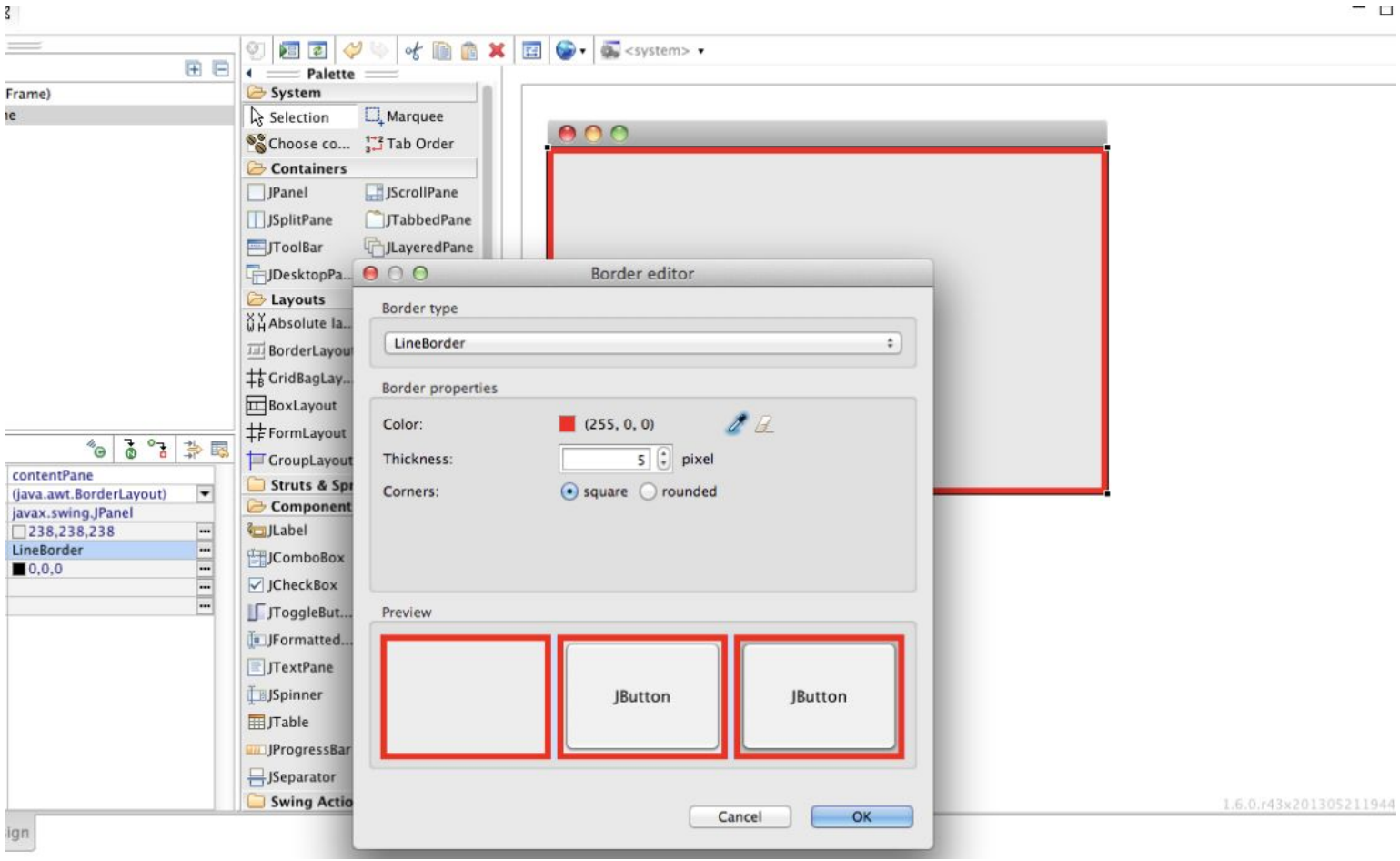
เรื่องที่ 1 การเขียนโปรแกรมเชิงวัตถุ โดยใช้โปรแกรม Eclipse

ENGCE174 การเขียนโปรแกรมเชิงวัตถุ (Object-oriented programming)

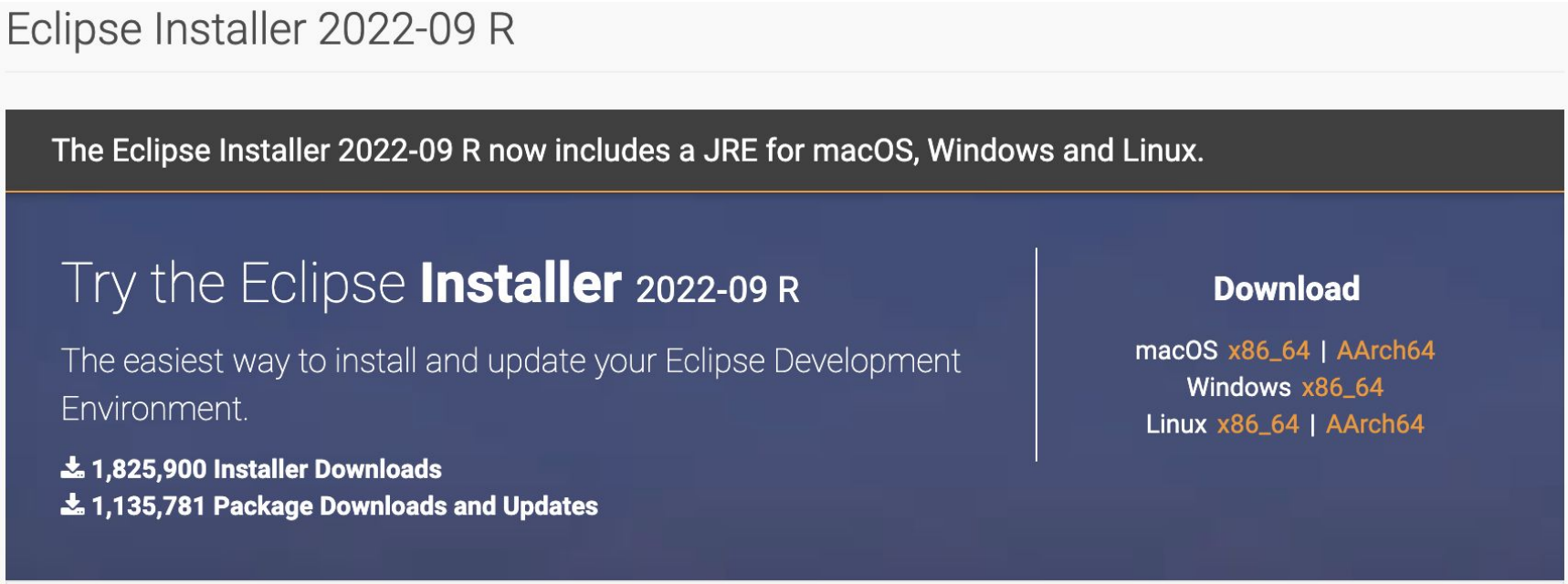
อ.กิตตินันท์ น้อยมณี (mr.kittinan@rmutl.ac.th)

Eclipse


Palette
System
Selection
Marquee
Choose co...
Tab Order
Containers
JPanel
JScrollPane
JSplitPane
JTabbedPane
JToolBar
JLayeredPane
Layouts
Absolute la...
BorderLayout
GridBagLay...
BoxLayout
FormLayout
GroupLayout
Struts & Spr
Component
JLabel
JComboBox
JCheckBox
JToggleBut...
JFormatted...
JTextPane
JSpinner
JTable
JProgressBar
JSeparator
Swing Actio
contentPane
(java.awt.BorderLayout)
javax.swing.JPanel
238,238,238
LineBorder
0,0,0
Border editor
Border type
LineBorder
Border properties
Color:
(255, 0, 0)
Thickness:
5
pixel
Corners:
square
rounded
Preview
JButton
JButton
Cancel
OK

Eclipse

https://www.eclipse.org/downloads/



5 Steps to Install Eclipse

We've recently introduced the Eclipse Installer, a new and more efficient way to install Eclipse. It is a proper installer (no zip files), with a self-extracting download that leads you through the installation process. For those who prefer not to use the Installer, the packages and zip files are still available on our package download page.

1. Download the Eclipse Installer

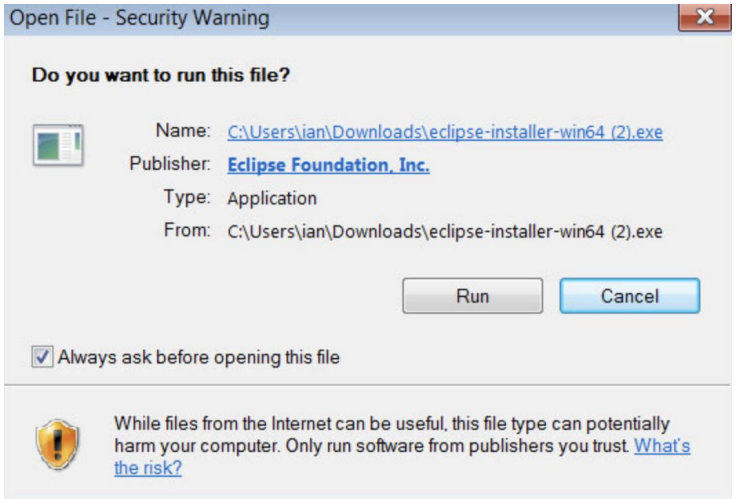
Download Eclipse Installer from http://www.eclipse.org/downloads

Eclipse is hosted on many mirrors around the world. Please select the one closest to you and start to download the Installer

2. Start the Eclipse Installer executable

For Windows users, after the Eclipse Installer executable has finished downloading it should be available in your download directory. Start the Eclipse Installer executable. You may get a security warning to run this file. If the Eclipse Foundation is the Publisher, you are good to select Run.

For Mac and Linux users, you will still need to unzip the download to create the Installer. Start the Installer once it is available.



3. Select the package to install

The new Eclipse Installer shows the packages available to Eclipse users. You can search for the package you want to install or scroll through the list.

Select and click on the package you want to install.

4. Select your installation folder

Specify the folder where you want Eclipse to be installed. The default folder will be in your User directory.

Select the ‘Install’ button to begin the installation.

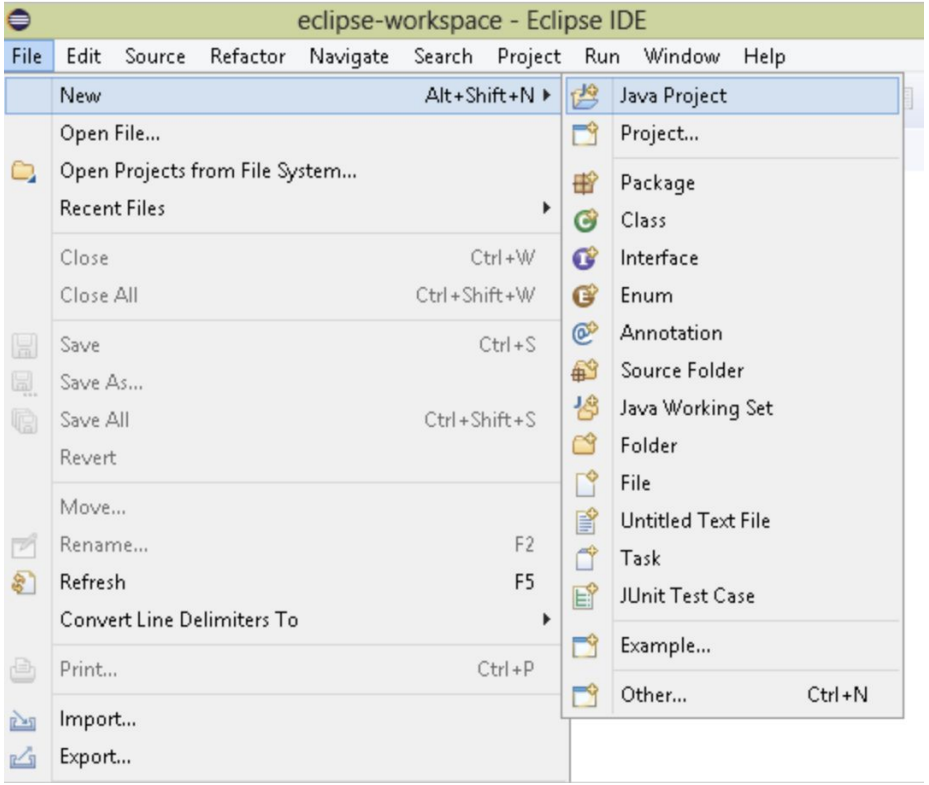
5. Launch Eclipse

Once the installation is complete you can now launch Eclipse. The Eclipse Installer has done it's work. Happy coding.

Eclipse IDE: Create And Run Your First Java Project

In this tutorial, we will learn all about creating a new Java project in the Eclipse IDE and working with it.

#1) Click on File -> New -> Java project.

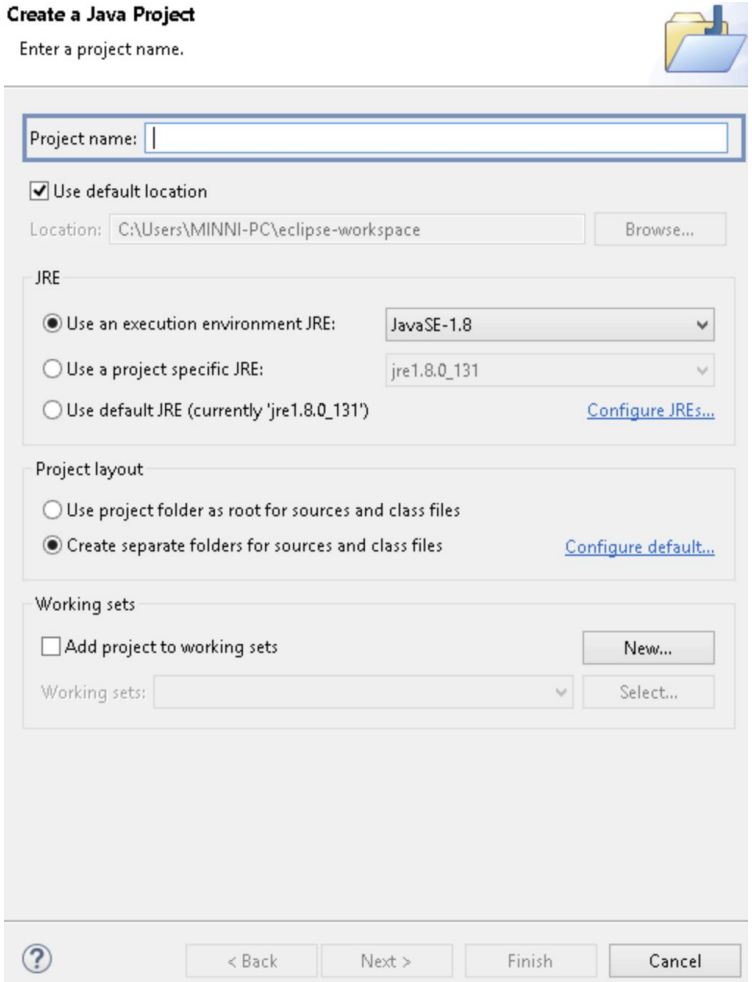


Eclipse IDE: Create And

Run Your First Java Project

#2) The following window will open:

Give a name to your project in the highlighted text field. While creating a formal project the name should have a logical sense, however as we are in the learning process, for now, you can name your project as “MyFirstProject” and leave the rest of the fields as default and click next.

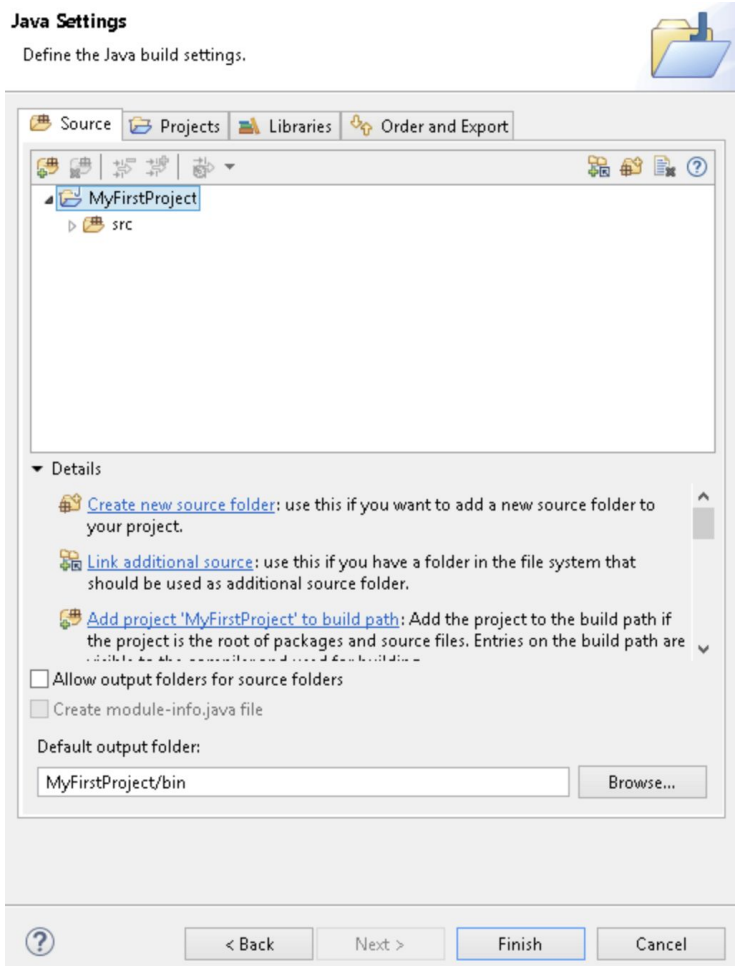


Eclipse IDE: Create And

Run Your First Java Project

#3) The following screen opens up:

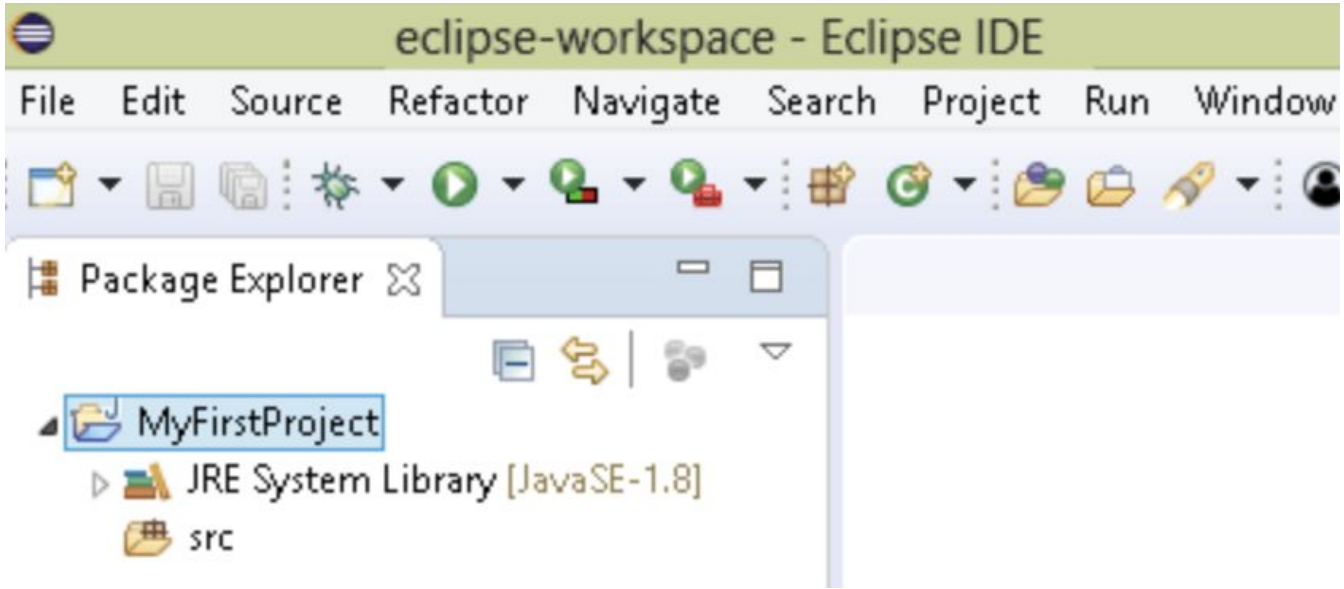
Leave all the options as it is and click on finish.



Eclipse IDE: Create And Run Your First Java Project

You will see the following structure created in your project explorer.

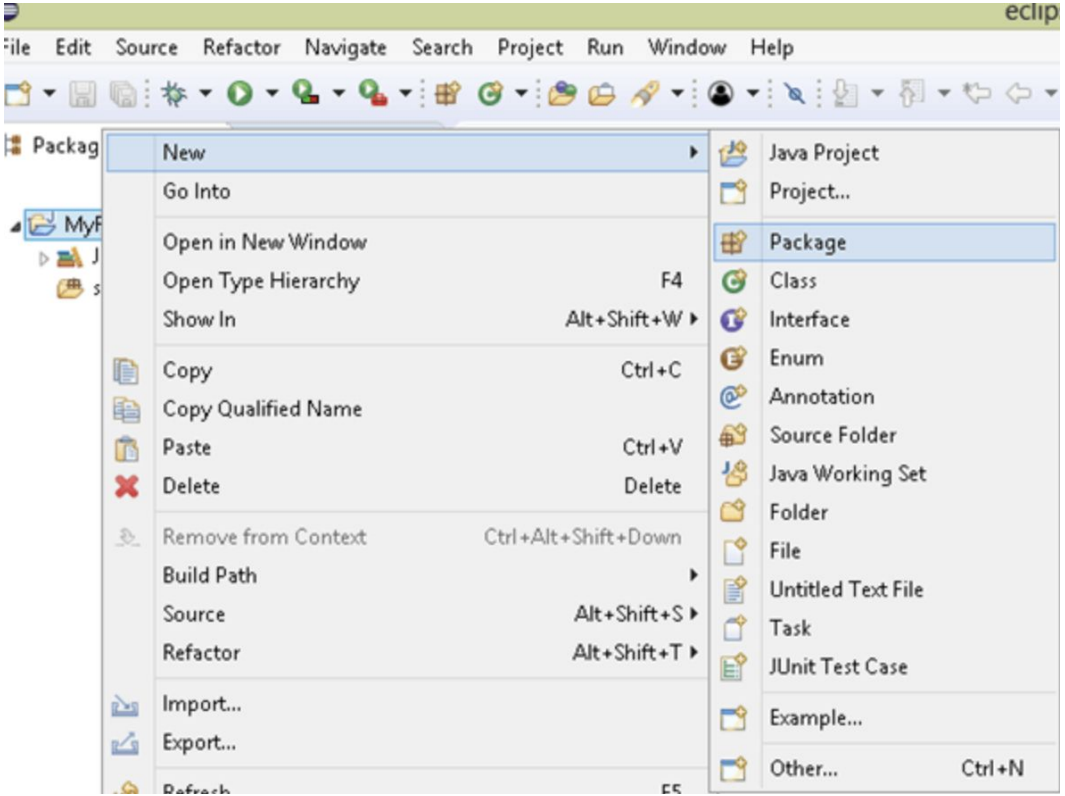
Here you can see two sub-items under your project name. The JRE system library is the default Java library that Eclipse adds to your project. This library provides Java support and without this, you cannot proceed with creating a Java project. The other folder is the “src” folder or the source folder. Your project structure will get created inside this.



Eclipse IDE: Create And Run Your First Java Project

#4) You now have your base project structure, let's add more to this. In Java, we have packages and classes. Packages provide logical separation for classes. Packages are just folders, whereas classes are where you write your actual code.

To create a package right-click on your project name as shown below.

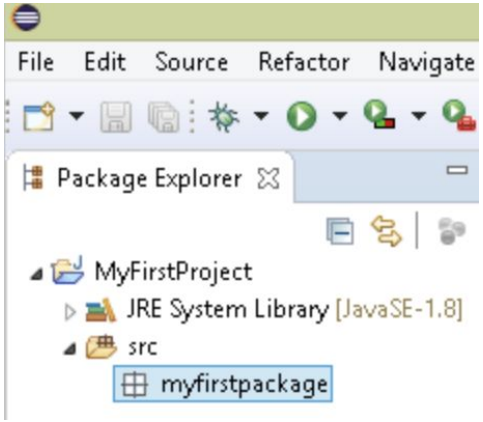


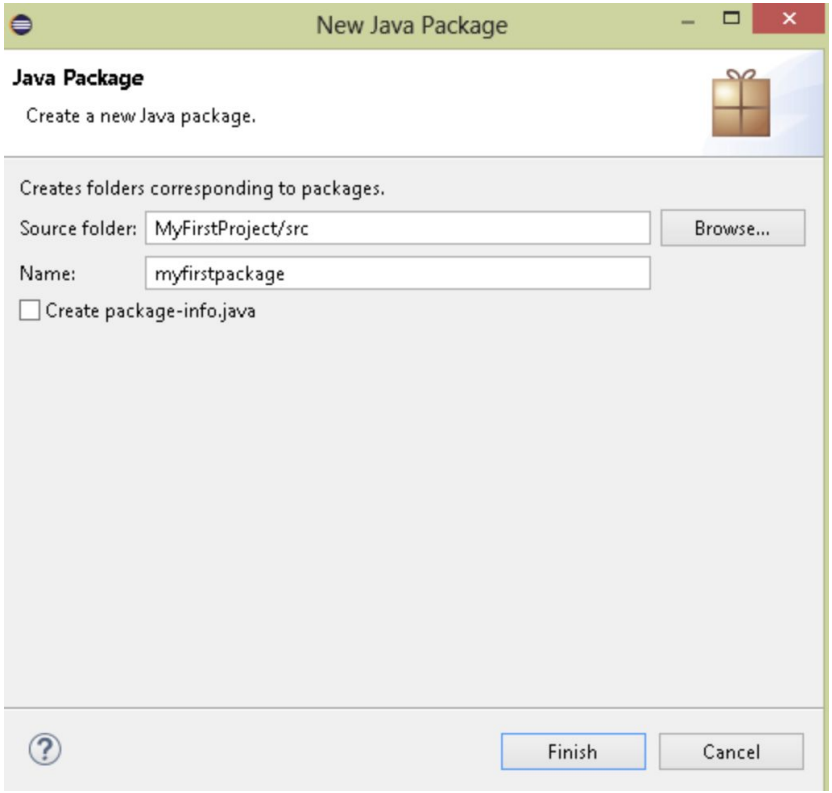
Eclipse IDE: Create And Run Your First Java Project

Click on the package:

Java says package names should be in a small case, so give any name such as “myfirstpackage” and click on finish.

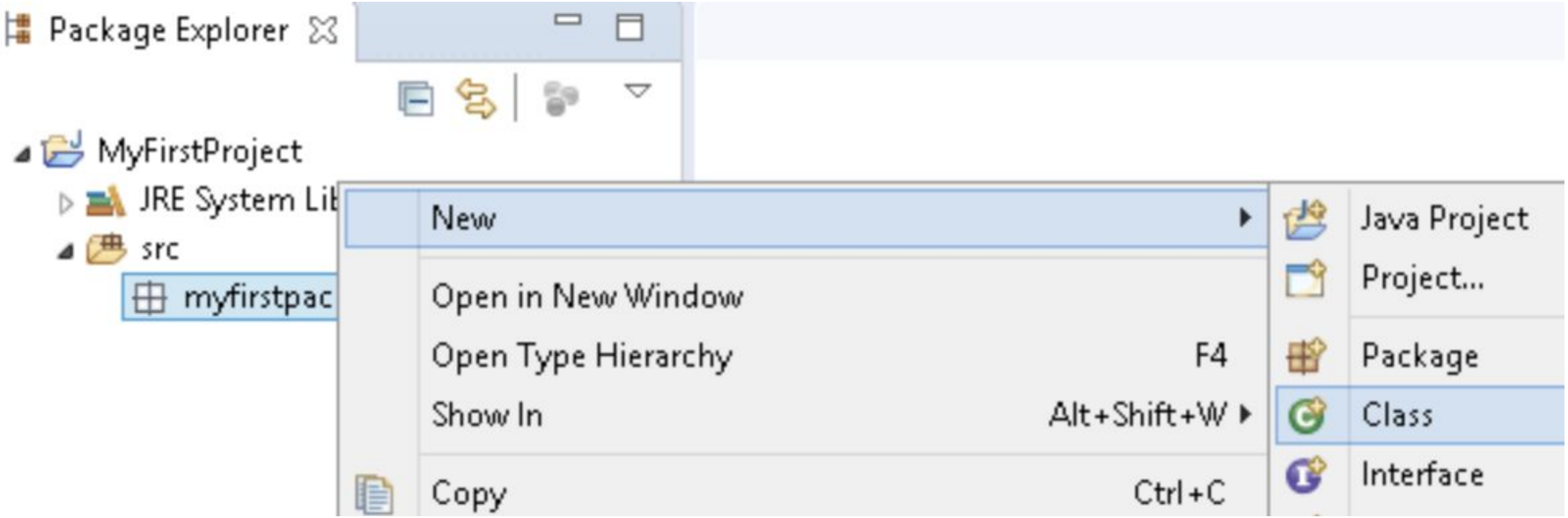
Check your Project Structure.





Eclipse IDE: Create And Run Your First Java Project

#5) Next, you need to create a class, and this is where your actual code will go. Right-click on the package name as shown below.

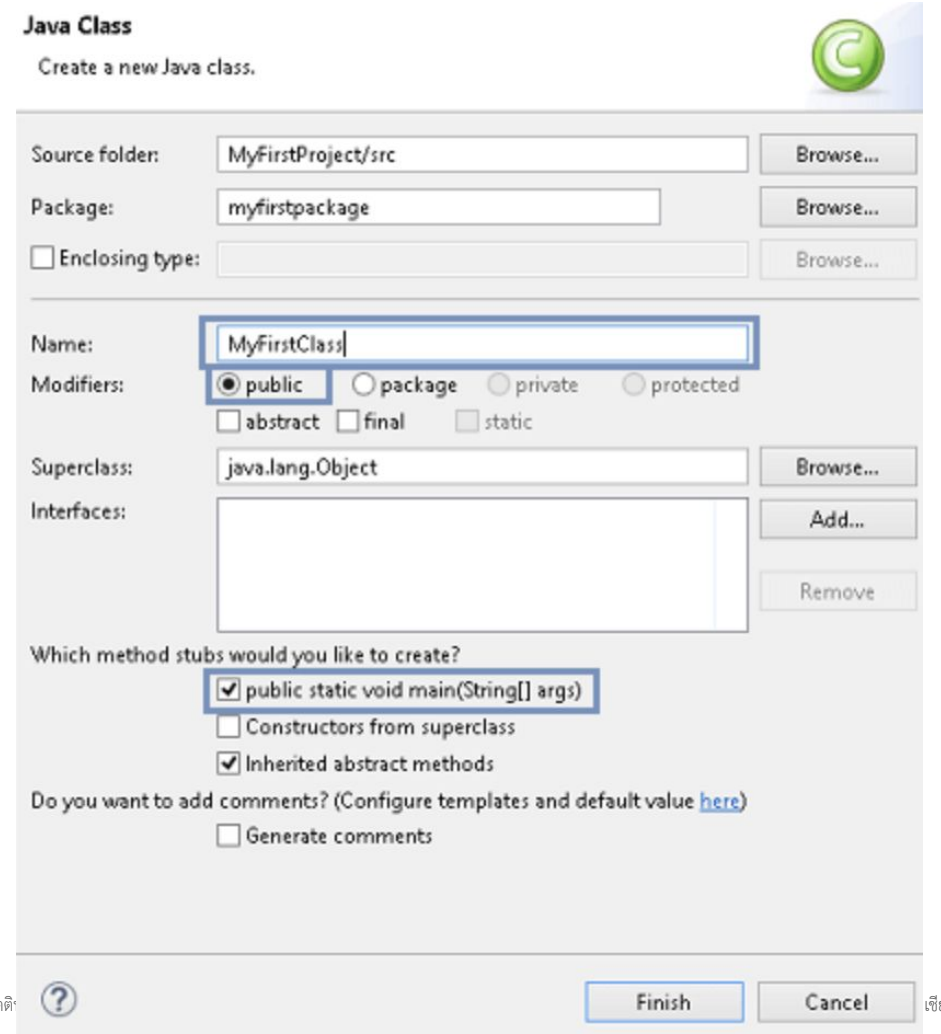


Eclipse IDE: Create And

Run Your First Java Project

Click on the class:

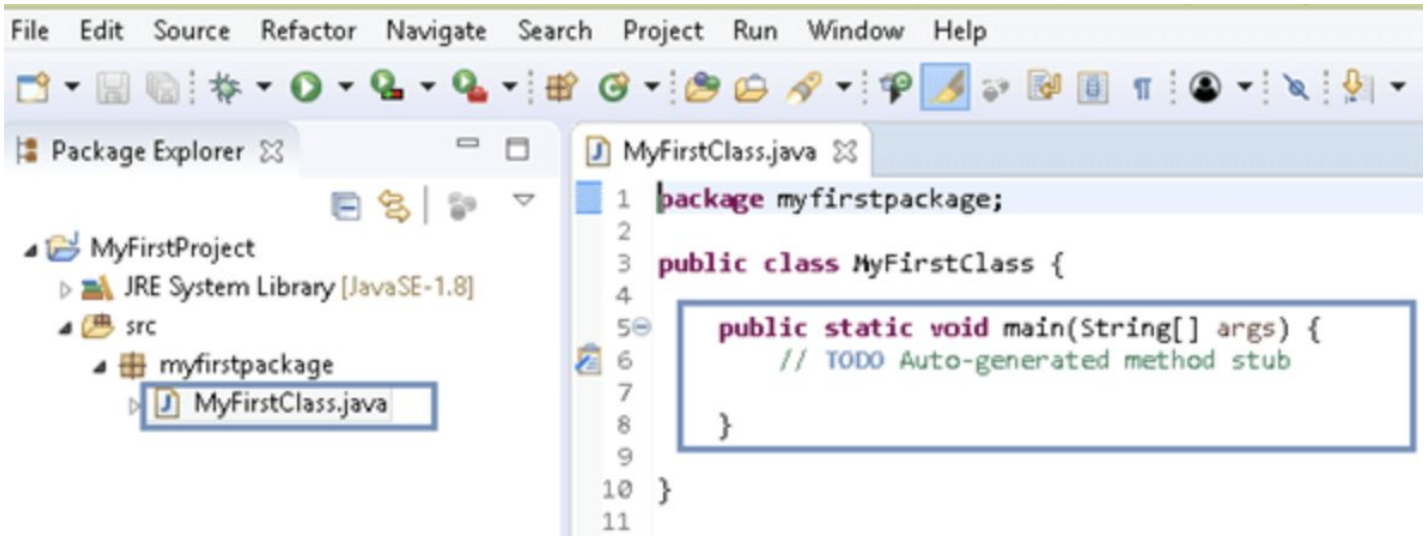
Give a name to your class and keep the modifier selected as public. While creating a simple Java project, ensure to check the checkbox for creating the method “public static void main(String[] args)”.



Eclipse IDE: Create And Run Your First Java Project

Click on finish, and now your project will look as shown below.

You can see that the class is created and as we selected the main method to be added, the method signature is added to the class. Similar to a class you can also create an interface by following the same steps and by selecting “New Interface”.



Eclipse IDE: Create And Run Your First Java Project

#6) As we have successfully created a project and class, let's put a simple print statement in it and run the same to check the output.

Write the below code and save:

MyFirstClass.java

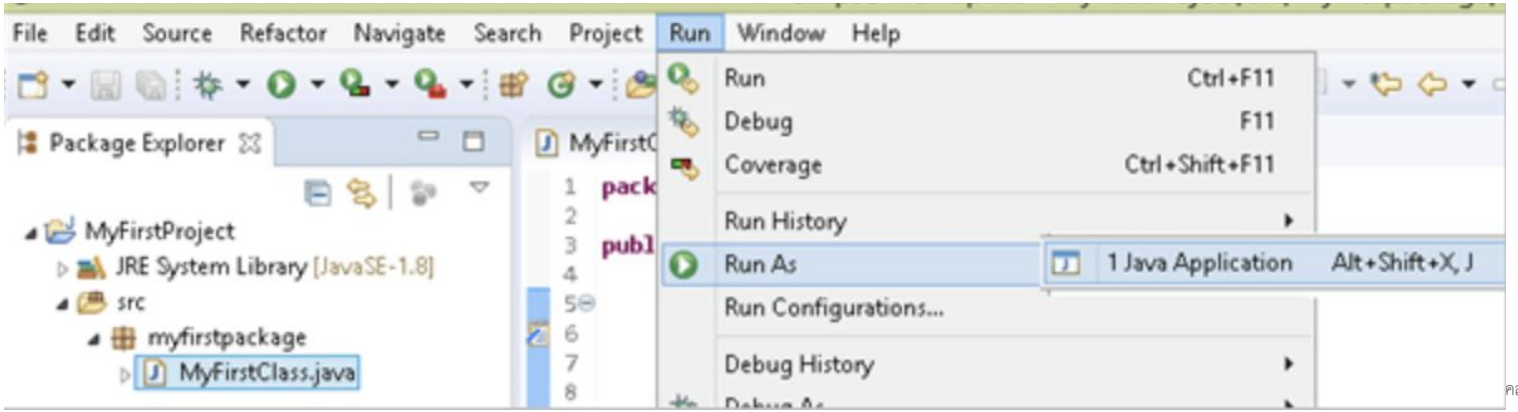
```
1  package myfirstpackage;
2
3  public class MyFirstClass {
4
5      public static void main(String[] args) {
6          // TODO Auto-generated method stub
7
8          System.out.println("This is my first code");
9
10     }
11
12 }
13
```

Eclipse IDE: Create And Run Your First Java Project

Once you have saved the file there are multiple places from where you can run it as a Java application.

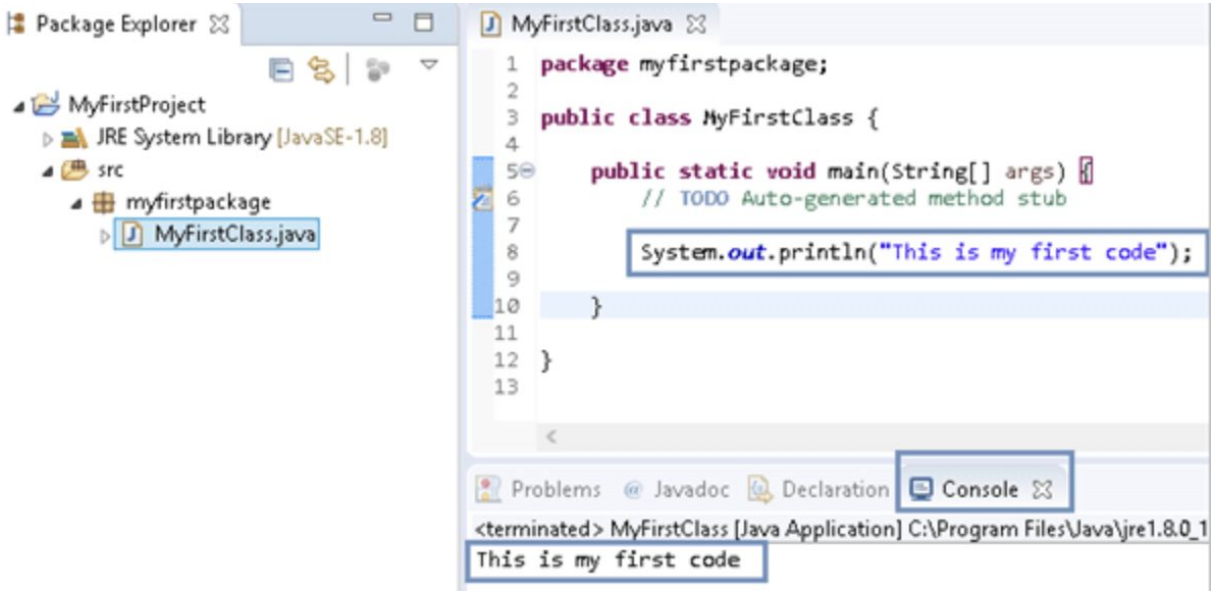
a) Keep your cursor on the class that you want to run and click on this icon:

c) Select the class name in the project explorer and then select the **Run option from the top menu -> select Run As and then click on Java application.**



Eclipse IDE: Create And Run Your First Java Project

#7) As soon as you run your class from any of the above places,



Eclipse IDE: Create And Run Your First Java Project

The console in Eclipse displays the runtime logs while your application is running. In a si

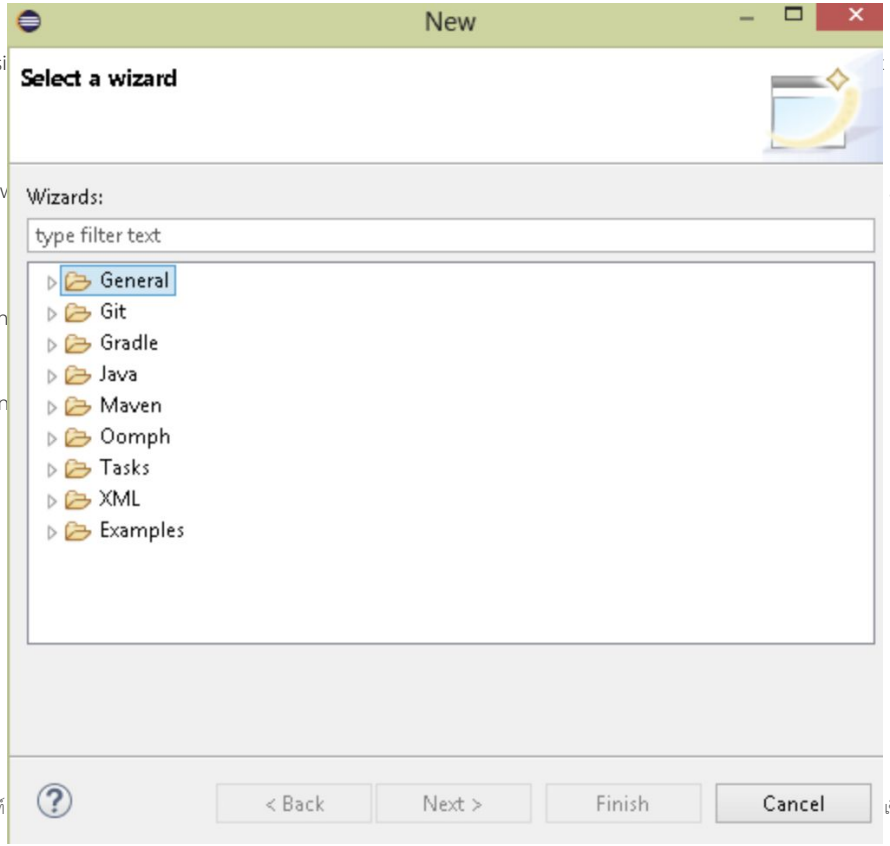
projects, there may be a lot more information displayed.

The console also displays error information if your program encounters any exceptions w and

see the output generated by your code.

Apart from classes, there are many other types of files that you can create in Eclipse wh

You can explore the types of files or types of projects that you can create by clicking on



Eclipse IDE: Create And Run Your First Java Project

Summary

In this tutorial, we learned to create a new and simple Java project in Eclipse and saw how to run it as a Java application.

We also saw that there are several types of files like XML's that you can create from Eclipse. Similarly, as you explore more, you will find many options that you can keep adding to your project.

In our upcoming tutorial, we will take a look at the common features that Eclipse offers to help you code in a better manner.