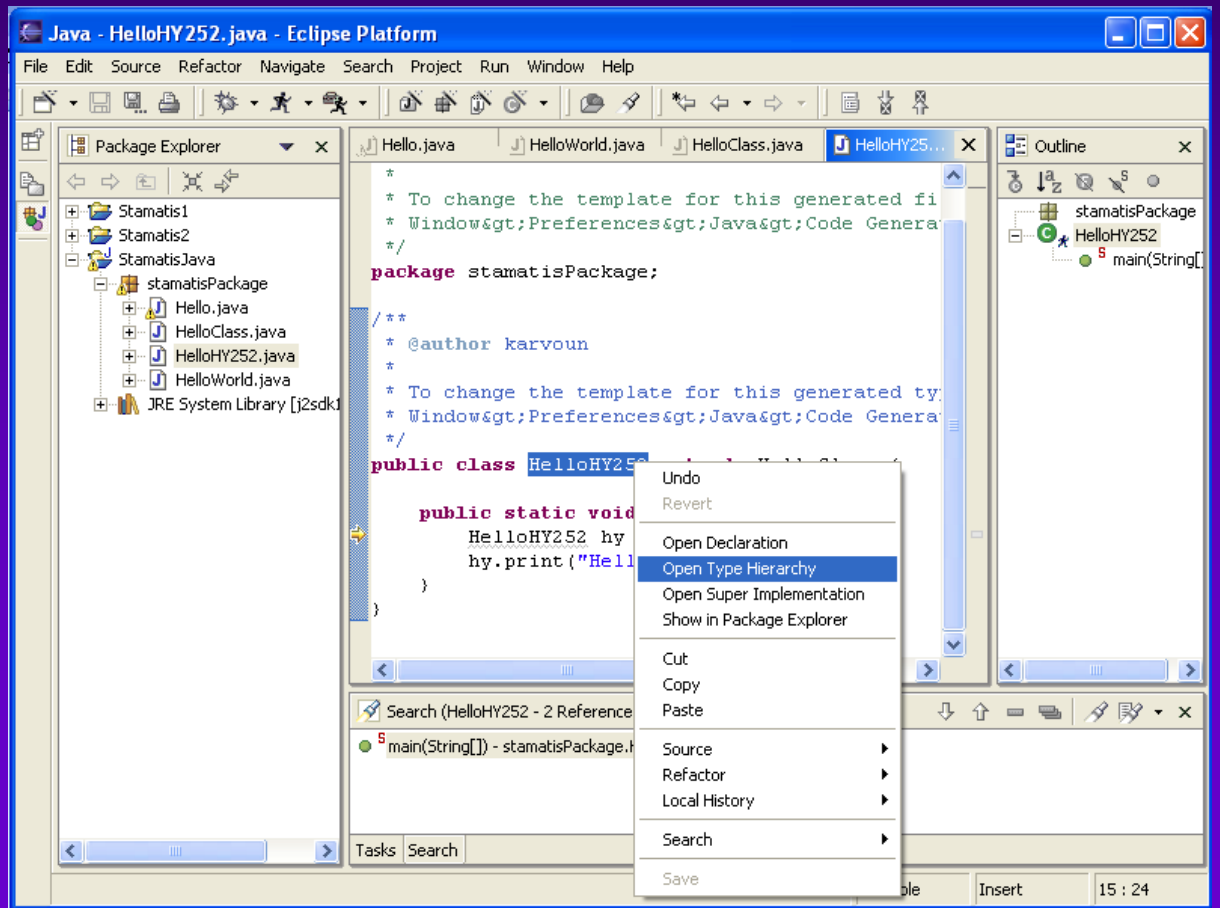


Class Hierarchies

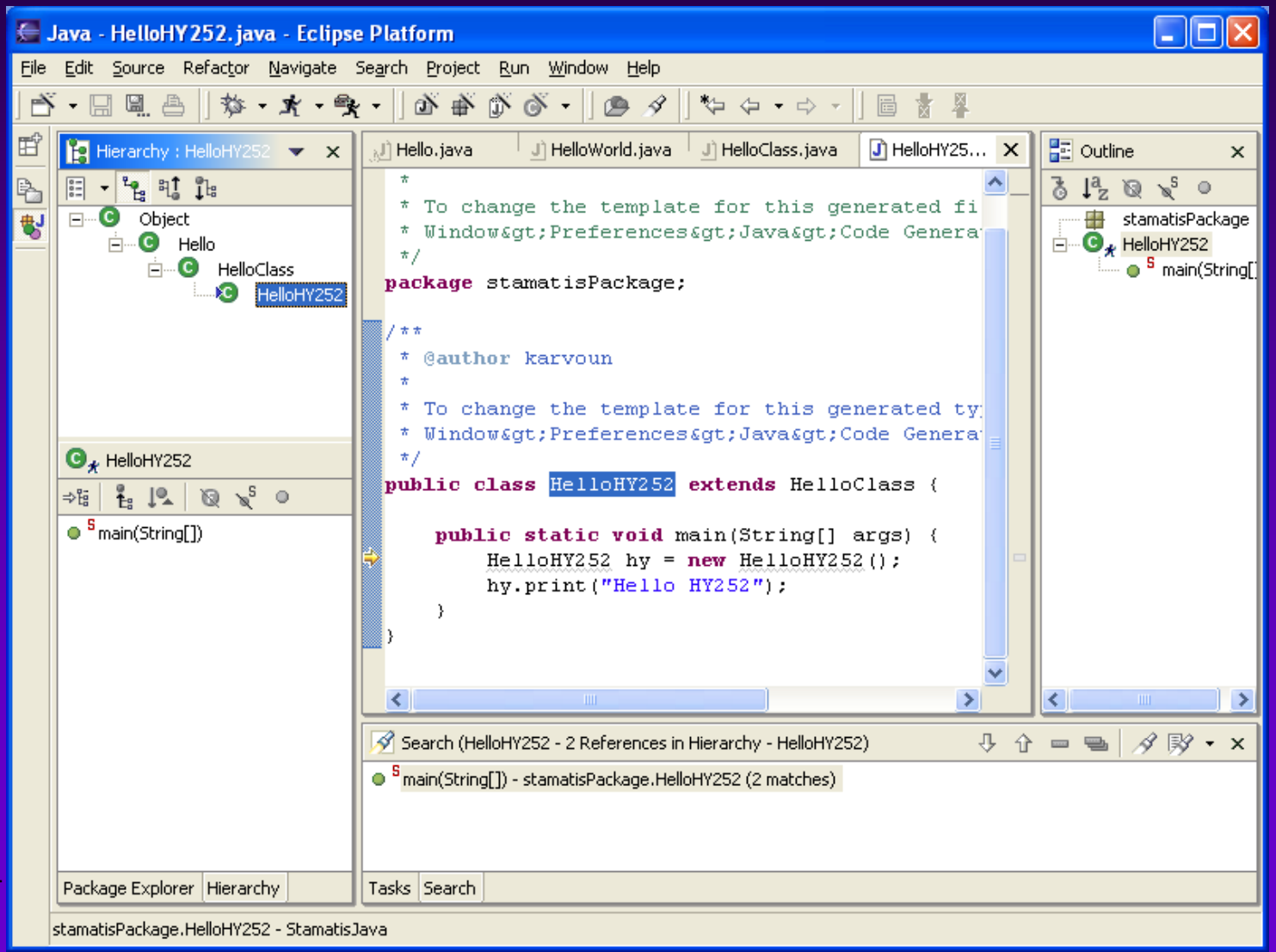


Class Hierarchies

- Select a type and from its pop-up select “Open Type Hierarchy”
- The Hierarchy View will appear

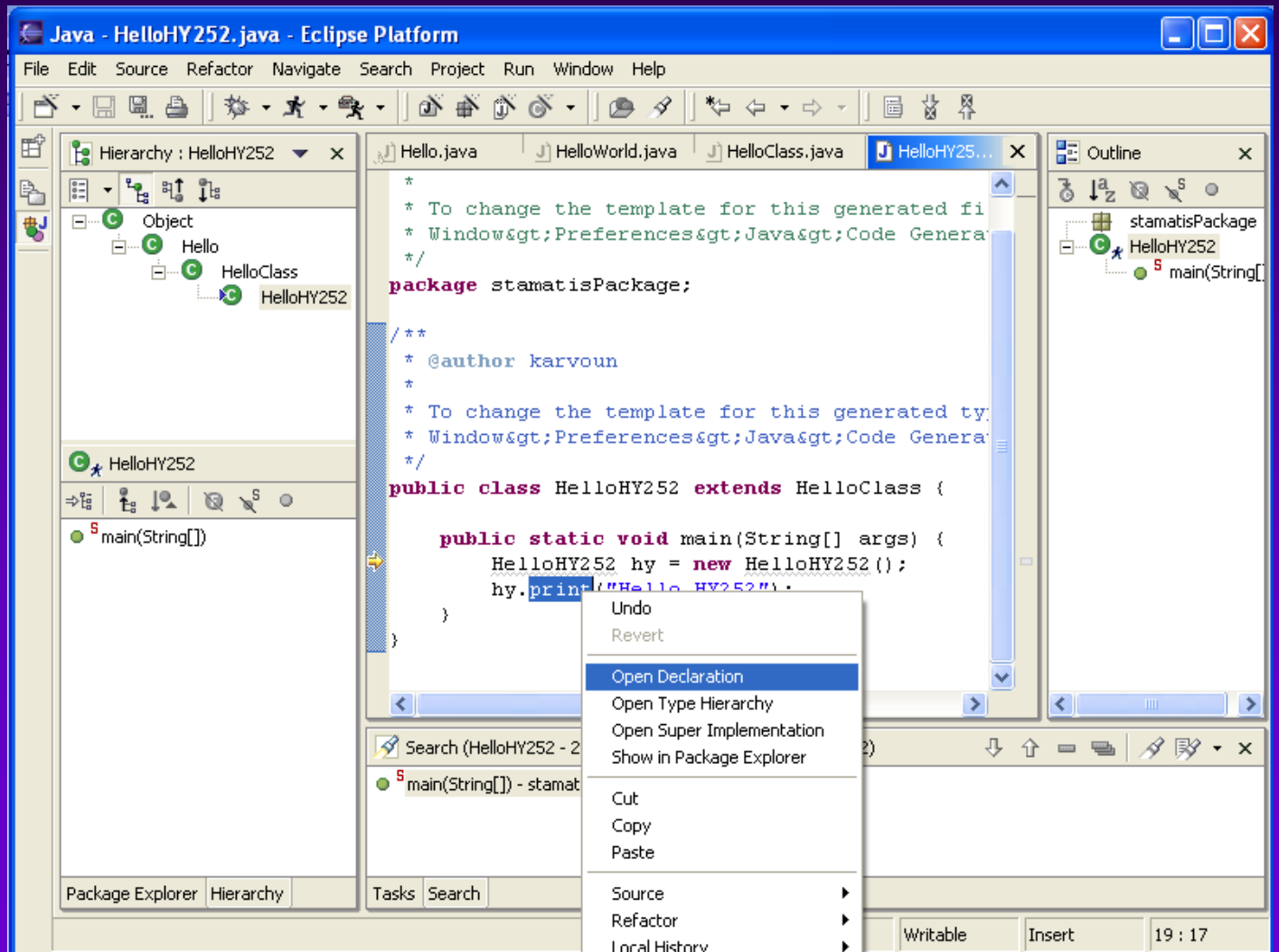


Class Hierarchies

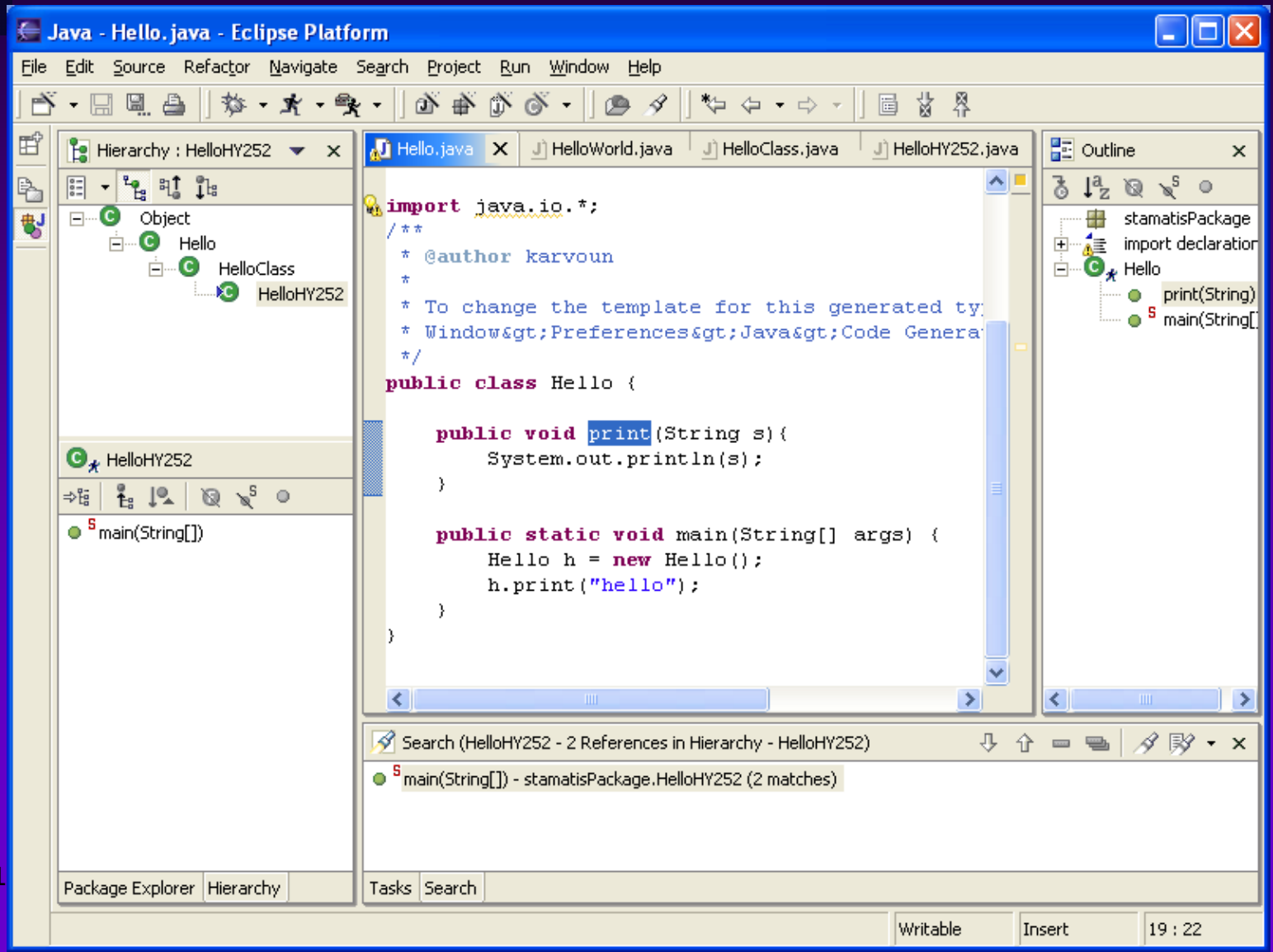


Search for Java Elements

- Declarations or References

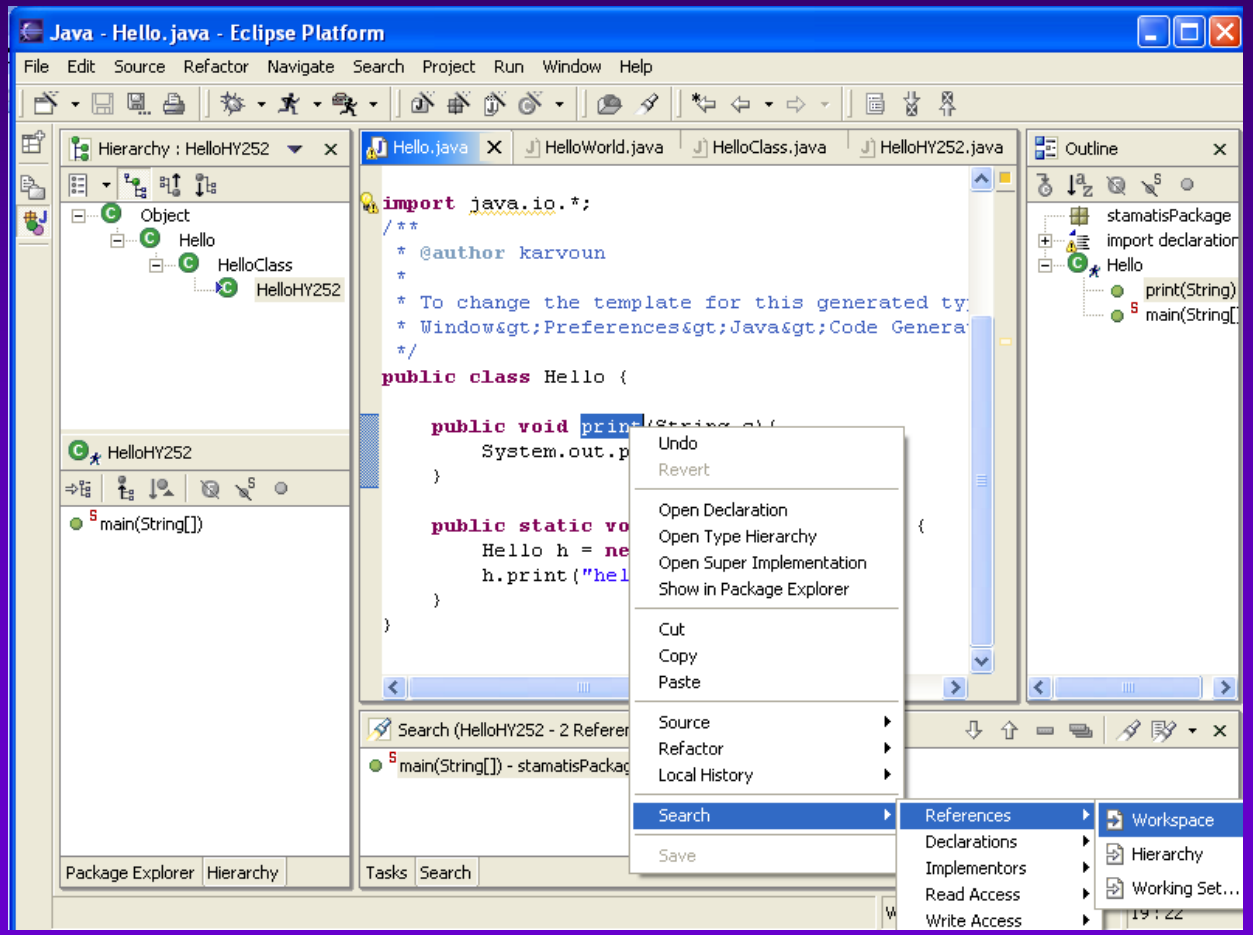


Search for Java Elements

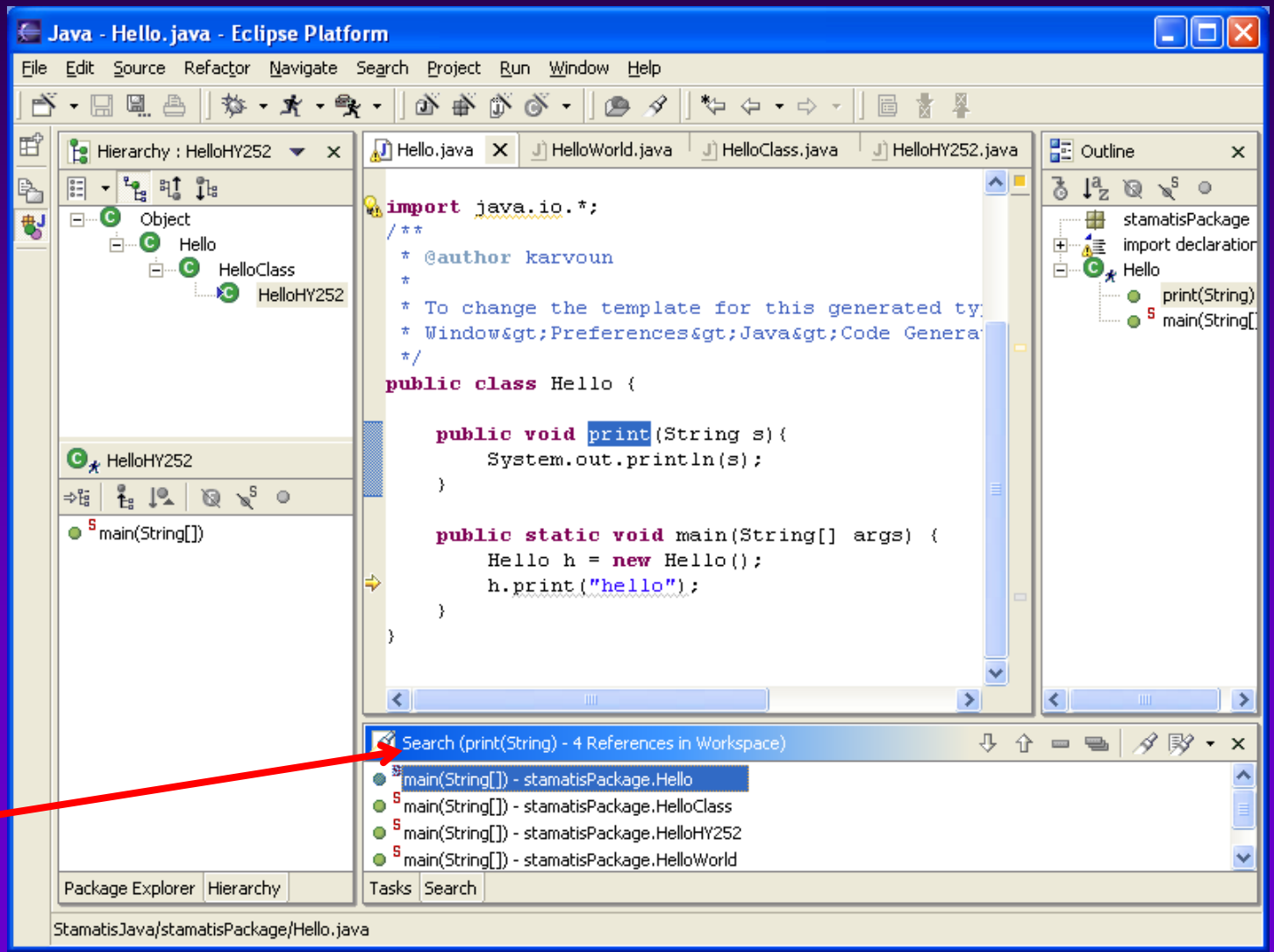


Search for Java Elements

- Select a type or method
- From its pop-up select Search > References > Workspace
- You will retrieve all it's references in the Search View



Search for Java Elements

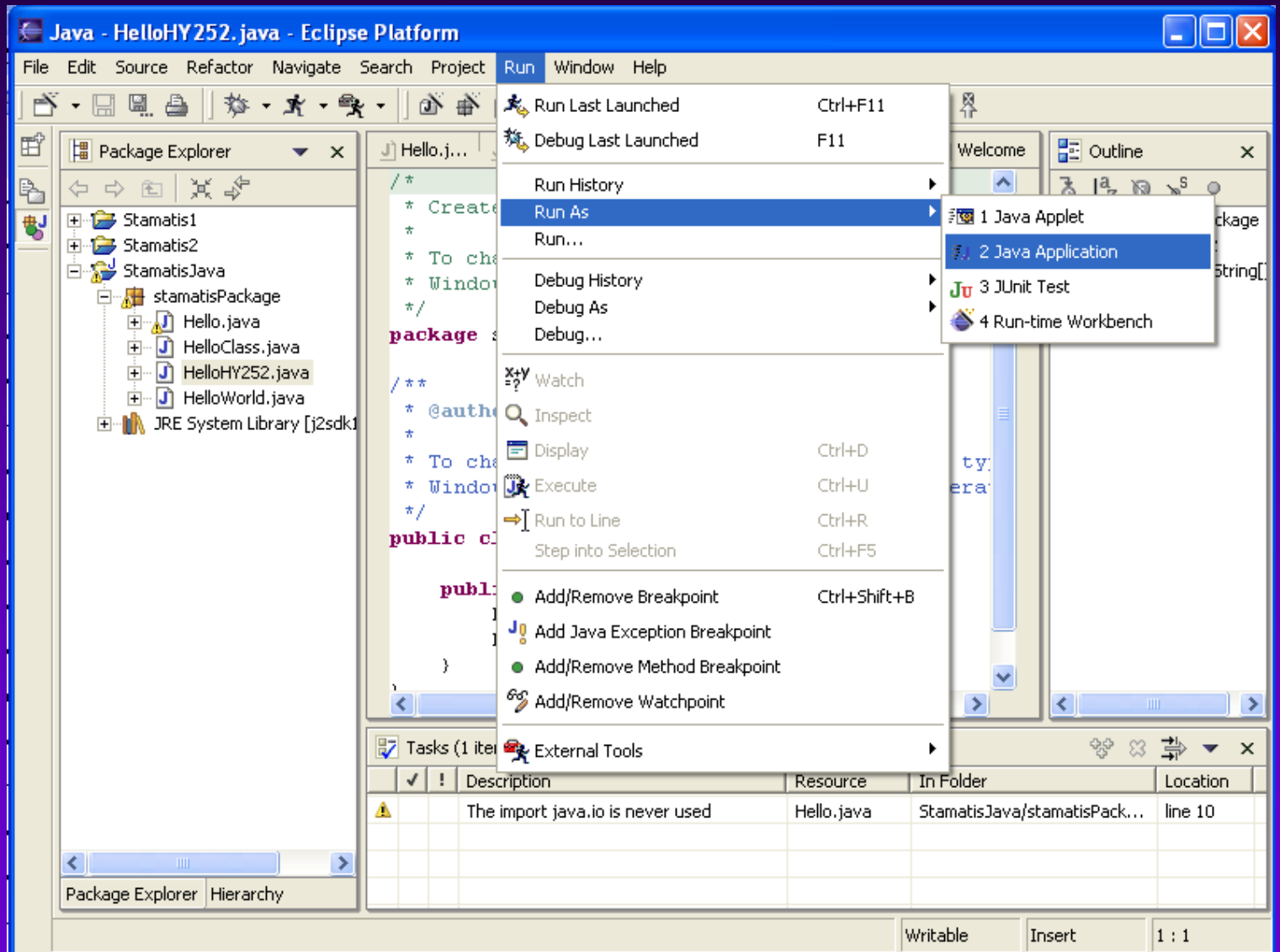


Search View

Run Java Programs

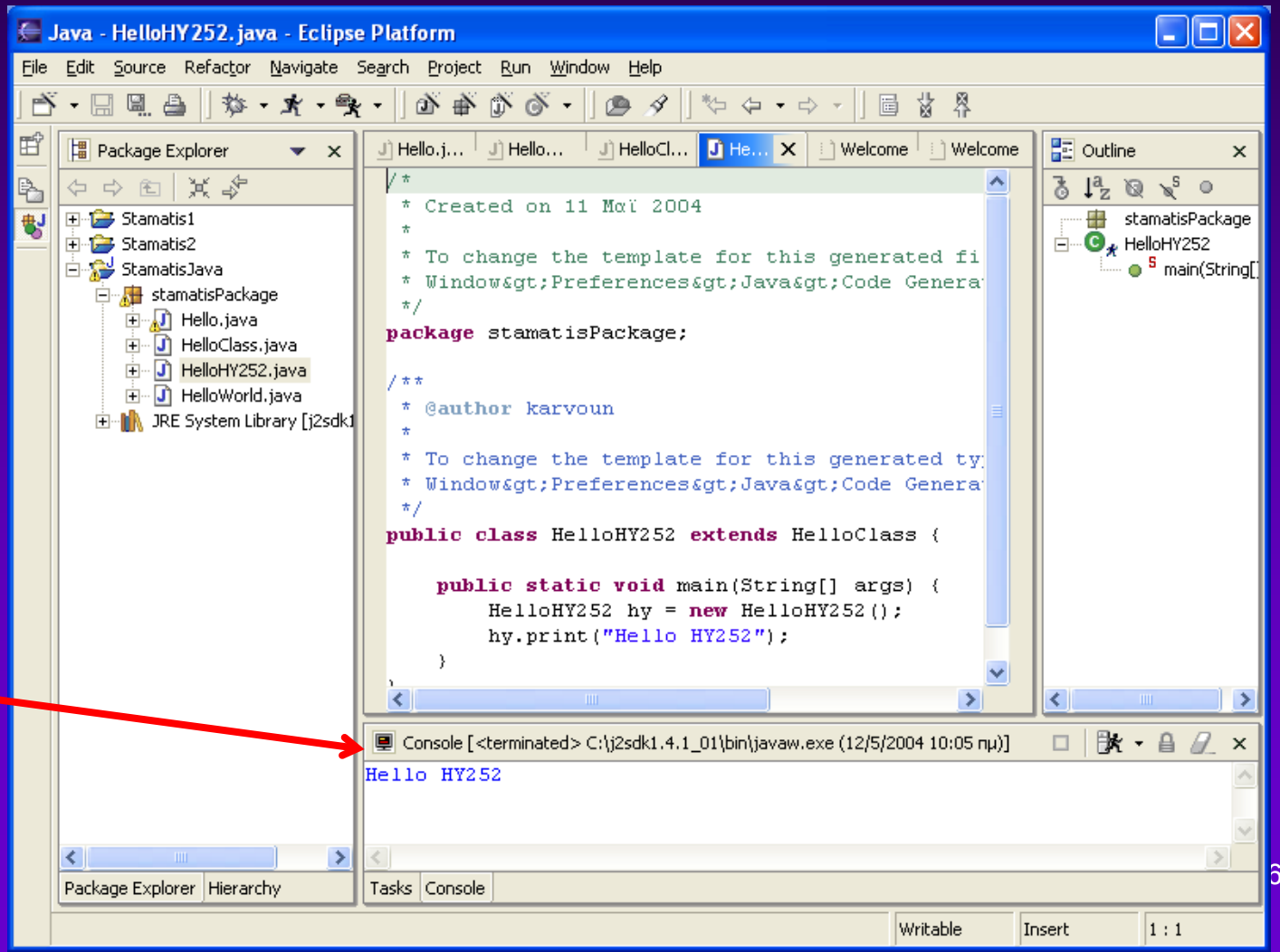
- Build is performed automatically upon resource modification
- If you want to do it explicitly select menu Project>Rebuild project
- Two ways to run a project:
  - Select menu Run > Run As > Java Application
  - Click on the “Run” icon in the toolbar

Run Java Programs



Run Java Programs

- The Console View shows up with the output!



Part4

UML

Eclipse UML Installation

- Exit eclipse
- Download EMF (Eclipse Modeling Framework) zip file from http://www.omondo.com/download/free/index.jsp
- Unzip EMF zip in %ECLIPSE_HOME% directory
- Re-start eclipse
- Exit eclipse

razy Browser - [Eclipse - Omondo - The Live UML Company - Download Free Edition]
Edit View Favorites History Groups Options Tools Tabs Help
http://www.omondo.com/download/free/index.jsp
1rstWAP contra - sports web cosmote - mail Customize Links ebPML.org Free Hotmail Google webmail-csd winbank Windows
tp://www.eclipse.o... Eclipse.org Main Pag... Eclipse.org Main Pag... Index of /home/liebe... Eclipse - Omondo - T... Eclipse - Omondo - T... Eclipse Java & UML i...

Zip File

Eclipse 2.1.1 is required.

- Get **Eclipse 2.1.1 R-2.1.1-200306271545**.

The **EMF (Eclipse Modeling Framework)** plugin is required.

- Get **EMF 1.1.0 Build 20030620 1105VL**.

Download this

The proper **GEF (Graphical Editor Framework)** plugin is included in our zip file.

Previous GEF installation should be uninstalled.

TYPE	VERSION	DATE	SIZE
EclipseUML	**1.2.1.20031103**	11/04/2003	9 932 806 Bytes
EclipseUML	**1.2.1.20030806**	08/06/2003	10 769 882 Bytes

Eclipse 2.1 is required.

- Get **Eclipse 2.1 R-2.1-200303272130**.

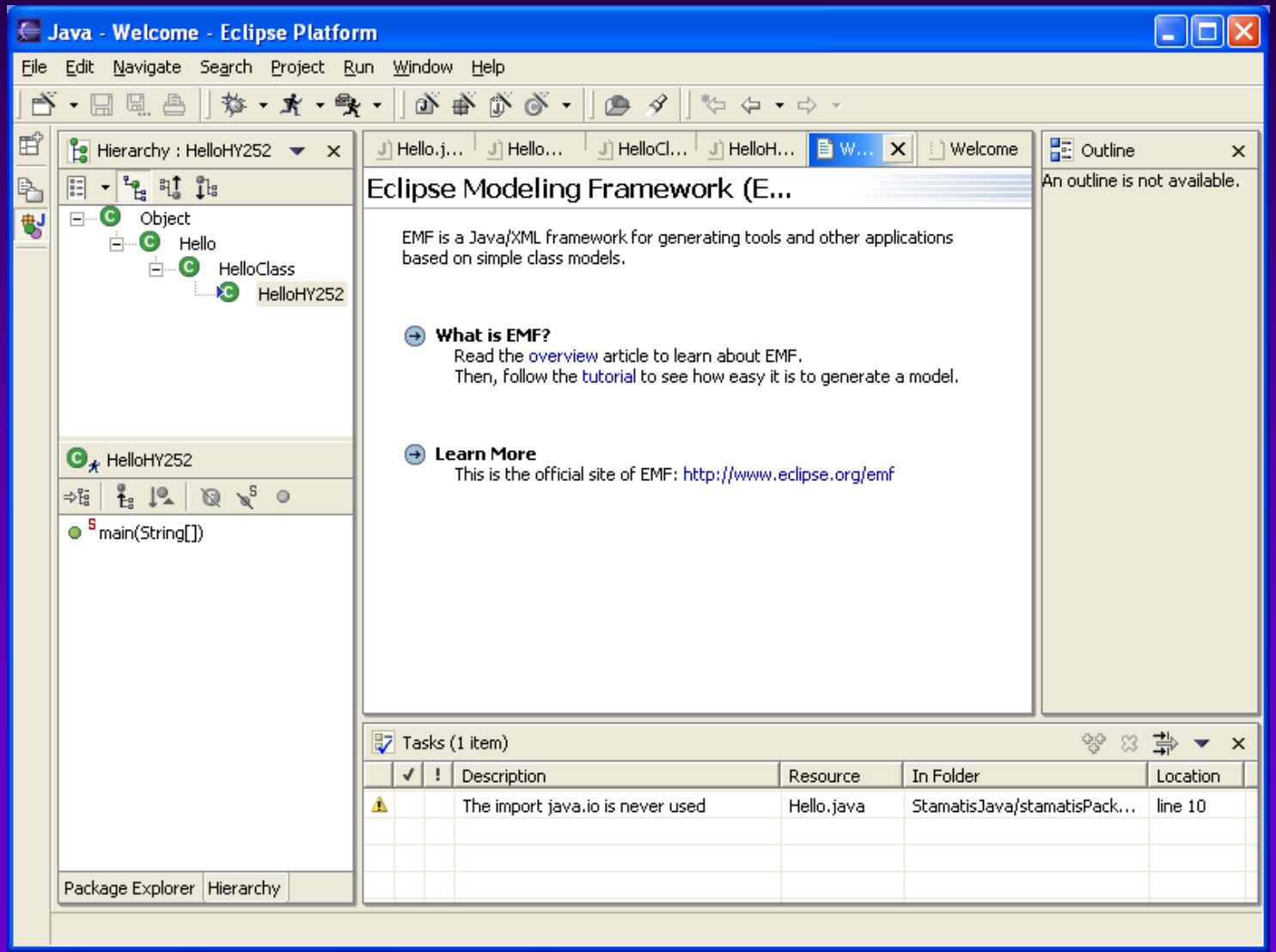
The **EMF (Eclipse Modeling Framework)** plugin is required.

- Get **EMF 1.1.0 Build 20030620 1105VL**.

The proper **GEF (Graphical Editor Framework)** plugin is included in our zip file.

100% Tab Counts:16 Internet

Eclipse UML Installation



Eclipse UML Installation

- Download Eclipse UML zip file from http://www.omondo.com/download/free/index.jsp
- Unzip Eclipse UML zip in %ECLIPSE_HOME% directory
- Re-start eclipse
- That's it!

- More details in:
  - http://www.omondo.com/faq/free/index.jsp#emfzipi
  - http://www.lifl.fr/~offroy/eclipse_HTML/internal/installation/Eclipse_Java_&_UML_install.html

razy Browser - [Eclipse - Omondo - The Live UML Company - Download Free Edition]
Edit View Favorites History Groups Options Tools Tabs Help
http://www.omondo.com/download/free/index.jsp
1rstWAP contra - sports web cosmote - mail Customize Links ebPML.org Free Hotmail Google webmail-csd winbank Windows
tp://www.eclipse.o... Eclipse.org Main Pag... Eclipse.org Main Pag... Index of /home/liebe... Eclipse - Omondo - T... Eclipse - Omondo - T... Eclipse Java & UML i...

Zip File

Eclipse 2.1.1 is required.

- Get **Eclipse 2.1.1 R-2.1.1-200306271545**.

The **EMF (Eclipse Modeling Framework)** plugin is required.

- Get **EMF 1.1.0 Build 20030620 1105VL**.

The proper **GEF (Graphical Editor Framework)** plugin is included in our zip file.

Download This

Previous GEF installation should be uninstalled.

TYPE	VERSION	DATE	SIZE
EclipseUML	**1.2.1.20031103**	11/04/2003	9 932 806 Bytes
EclipseUML	**1.2.1.20030806**	08/06/2003	10 769 882 Bytes

Eclipse 2.1 is required.

- Get **Eclipse 2.1 R-2.1-200303272130**.

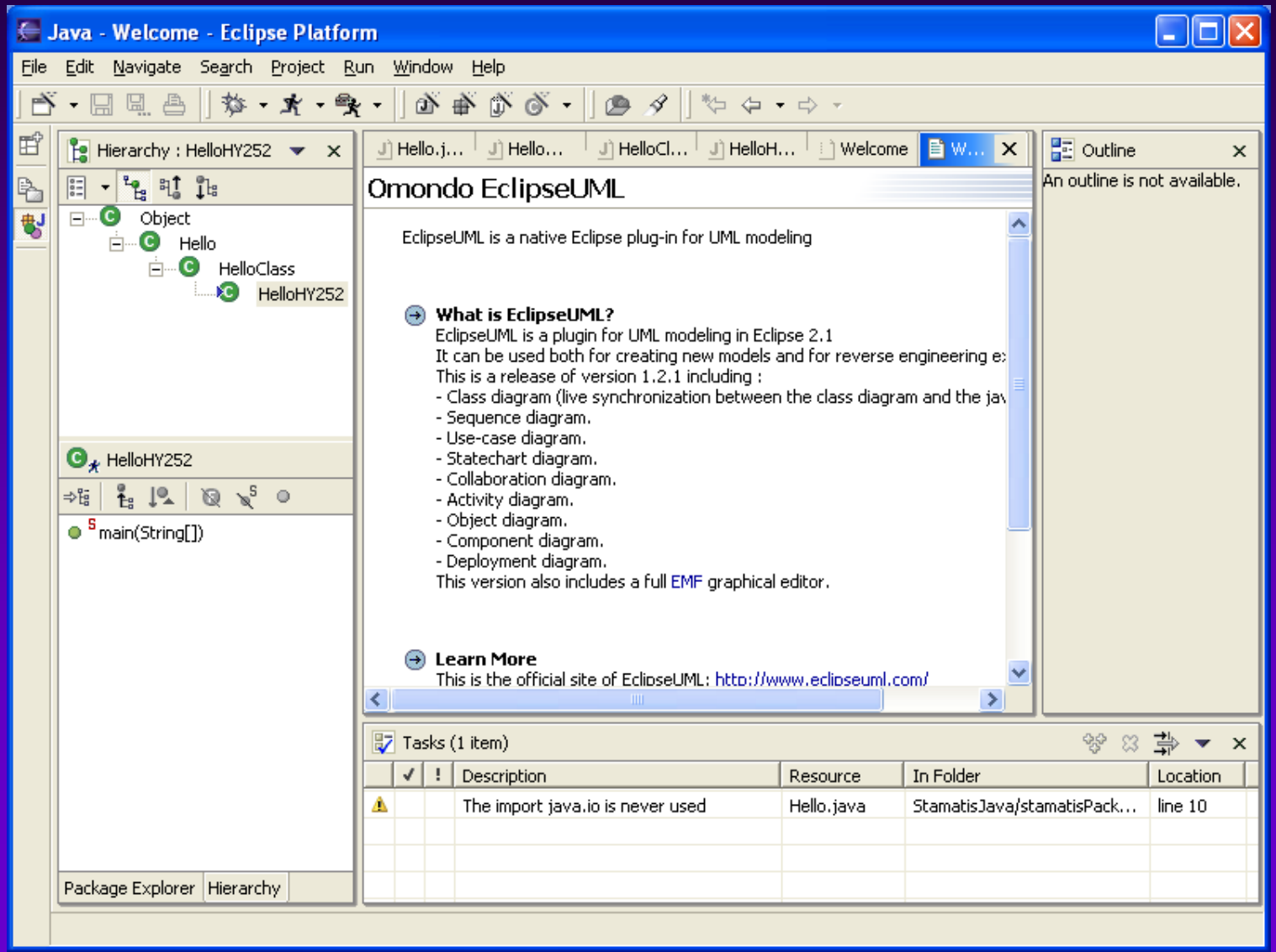
The **EMF (Eclipse Modeling Framework)** plugin is required.

- Get **EMF 1.1.0 Build 20030620 1105VL**.

The proper **GEF (Graphical Editor Framework)** plugin is included in our zip file.

100% Tab Counts:16 Internet

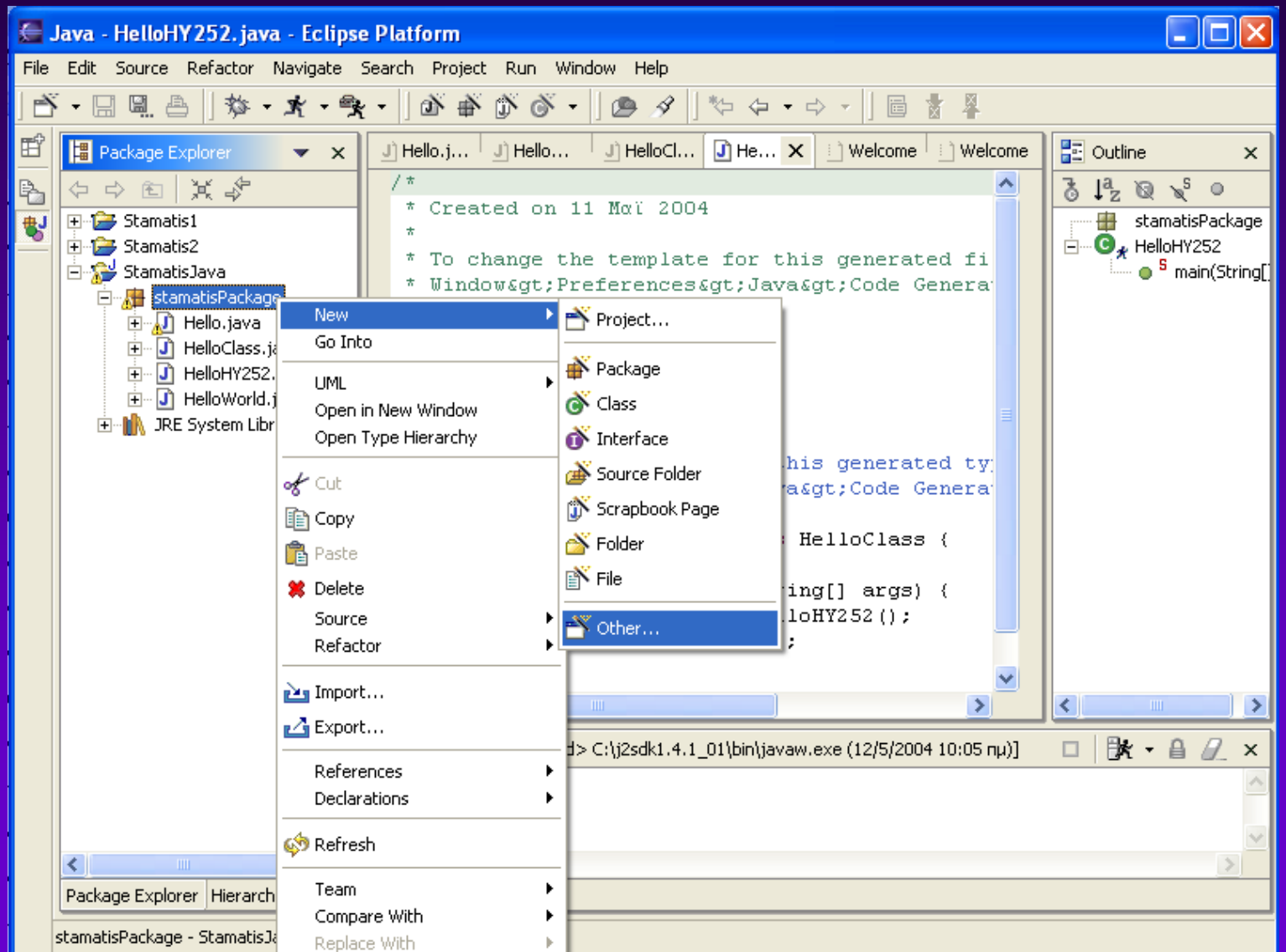
Eclipse UML Installation



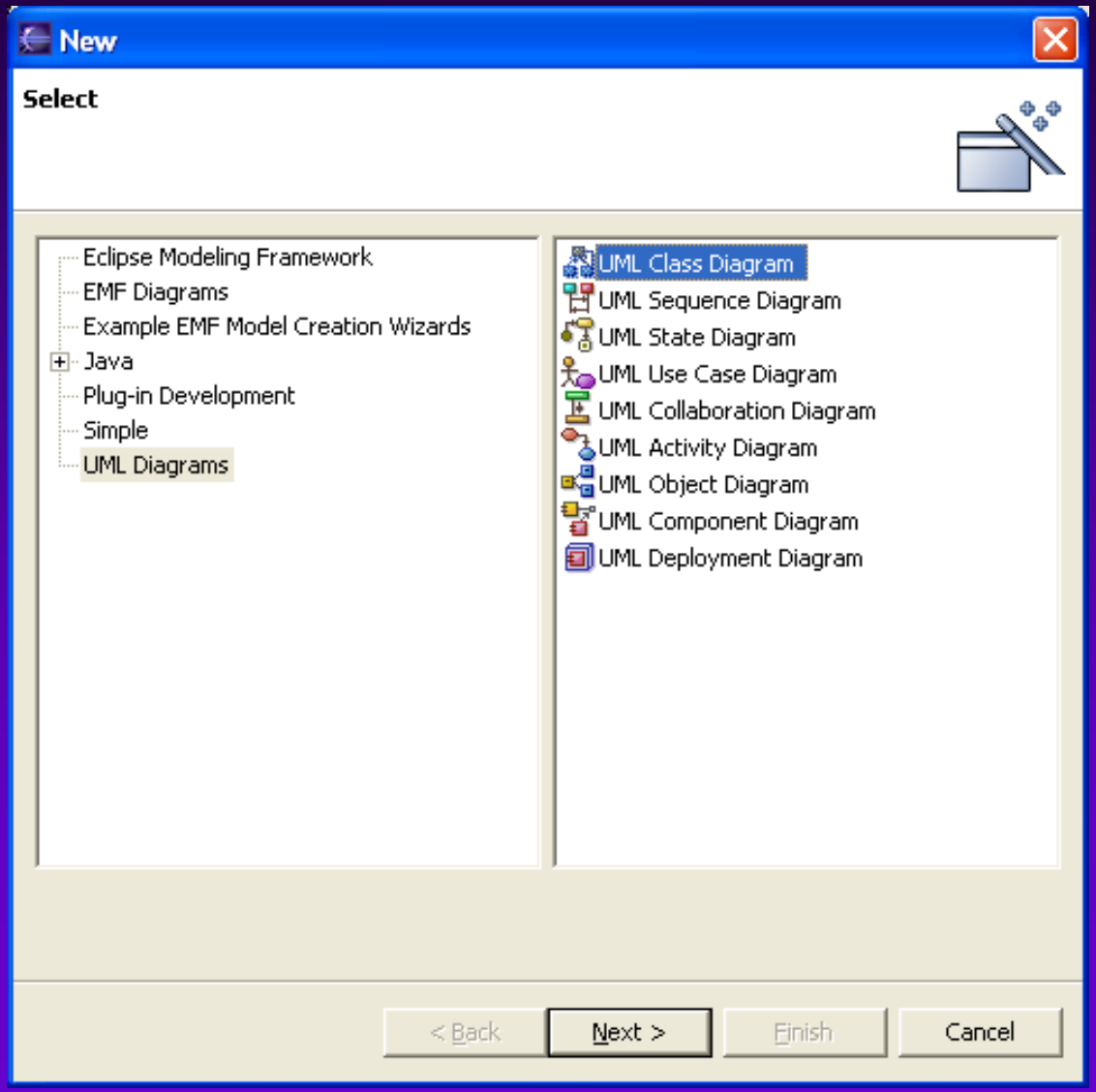
Create your first class diagram

- Select a package
- From it context menu (right-click) select New>Other>UML->UML Class Diagram

Create your first class diagram

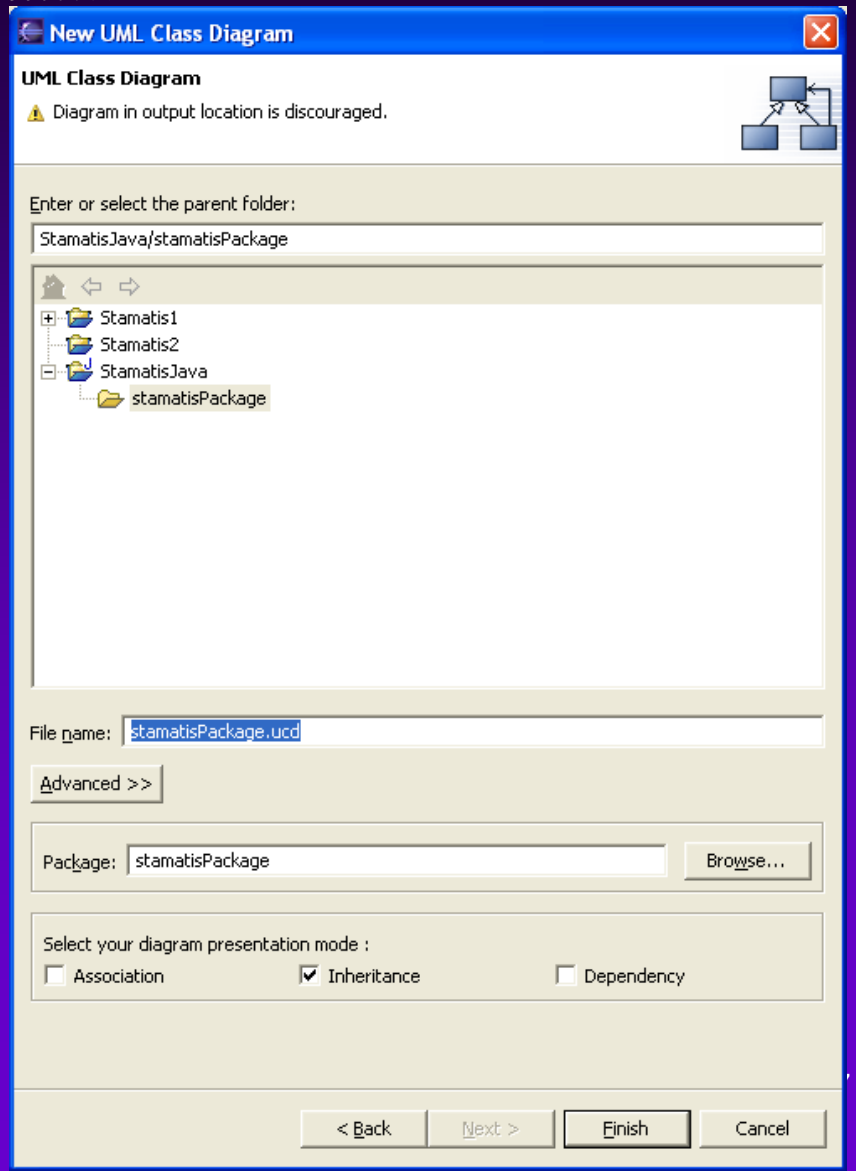


Create your first class diagram



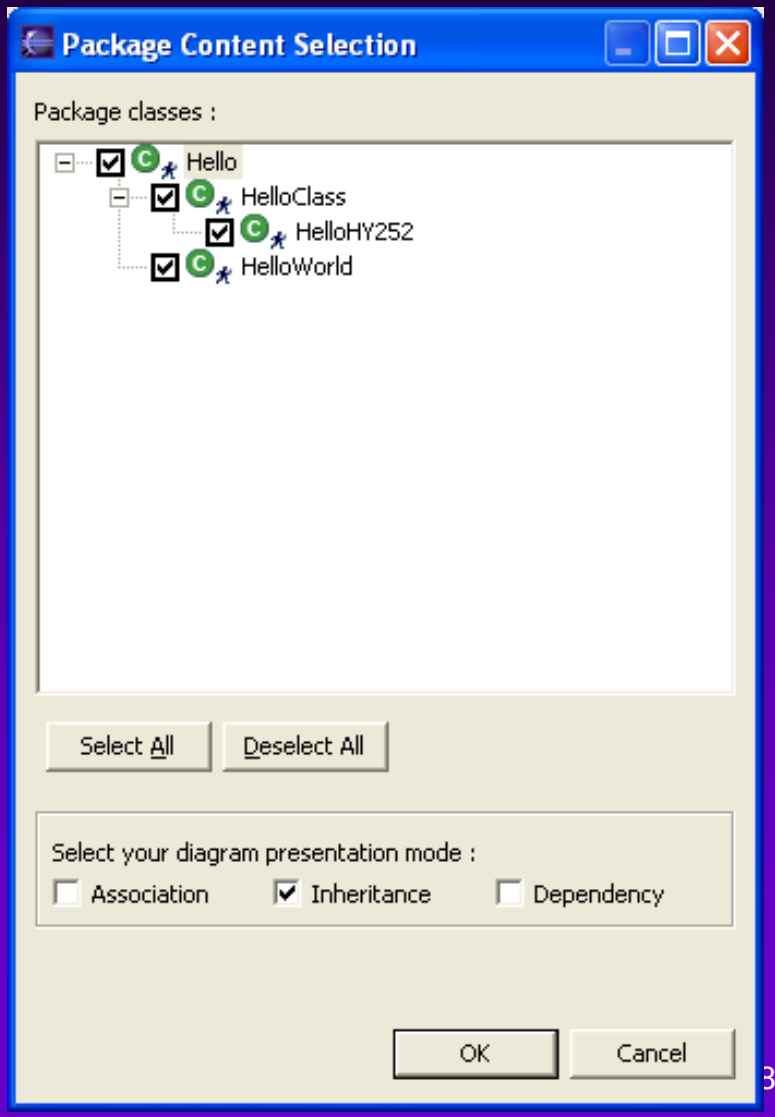
Create your first class diagram

- Choose the kind of relationship you want to be depicted
  - Association
  - Inheritance
  - Dependency



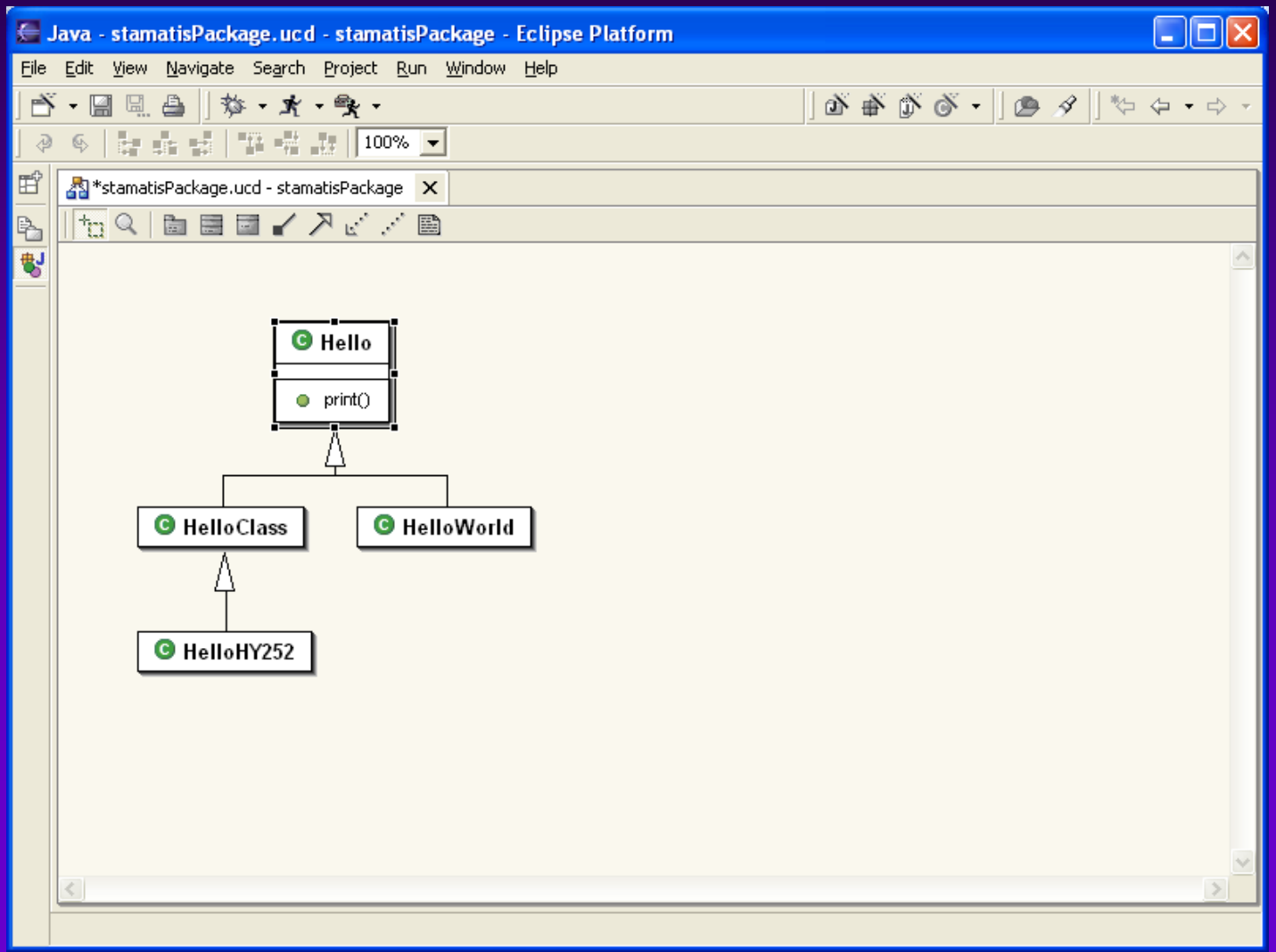
Create your first class diagram

- Choose the classes that will participate in the diagram



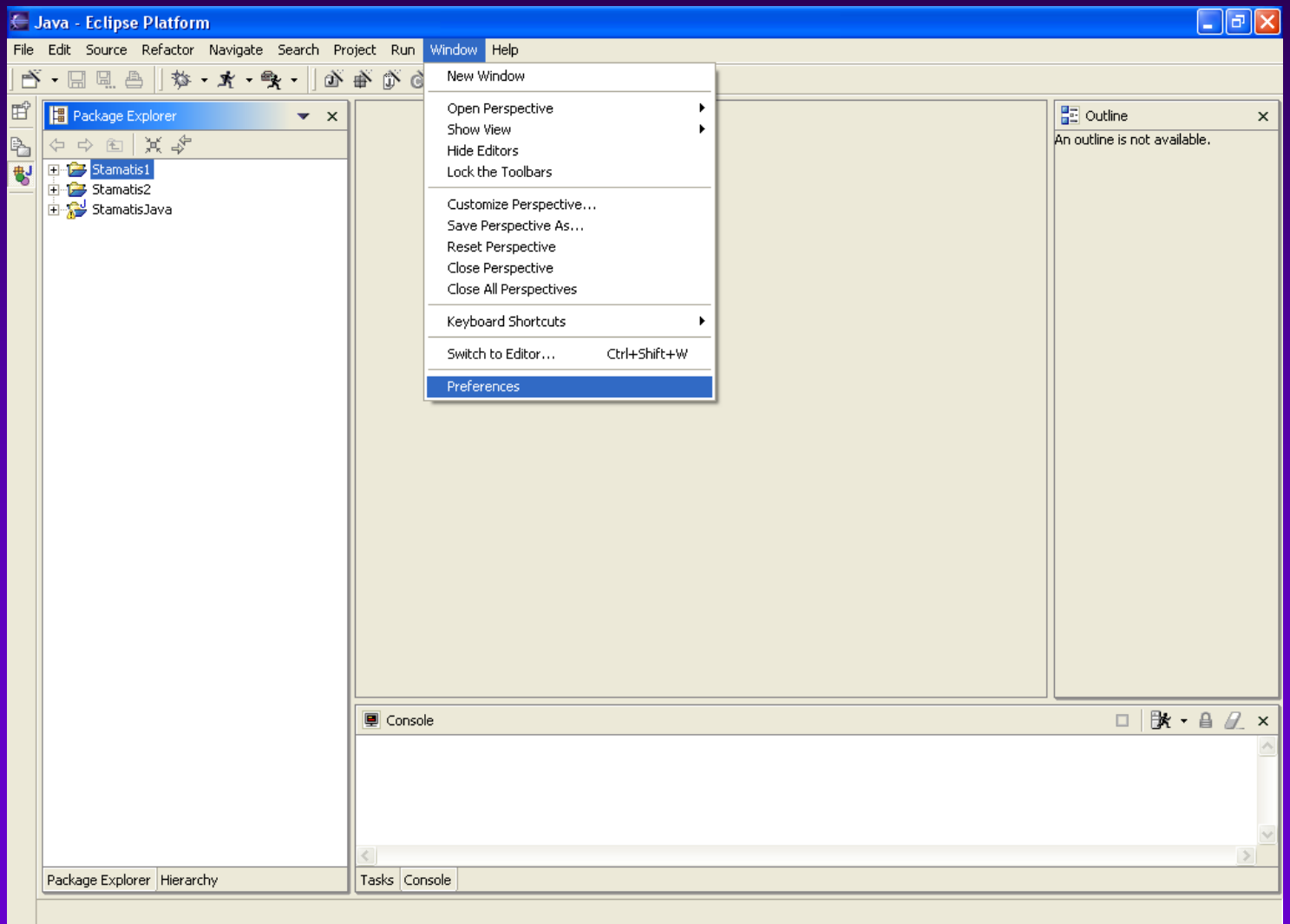
Create your first class diagram

- Here is it!



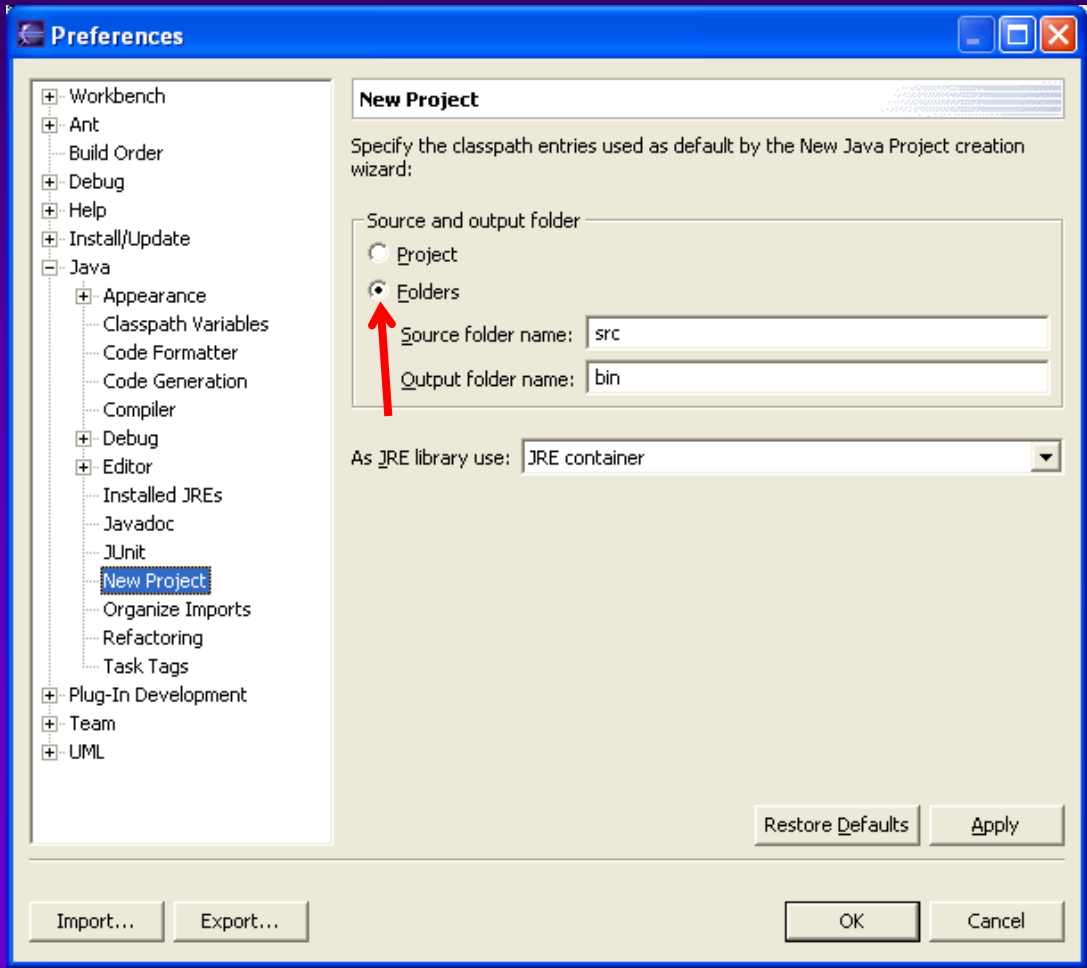
Prepare Workbench for UML design

- Select menu Window > Preferences



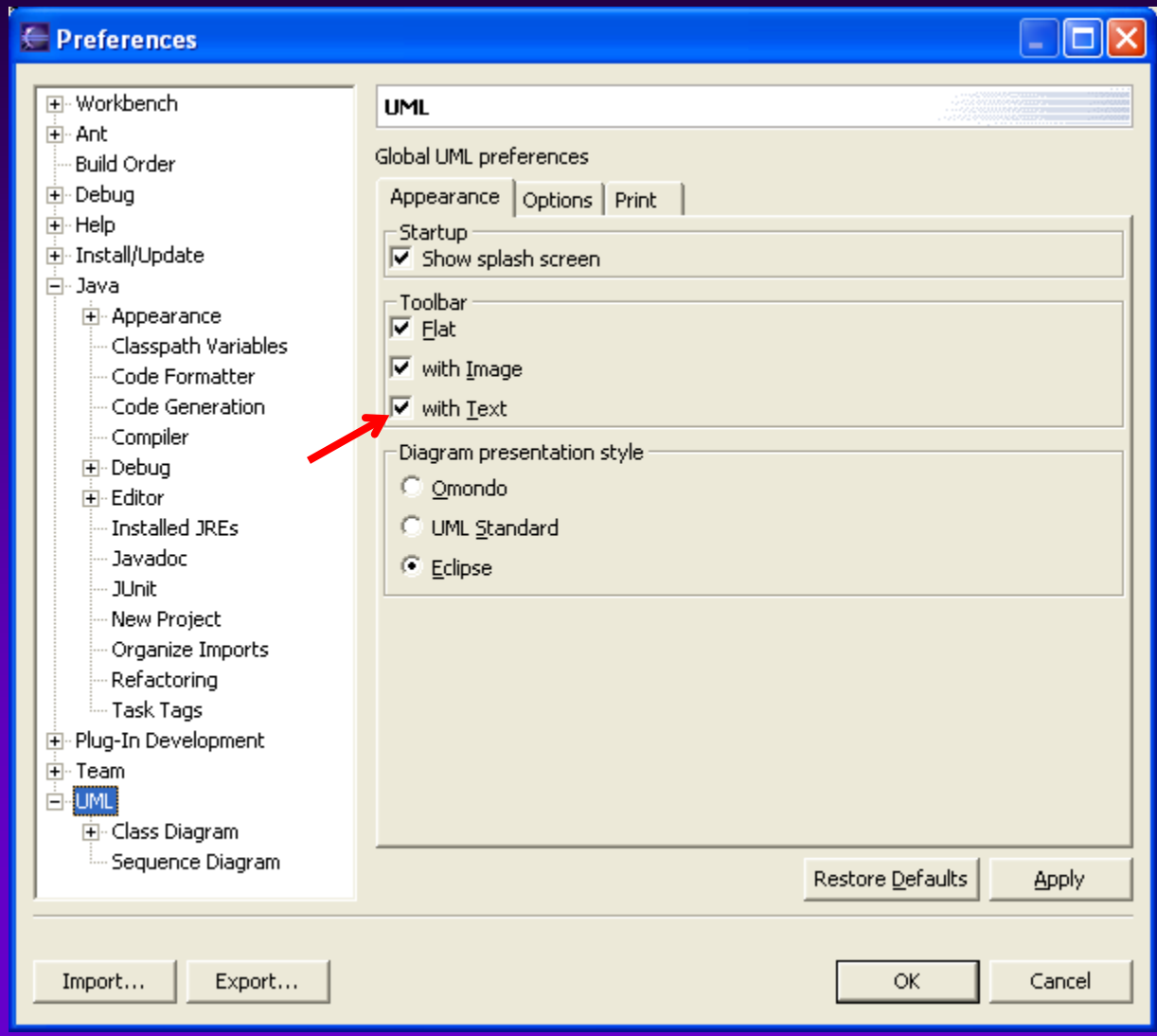
Prepare Workbench for UML design

- We will create a src and a bin output for the UML project
- Select Java > New Projects
- Click on “Folders” checkbox



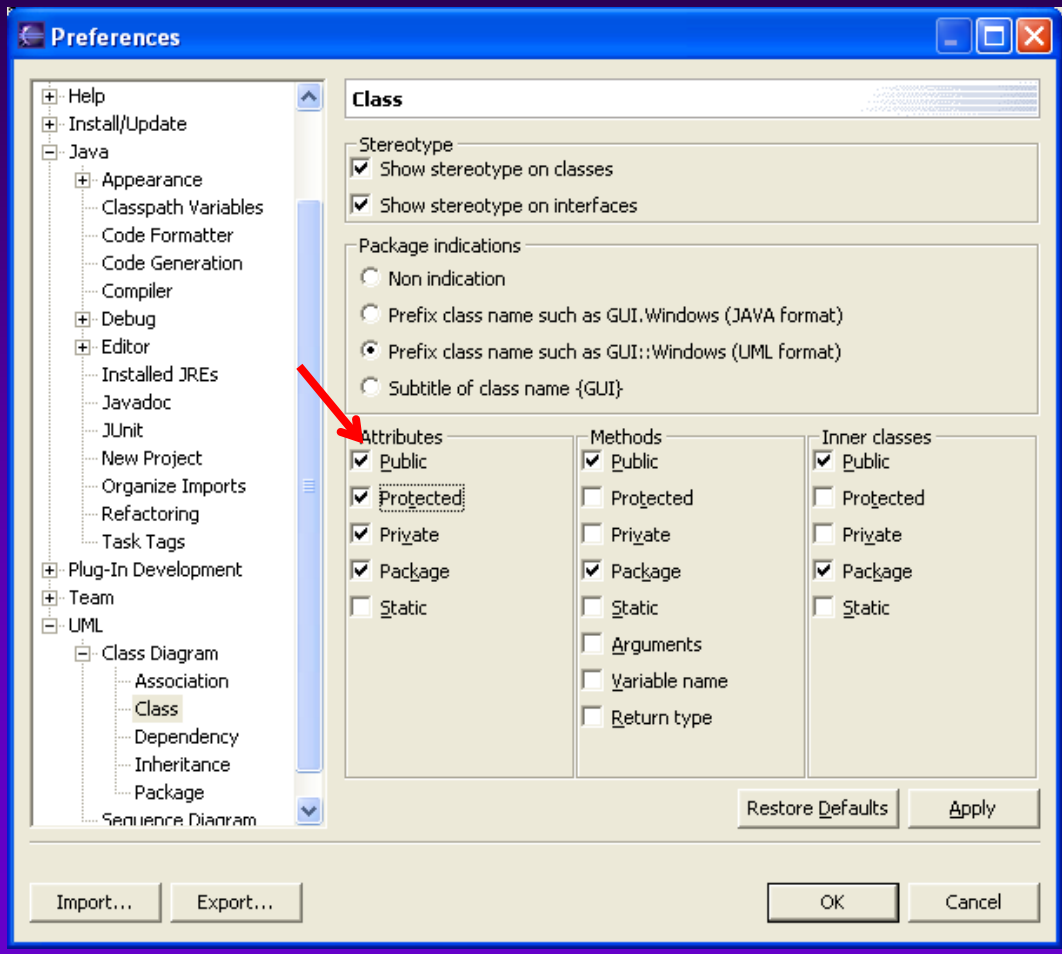
Prepare Workbench for UML design

- Select “UML”
- Click on checkbox “with text”



Prepare Workbench for UML design

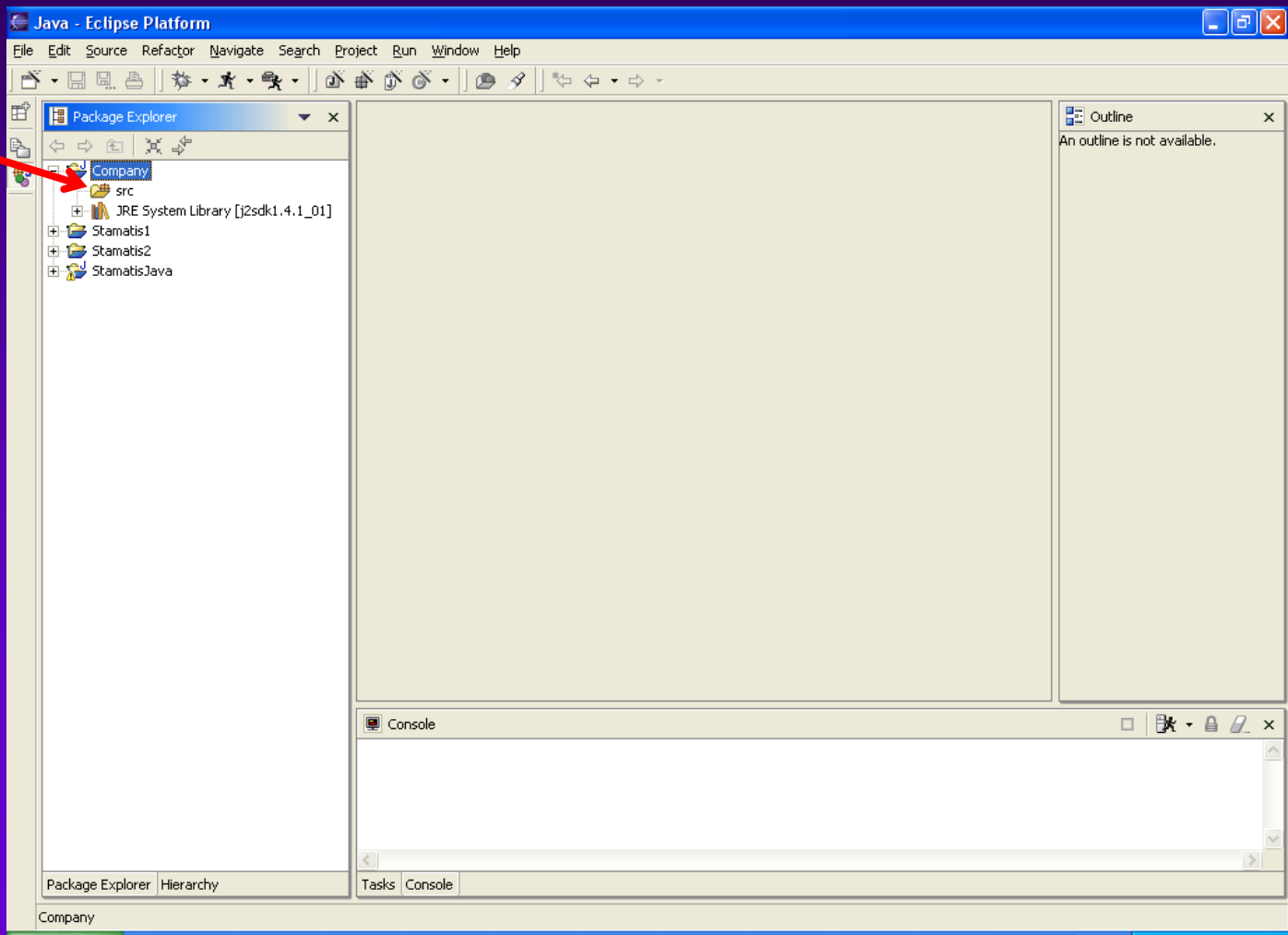
- Select UML>Class Diagrams>Class
- In attributes pane check boxes “Public”, ”Private”, “Protected” and “Package”



Eclipse UML Class Diagrams

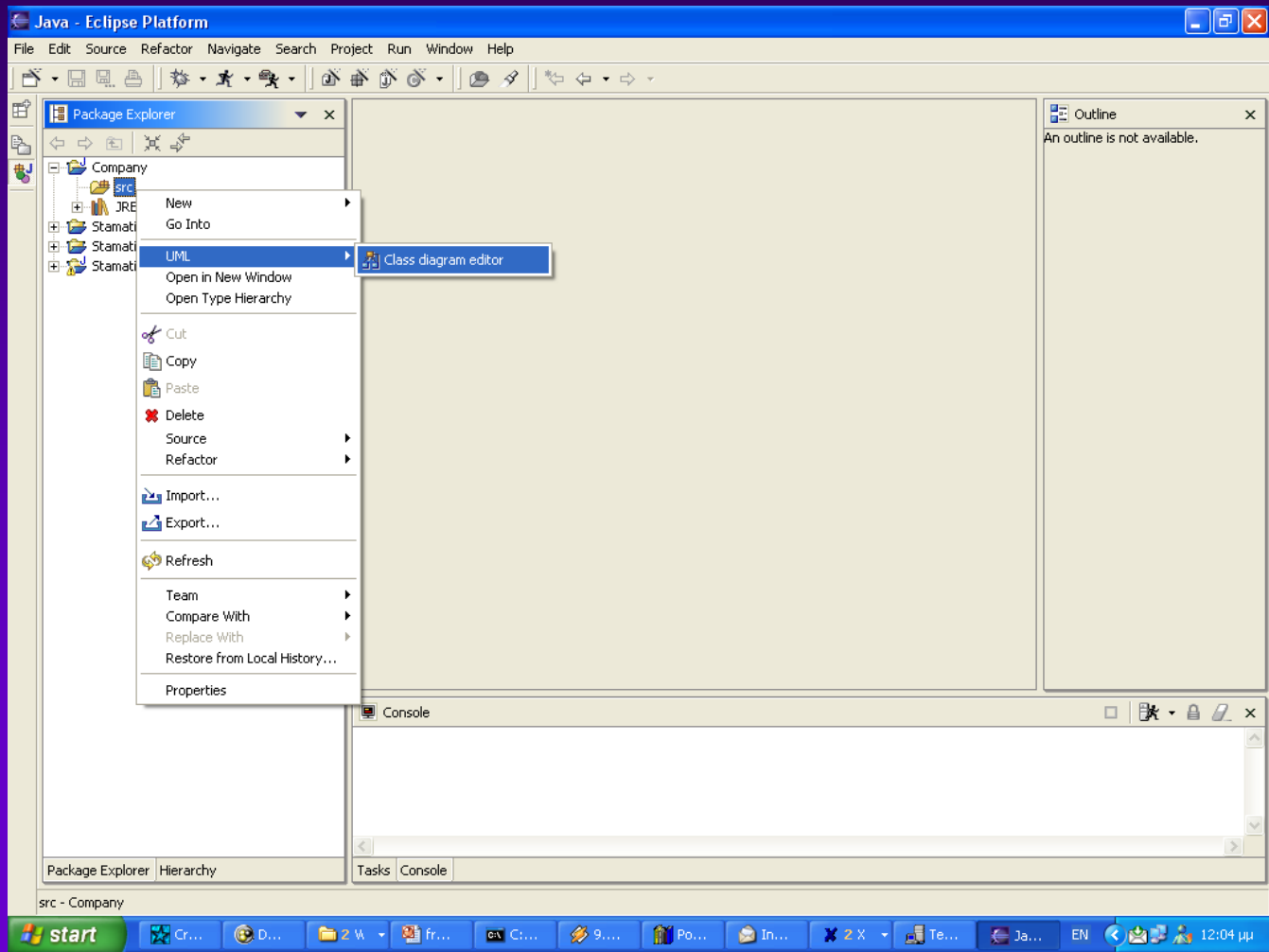
- Select Java Perspective
- Create a new Java project named “Company”

The src folder is created automatically

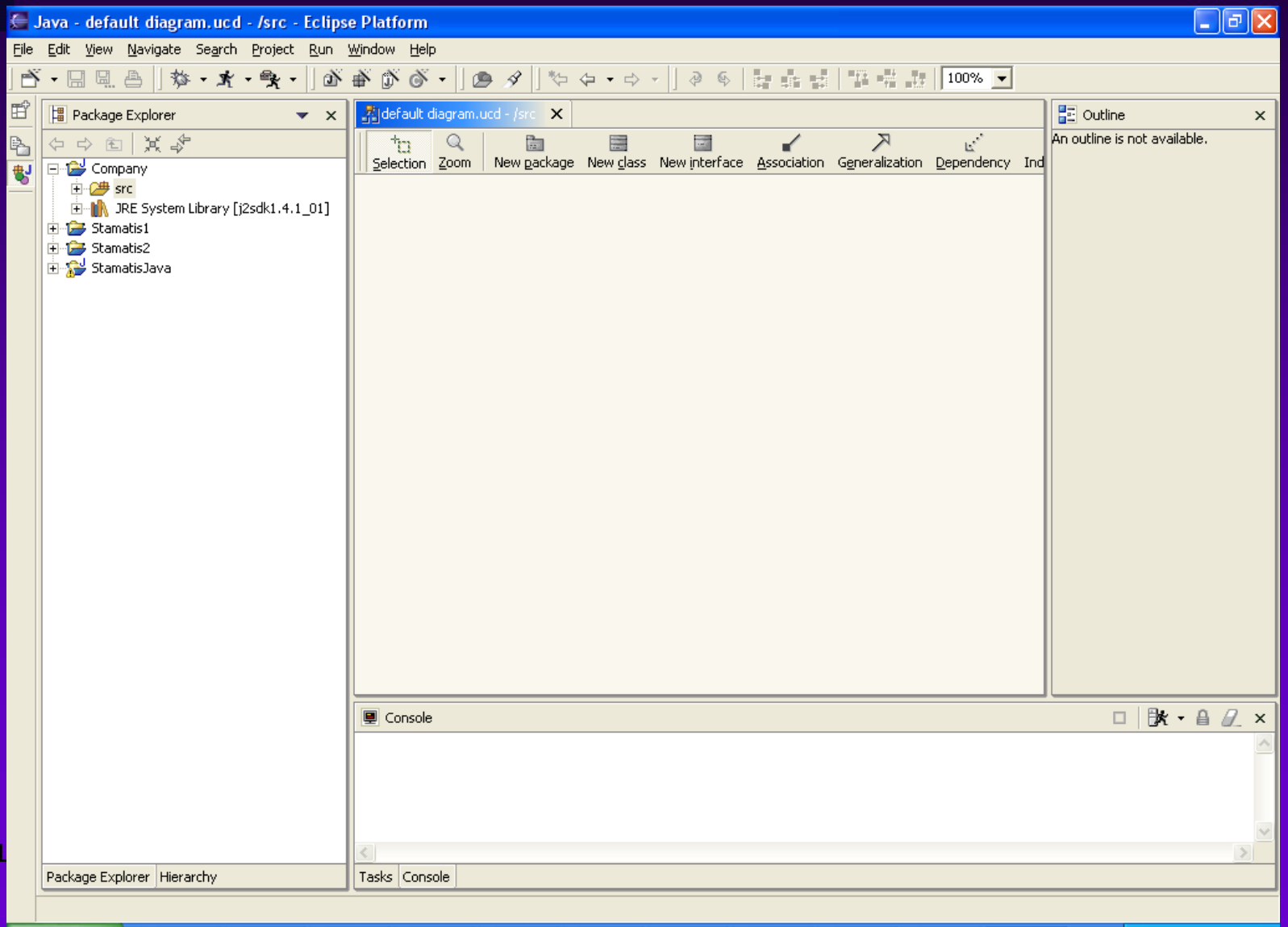


Eclipse UML Class Diagrams

- Select src folder
- From it's pop-up menu select UML>Class diagram editor

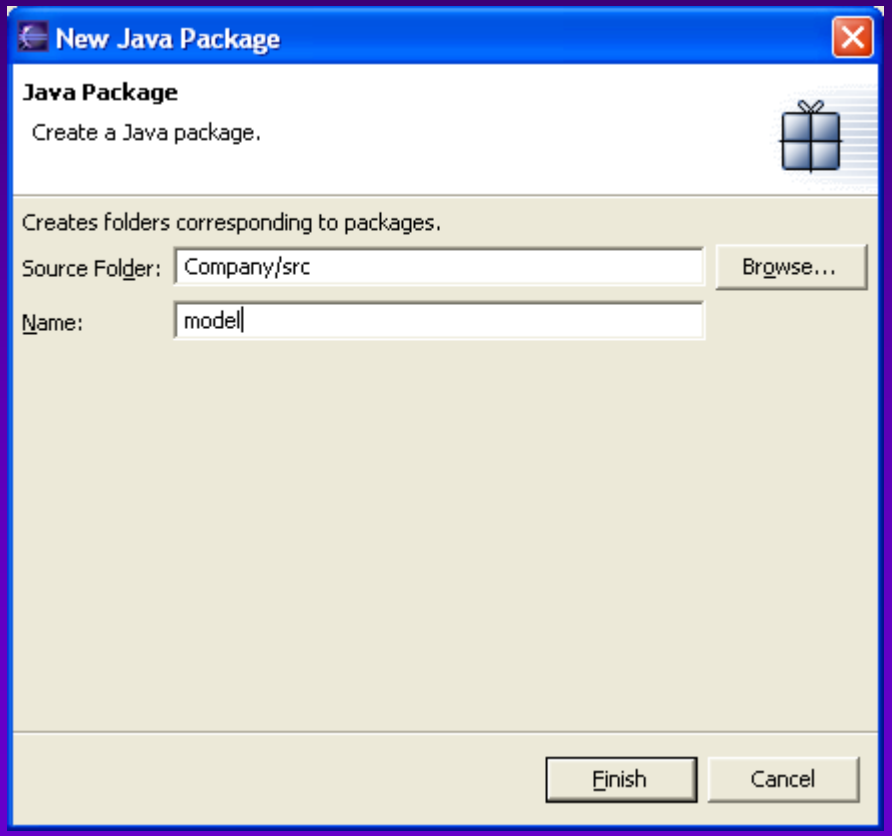


Eclipse UML Class Diagrams

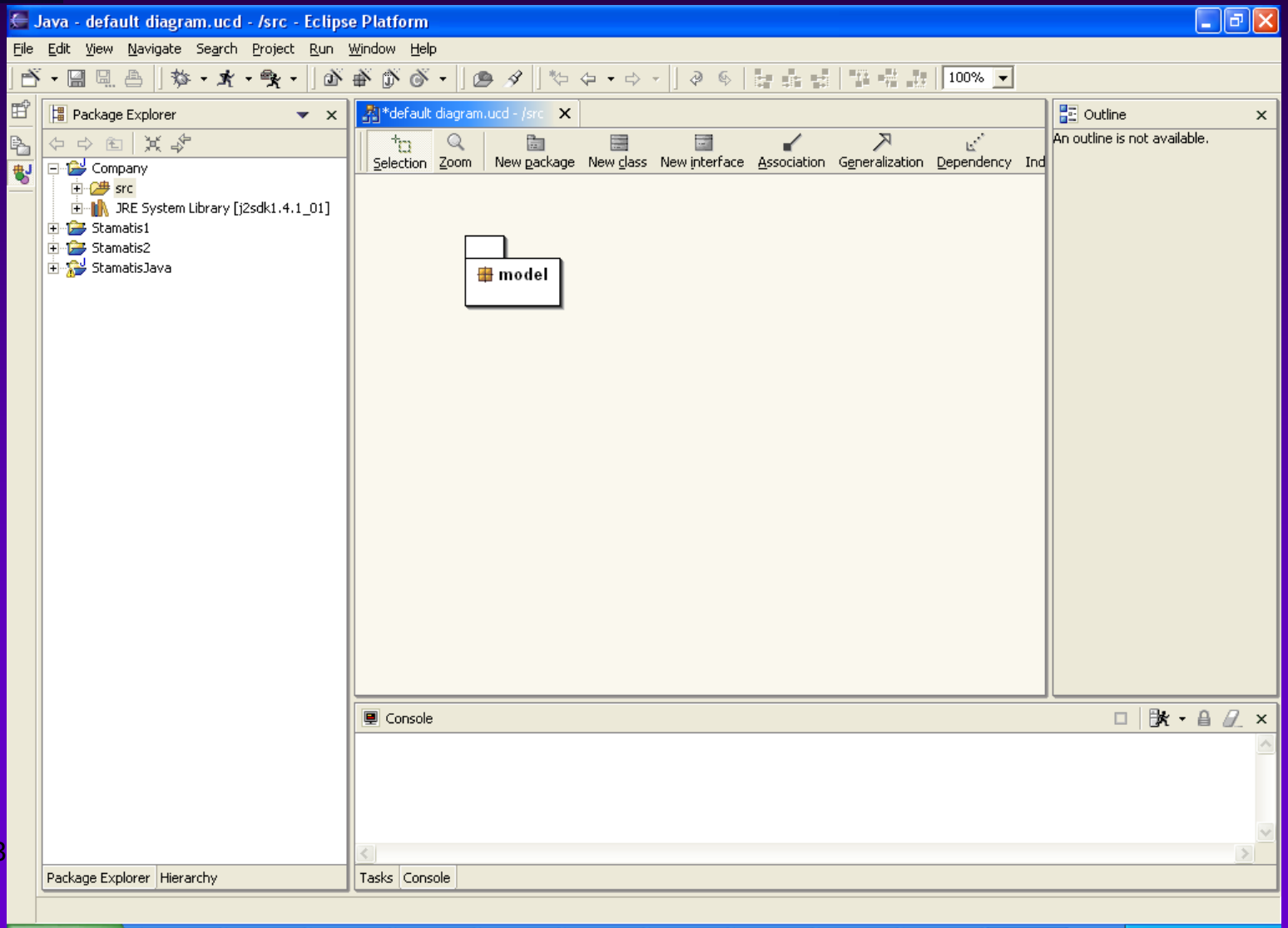


Eclipse UML Class Diagrams

- Create a New Package.
- Select “new package” in the toolbar and click on the diagram.
- Name it “model”

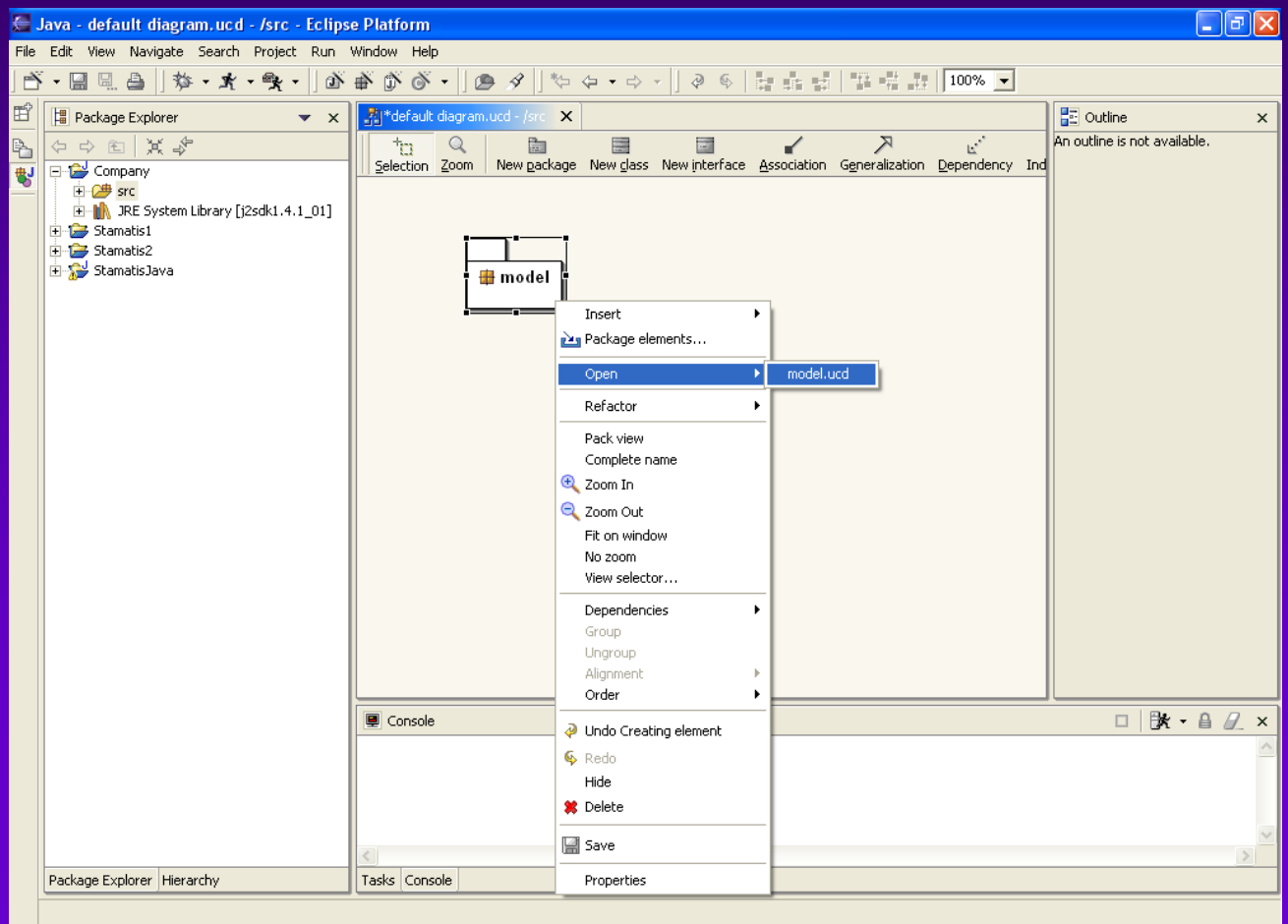


Eclipse UML Class Diagrams

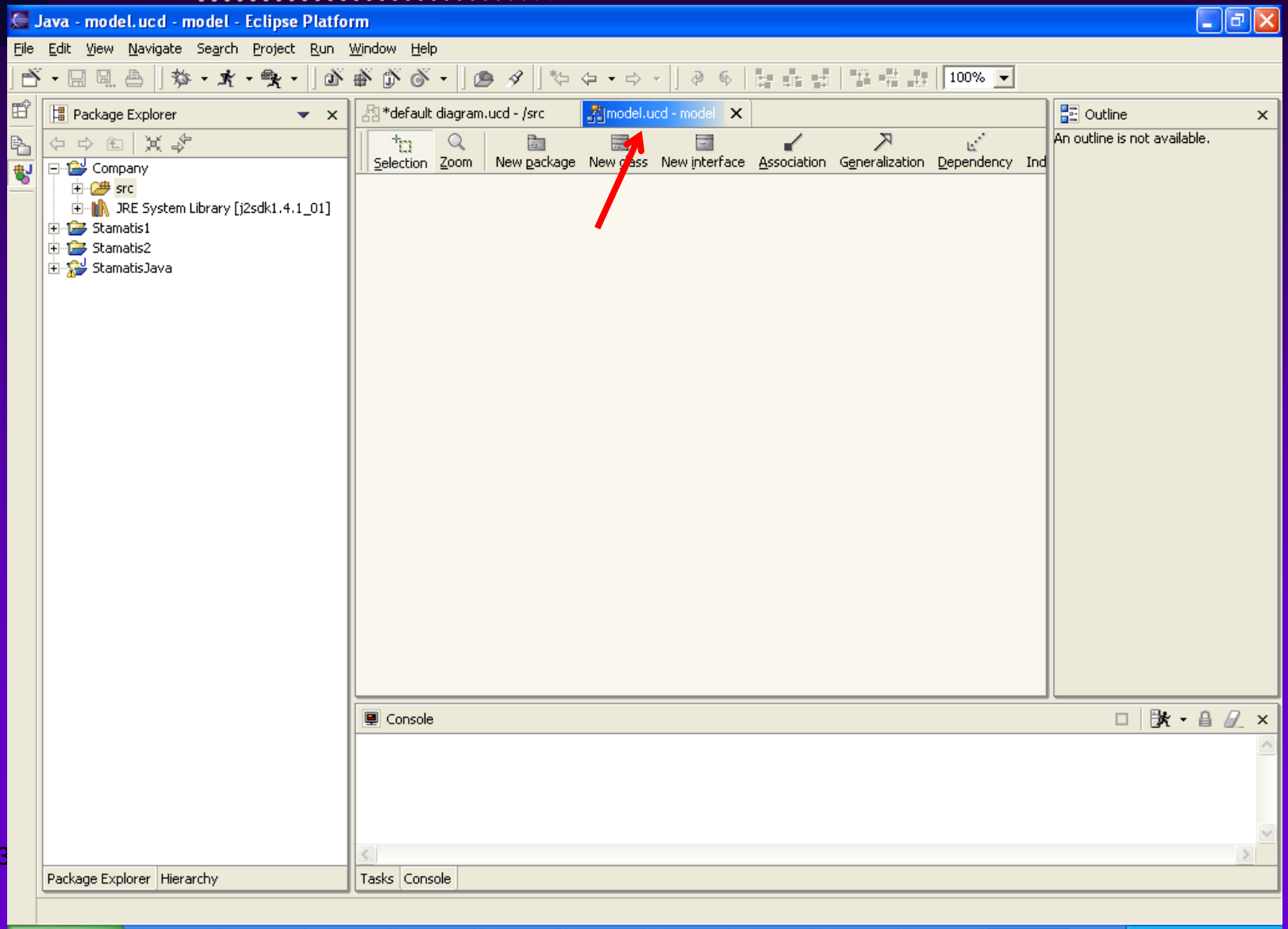


Eclipse UML Class Diagrams

- Create a new Class Diagram inside the “model” Package
- Right-click on model then Open>model.ucd (ucd is the class diagram extension).

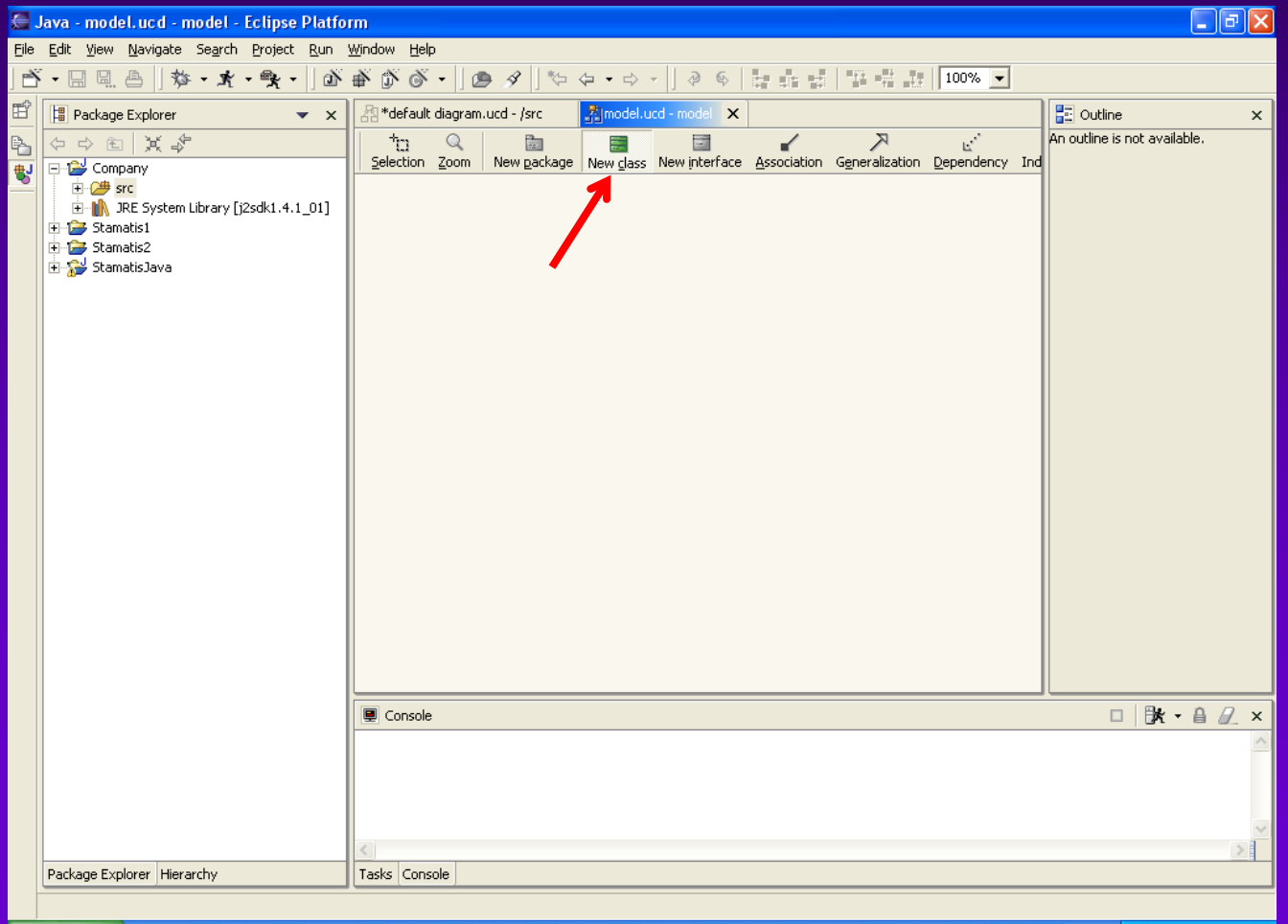


Eclipse UML Class Diagrams



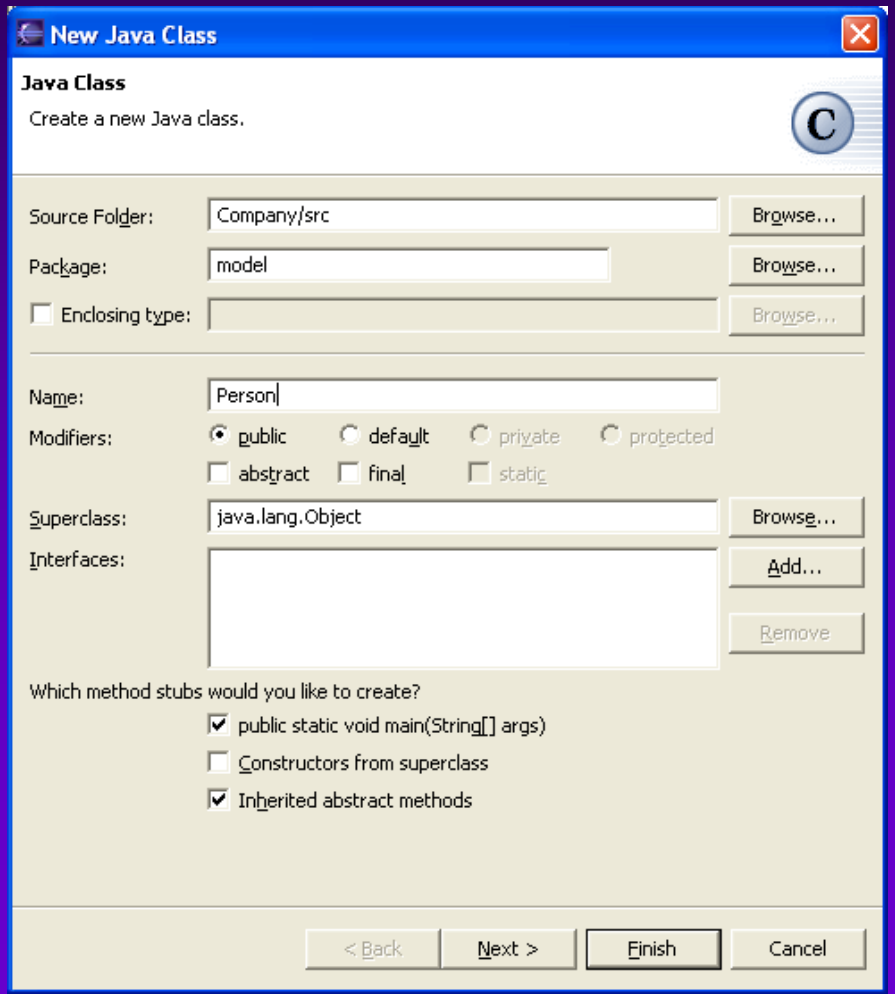
Eclipse UML Class Diagrams

- Create a New class
- Select New class in the toolbar and click on the diagram.

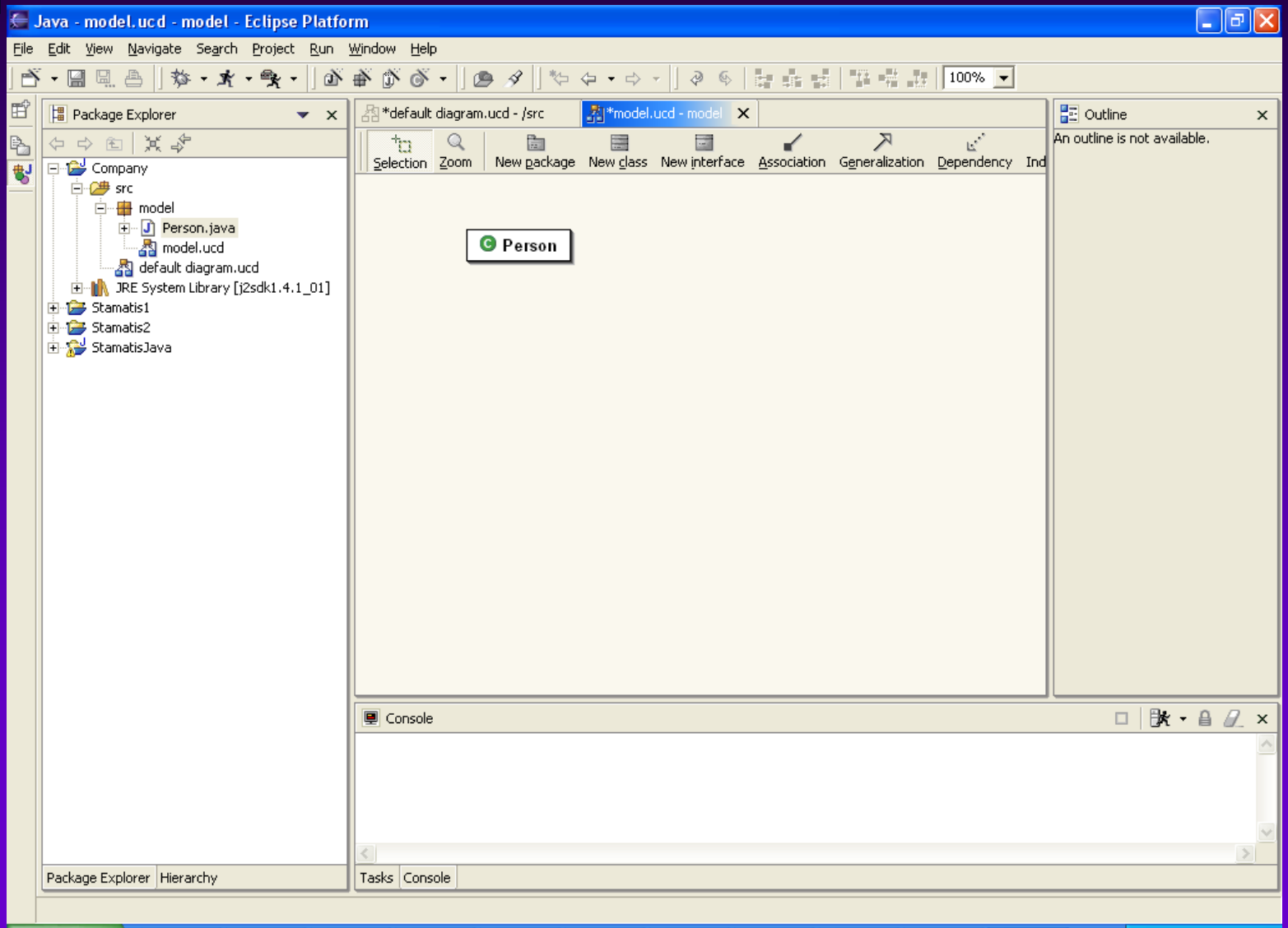


Eclipse UML Class Diagrams

- Enter the name of the New Java Class and check
- Enter Person and click on the Finish button.

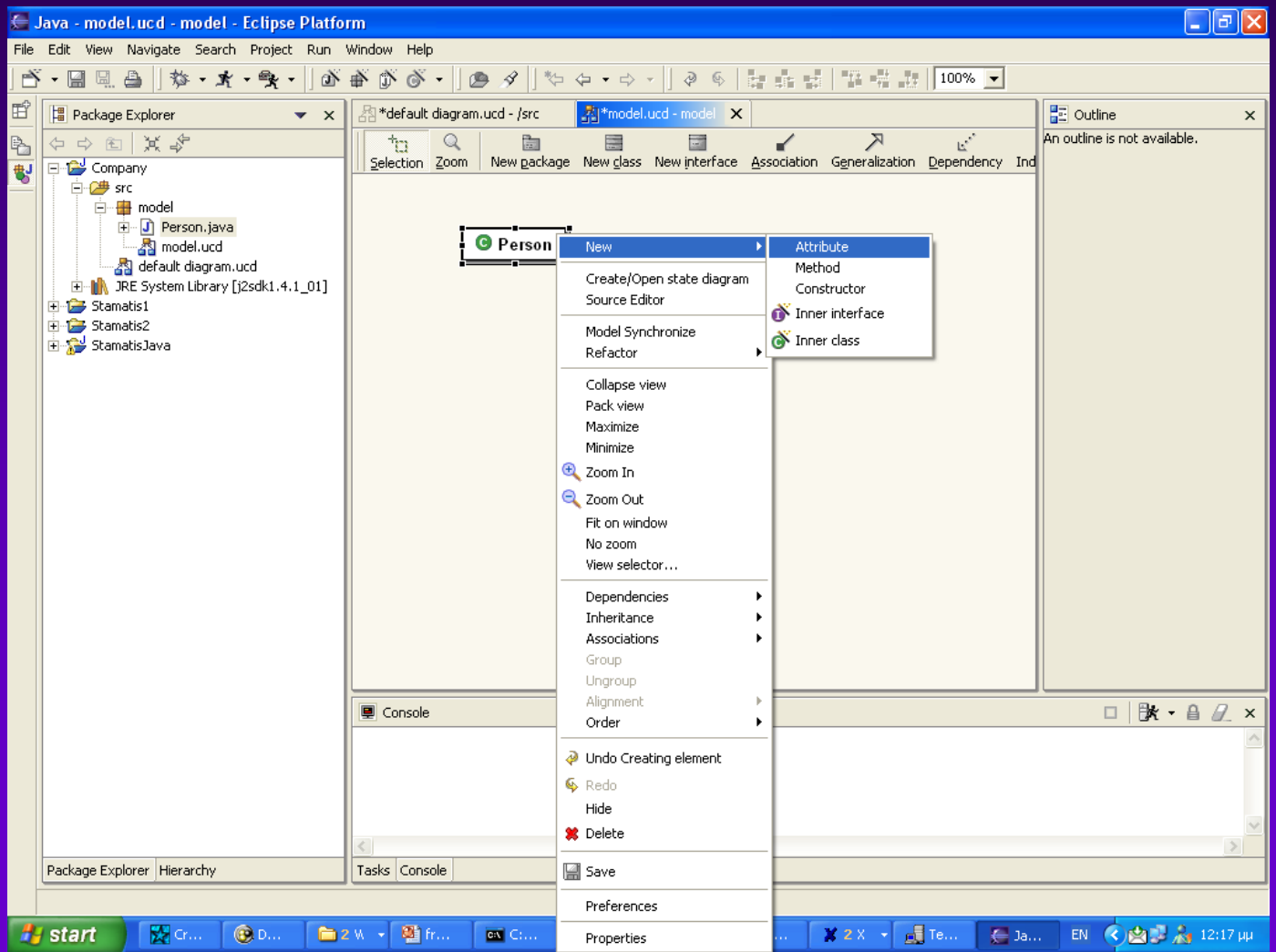


Eclipse UML Class Diagrams



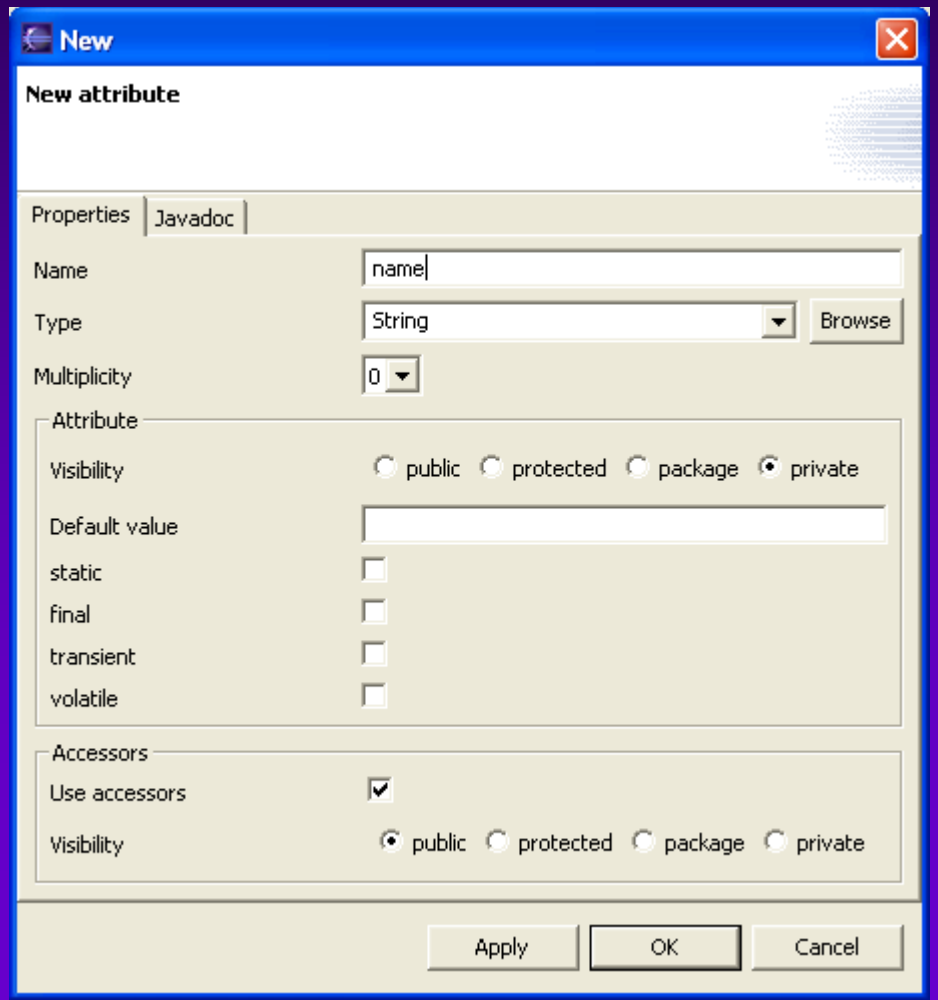
Eclipse UML Class Diagrams

- We are going to add two new Attributes.
- Click on Person>New>Attribute.

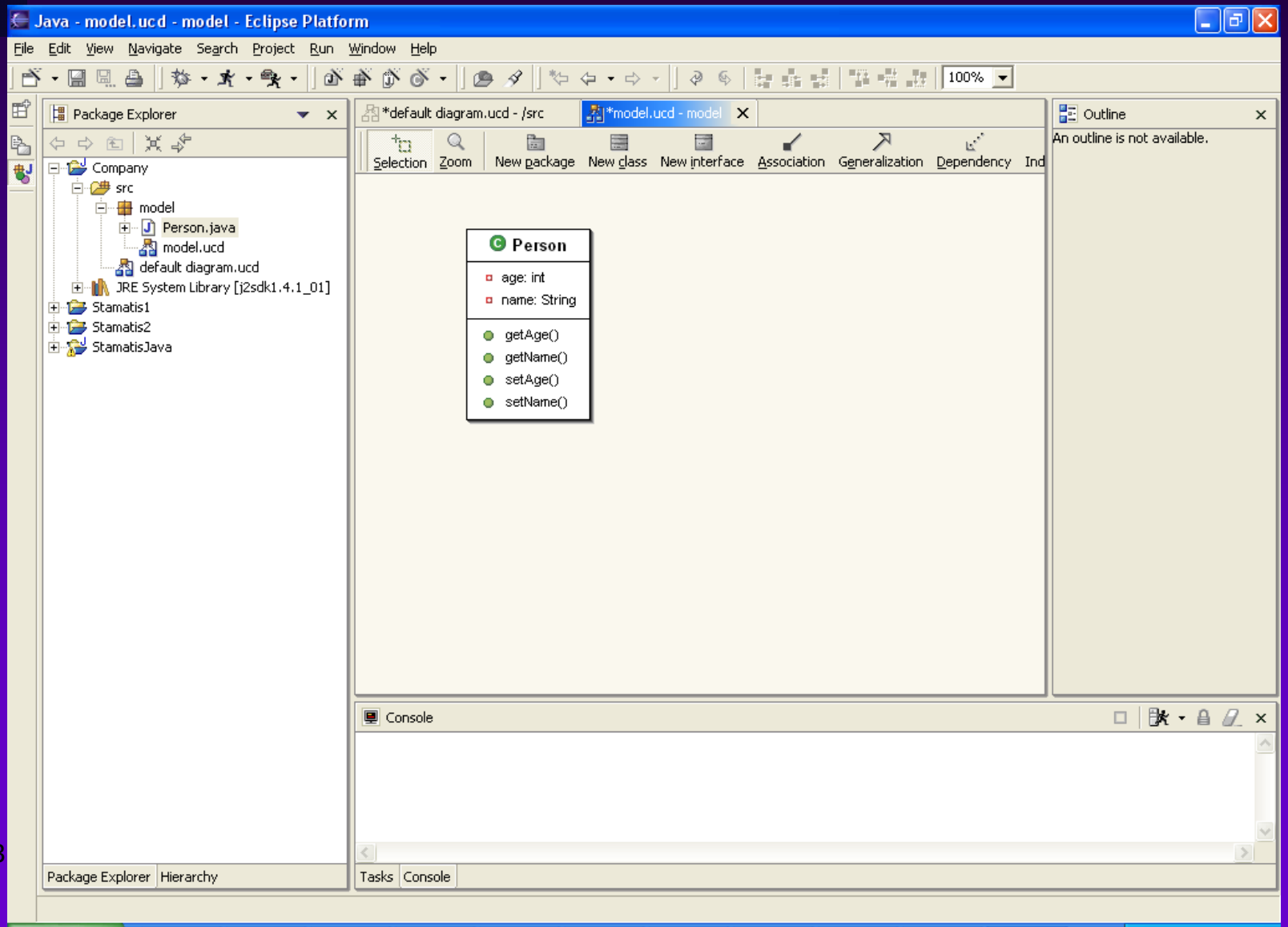


Eclipse UML Class Diagrams

- Enter Attributes properties: "age: int" and "name: String"

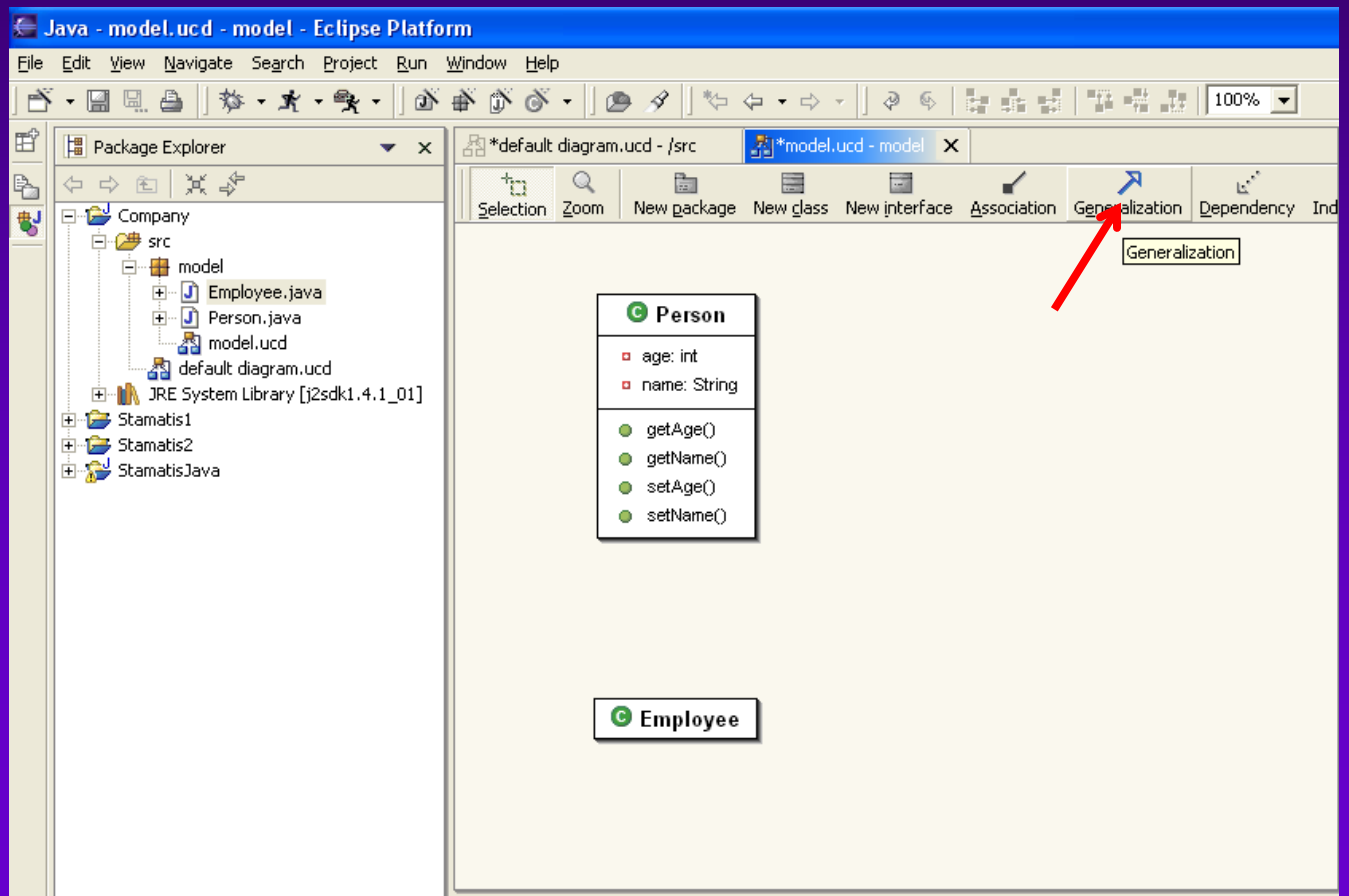


Eclipse UML Class Diagrams



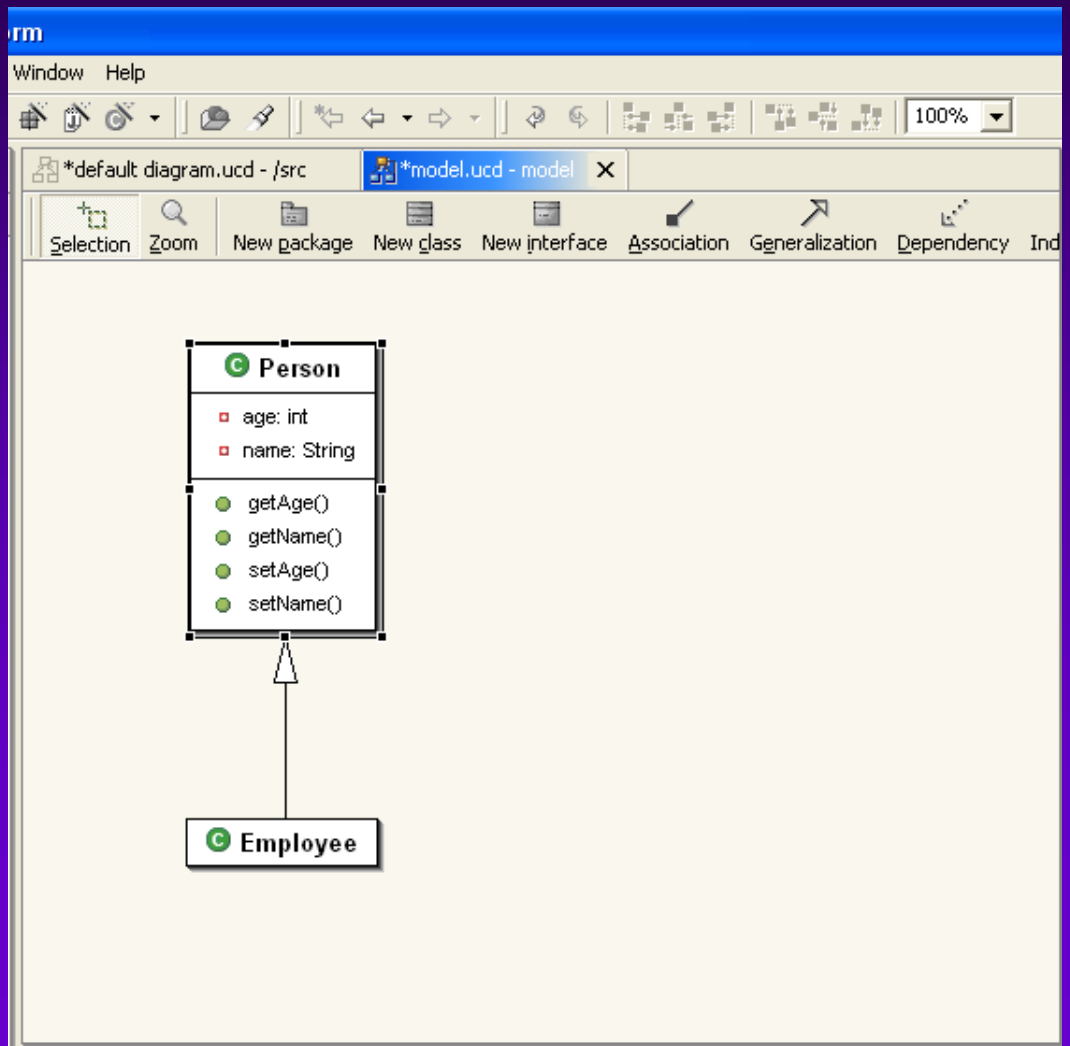
Eclipse UML Class Diagrams

- Create a New class named Employee.
- We would like to add inheritance from Employee to Person.
- Select Generalization in the Toolbar



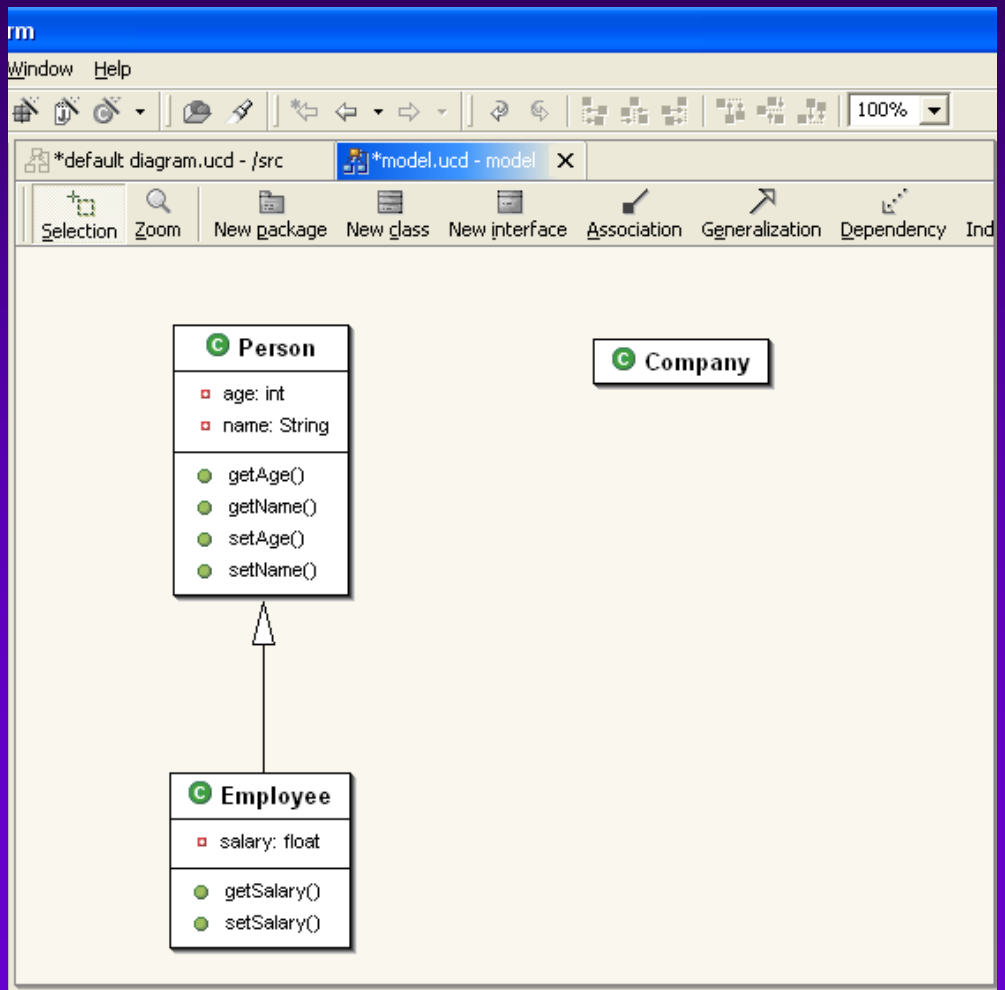
Eclipse UML Class Diagrams

- Drag the Inheritance from Employee to Person.



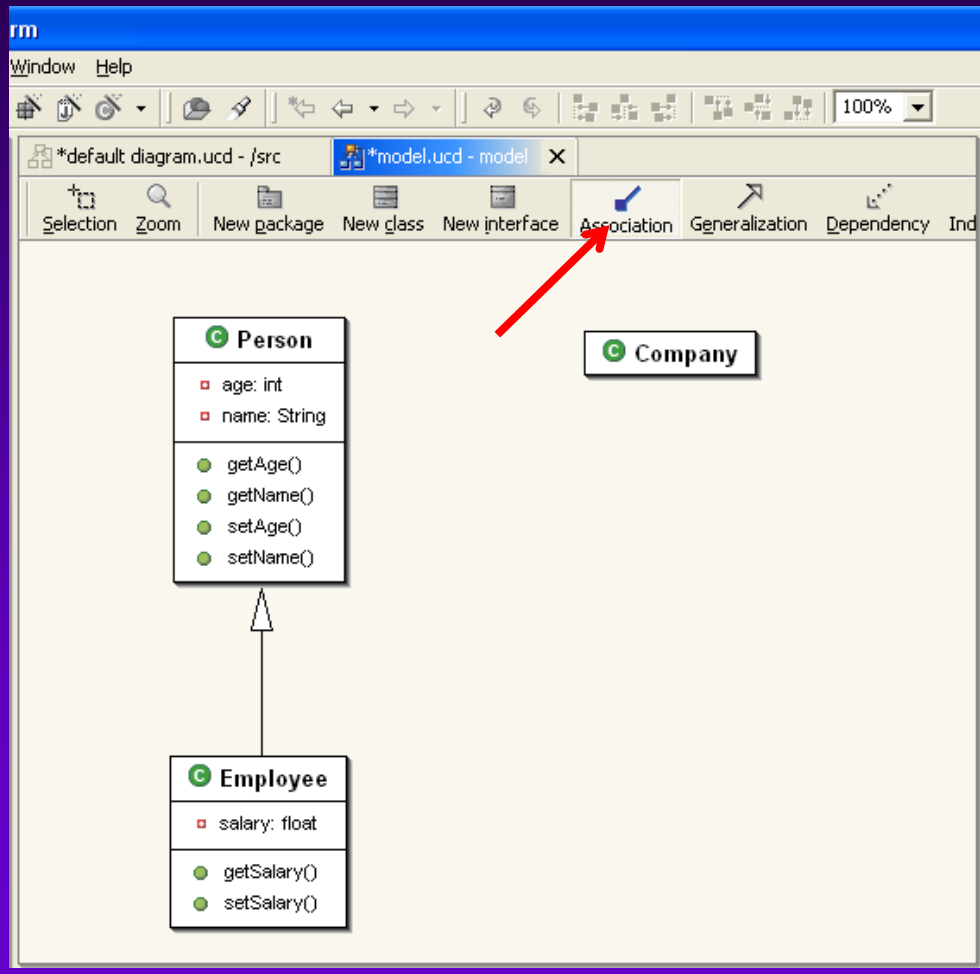
Eclipse UML Class Diagrams

- Add "salary: float" attribute to Employee
- Create a New class named Company.

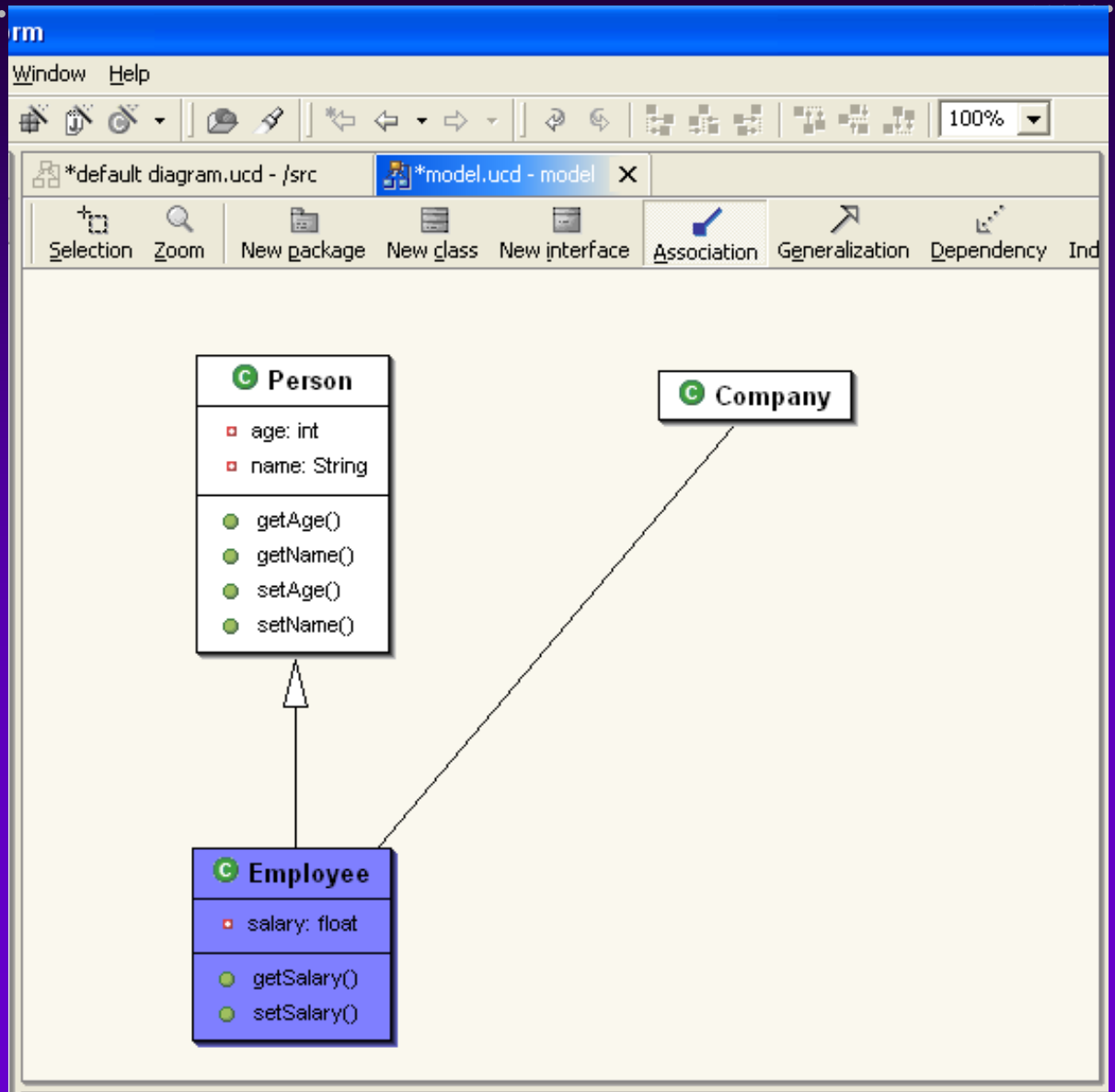


Eclipse UML Class Diagrams

- We want to create a new Association between Company and Employee.
- Select Association in the Toolbar then drag from Company to Employee.

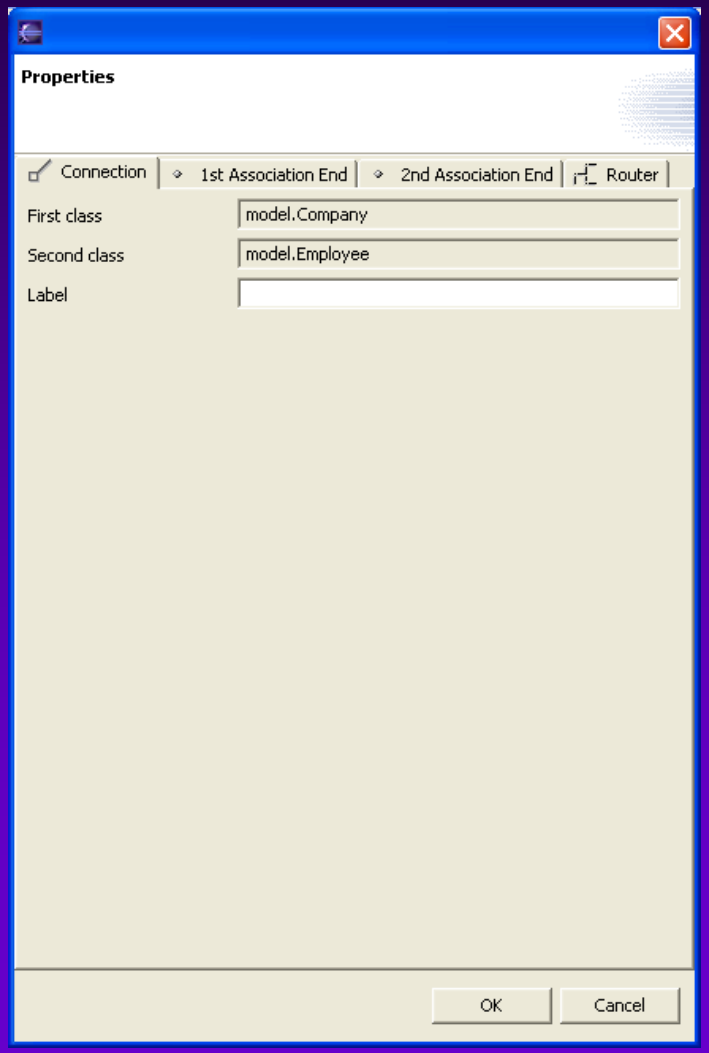


Eclipse UML Class Diagrams



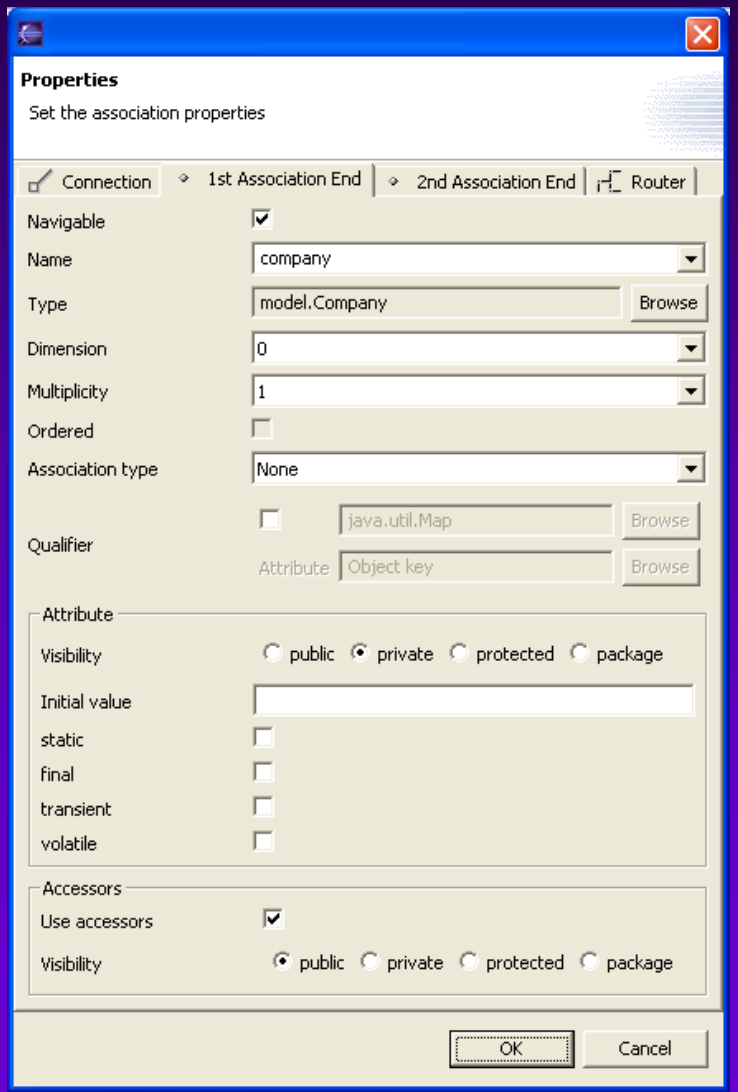
Eclipse UML Class Diagrams

- It is possible to add a name to the Label, but we will not in this example.



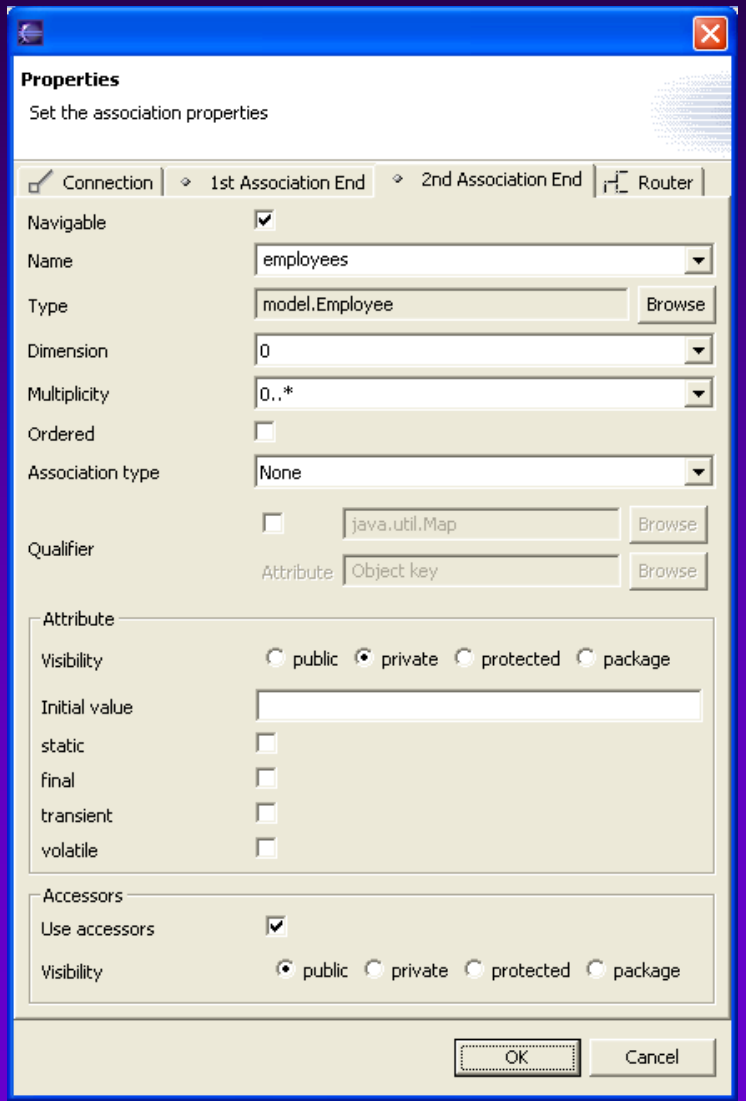
Eclipse UML Class Diagrams

- Select the "1st Association End" tab
- Set multiplicity to 1

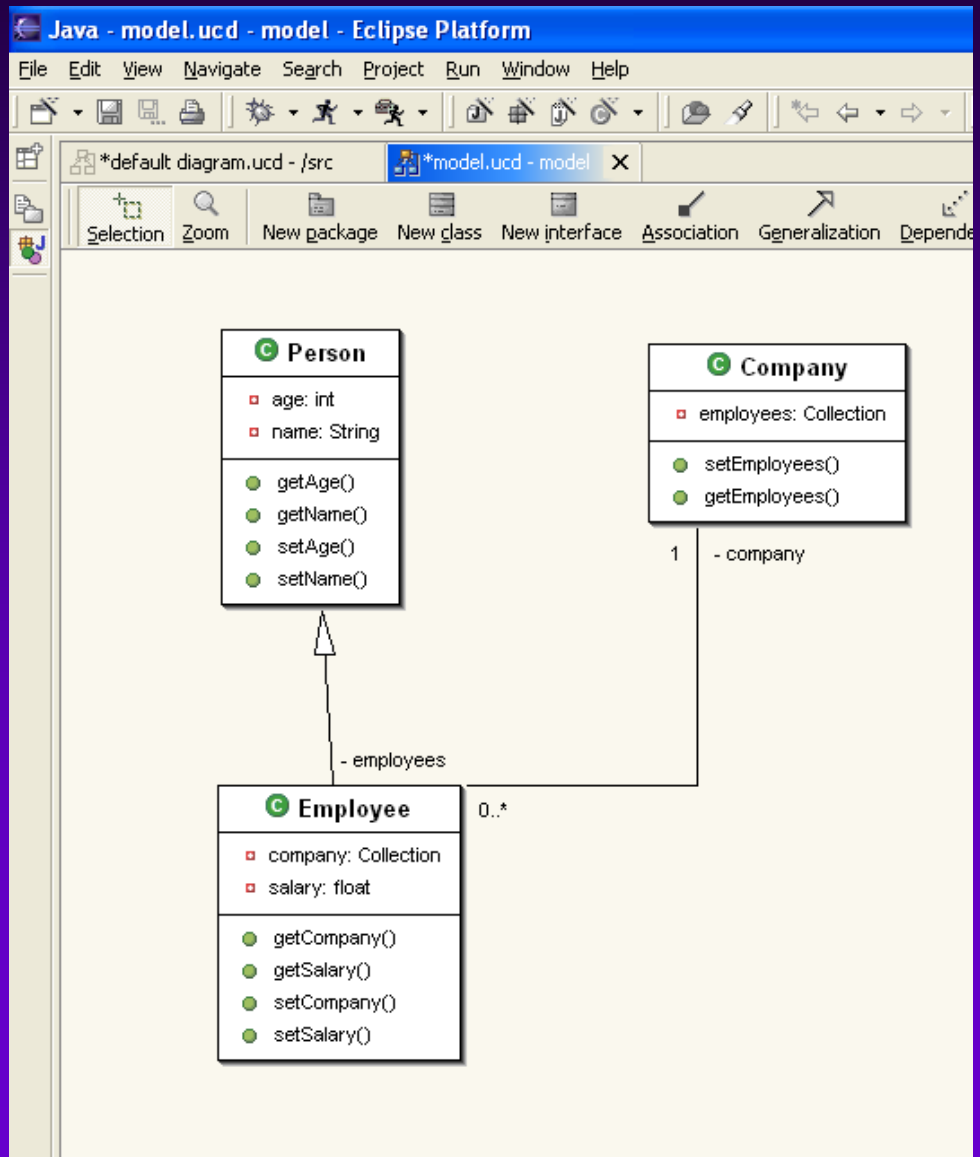


Eclipse UML Class Diagrams

- Select the "2nd Association End" tab
- Set name "employees"
- Set multiplicity "0..*"

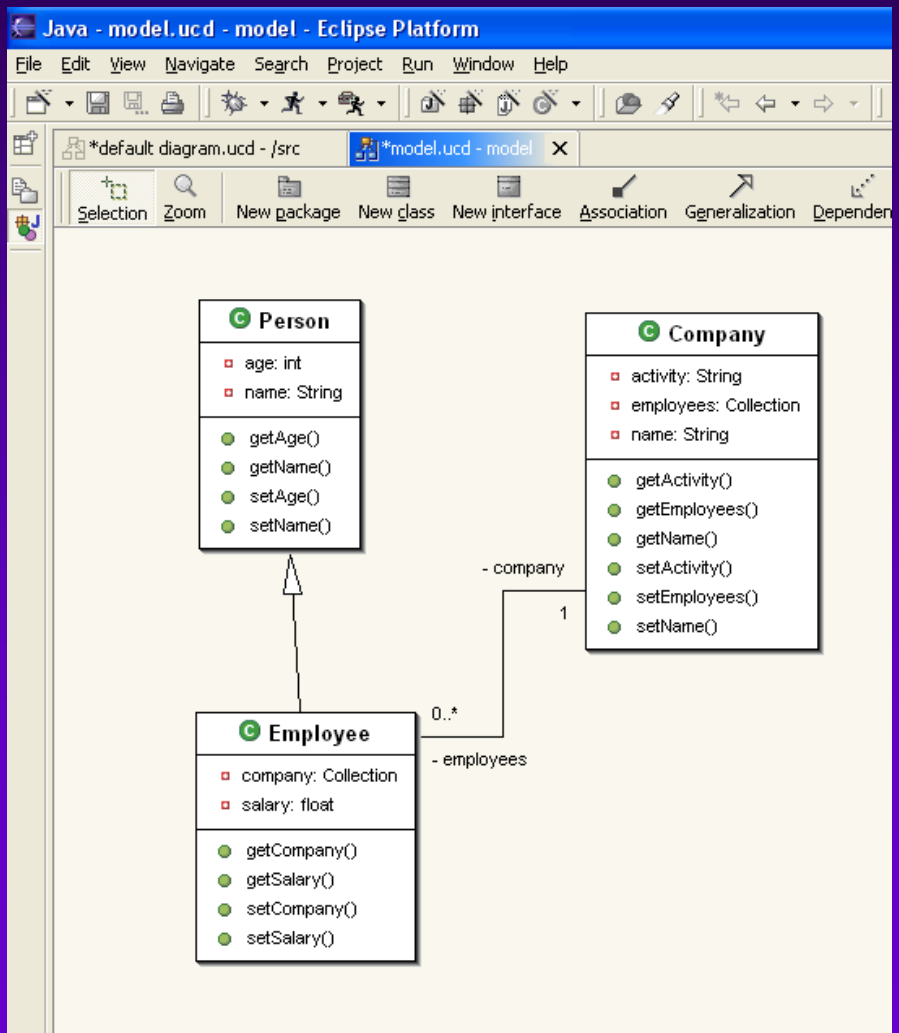


Eclipse UML Class Diagrams



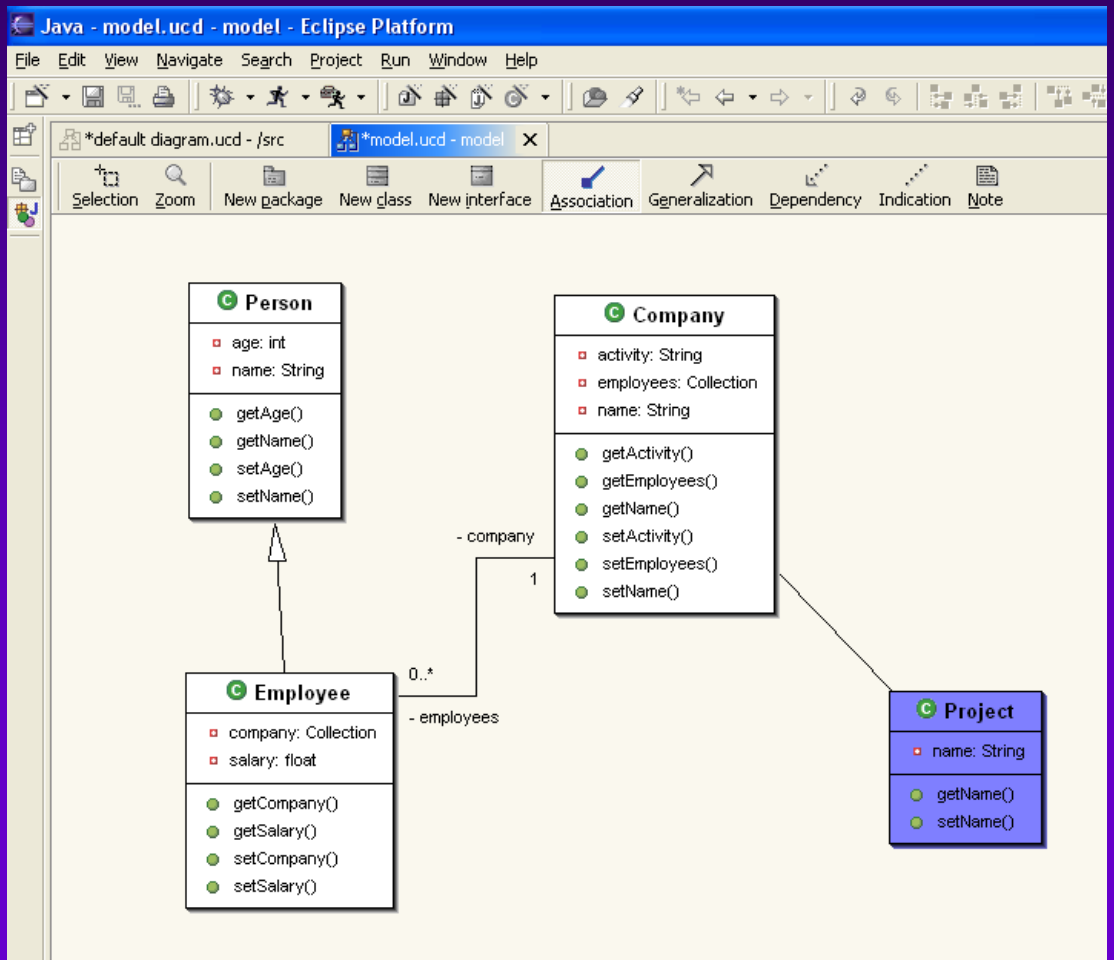
Eclipse UML Class Diagrams

- Click on Company and create two new Attributes: "name: String" and "activity: String"



Eclipse UML Class Diagrams

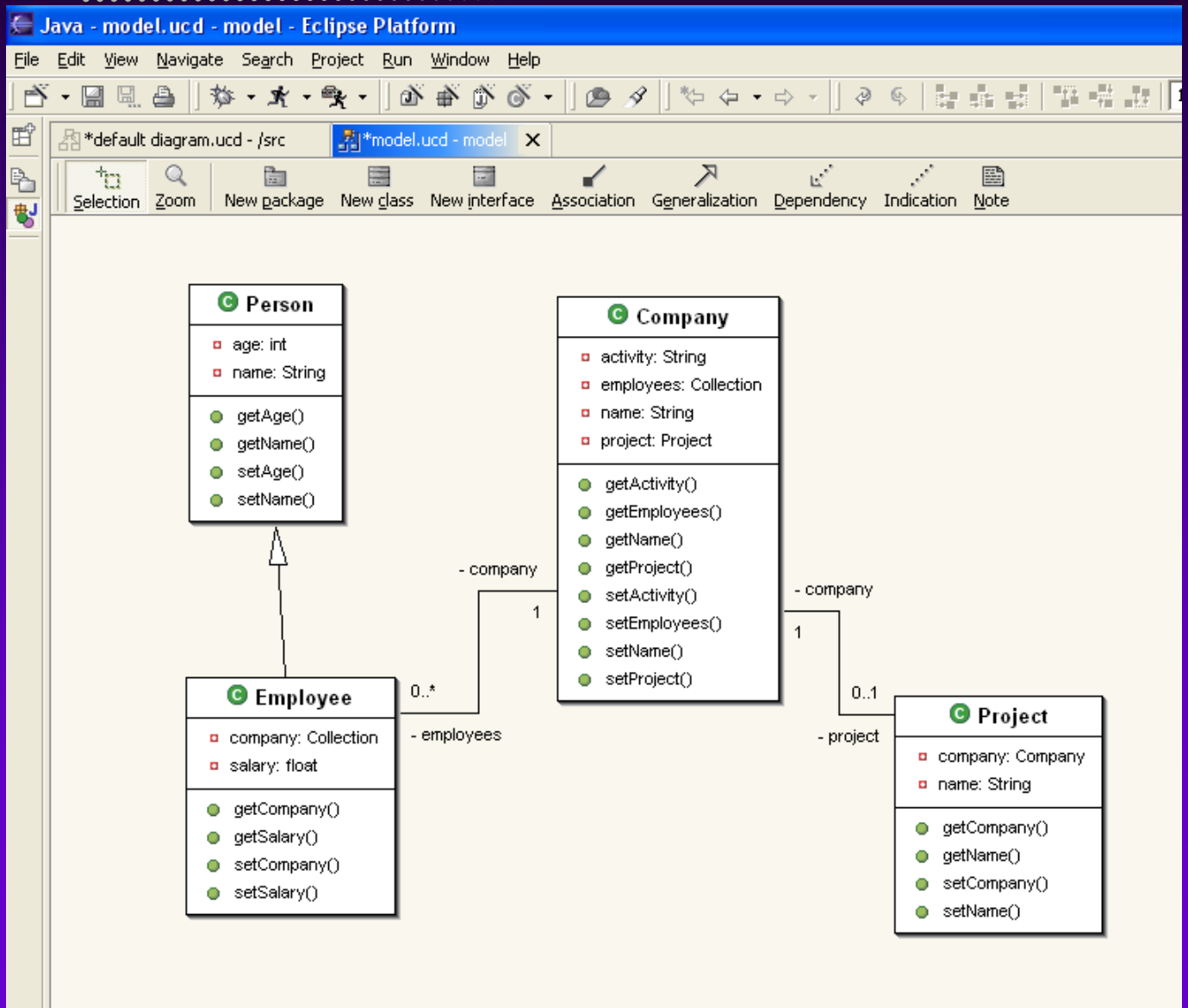
- Create a New class named Project
- Add attribute “name:String: to class Project
- Add an association between Company and Project.



Eclipse UML Class Diagrams

- In association properties
  - Label: leave blank
  - 1st associations end:
    - Name: company
    - Multiplicity:1
  - 2nd associations end:
    - Name: project
    - Multiplicity:0..1

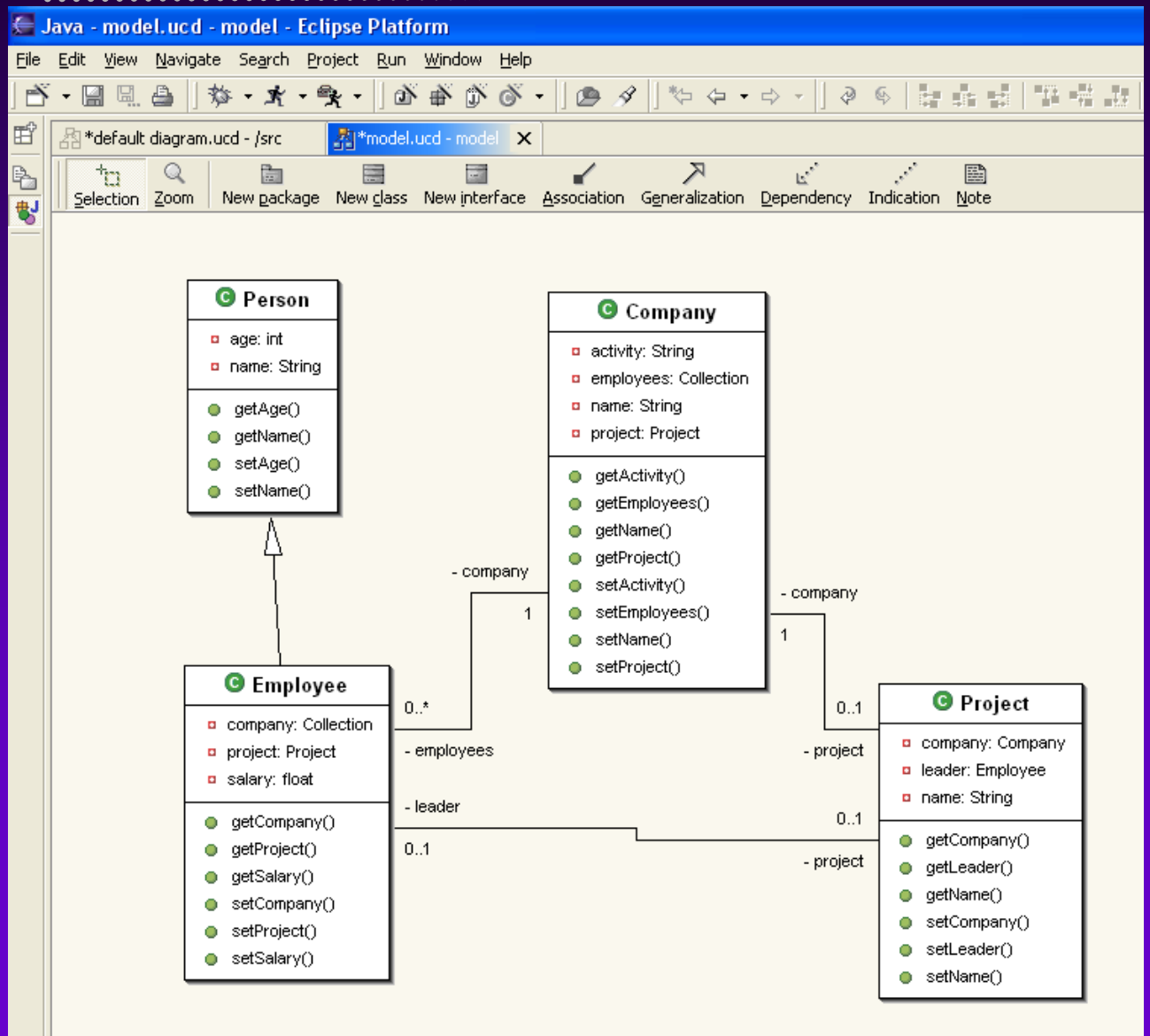
Eclipse UML Class Diagrams



Eclipse UML Class Diagrams

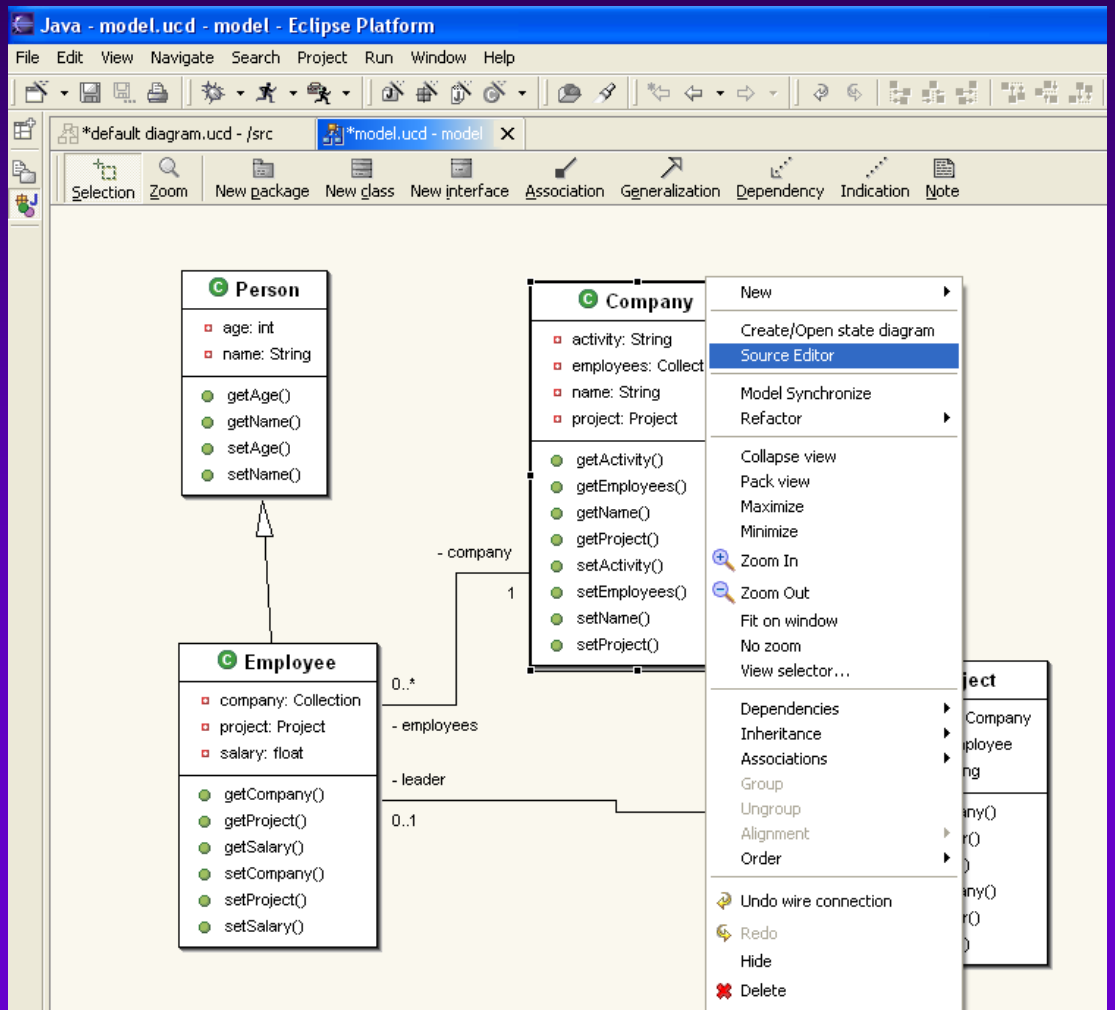
- Add an Association between Employee and Project
- Select Association and drag from Employee to Project
- In the properties menu:
  - Label: leave blank
  - 1st Association End: Name: leader, Dimension 0, Multiplicity 0..1
  - 2nd Association End: Name: Project, Dimension 0, Multiplicity 0..1

Eclipse UML Class Diagrams

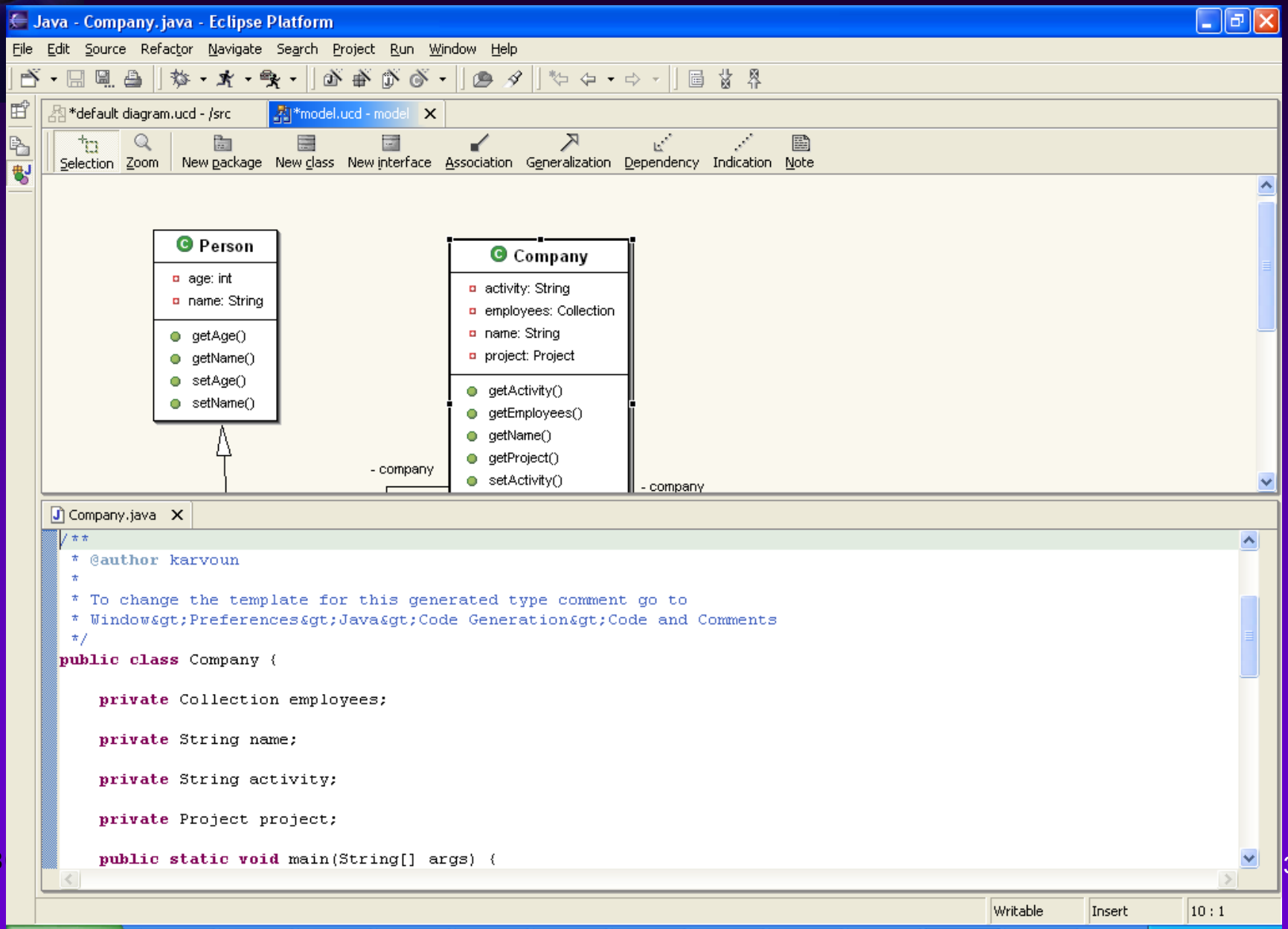


Show code

- Select a Class
- From the pop-up menu select “Source editor”



Show code


Java - Company.java - Eclipse Platform
File Edit Source Refactor Navigate Search Project Run Window Help
*default diagram.ucd - /src
*model.ucd - model
Selection Zoom New package New class New interface Association Generalization Dependency Indication Note
Person
age: int
name: String
getAge()
getName()
setAge()
setName()
Company
activity: String
employees: Collection
name: String
project: Project
getActivity()
getEmployees()
getName()
getProject()
setActivity()
- company
- company
Company.java
/**
* @author karvoun
*
* To change the template for this generated type comment go to
* Window>Preferences>Java>Code Generation>Code and Comments
*/
public class Company {
private Collection employees;
private String name;
private String activity;
private Project project;
public static void main(String[] args) {
Writable
Insert
10 : 1

The end

- Experiment!!!
- Eclipse Tutorials:
  - http://www.eclipse.org/
  - http://www.3plus4software.de/eclipse/index_en.html
  - http://www.omondo.com/
  - http://www.tutorial-omondo.com/jdo/classdiagram.html