

# kSelenium 终极自动化测试环境搭建（一）

## Selenium+Eclipse+Junit+TestNG

### 第一步 安装 JDK

Jdk1.7.

下载地址:

<http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>

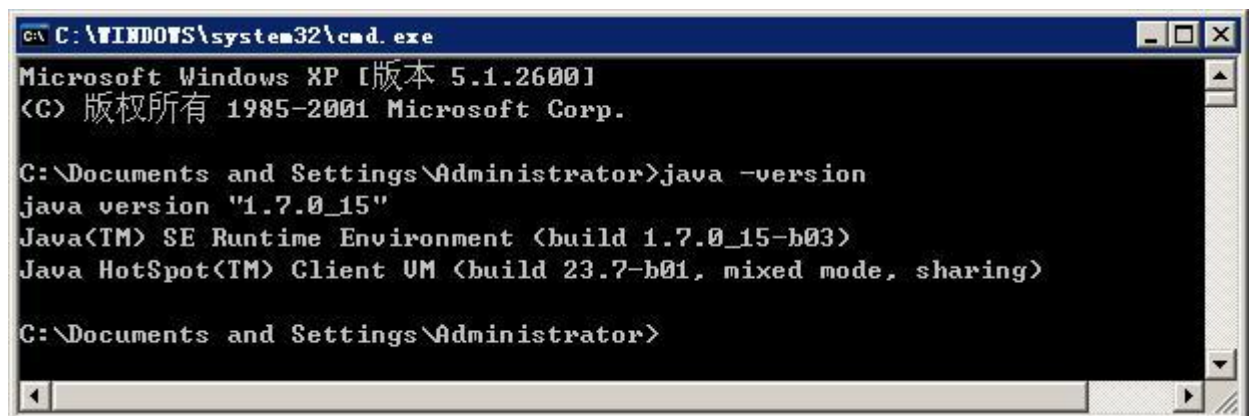
一路猛击‘下一步’，OK。安装完成后配置环境变量:

JAVA\_HOME = E:\Java\Java\jdk1.7.0\_15

PATH = %JAVA\_HOME%\bin

CLASSPATH = .;%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\lib\tools.jar

配置完环境变量后，CMD 命令行输入: java -version，返回如下结果，则表示安装成功:



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [版本 5.1.2600]
(C) 版权所有 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>java -version
java version "1.7.0_15"
Java(TM) SE Runtime Environment (build 1.7.0_15-b03)
Java HotSpot(TM) Client VM (build 23.7-b01, mixed mode, sharing)

C:\Documents and Settings\Administrator>
```

### 第二步 下载 Eclipse

下载地址: <http://www.eclipse.org/download/>

最新的 Eclipse Standard 4.3, 198 MB, 下载的都是不用安装的，解压出来后直接用。

### 第三步 下载 Selenium IDE、SeleniumRC、IEDriverServer、SeleniumClient

#### Drivers

下载地址: <http://www.seleniumhq.org/download/>

- 1、 Selenium IDE: [selenium-ide-2.2.0.xpi](#) 用来在 Firefox 上录制脚本。
- 2、 Selenium RC: [selenium-server-standalone-2.33.0.jar](#) 模拟服务器端, 不可少。
- 3、 IEDriverServer: [DriverServer\\_Win32\\_2.33.0.zip](#) IE 驱动, Firefox 和 chorm 不用驱动。
- 4、 Selenium Client Drivers: [selenium-java-2.33.0.zip](#) 模拟 Selenium 客户端。

## Downloads

Below is where you can find the latest releases of all the Selenium components. You can also find a list of [previous releases](#), [source code](#), and additional information for [Maven users](#) (Maven is a popular Java build tool).

### Selenium IDE

Selenium IDE is a Firefox plugin which records and plays back user interactions with the browser. Use this to either create simple scripts or assist in exploratory testing. It can also export Remote Control or WebDriver scripts, though they tend to be somewhat brittle and should be overhauled into some sort of Page Object-y structure for any kind of resiliency.

Download latest released version [2.2.0](#) released on 06/July/2013 or view the [Release Notes](#) and then [install some plugins](#).

**selenium-ide-2.2.0.xpi**

Download version under development [unreleased](#)

### Selenium Server (formerly the Selenium RC Server)

The Selenium Server is needed in order to run either Selenium RC style scripts or Remote Selenium Webdriver ones. The 2.x server is a drop-in replacement for the old Selenium RC server and is designed to be backwards compatible with your existing infrastructure.

Download version [2.33.0](#) **selenium-server-standalone-2.33.0.jar**

To use the Selenium Server in a Grid configuration [see the wiki page](#).

### The Internet Explorer Driver Server

This is required if you want to make use of the latest and greatest features of the WebDriver InternetExplorerDriver. Please make sure that this is available on your \$PATH (or %PATH% on Windows) in order for the IE Driver to work as expected.

Download version 2.33.0 for (recommended) [32 bit Windows IE](#) or [64 bit Windows IE](#)

### Selenium Client & WebDriver Language Bindings **IEDriverServer\_Win32\_2.33.0.zip**

In order to create scripts that interact with the Selenium Server (Selenium RC, Selenium Remote Webdriver) or create local Selenium WebDriver script you need to make use of language-specific client drivers. These languages include both 1.x and 2.x style clients.

While language bindings for

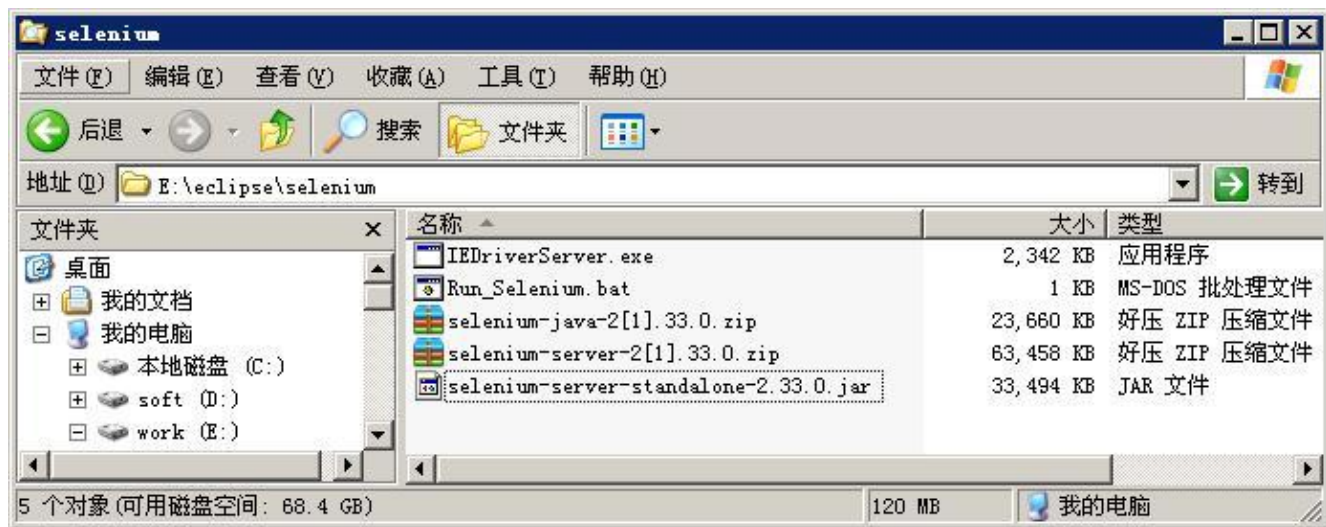
[other languages exist](#), these are the core ones that are supported by the main project hosted on google code.

#### Language Client Version Release Date

Language	Client Version	Release Date	Download	Change log	API docs
Java	2.33.0	2013-05-22	<a href="#">Download</a>	<a href="#">Change log</a>	<a href="#">Javadoc</a>
C#	2.33.0	2013-05-22	<a href="#">Download</a>	<a href="#">Change log</a>	<a href="#">API docs</a>
Ruby	2.32.0	2013-04-09	<a href="#">Download</a>	<a href="#">Change log</a>	<a href="#">API docs</a>
Python	2.33.0	2013-05-22	<a href="#">Download</a>	<a href="#">Change log</a>	<a href="#">API docs</a>

**selenium-java-2.33.0.zip**

这里，我将下载得到的所有文件，全存放在 E:\eclipse\selenium 下面，方便管理：



#### 第四步 下载 Firefox

下载地址：<http://www.firefox.com.cn/download/>

下载得到文件：Firefox-latest.exe

#### 第五步 安装 IDE、Firebug、Xpath checker、Xpath finder

安装完 Firefox 后，打开 Firefox，把前面下载的 selenium-ide-2.2.0xpi 拖放到 Firefox，弹出下图后，安装即可。





Firebug、Xpath checker、Xpath finder, 打开 firefox 浏览器, 选择工具——附加组件, 打开附加组件管理器页面, 搜索 firebug、Xpath。

将查询到的 firebug、xpath checker、xpath finder 都装上, 重启浏览器后生效:



SeleniumIDE、Firebug 和 xpath 的用法, 可以百度 Selenium 私房菜 (新手入门教程).pdf, 里面有很好的说明。

## 第六步 启动 SeleniumRC

启动 seleniumRC 的方法:

cmd 命令行进入 selenium-server-standalone-2[1].33.0.jar 存放目录, 输入如下命令  
java -jar selenium-server-standalone-2[1].12.0.jar

```
C:\WINDOWS\system32\cmd.exe - java -jar selenium-server-standalone-2.33.0.jar
(C) 版权所有 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>E:

E:\>cd eclipse\selenium

E:\eclipse\selenium>java -jar selenium-server-standalone-2.33.0.jar
七月 20, 2013 12:47:07 上午 org.openqa.grid.selenium.GridLauncher main
INFO: Launching a standalone server
00:47:07.781 INFO - Java: Oracle Corporation 23.7-b01
00:47:07.781 INFO - OS: Windows XP 5.1 x86
00:47:07.781 INFO - v2.33.0, with Core v2.33.0. Built from revision 4e90c97
00:47:07.968 INFO - RemoteWebDriver instances should connect to: http://127.0.0.
1:4444/wd/hub
00:47:07.968 INFO - Version Jetty/5.1.x
00:47:07.968 INFO - Started HttpContext[/selenium-server/driver,/selenium-server
/driver]
00:47:07.968 INFO - Started HttpContext[/selenium-server,/selenium-server]
00:47:07.968 INFO - Started HttpContext[/,/]
00:47:08.031 INFO - Started org.openqa.jetty.jetty.servlet.ServletHandler@3492fc
00:47:08.031 INFO - Started HttpContext[/wd,/wd]
00:47:08.031 INFO - Started SocketListener on 0.0.0.0:4444
00:47:08.031 INFO - Started org.openqa.jetty.jetty.Server@d9205
```

为了方便，可以将启动命令写一个 bat 来执行，Run\_selenium.bat，内容如下：

```
@echo off

cd E:\eclipse\selenium

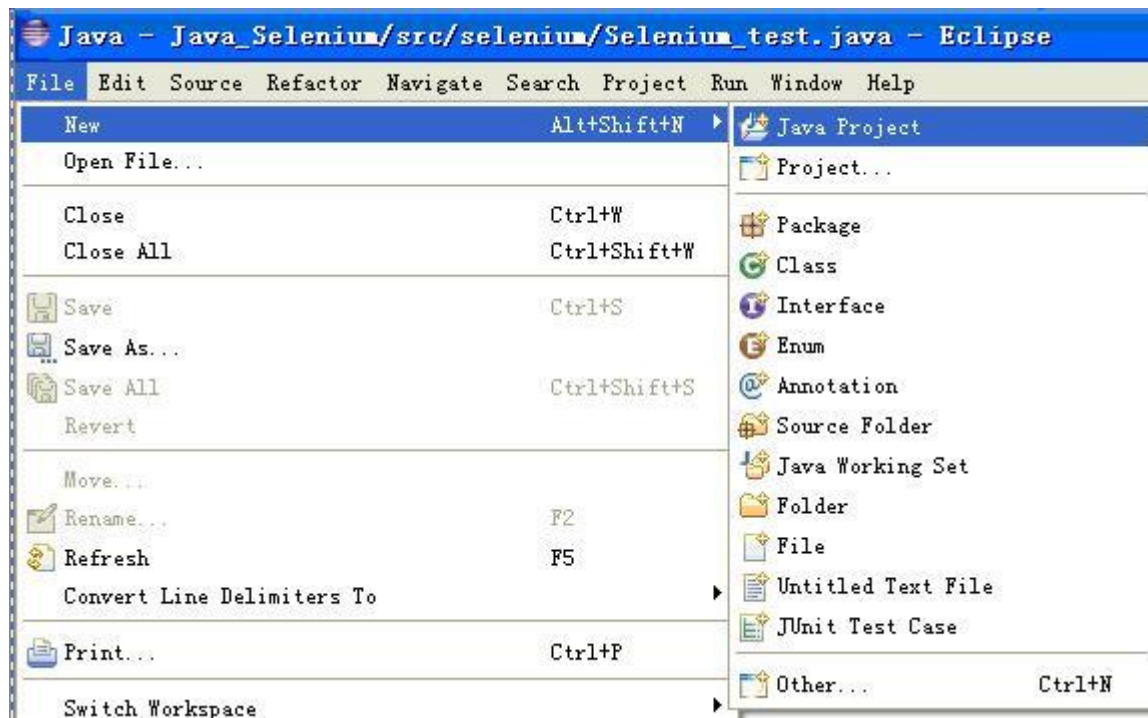
E:

java -jar selenium-server-standalone-2.33.0.jar
```

## 第七步 Eclipse 执行 Selenium 的 Java 实例

### -----7.1

打开 Eclipse，新建一个工程 File—new—Java Project



## -----7.2

输入工程名: Selenium, next



## -----7.3

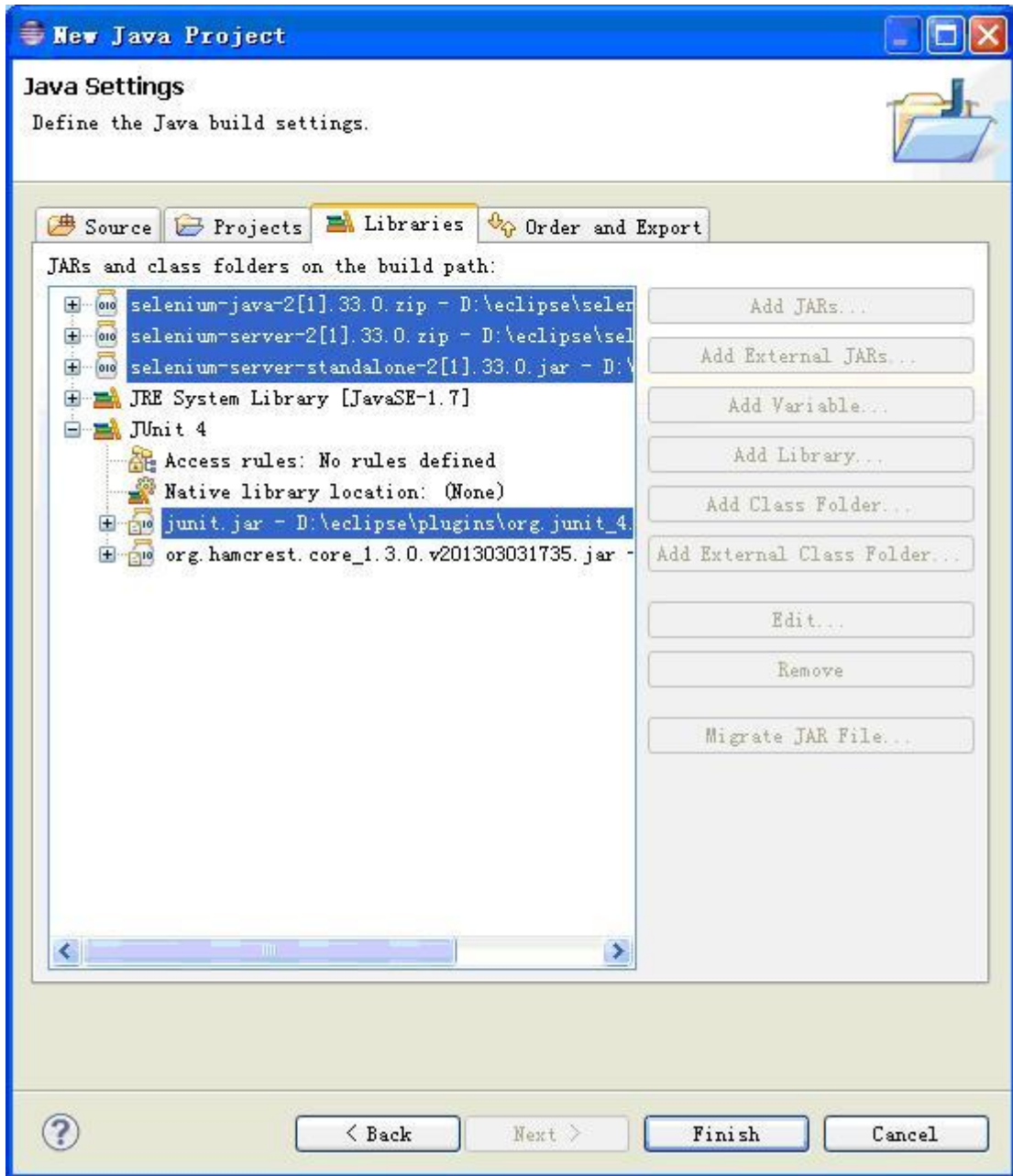
接下来, 窗口进入 Java Settings, 选择 Libraries, 点击 Addlibrary.

引用 Junit4 的 Jar 包(E:\eclipse\plugins\org.junit\_4.11.0.v2XXXX).

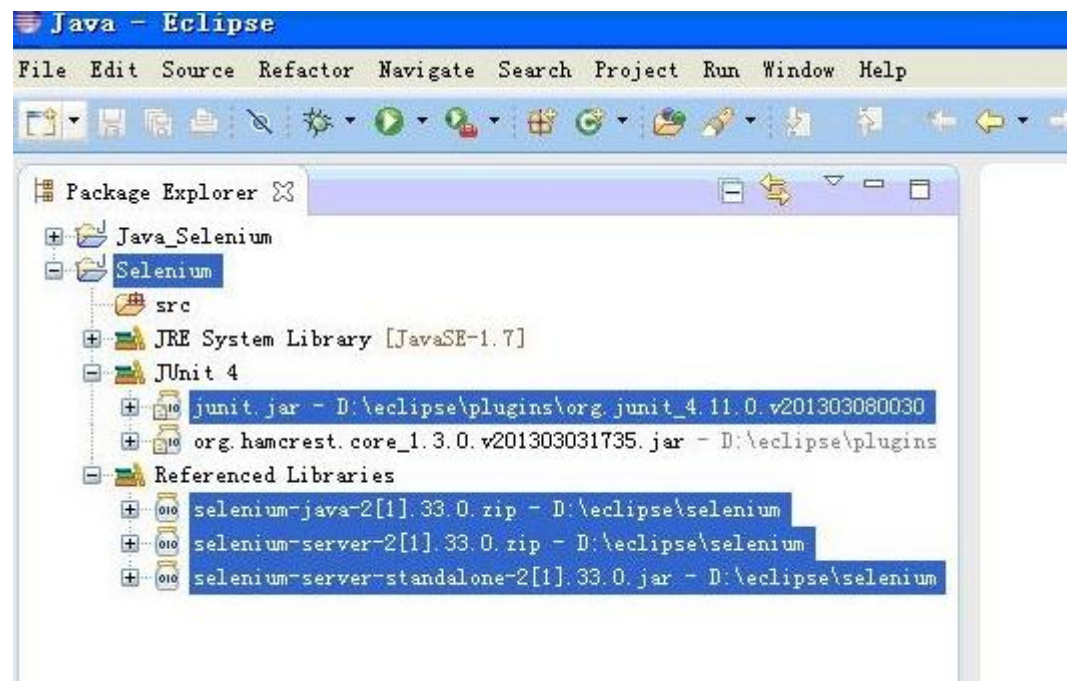
然后点击 Add External Jars...

引用 Selenium 相关的包(E:\eclipse\selenium),最终 Libraries 如下:





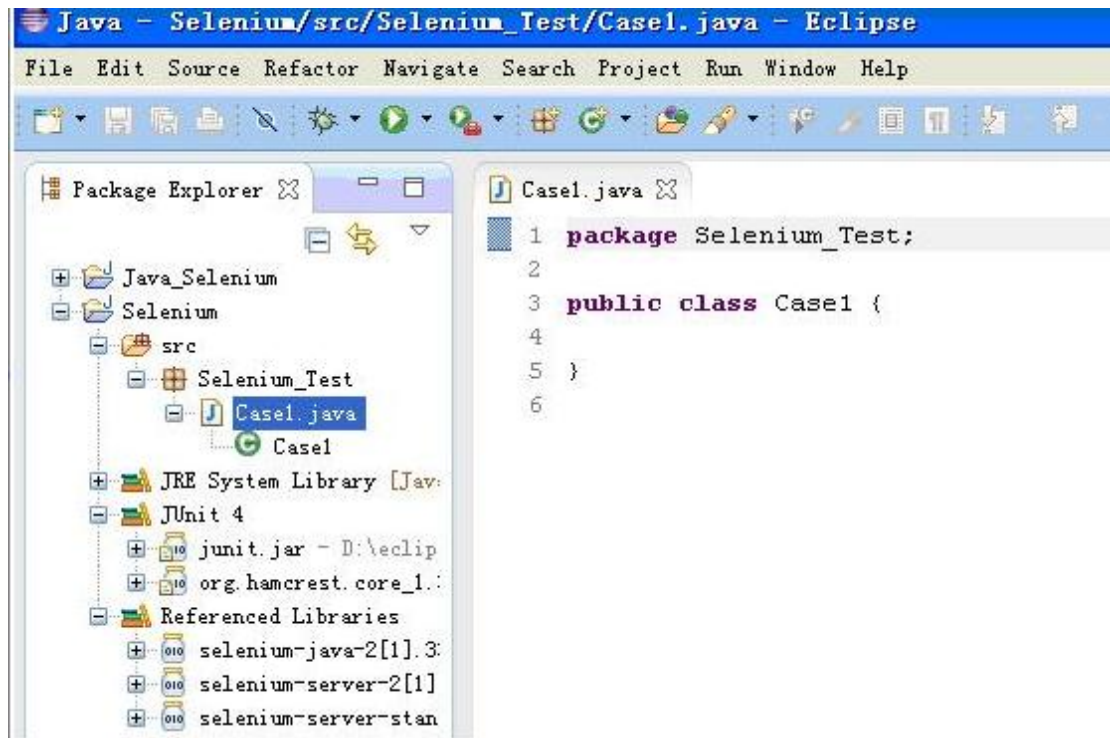
完成后，Java 视图如下：



#### -----7.4

右击 src，new->package 新建一个包 Selenium\_Test，

再右击包 Selenium\_Test，new->class，新建一个 Class 类 Case1.java，最终效果如下：



## -----7.5

下面我们来用 IE 浏览器执行一个实例，修改 **Case1.java**，代码如下：

```
package Selenium_Test;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.ie.InternetExplorerDriver;
import org.openqa.selenium.remote.DesiredCapabilities;

public class Casel {

public static void main(String[] args) {

    System.setProperty("webdriver.ie.driver",

        "E:\\eclipse\\selenium\\IEDriverServer.exe");//注意这里 IEDriverServer.
exe 的文件存放路径

    DesiredCapabilities ieCapabilities = DesiredCapabilities

        .internetExplorer();

    ieCapabilities

        .setCapability(

            InternetExplorerDriver.INTRODUCE_FLAKINESS_BY_IGNORING_SECURITY_DO
MAINS,
```

```
        true);  
  
        WebDriver driver = new InternetExplorerDriver(ieCapabilities);  
        driver.get("http://www.google.com.hk");  
        WebElement element = driver.findElement(By.name("q"));  
        element.sendKeys("hello Selenium!");  
        element.submit();  
  
        try {  
            Thread.sleep(3000);  
        } catch (InterruptedException e) {  
            e.printStackTrace();  
        }  
        System.out.println("Page title is: " + driver.getTitle());  
        driver.quit();  
    }  
}
```

## -----7.6

运行 Run\_selenium.bat, 启动 Selenium RC 服务器。

然后右击 Case1.Java, Run As—>Java Application, 执行成功结果如下:

```
*Casel.java ☒
10
11
12 public static void main(String[] args) {
13
14     System.setProperty("webdriver.ie.driver", "E:\\eclipse\\seleni
15     DesiredCapabilities ieCapabilities = DesiredCapabilities.inte
16     ieCapabilities.setCapability(InternetExplorerDriver.INTRODUCE
17     WebDriver driver = new InternetExplorerDriver(ieCapabilities)
18
19
20     driver.get("http://www.google.com.hk");
21     WebElement element = driver.findElement(By.name("q"));
22     element.sendKeys("hello Selenium!");
23     element.submit();
24     try {
25         Thread.sleep(3000);
26     } catch (InterruptedException e) {
27         e.printStackTrace();
28     }
29     System.out.println("Page title is: " + driver.getTitle());
30
31     driver.quit();
32 }
33
34 }
```

Problems @ Javadoc Declaration Console ☒

<terminated> Casel (1) [Java Application] E:\Java\jdk1.7.0\_15\bin\javaw.exe (2013年7月19日 下午8:25  
INFO: I/O exception (org.apache.http.NoHttpResponseException) caught when  
七月 19, 2013 8:25:46 下午 org.apache.http.impl.client.DefaultRequestDirecto  
INFO: Retrying request  
Page title is: hello Selenium! - Google 搜索

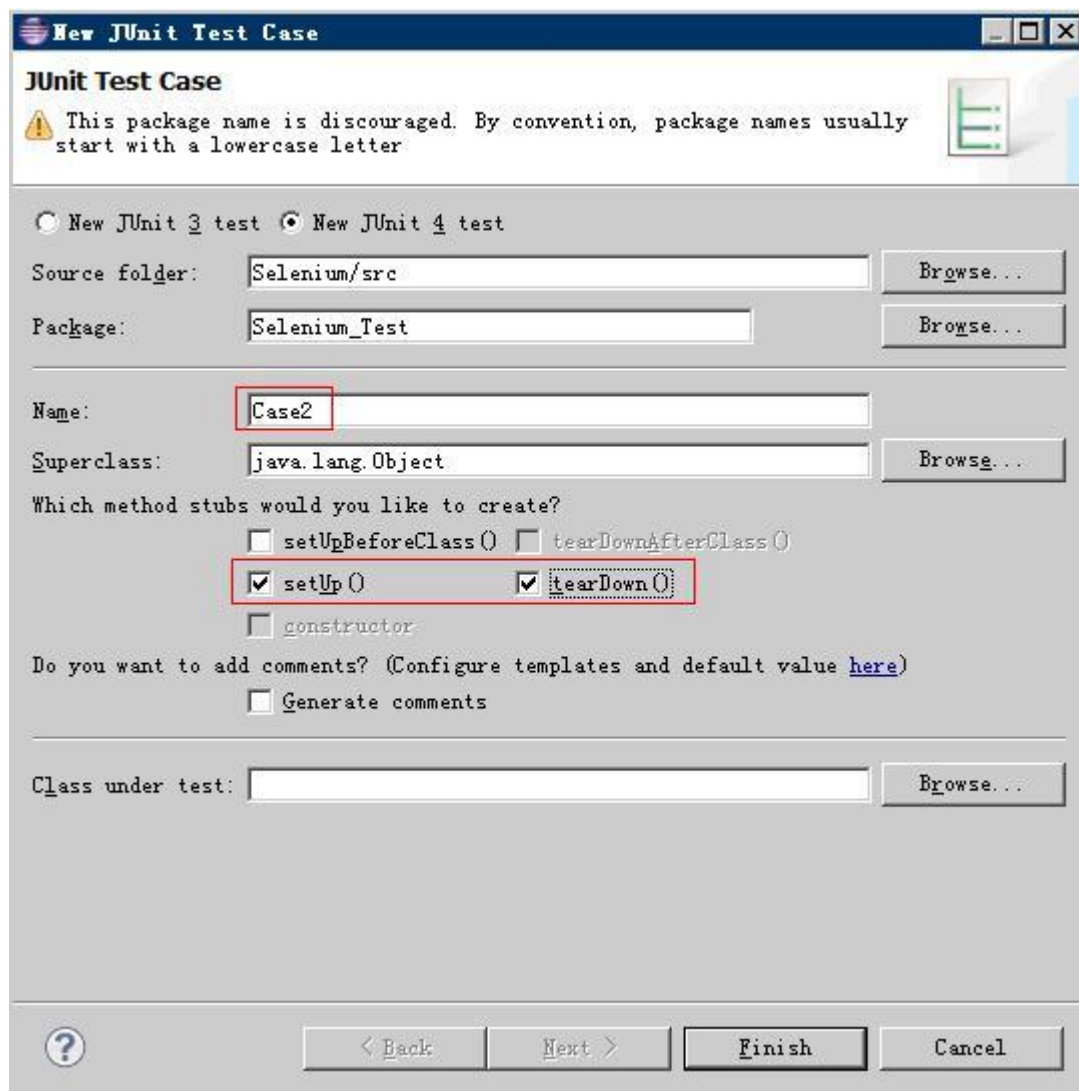


下面我们通过 Junit 来运行脚本，脚本需要修改一下，因为 Junit 的 Java 文件有它自己的格式。

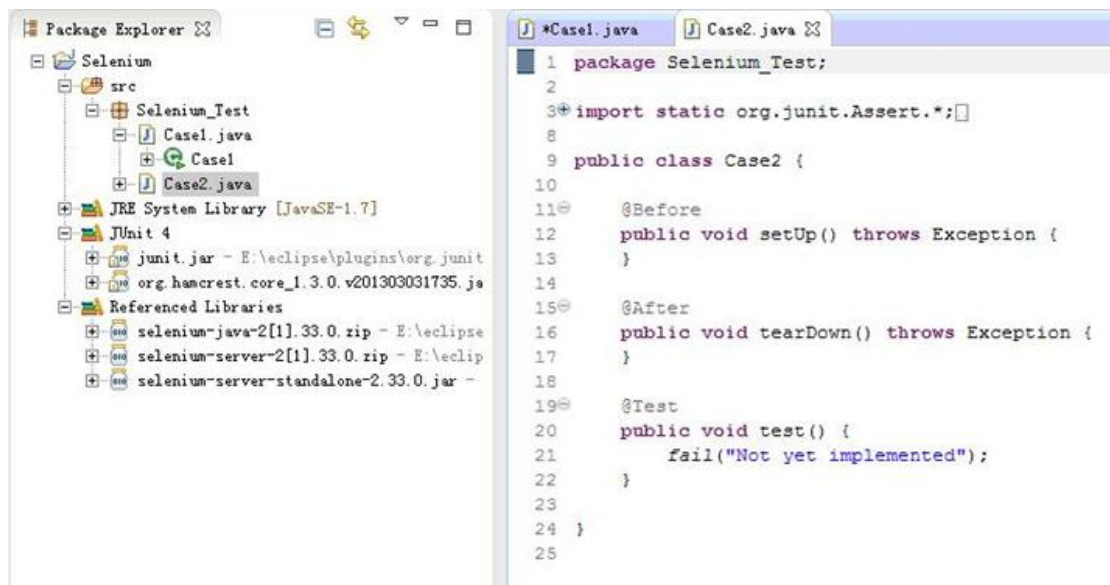
## 第八步 Eclipse 通过 Junit 执行 Selenium 的 Java 实例

### -----8.1

右击 Selenium\_Test, new->JUnit test case 新建一个 Case2.java。



完成后如下：



The screenshot shows the Eclipse IDE interface. On the left, the Package Explorer displays a project named 'Selenium' with a source folder 'src'. Inside 'src', there is a package 'Selenium\_Test' containing three files: 'Case1.java', 'Case1', and 'Case2.java'. Below the source files, the JRE System Library [JavaSE-1.7] and JUnit 4 are listed, including 'junit.jar' and 'org.hamcrest.core\_1.3.0.v201303031735.jar'. Under 'Referenced Libraries', there are three Selenium-related JAR files: 'selenium-java-2[1].33.0.zip', 'selenium-server-2[1].33.0.zip', and 'selenium-server-standalone-2.33.0.jar'.

On the right, the editor shows the code for 'Case2.java'. The code is as follows:

```
1 package Selenium_Test;
2
3 import static org.junit.Assert.*;
4
5
6
7
8
9 public class Case2 {
10
11     @Before
12     public void setUp() throws Exception {
13     }
14
15     @After
16     public void tearDown() throws Exception {
17     }
18
19     @Test
20     public void test() {
21         fail("Not yet implemented");
22     }
23
24 }
25
```

## -----8.2

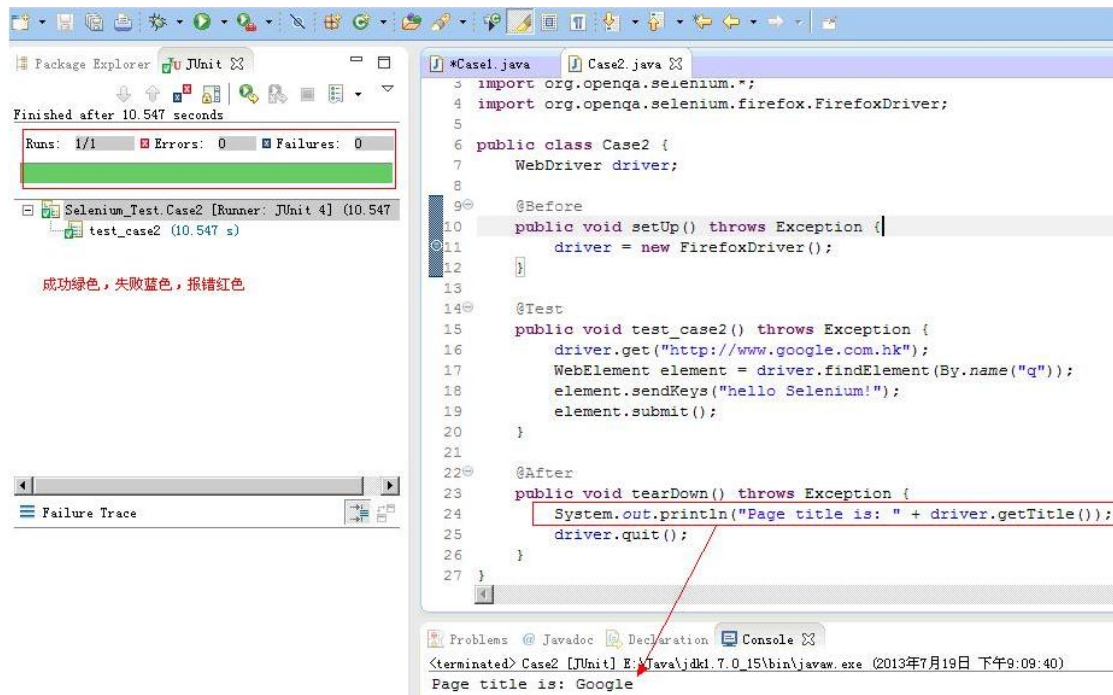
修改 `Case2.java` 代码如下:

```
package Selenium_Test;
import org.junit.*;
import org.openqa.selenium.*;
import org.openqa.selenium.firefox.FirefoxDriver;
public class Case2 {
    WebDriver driver;
    @Before
    public void setUp() throws Exception {
        driver = new FirefoxDriver();
    }
    @Test
    public void test_case2() throws Exception {
        driver.get("http://www.google.com.hk");
        WebElement element = driver.findElement(By.name("q"));
        element.sendKeys("hello Selenium!");
        element.submit();
    }
    @After
    public void tearDown() throws Exception {
        System.out.println("Page title is: " + driver.getTitle());
        driver.quit();
    }
}
```

### -----8.3

运行 Run\_selenium.bat, 启动 Selenium RC 服务器 (前面 RC 启动后若未关闭, 则无需启动多个)。

右击 Case2.java, Run As—>JUnit Test, 执行成功结果如下:

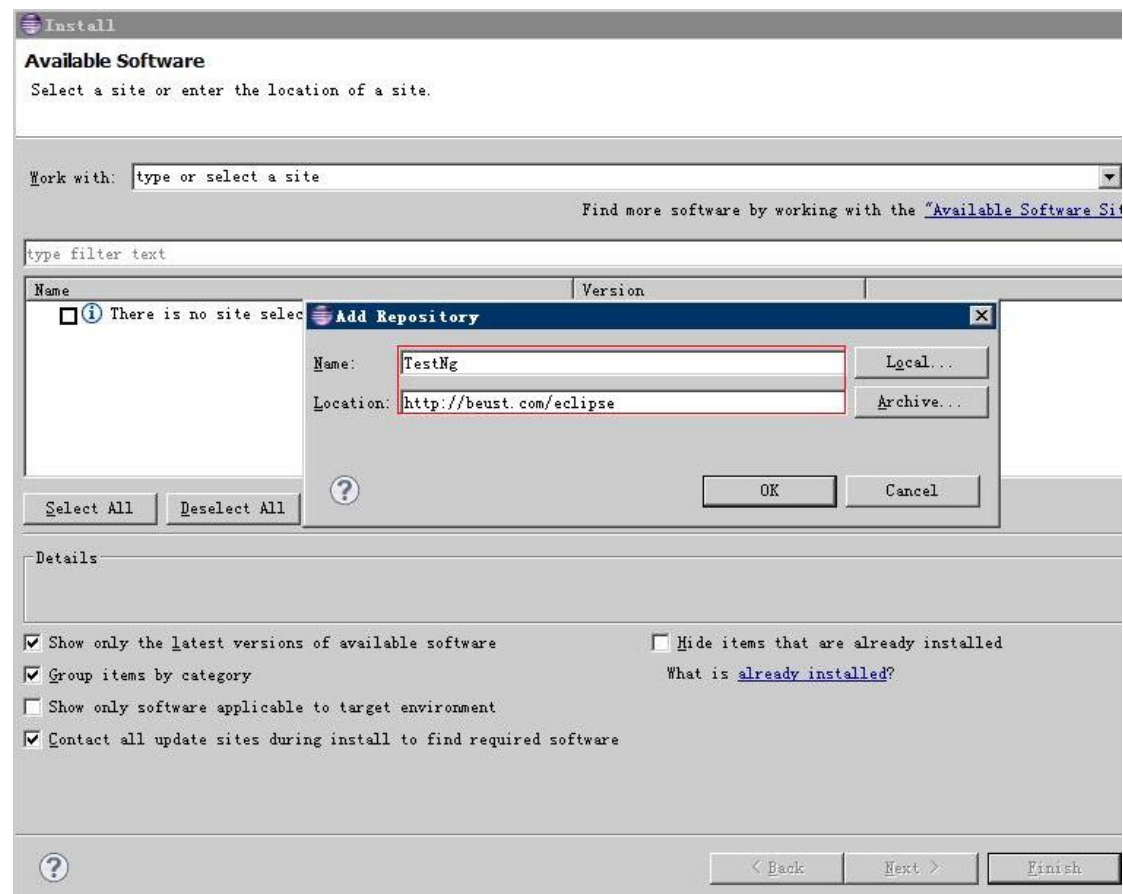


## 第九步 Eclipse 通过 TestNG 执行 Selenium 的 Java 实例

### -----9.1

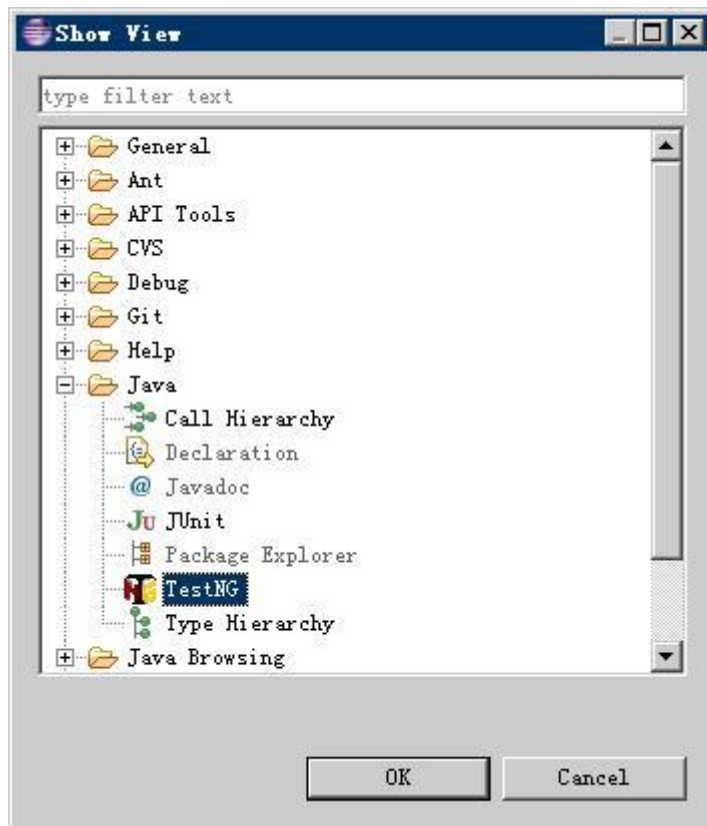
#### 安装 TestNG

在 Eclipse 中, 点击 Help -> Install new software, 在 add 栏中输入 <http://beust.com/eclipse>, 在下面就会看到 TestNG. 选中点击安装, 按下一步直到安装完, 在线安装会有点很慢。



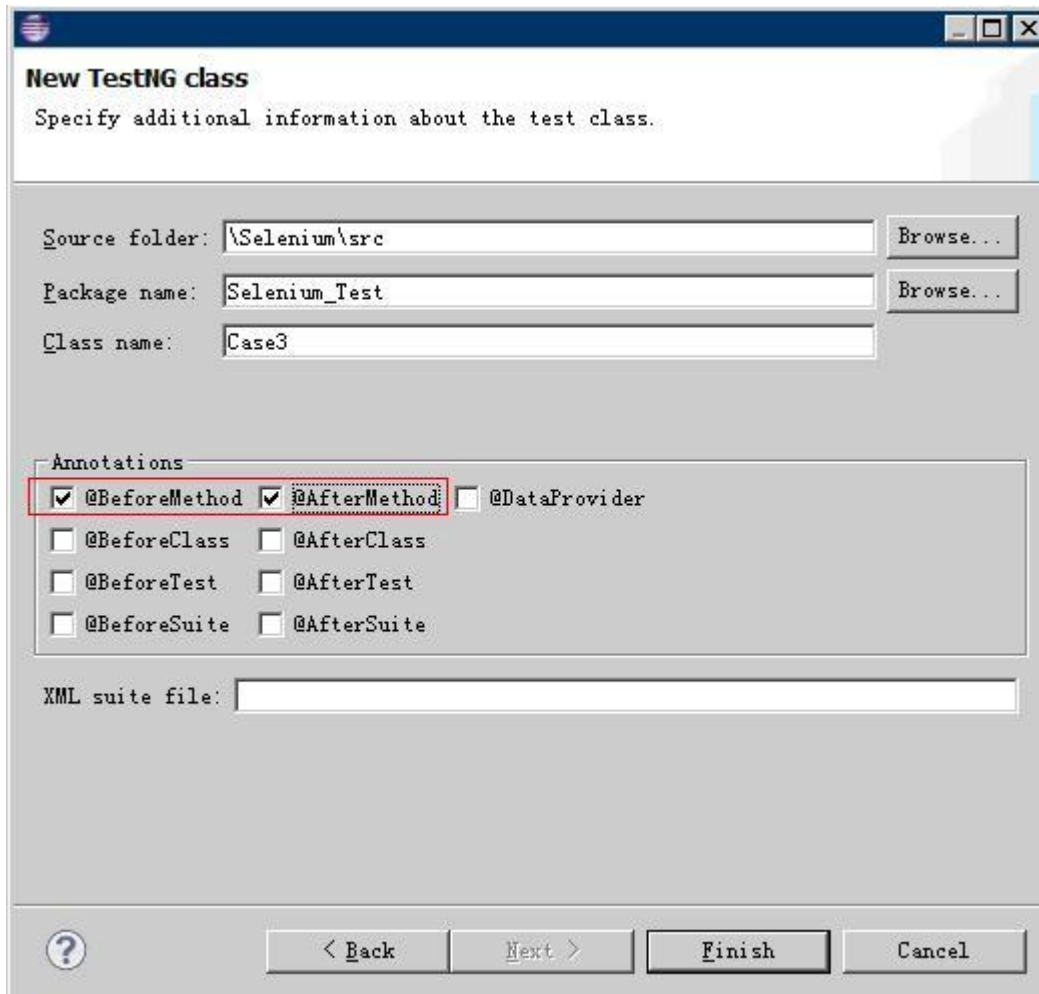


安装完重启 Eclipse 后，在 window->Show View->other 里面选中 Java->TestNG，就会出现 TestNG 选项了。

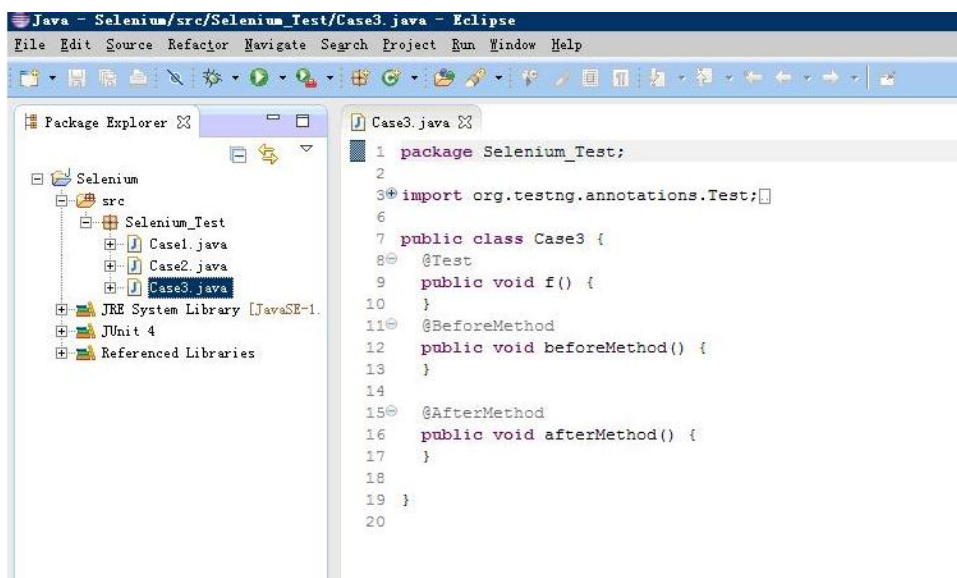


## -----9.2

右击包 Selenium\_Test,new->other->TestNG 新建一个 TestNG 的测试类 Case3.java。



完成后如下:



修改 **Case3.java** 脚本内容如下:

```
package Selenium_Test;

import org.testng.annotations.Test;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.AfterMethod;

import org.openqa.selenium.firefox.FirefoxDriver;

public class Case3 {

    WebDriver driver;

    @BeforeMethod

    public void beforeMethod() {

    }

    @AfterMethod

    public void afterMethod() {

        System.out.println("Page title is: " + driver.getTitle());

        driver.quit();

    }

    @Test

    public void test_case3() {

        driver = new FirefoxDriver();

        driver.get("http://www.google.com.hk");

        WebElement element = driver.findElement(By.name("q"));

        element.sendKeys("hello Selenium!");

        element.submit();

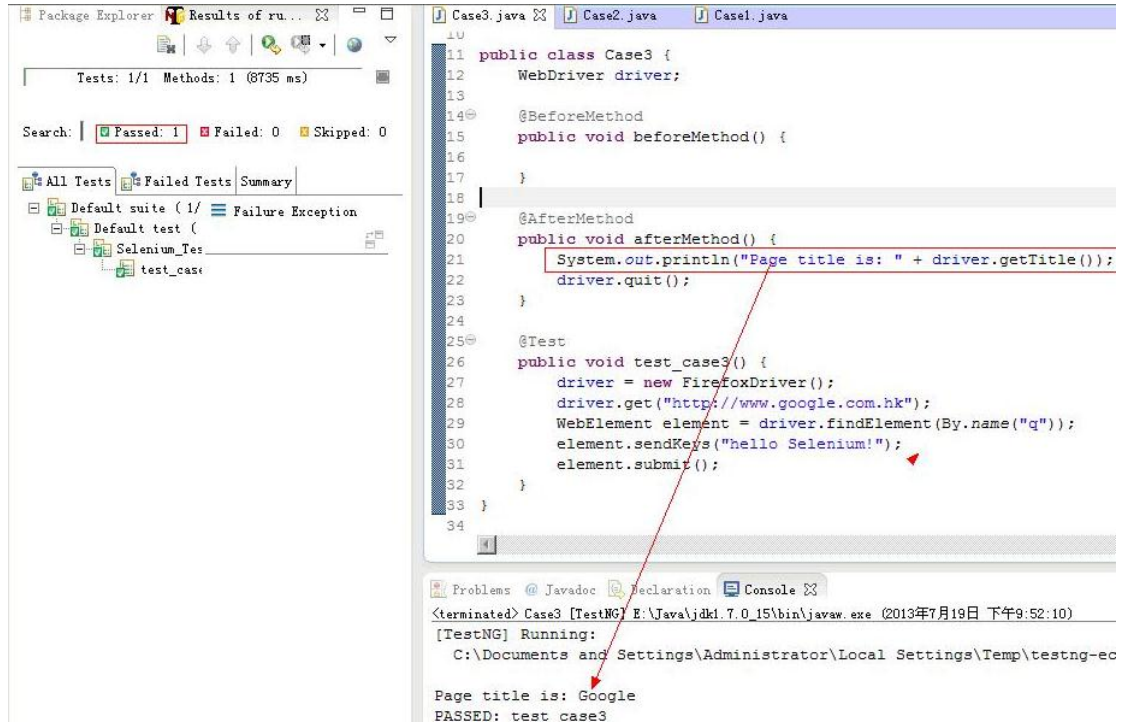
    }

}
```

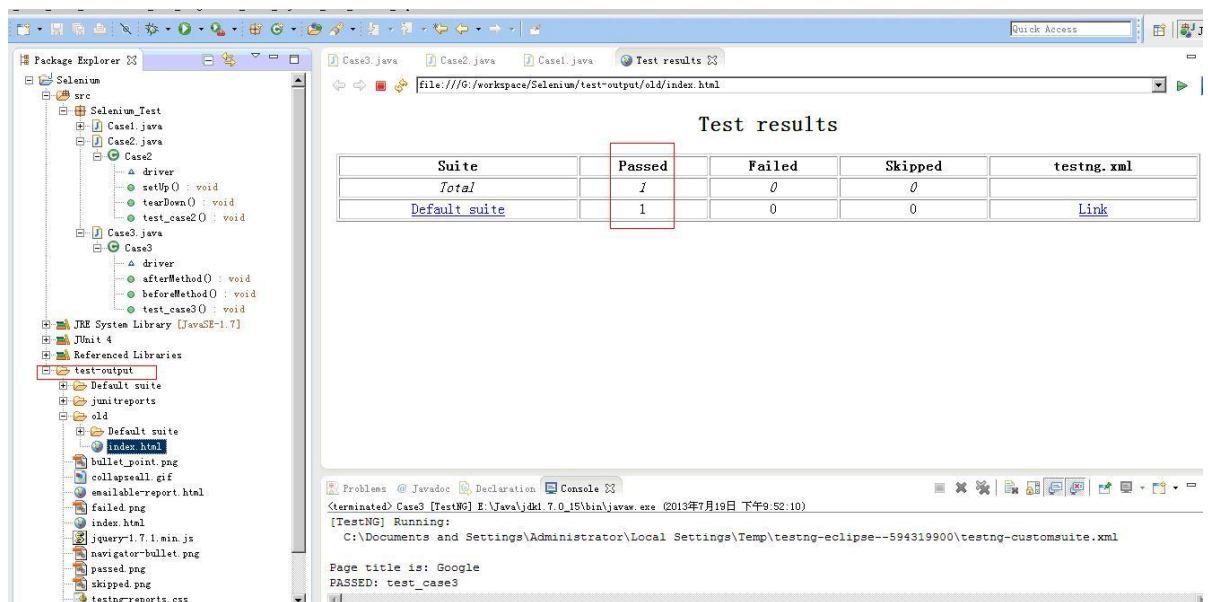
### -----9.3

运行 Run\_selenium.bat, 启动 Selenium RC 服务器。

右击 Case3.java, Run as->TestNG Test, 执行成功结果如下:



执行完, 会生成一个 test-output 文件夹, 文件夹下面的 index.html 就是测试报告, 如下:



以上是在 Eclipse 下如何搭建 Selenium 的测试环境，包括直接执行.java，通过 Junit 执行.java，通过 TestNG 执行.java。

## Selenium 终极自动化测试环境搭建（二）

### Selenium+Eclipse+Python

前面举例了 Selenium+Eclipse+Junit+TestNG 自动化测试环境的搭建，在前一篇的基础上，下面再举例 Selenium+Eclipse+Python 测试环境搭建。

#### 第一步：安装 Python

根据下面的地址，直接一键安装，全部默认方式。

下载地址：<http://www.python.org/ftp/python/2.7.5/python-2.7.5.msi>

安装到 C:\Python27，设置 Python 环境变量，Path = E:\Python27;（前面安装 JDK 时，已经有了 Path 环境变量，这里直接加在最前面即可，注意 Python27 后面的';'）

#### 第二步：安装 Python 的 SetupTools

其实 SetupTools 就是一个帮助你安装第三方工具包的增强工具软件，根据下面的地址下载，然后按下一步一键安装。

setuptools-0.6c11.win32-py2.7.exe.exes

下载地址：

<http://pypi.python.org/packages/2.7/s/setuptools/setuptools-0.6c11.win32-py2.7.exe#md5=57e1e64f6b7c7f1d2eddfc9746bbaf20>

#### 第三步：安装 Python 的包管理工具 pip—有点类似 SetupTools ,但是比它强大

利用第二步安装的 SetupTools 进行安装，打开 DOS 界面，进入到目录：

C:\Python27\Scripts， 然后敲入命令： easy\_install pip， 等待完成就 OK。



```
C:\Python27\Scripts>easy_install pip
Searching for pip
Reading http://pypi.python.org/simple/pip/
Reading http://pip.openplans.org
Reading http://www.pip-installer.org
Best match: pip 1.1
Downloading http://pypi.python.org/packages/source/p/pip/pip-1.1.tar.gz#md5=62a9
f08dd5dc69d76734568a6c040508
Processing pip-1.1.tar.gz
Running pip-1.1\setup.py -q bdist_egg --dist-dir c:\docume~1\admini~1\locals~1\te
mp\easy_install-rrs8ls\pip-1.1\egg-dist-temp-vkxjfp
warning: no files found matching '*.html' under directory 'docs'
warning: no previously-included files matching '*.txt' found under directory 'do
cs\_build'
no previously-included directories found matching 'docs\_build\_sources'
Adding pip 1.1 to easy-install.pth file
Installing pip-script.py script to C:\Python27\Scripts
Installing pip.exe script to C:\Python27\Scripts
Installing pip.exe.manifest script to C:\Python27\Scripts
Installing pip-2.7-script.py script to C:\Python27\Scripts
Installing pip-2.7.exe script to C:\Python27\Scripts
Installing pip-2.7.exe.manifest script to C:\Python27\Scripts

Installed c:\python27\lib\site-packages\pip-1.1-py2.7.egg
Processing dependencies for pip
Finished processing dependencies for pip

C:\Python27\Scripts>
```

#### 第四步：安装基于 Python 的 Selenium 包

打开 DOS 界面，进入到目录： C:\Python27\Scripts

然后敲入命令： pip install selenium 或者 pip install -U selenium（用后一个貌似报错，用前一个可安装。）

```
C:\Python27\Scripts>pip install selenium
Downloading/unpacking selenium
  Downloading selenium-2.20.0.tar.gz (4.2Mb): 93% 3.9Mb
```

安装时可能会有一些警告，暂不用管，安装完后如下，

```
C:\Python27\Scripts>pip install selenium
Downloading/unpacking selenium
  Downloading selenium-2.20.0.tar.gz (4.2Mb): 4.2Mb downloaded
  Running setup.py egg_info for package selenium
    C:\Python27\lib\distutils\dist.py:267: UserWarning: Unknown distribution option: 'src_root'
      warnings.warn(msg)

    warning: no files found matching 'docs\api\py\index.rst'
Installing collected packages: selenium
  Running setup.py install for selenium
    C:\Python27\lib\distutils\dist.py:267: UserWarning: Unknown distribution option: 'src_root'
      warnings.warn(msg)

    warning: no files found matching 'docs\api\py\index.rst'
Successfully installed selenium
Cleaning up...
```

#### 第五步：验证 Selenium 安装是否成功

在记事本中编写下面的代码：（保存为 `pytest.py`，然后双击直接运行即可！）

```
from selenium import webdriver
browser = webdriver.Firefox()
browser.get("http://www.yahoo.com")
assert "Yahoo!" in browser.title
browser.close()
```

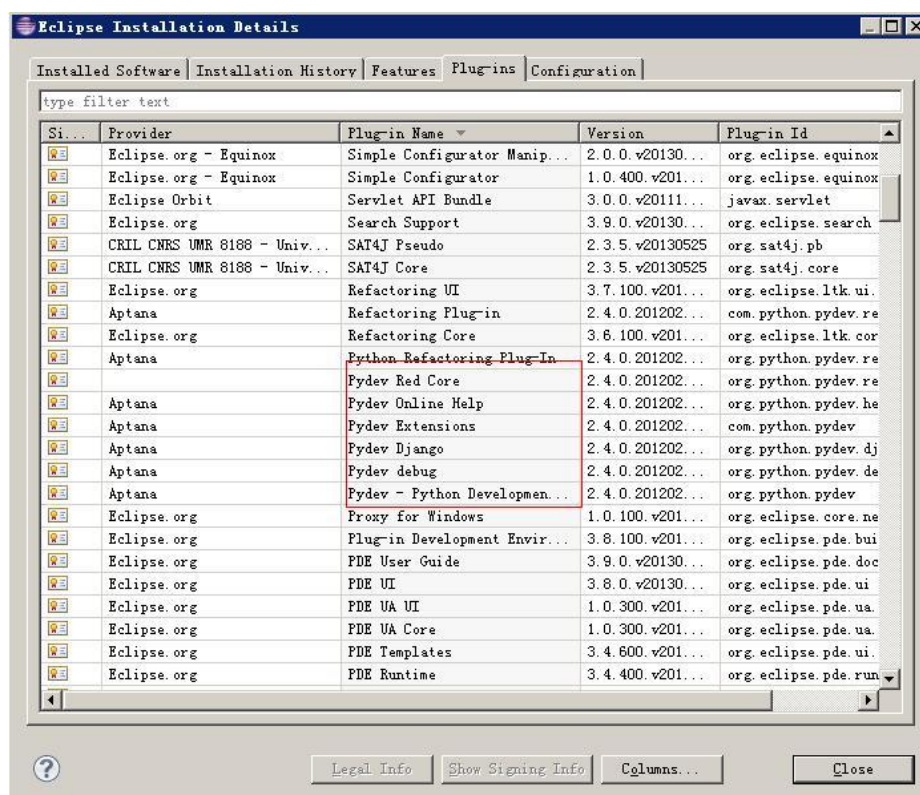
如果代码运行成功，就表示 **Selenium** 安装成功了！ **Very Good!**

## 第六步：python 的开发环境配置-Eclipse-PyDev 插件安装

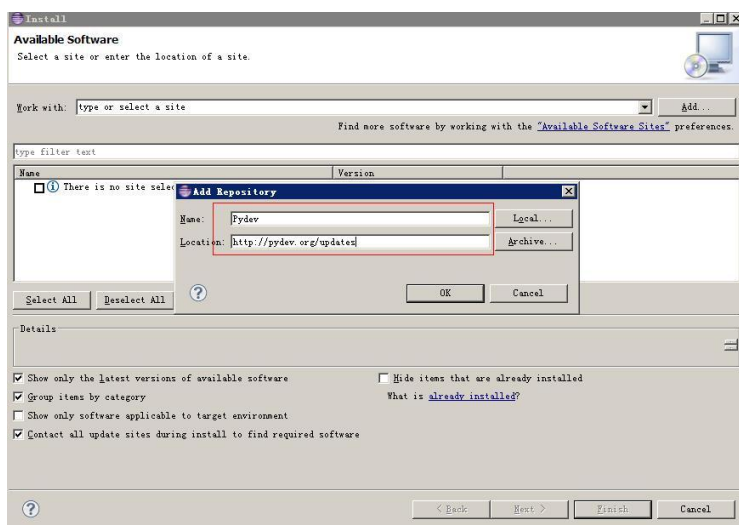
安装 PyDev 插件的两种安装方法：

1、百度搜索 PyDev 2.4.0.zip，下载后解压，得到 Plugins 和 Feature 文件夹，复制两文件夹到 Eclipse 目录，覆盖即可。

完成后重启 Eclipse，若在 Eclipse 菜单 Help->About Eclipse->Installation Detail->Plug-ins，能看到 PyDev 组件，则表示安装成功。



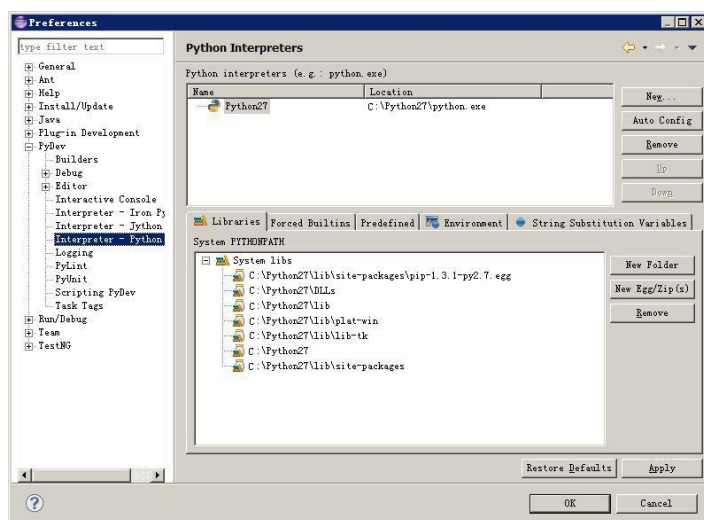
2、直接在 Eclipse 中选择菜单：Help—Install New Software..—Add，输入 <http://pydev.org/updates>，下载并安装。



## 配置 PyDev

安装好 PyDev 之后，需要配置 Python/Jython 解释器，配置过程很简单。

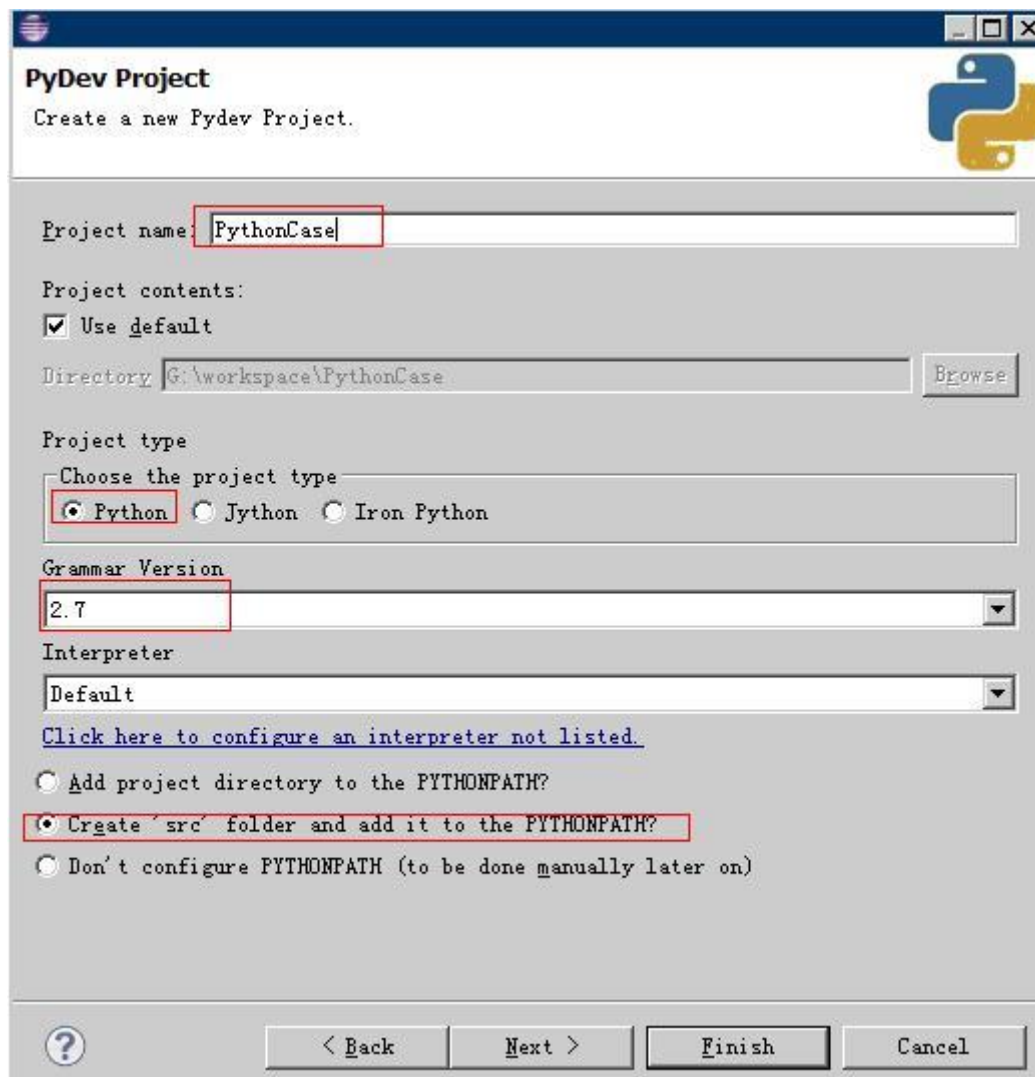
在 Eclipse 菜单栏中，选择 Window > Preferences > Pydev > Interpreter - Python，在这里配置 Python/解释器，添加已安装的解释器。这里，Python 安装在 C:\Python27 路径下。单击 New，选择 Python 解释器 python.exe，打开后显示出一个包含很多复选框的窗口，选择需要加入系统 **PYTHONPATH** 的路径，单击 **OK**。



## 第七步：执行 Selenium 实例

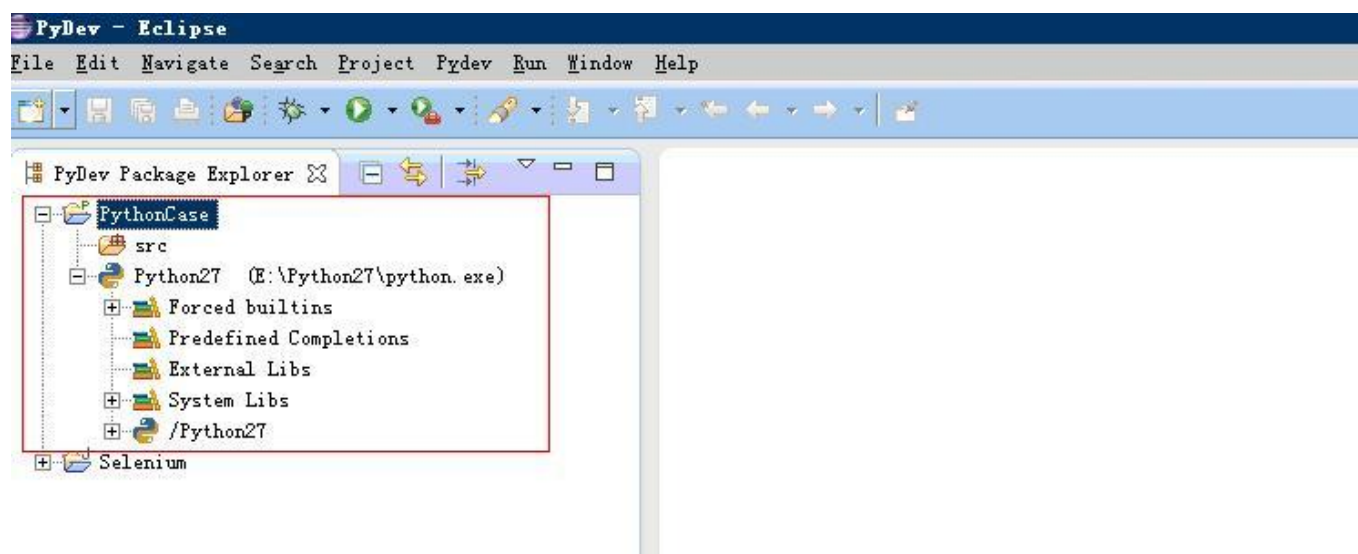
下面，我们来创建一个 python 项目。

在 Eclipse 菜单栏中，选择 File > New > Project > Pydev > Pydev Project，新建项目：PythonCase，单击 Next。





完成后如下：



## 创建 Python 包和模块

接下来，在刚创建的项目中开始创建 Python 包和模块。

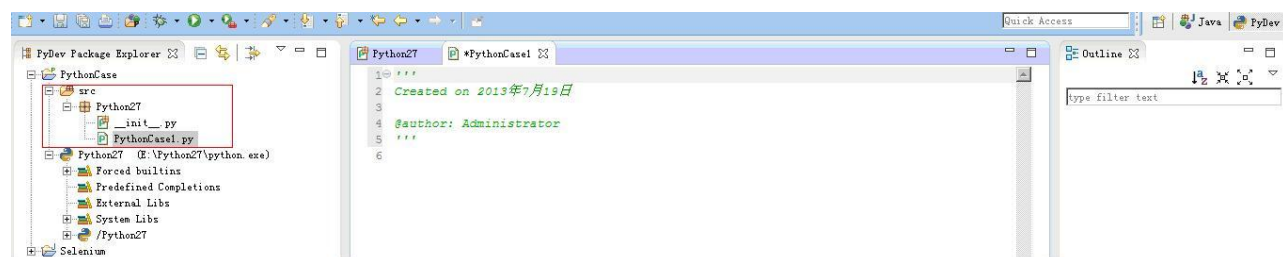
进入 Pydev 透视图，在 Python Package Explorer 中，右键单击 src，选择 New->Pydev Package，输入 Package 名称 Python27。

单击 Finish，Python 包就创建好了，此时，自动生成\_\_init\_\_.py 文件，该文件不包含任何内容。

注意：

如果在创建项目的时候没有选中“Create default src folder and add it to the pythonpath”复选框，则需要通过 File > New > Other > Source Folder 手动创建一个源代码文件夹 src。

创建完 Pydev Package 后，右键单击创建的包，选择 New->Pydev Module，输入模块名称 PythonCase1.py Finish。这样，Python 模块就建成了。



修改 `PythonCase1.py` 内容如下:

```
#!/usr/bin/env python
#-*- coding=utf-8 -*-

from selenium import webdriver

if __name__ == "__main__":

    driver = webdriver.Firefox()

    driver.implicitly_wait(30)

    driver.get("http://www.google.com.hk")

    driver.find_element_by_name("q").send_keys("hello Selenium!")

    driver.find_element_by_name("q").submit()

    print 'Page title is:',driver.title

    driver.quit()
```

### 执行脚本

运行 `Run_selenium.bat`, 启动 Selenium RC 服务器。右击 `PythonCase1.py`, Run As->Python Run, 执行成功结果如下:

