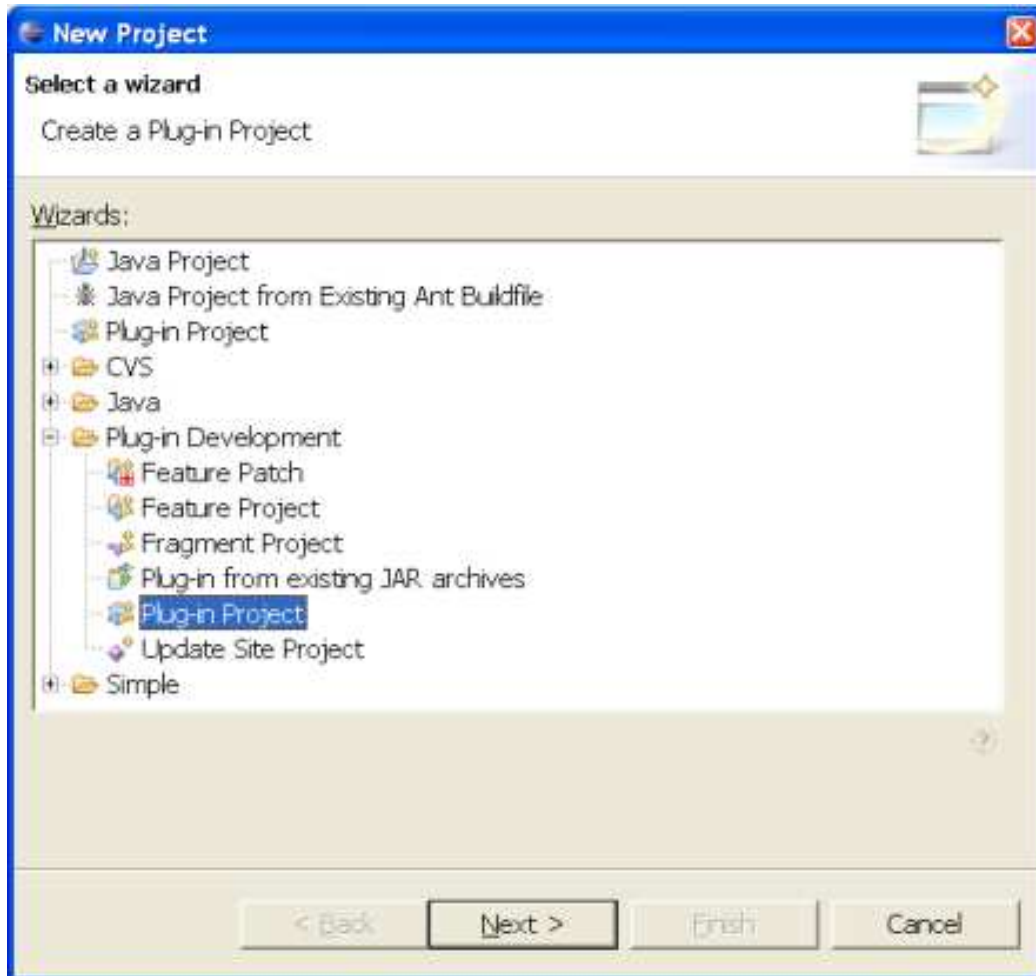


## Lab 3 Creating a 'Hello World' Plug-in

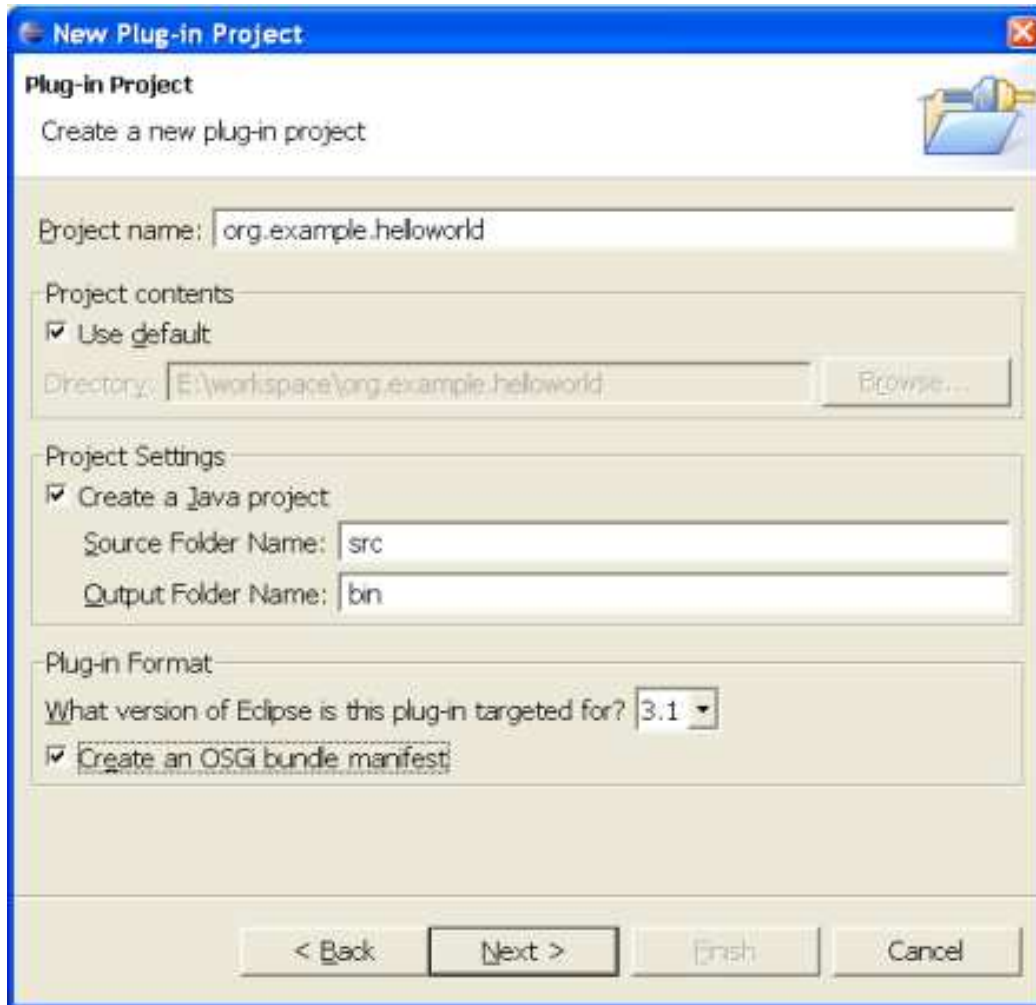
1. Select **File>New>Project** from the menu bar and select Plug-in Project as shown and click on the **next** button.



2. Enter the project name: 'org.example.helloworld' , leaving the Project settings as default, and select the **Next** button.

**Source Folder** lets you specify the subdirectory where the Java of your project are kept.

**Output Folder** specifies where Eclipse place the generated class files.



The screenshot shows the 'New Plug-in Project' dialog box in Eclipse IDE. The dialog is titled 'New Plug-in Project' and contains the following fields and options:

- Project name:** org.example.helloworld
- Project contents:**
  - Use default
  - Directory:** E:\workspace\org.example.helloworld (with a 'Browse...' button)
- Project Settings:**
  - Create a Java project
  - Source Folder Name:** src
  - Output Folder Name:** bin
- Plug-in Format:**
  - What version of Eclipse is this plug-in targeted for?** 3.1 (dropdown menu)
  - Create an OSGI bundle manifest

At the bottom of the dialog, there are four buttons: '< Back', 'Next >', 'Finish', and 'Cancel'.

3. In the next window use the default content as follows:

**New Plug-in Project**

**Plug-in Content**  
Enter the data required to generate the plug-in.

**Plug-in Properties**

Plug-in ID:

Plug-in Version:

Plug-in Name:

Plug-in Provider:

Classpath:

**Plug-in Class**

Generate the Java class that controls the plug-in's life cycle  
Class Name:

This plug-in will make contributions to the UI

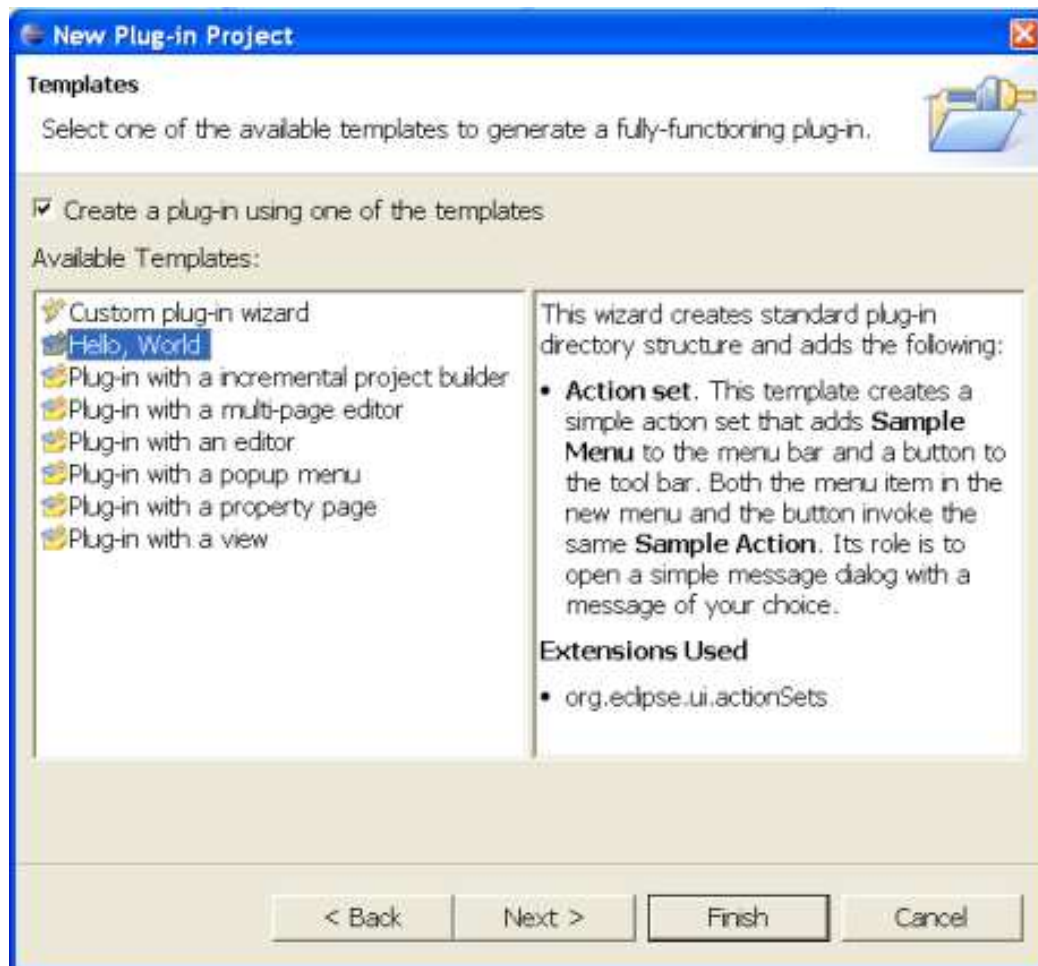
**Rich Client Application**

Would you like to create a rich client application?  Yes  No

< Back    Next >    Finish    Cancel

Click **Next** .

4. The next window enables you to select the plug-in code generator.  
Firstly enable **Create a plug-in project using a code generation wizard**.  
In the case of **Hello World** a sample code generator is supplied. Select this.  
Select **Next**.



5. Use the default content as below and select **Finish**

**New Hello World plug-in project**

**Sample Action Set**

This template will generate a sample action set extension with a menu, a menu item and a tool bar button.

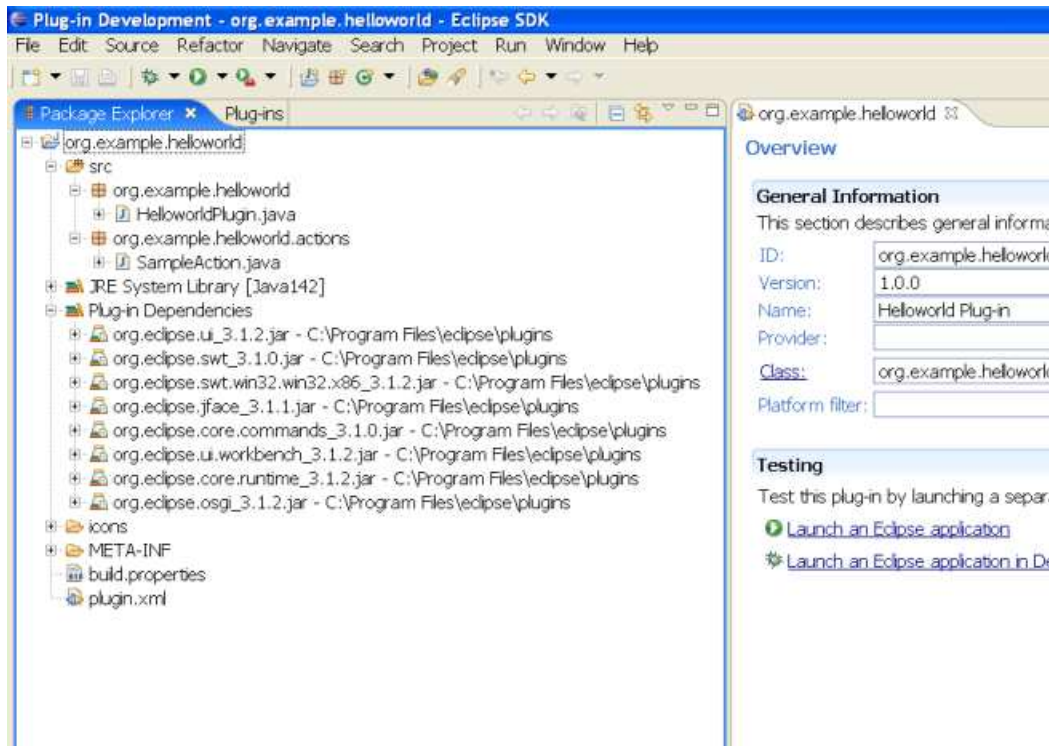
Java Package Name:

Action Class Name:

Message Box Text:

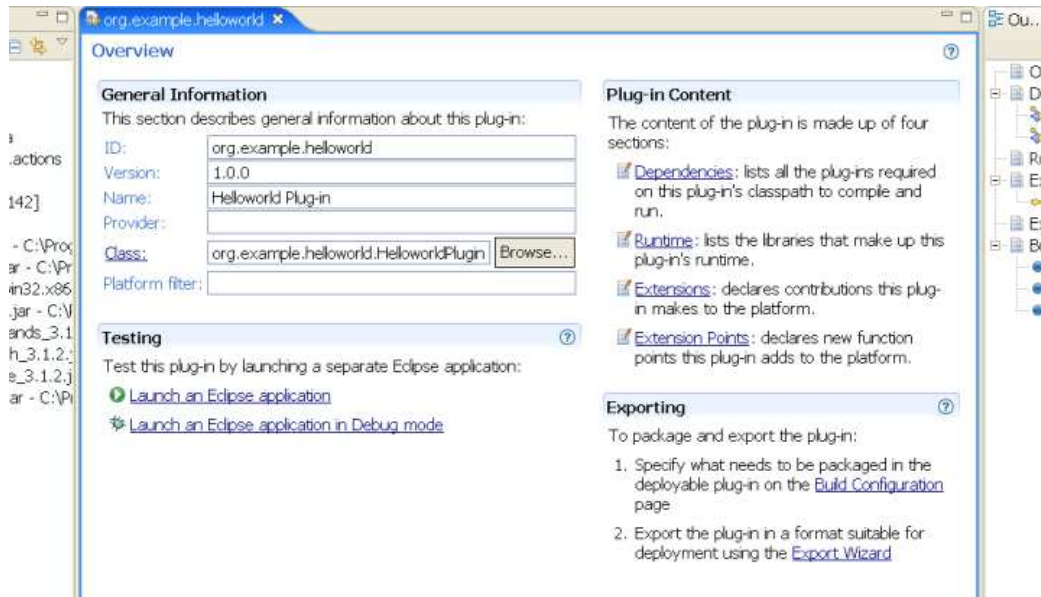
< Back   Next >   Finish   Cancel

6. The plug-in has been created. The file structure ...



7. The plug-in can be run either by deploying the created files into the plug-ins directory, which is the subject of Lab 4, or run in a new spawned Eclipse image. The latter approach is used for plug-in development and is described here:

1. Select **Run > Run As > Eclipse Application**.



8. Click on the **Sample Menu**, then **Sample Action** or Click on the Eclipse Icon circled below;

