Open2Test Test Automation Framework for Selenium Web Driver – Quick Start Guide

Version 1.0

Apr 2013

Table of Contents

1 PURPOSE

This document explains the support settings and how to get started with keyword-driven scripting using Selenium WebDriver.

2 Environment Setup

The following jar files are required to run the Open2Test Selenium Webdriver Framework.

- jexcelapi_2_6_12 or higher
- junit-4.9 or higher
- selenium-server-standalone-2.7.0 or higher

3 Framework at Glance

Input Files:

a. Selenium Utility Excel:

The TestSuite, TestScript, Object Repository and Report Folder location should be mentioned in Selenium_Utility excel.

File\Folder Name	Location
Test Suite	D:/path/Test_Suite1.xls
Test Script	D:/path/
Object Repository	D:/path/TestData/ObjectRepository.xls
Summary Report	D:/path/Test Reports/
Screen Shot Report	D:/path/Test Reports/ScreenShot_Report/
Detailed Report	D:/path/Test Reports/Detailed_Report/

Note: The selenium utility excel file should be placed in

D:\Selenium_Utility.xls.If you wish to place the utility file in some other location, you must update the path in the ReadUtilFile() function in the framework.

b. Object Repository Excel:

Collect the properties of objects and define it in Object_Repository.xls like below

Object Name	Object Type	Parent	ObjectPath
Welcome to Open2Test.org	Page	0	name=
logo.jpg	Image	1	xpath=/html[1]/body[1]/div[1]/div[1]/img[1]
Home	Link	1	link=Home
About Us	Link	1	link=About Us
Tech Docs	Link	1	link=Tech Docs
News	Link	1	link=News
Roadmap	Link	1	link=Roadmap
Comment	Link	1	link=Comment
Support	Link	1	link=Support
Contact Us	Link	1	link=Contact Us
banner.jpg	Image	1	xpath=/html[1]/body[1]/div[1]/div[2]/div[1]/img[1]
Learn More	Link	1	link=Learn More
Learn More_1	Link	1	link=Learn More
Name	Textbox	1	id=name
emailID	Textbox	1	id=emailID
selectframework	ComboBox	1	xpath=//select
Agreement	Textbox	1	id=Agreement

Terms & Conditions	Link	1	link=Terms & Conditions
Input	Textbox	1	id=submit
new.gif	Image	1	xpath=/html[1]/body[1]/div[1]/div[2]/div[2]/div[2]/p[1]/img[1]
More news	Link	1	link=More news
Sitemap	Link	1	link=Sitemap
Privacy Policy	Link	1	link=Privacy Policy

c. Test suite Excel:

Mention the test scripts in the 'Test suite' excel in the order you wish to execute them.

Run	Test Scripts
r	Mantis_Create Project
r	Mantis_Report Issue
r	Mantis_View Issue and Delete Project
r	Mantis_Fail
r	Call Tariff Addition and Amendment

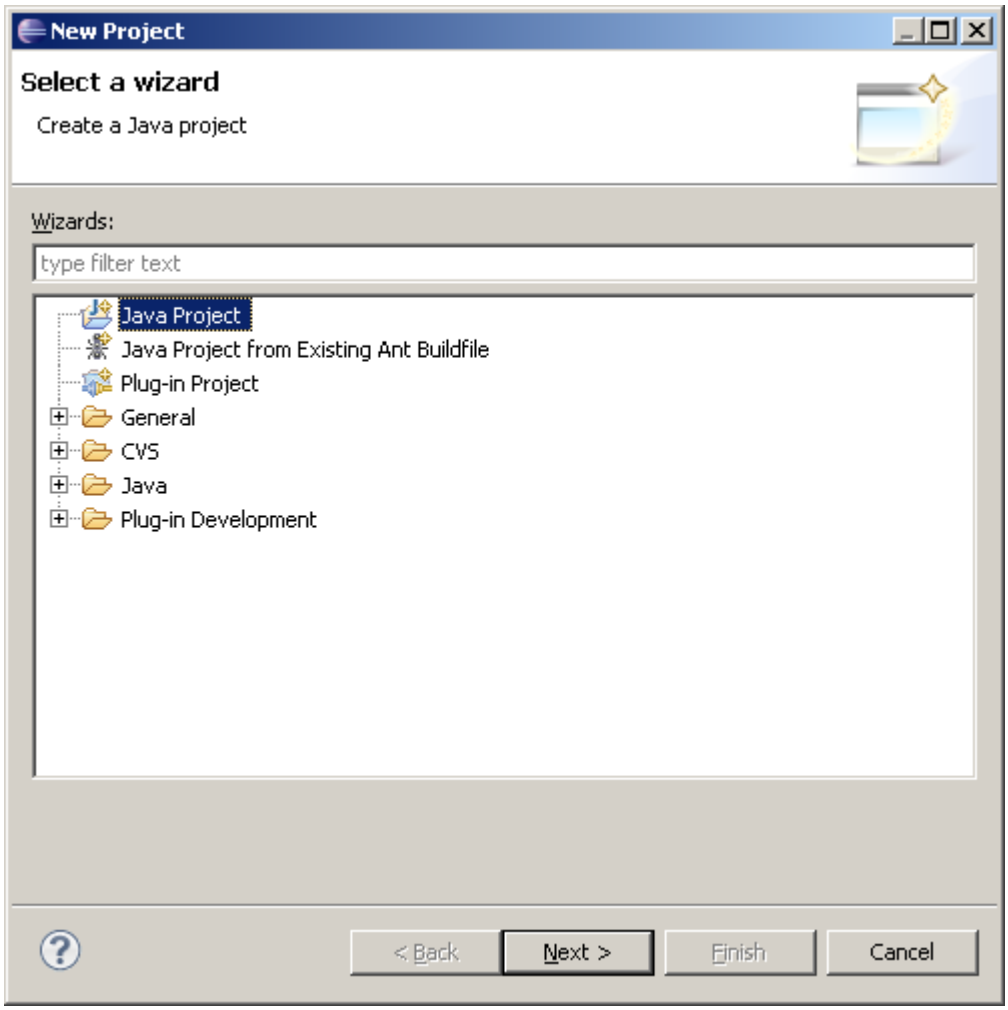
d. Test Script Excel

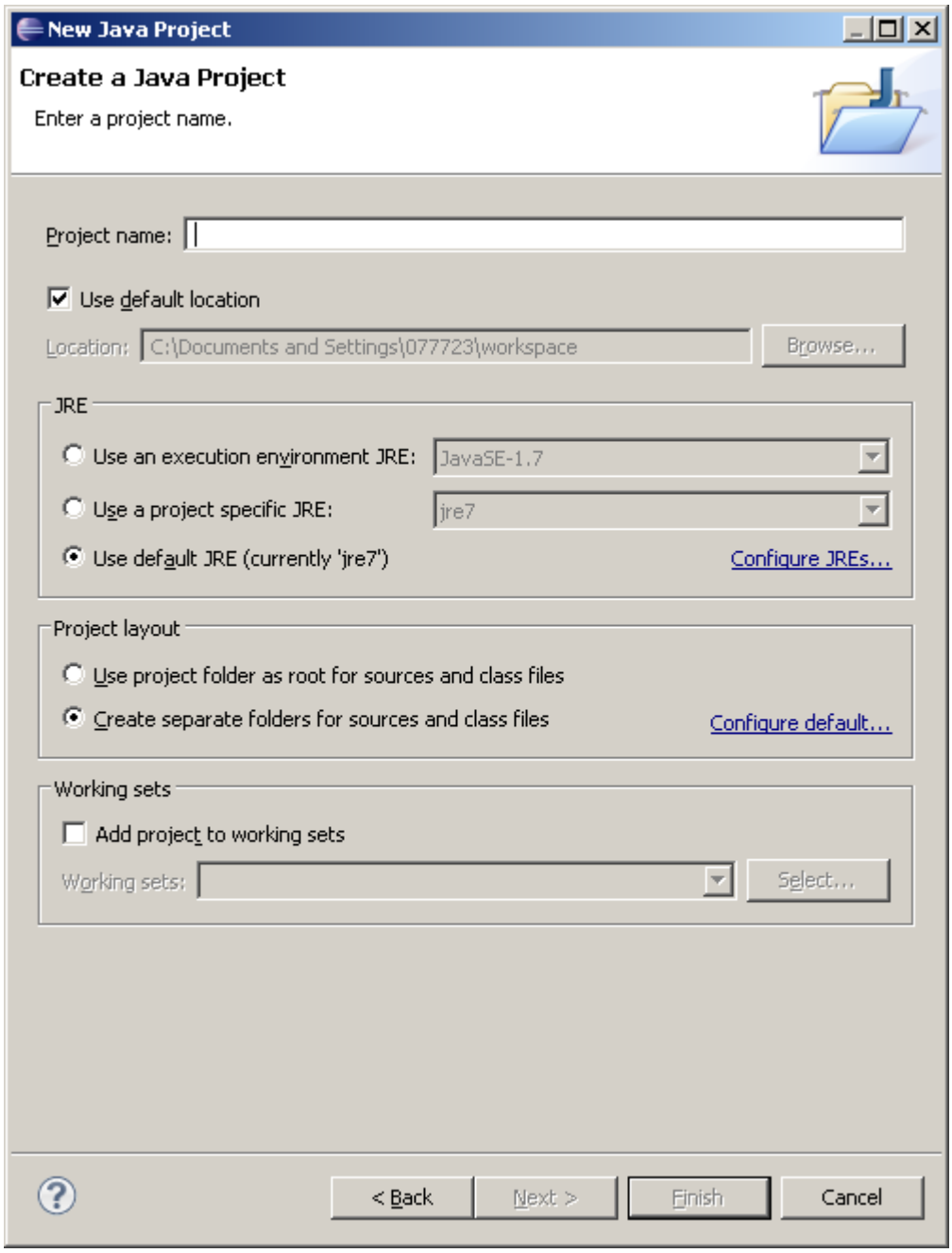
Write the keywords for the test script like below

Run	Keyword	ObjectDetails	Action	Action2	Comments
r	LaunchApp	www.open2test.org			
r	importdata	D:\001_D_Files\path\TestData\TestData.xls	Sheet1		
r	loop	3			
r	perform	Textbox;name	set:dt_username		
r	perform	Textbox;emailID	set:dt_email		
r	check	Textbox;emailID	enabled:true		
	call				
r	perform	ComboBox;selectframework	select:Web Framework (V2) for Selenium		
r	perform	Textbox;Agreement	click		
	perform	Link;Terms & Conditions	click		
r	perform	Button;input	click		
r	screencapture				
r	Endloop				

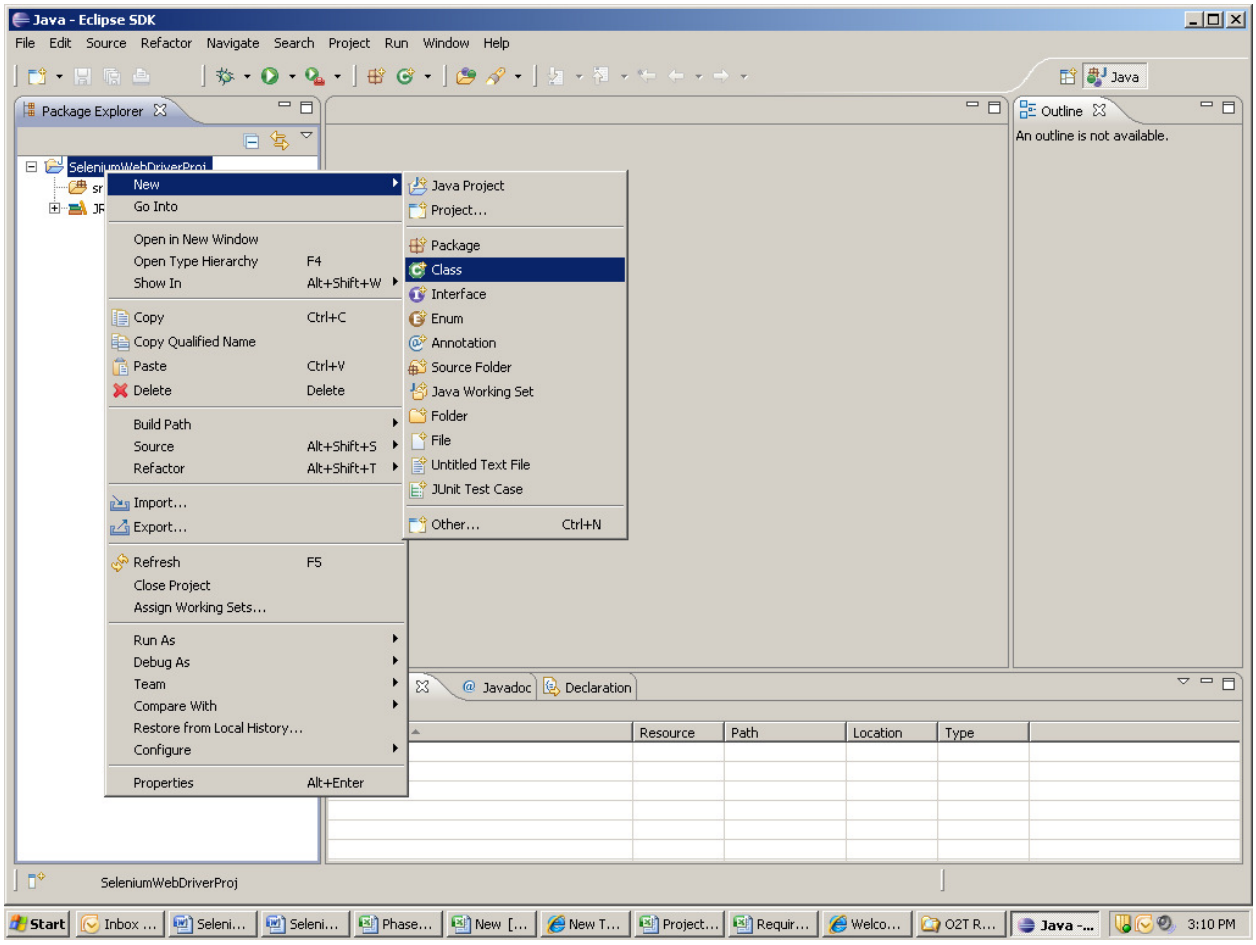
3. Project Setup in Eclipse

Create a Java project in Eclipse

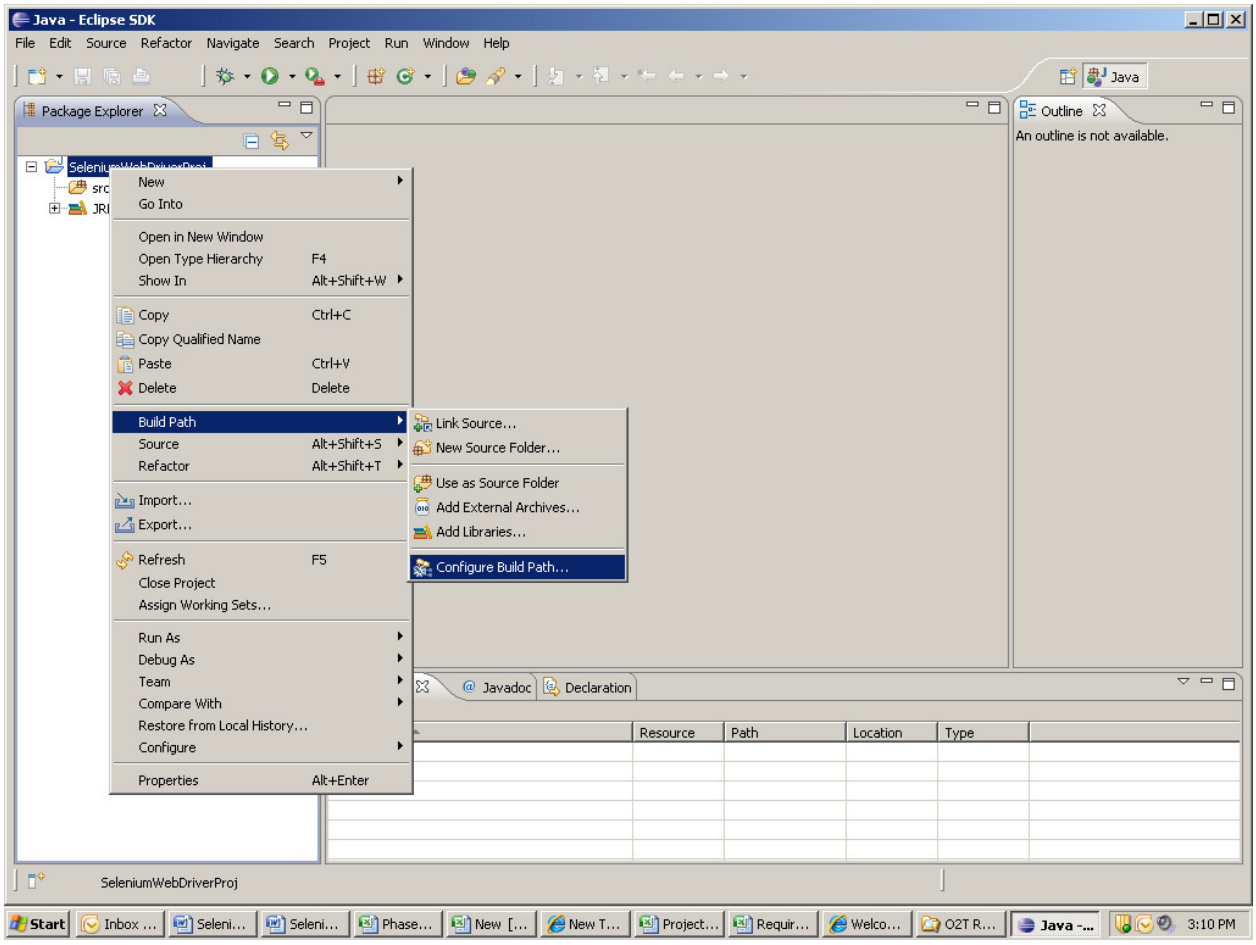


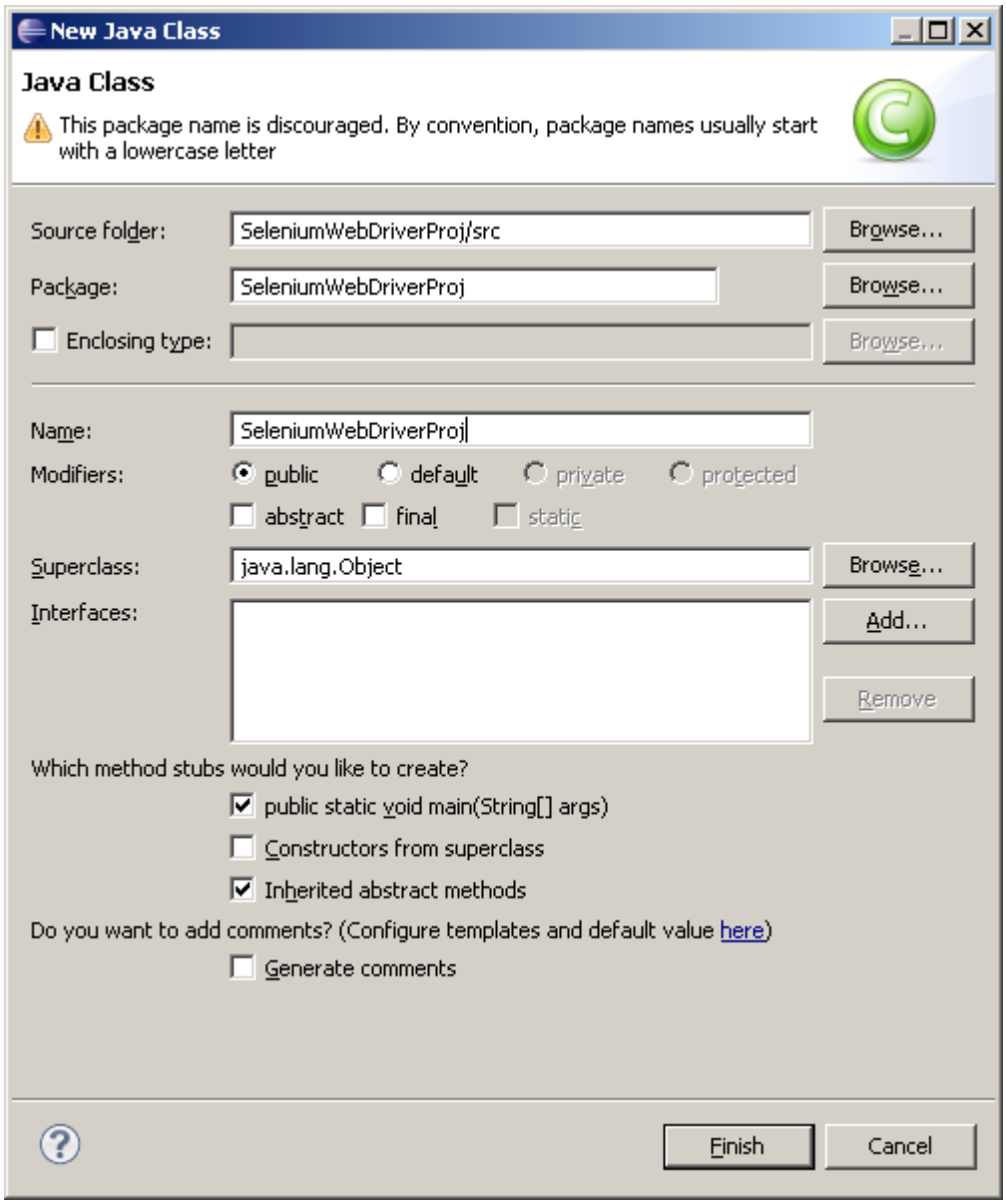


Note: Please note that JavaSE-1.7 has to be used

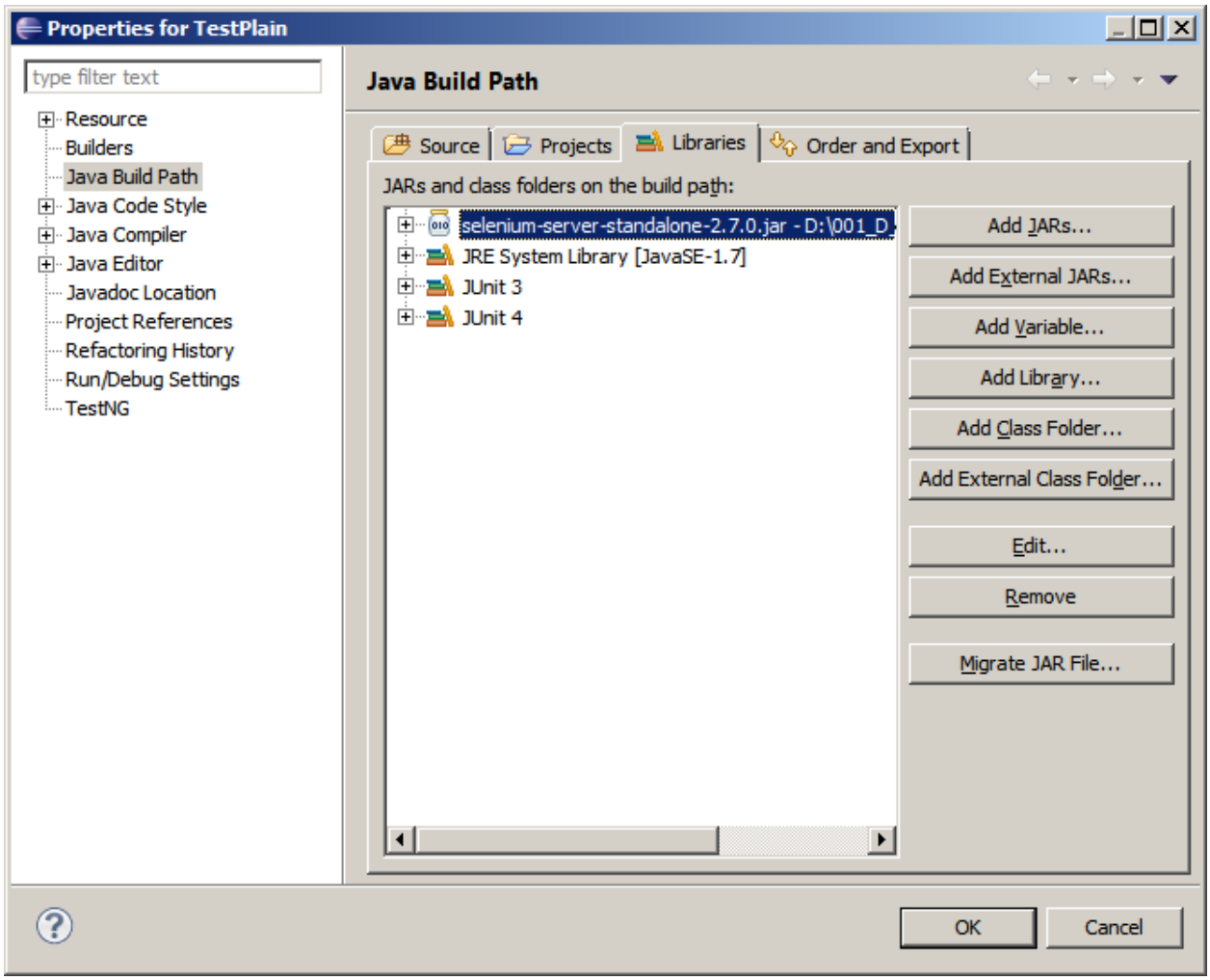


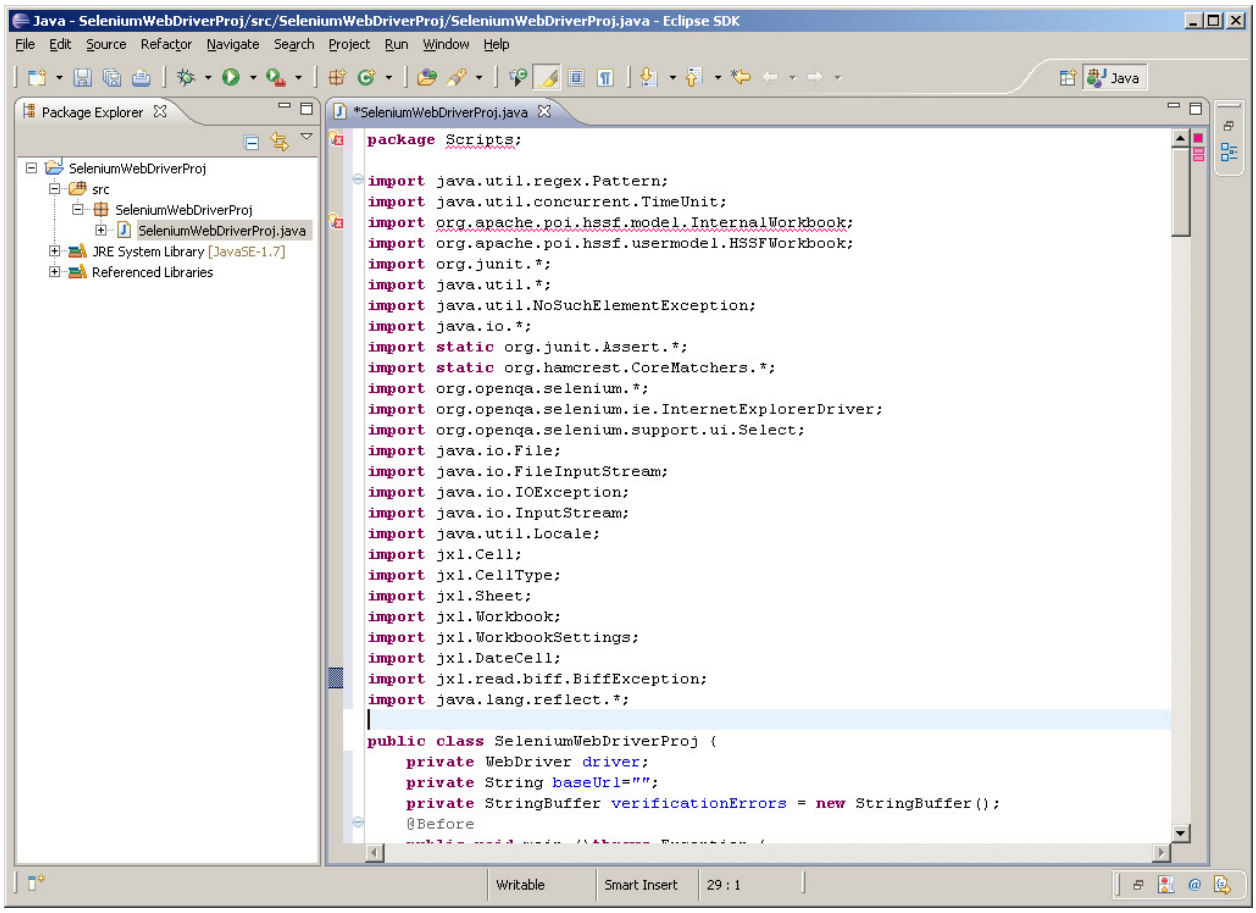
Add the required jar files to the library:


Java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Outline
An outline is not available.
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Build Path
Source Alt+Shift+S
Refactor Alt+Shift+T
Import...
Export...
Refresh F5
Close Project
Assign Working Sets...
Run As
Debug As
Team
Compare With
Restore from Local History...
Configure
Properties Alt+Enter
Link Source...
New Source Folder...
Use as Source Folder
Add External Archives...
Add Libraries...
Configure Build Path...
Javadoc
Declaration
Resource
Path
Location
Type
SeleniumWebDriverProj
Start
3:10 PM



Remove the contents of the new class and Copy the framework into the new class file


Properties for TestPlain
type filter text
Resource
Builders
Java Build Path
Java Code Style
Java Compiler
Java Editor
Javadoc Location
Project References
Refactoring History
Run/Debug Settings
TestNG
Java Build Path
Source
Projects
Libraries
Order and Export
JARs and class folders on the build path:
selenium-server-standalone-2.7.0.jar - D:\001_D
JRE System Library [JavaSE-1.7]
JUnit 3
JUnit 4
Add JARs...
Add External JARs...
Add Variable...
Add Library...
Add Class Folder...
Add External Class Folder...
Edit...
Remove
Migrate JAR File...
OK
Cancel


Java - SeleniumWebDriverProj/src/SeleniumWebDriverProj/SeleniumWebDriverProj.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run Window Help
Java
Package Explorer
SeleniumWebDriverProj
src
SeleniumWebDriverProj
SeleniumWebDriverProj.java
JRE System Library [JavaSE-1.7]
Referenced Libraries
*SeleniumWebDriverProj.java
package Scripts;
import java.util.regex.Pattern;
import java.util.concurrent.TimeUnit;
import org.apache.poi.hssf.model.InternalWorkbook;
import org.apache.poi.hssf.usermodel.HSSFWorkbook;
import org.junit.*;
import java.util.*;
import java.util.NoSuchElementException;
import java.io.*;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.*;
import org.openqa.selenium.*;
import org.openqa.selenium.ie.InternetExplorerDriver;
import org.openqa.selenium.support.ui.Select;
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.InputStream;
import java.util.Locale;
import jxl.Cell;
import jxl.CellType;
import jxl.Sheet;
import jxl.Workbook;
import jxl.WorkbookSettings;
import jxl.DateCell;
import jxl.read.biff.BiffException;
import java.lang.reflect.*;
public class SeleniumWebDriverProj {
private WebDriver driver;
private String baseUrl="";
private StringBuffer verificationErrors = new StringBuffer();
@Before
Writable
Smart Insert
29 : 1

4 Usage of Keywords

The keywords for the test scenario should be entered in the Sheet1 of the Microsoft Excel placed in the Test Scripts folder. The syntax for the keywords can be referred from Selenium Keywords Reference Dictionary Document.

5 Test Results for a Keyword-Driven Script

Test execution results can be viewed and analyzed as soon as the run session ends. To access the test results, go to the Test_Reports folder customized using the test automation framework. Two folders will be available: one showing the summary report for test suite execution, and anoter folder called Detailed_Report that displays the detailed step-wise test results for each test script. A screenshot will be available for the failure scripts under ScreenShot_Report.