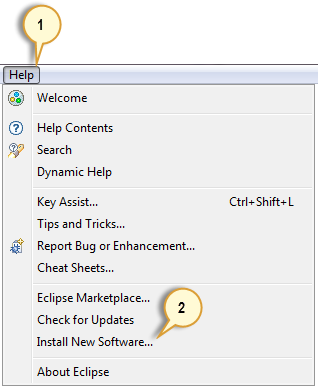
Selenium Test NG and Log4j

**Installing TestNG in Eclipse**

**Step 1**

* Launch Eclipse.
* On the menu bar, click Help.
* Choose the "Install New Software..." option.

[](http://cdn.guru99.com/images/step_1-0046.png)

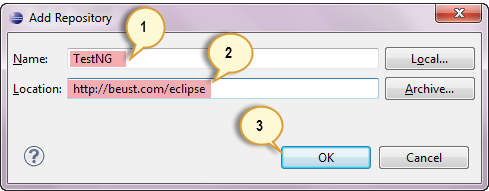
**Step 2**

In the Install dialog box, click the Add button

[How TestNG makes Selenium tests easier](http://cdn.guru99.com/images/step_2-0047.png)

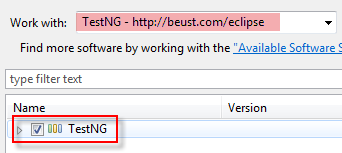
**Step 3**

1. In "Name", type TestNG.
2. In "Location", type <http://beust.com/eclipse>.
3. Click OK

[](http://cdn.guru99.com/images/step_3-0048.png)

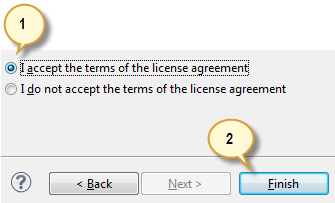
**Step 4**

* Notice that "TestNG - <http://beust.com/eclipse>" was populated onto the "Work with:" textbox.
* Check the "TestNG" check box as shown below, then click Next.
* Note: In the latest Eclipse (Kepler) you don't have a checkbox for TestNG, instead you click on question mark (help) icon which will open up the form, and you can select all and installation will continue as for the remaining instructions. Thanks Jana for the tip!

[](http://cdn.guru99.com/images/step_4-0049.png)

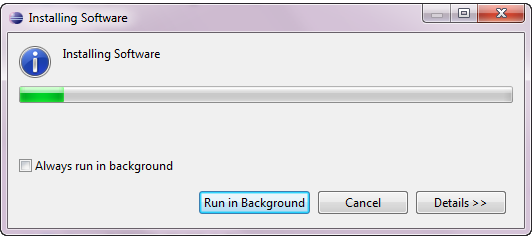
**Step 5**

* Click Next again on the succeeding dialog box until you reach the License Agreement dialog.
* Click "I accept the terms of the license agreement" then click Finish.

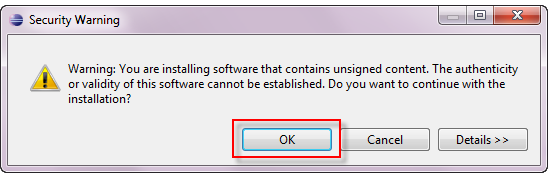
[](http://cdn.guru99.com/images/step_5-0050.png)

**Step 6**

Wait for the installation to finish

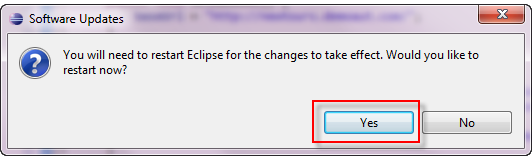
[](http://cdn.guru99.com/images/step_6-0052.png)

If you encounter a Security warning, just click OK

[](http://cdn.guru99.com/images/step_6_b-0051.png)

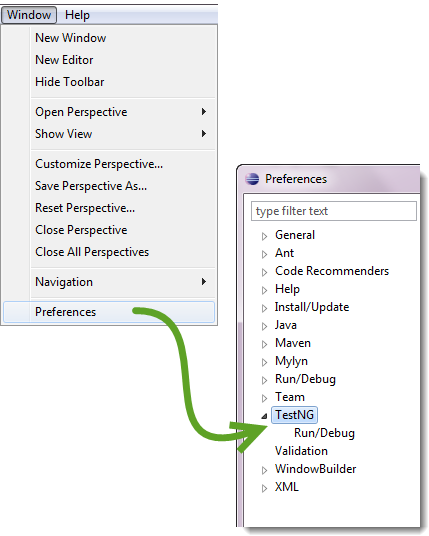
**Step 7**

When Eclipse prompts you for a restart, just click Yes.

[](http://cdn.guru99.com/images/step_7-0053.png)

**Step 8**

After the restart, verify if TestNG was indeed successfully installed. Click Window > Preferences and see if TestNG is included on the Preferences list.

[](http://cdn.guru99.com/images/step_8-0056.png)

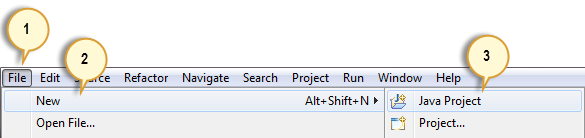
**First test case using annotations**

Before we create a test case, we should first setup a new TestNG Project in Eclipse and name it as "FirstTestNGProject".

**Setting up a new TestNG Project**

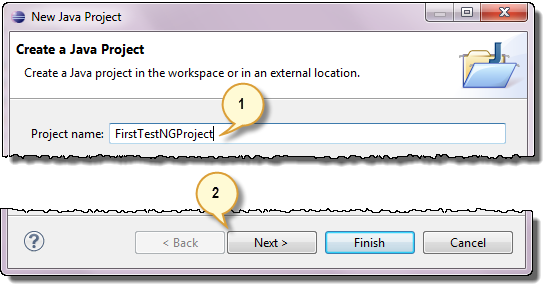
**Step 1**

Click File > New > Java Project

[](http://cdn.guru99.com/images/step_1-0034.png)

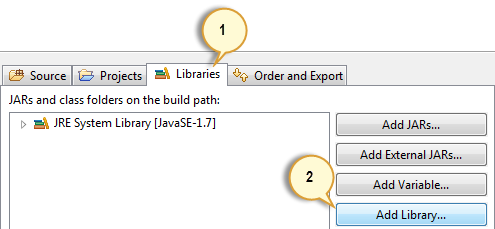
**Step 2**

Type "FirstTestNGProject" as the Project Name then click Next.

[](http://cdn.guru99.com/images/step_2-0037.png)

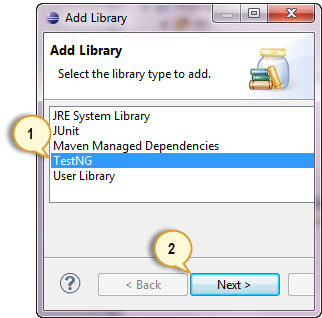
**Step 3**

We will now start to import the TestNG Libraries onto our project. Click on the "Libraries" tab, and then "Add Library…"

[](http://cdn.guru99.com/images/step_3-0038.png)

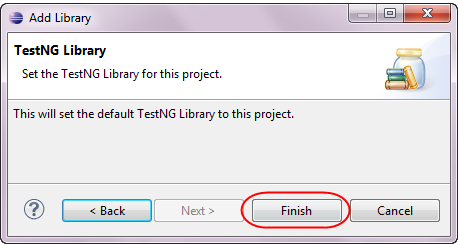
**Step 4**

On the Add Library dialog, choose "TestNG" and click Next.

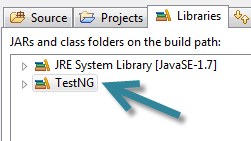
[](http://cdn.guru99.com/images/step_4-0039.png)

**Step 5**

Click Finish.

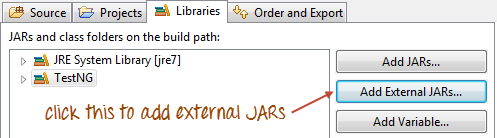
[](http://cdn.guru99.com/images/step_5-0041.png)

You should notice that TestNG is included on the Libraries list.

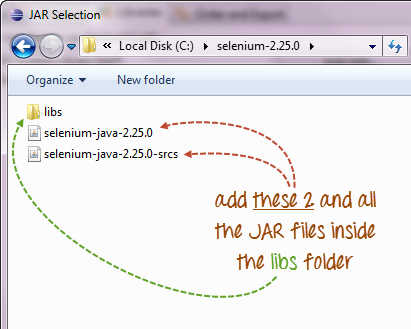
[](http://cdn.guru99.com/images/step_5_b-0040.png)

**Step 6**

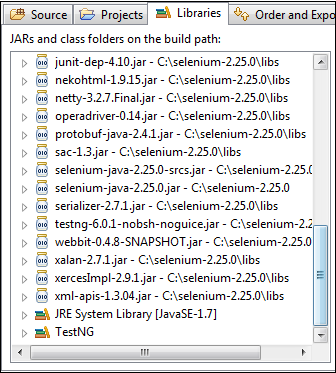
We will now add the JAR files that contain the Selenium API. These files are found in the Java client driver that we downloaded from <http://docs.seleniumhq.org/download/> when we were installing Selenium and Eclipse in the previous chapters.

[](http://cdn.guru99.com/images/step_6a.png)

Then, navigate to where you have placed the Selenium JAR files.

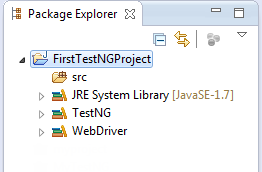
[](http://cdn.guru99.com/images/step_6b.png)

After adding the external JARs, your screen should look like this.

[](http://cdn.guru99.com/images/step_6c.png)

**Step 7**

Click Finish and verify that our FirstTestNGProject is visible on Eclipse's Package Explorer window.

[](http://cdn.guru99.com/images/step_7-0045.png)

**Creating a New TestNG Test File**

Now that we are done setting up our project, let us create a new TestNG file.

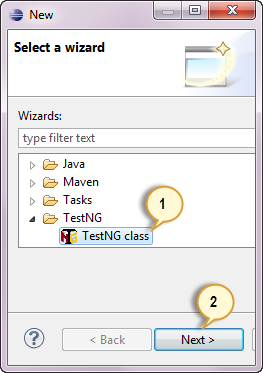
**Step 1**

Right-click on the "src" package folder then choose New > Other…

[](http://cdn.guru99.com/images/step_1-0029.png)

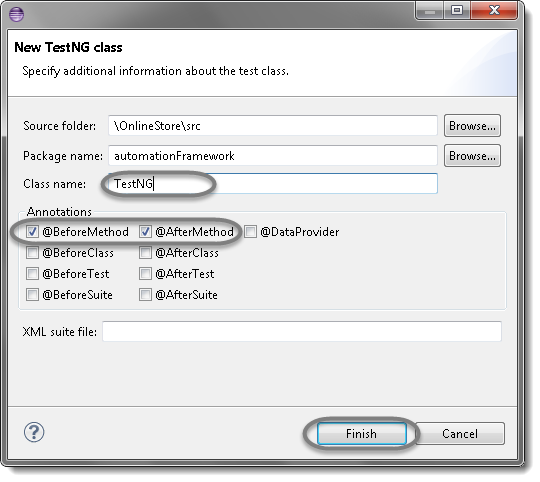
**Step 2**

Click on the TestNG folder and select the "TestNG class" option. Click Next.

[](http://cdn.guru99.com/images/step_2-0030.png)

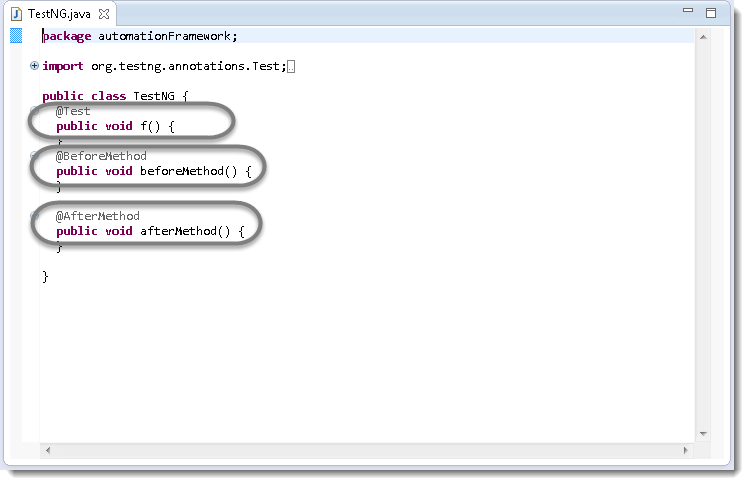
**Step 3**

Type the values indicated below on the appropriate input boxes and click Finish. Notice that we have named our Java file as "FirstTestNGFile".



**Step 4**

Eclipse should automatically create the template for our TestNG file shown below.



**Step 5**

Now it is the time to write the first TestNG test case.

**import** java.util.concurrent.TimeUnit;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.firefox.FirefoxDriver;

**import** org.testng.annotations.Test;

**import** org.testng.annotations.BeforeMethod;

**import** org.testng.annotations.AfterMethod;

**public** **class** FirstTestNGFile {

**public** WebDriver driver;

@Test

**public** **void** main() {

// Find the element that's ID attribute is 'account'(My Account)

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a")).click();

// Find the element that's ID attribute is 'log' (Username)

// Enter Username on the element found by above desc.

driver.findElement(By.*xpath*(".//\*[@id='Email']")).sendKeys("test123@gmail.com");

// Find the element that's ID attribute is 'pwd' (Password)

// Enter Password on the element found by the above desc.

driver.findElement(By.*xpath*(".//\*[@id='Password']")).sendKeys("test@123");

// Now submit the form. WebDriver will find the form for us from the element

driver.findElement(By.*xpath*("html/body/div[5]/div[3]/div/div/div/div[2]/div[1]/div[2]/form/div[3]/input")).click();

// Print a Log In message to the screen

System.***out***.println(" Login Successfully, now it is the time to Log Off buddy.");

// Find the element that's ID attribute is 'account\_logout' (Log Out)

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a"));

}

@BeforeMethod

**public** **void** beforeMethod() {

// Create a new instance of the Firefox driver

driver = **new** FirefoxDriver();

//Put a Implicit wait, this means that any search for elements on the page could take the time the implicit wait is set for before throwing exception

driver.manage().timeouts().implicitlyWait(10, TimeUnit.***SECONDS***);

//Launch the Online Store Website

driver.get("http://demo.nopcommerce.com/");

}

@AfterMethod

**public** **void** afterMethod() {

// Close the driver

driver.quit();

}

}

**Step 3**

Run the test by right click on the test case script and select **Run As** >**TestNG Test**.

**Log4j and LogExpert with Selenium Tutorial**

**Why use Log4j?**

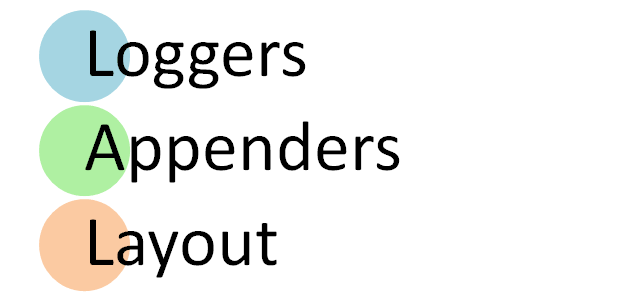
It is an open source

With Log4j, it is possible to store the flow details of our Selenium Automation in a file or databases

Log4j is used for large as well as small projects

In Log4j, we use log statements rather than SOPL statements in the code to know the status of a project while it is executing

**Log4j has three principal components**

[](http://cdn.guru99.com/images/5-2015/050115_0810_TutorialonL1.png)

1. **Loggers**: It is responsible for logging information. To implement loggers into a project following steps need to be performed -

* **Create an instance for logger class**: Logger class is a Java-based utility that has got all the generic methods already implemented to use log4j
* **Define the log level**: Primarily there are five kinds of log levels
  1. All - This level of logging will log everything ( it turns all the logs on )
  2. DEBUG – print the debugging information and is helpful in development stage
  3. INFO – print informational message that highlights the progress of the application
  4. WARN – print information regarding faulty and unexpected system behavior.
  5. ERROR – print error message that might allow system to continue
  6. FATAL – print system critical information which are causing the application to crash
  7. OFF – No logging

1. **Appenders**: It is used to deliver LogEvents to their destination. It decides what will happen with log information. In simple words, it is used to write the logs in file. Following are few types of Appenders
   1. ConsoleAppender logs to standard output
   2. File appender prints logs to some file
   3. Rolling file appender to a file with maximum size

Note: In log4j properties we can call appender with any name. There are other appenders as well but we will restrict to these few.

1. **Layouts**: It is responsible for formatting logging information in different styles.

The Logger class provides different methods to handle logging activities. It provides two static methods for obtaining a Logger Object.

* **Public static Logger getRootLogger()**
* **Public static Logger getLogger(String name)**

**How log4j is configured?**

To configure log4j we have to decide which appender to implement. Accordingly, parameters of appender will be set.

* We will use DEBUG level and RollingFileAppender
* We will do two configurations or logs,
  + First: root logger, that will write all system generated logs in file name i.e. Selenium.logs
  + Second: Will write the information generated by manual commands in code into the file name- Manual.logs
* Layout will be PatternLayout

#Root logger

log4j.rootLogger=DEBUG,file

log4j.appender.file=org.apache.log4j.RollingFileAppender

log4j.appender.file.File=D:\\eclipse\\ProjWork\\SeleniumTestNG\\src\\Selenium.logs

log4j.appender.file.maxFileSize=900KB

log4j.appender.file.maxBackupIndex=5

log4j.appender.file.layout=org.apache.log4j.PatternLayout

log4j.appender.file.layout.ConversionPattern=%d{ABSOLUTE} %5p %c<strong>**{1}**</strong>:%L - %m%n

log4j.appender.file.Append=false

#Application Logs

log4j.logger.FirstTestNGFile=DEBUG, dest1

log4j.appender.dest1=org.apache.log4j.RollingFileAppender

log4j.appender.dest1.maxFileSize=900KB

log4j.appender.dest1.maxBackupIndex=6

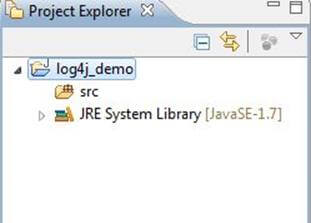
log4j.appender.dest1.layout=org.apache.log4j.PatternLayout

log4j.appender.dest1.layout.ConversionPattern=%d{dd/MM/yyyy HH:mm:ss} %c %m%n

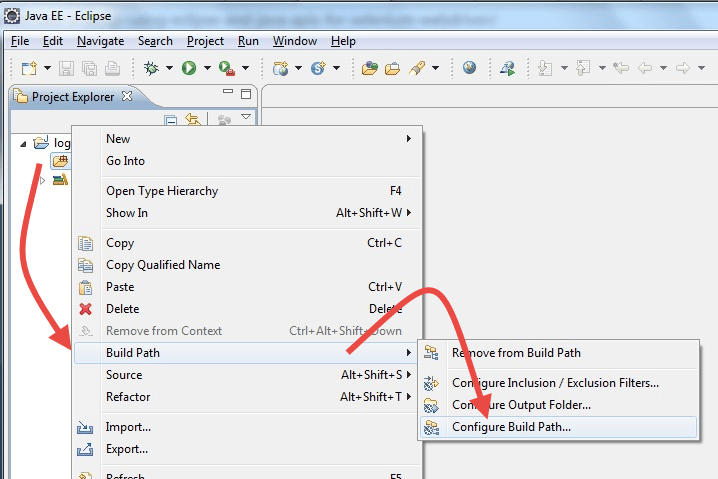
log4j.appender.dest1.File=D:\\eclipse\\ProjWork\\SeleniumTestNG\\src\\Manual.logs

log4j.appender.dest1.Append=false

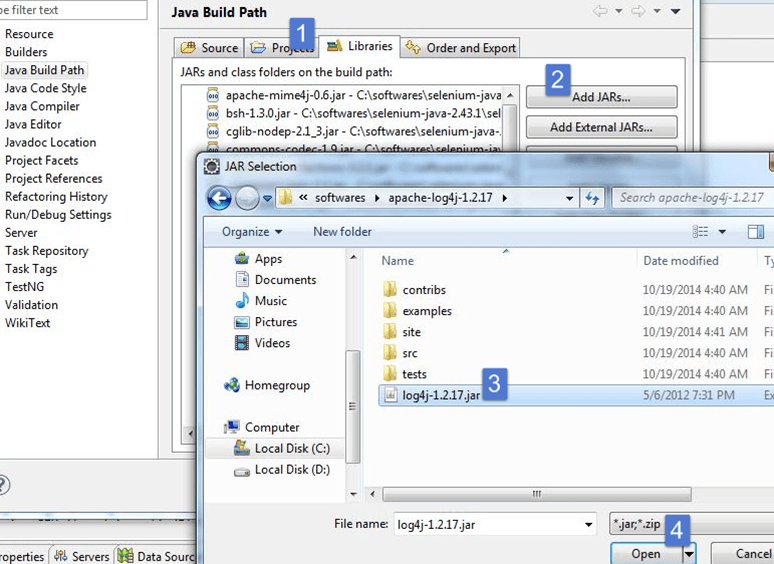
**Step 1)** In Eclipse create a new project with name log4j\_demo

[](http://cdn.guru99.com/images/5-2015/050115_0810_TutorialonL2.jpg)

**Step 2)**Right click on src -> Build Path -> Configure Build Path

[](http://cdn.guru99.com/images/5-2015/050115_0810_TutorialonL3.png)

**Step 2)**Click on Librariries and Add Log4J Lirabry. You can download it from <https://logging.apache.org/log4j/1.2/download.html>

[](http://cdn.guru99.com/images/5-2015/050115_0810_TutorialonL4.png)

**Step 3)**Create a new file. This file will include all the log4j configuration

1. Right click on src -> New -> Other -> General -> File
2. Give the file name as "log4j.properties"
3. Click on Finish

Create two more files and give them names such as Selenium.logs and Manual.logs. These files will contain all the logs created by system and manually logged statements

[](http://cdn.guru99.com/images/5-2015/050115_0810_TutorialonL5.jpg)

**Step 4)**In log4j.properties copy the entire configuration.

#Root logger

log4j.rootLogger=DEBUG,file

log4j.appender.file=org.apache.log4j.RollingFileAppender

log4j.appender.file.File=D:\\eclipse\\ProjWork\\SeleniumTestNG\\src\\Selenium.logs

log4j.appender.file.maxFileSize=900KB

log4j.appender.file.maxBackupIndex=5

log4j.appender.file.layout=org.apache.log4j.PatternLayout

log4j.appender.file.layout.ConversionPattern=%d{ABSOLUTE} %5p %c<strong>**{1}**</strong>:%L - %m%n

log4j.appender.file.Append=false

#Application Logs

log4j.logger.FirstTestNGFile=DEBUG, dest1

log4j.appender.dest1=org.apache.log4j.RollingFileAppender

log4j.appender.dest1.maxFileSize=900KB

log4j.appender.dest1.maxBackupIndex=6

log4j.appender.dest1.layout=org.apache.log4j.PatternLayout

log4j.appender.dest1.layout.ConversionPattern=%d{dd/MM/yyyy HH:mm:ss} %c %m%n

log4j.appender.dest1.File=D:\\eclipse\\ProjWork\\SeleniumTestNG\\src\\Manual.logs

log4j.appender.dest1.Append=false

**Step 5)**Copy the following code in to the main class

**package** FirstPackage;

**import** java.util.concurrent.TimeUnit;

**import** junit.framework.~~Assert~~;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

**import** org.openqa.selenium.firefox.FirefoxDriver;

**import** org.openqa.selenium.interactions.Actions;

**import** org.openqa.selenium.support.ui.ExpectedConditions;

**import** org.openqa.selenium.support.ui.WebDriverWait;

**import** org.testng.annotations.AfterClass;

**import** org.testng.annotations.AfterTest;

**import** org.testng.annotations.BeforeClass;

**import** org.testng.annotations.BeforeTest;

**import** org.testng.annotations.Test;

**import** org.testng.annotations.BeforeMethod;

**import** org.testng.annotations.AfterMethod;

**import** org.testng.asserts.Assertion;

**import** org.apache.log4j.Logger;

**public** **class** FirstTestNGFile {

**public** WebDriver driver;

Logger log = Logger.*getLogger*("FirstTestNGFile");

@Test(priority =3)

**public** **void** OpenSoftware() {

log.debug("@Test:- Test OpenSoftware");

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[2]/div[1]/a/img")).click();

log.debug("Go to Home");

driver.findElement(By.*linkText*("Computers")).click();

**new** Actions(driver).moveToElement(driver.findElement(By.*linkText*("Software"))).click().perform();

log.debug("Click on Computers -> Software");

driver.findElement(By.*xpath*("html/body/div[5]/div[3]/div[2]/div[2]/div/div[2]/div[3]/div/div[3]/div/div[1]/a/img")).click();

log.debug("Open Windows 8 Pro Software item");

}

@Test(priority =2)

**public** **void** CellPhones() {

log.debug("@Test:- Test cell-phones");

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[2]/div[1]/a/img")).click();

log.debug("Go to Home");

driver.findElement(By.*linkText*("Electronics")).click();

**new** Actions(driver).moveToElement(driver.findElement(By.*linkText*("Cell phones"))).click().perform();

log.debug("Click on Electronics -> Cell phones");

driver.findElement(By.*xpath*("html/body/div[5]/div[3]/div[2]/div[2]/div/div[2]/div[3]/div/div[3]/div/div[1]/a/img")).click();

log.debug("Open Mobile Nokia Lumia 1020");

}

@Test (priority =1)

**public** **void** adidasShoes() {

log.debug("@Test:- Test adidasShoes");

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[2]/div[1]/a/img")).click();

log.debug("Go to Home");

driver.findElement(By.*linkText*("Apparel")).click();

**new** Actions(driver).moveToElement(driver.findElement(By.*linkText*("Shoes"))).click().perform();

log.debug("Click on Apparel -> Shoes");

driver.findElement(By.*xpath*("html/body/div[5]/div[3]/div[2]/div[2]/div/div[2]/div[3]/div/div[1]/div/div[1]/a/img")).click();

log.debug("Open adidas Consortium Campus 80s Running Shoes");

}

@BeforeClass

**public** **void** beforeClass() {

log.debug("@BeforeClass");

driver = **new** FirefoxDriver();

log.debug("new FirefoxDriver()");

//Put a Implicit wait, this means that any search for elements on the page could take the time the implicit wait is set for before throwing exception

driver.manage().timeouts().implicitlyWait(10, TimeUnit.***SECONDS***);

//Launch the Online Store Website

driver.get("http://demo.nopcommerce.com/");

log.debug("Open 'http://demo.nopcommerce.com/' ");

}

@AfterClass

**public** **void** afterClass() {

log.debug(" @AfterClass");

//Close the driver

driver.quit();

log.debug("Close Firefox");

}

@BeforeMethod

**public** **void** beforeMethod() {

// Create a new instance of the Firefox driver

log.debug("@BeforeTest");

driver.manage().timeouts().implicitlyWait(15, TimeUnit.***SECONDS***);

driver.navigate().to("http://demo.nopcommerce.com/");

/\*WebDriverWait wait = new WebDriverWait(driver, 10);

WebElement element = wait.until(ExpectedConditions.elementToBeClickable(By.id("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a")));

element.click();

\*/

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a")).click();

// Find the element that's ID attribute is 'account'(My Account)

log.debug("driver.findElement(By.xpath('html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a')).click();");

// Enter Username on the element found by above desc.

driver.findElement(By.*xpath*(".//\*[@id='Email']")).sendKeys("test123@gmail.com");

log.debug("driver.findElement(By.xpath('.//\*[@id='Email']')).sendKeys('test123@gmail.com');");

// Find the element that's ID attribute is 'pwd' (Password)

// Enter Password on the element found by the above desc.

driver.findElement(By.*xpath*(".//\*[@id='Password']")).sendKeys("test@123");

log.debug("driver.findElement(By.xpath('.//\*[@id='Password']')).sendKeys('test@123');");

// Now submit the form. WebDriver will find the form for us from the element

driver.findElement(By.*xpath*("html/body/div[5]/div[3]/div/div/div/div[2]/div[1]/div[2]/form/div[3]/input")).click();

log.debug("Click on Login Button");

log.debug("Check for login status");

**boolean** present = driver.findElements(By.*xpath*("html/body/div[5]/div[3]/div/div/div/div[2]/div[1]/div[2]/form/div[2]/div/ul/li")).size() > 0;;

**if**(!present)

{

// Print a Log In message to the screen

System.***out***.println(" Login Successfully, now it is the time to Log Off buddy.");

log.debug("Login Successfully, User is Avilable");

// Find the element that's ID attribute is 'account\_logout' (Log Out)

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[1]/a"));

log.debug("driver.findElement(By.xpath('html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a'));");

}

**else**

{

log.debug("Login status: Failed, Creating New User");

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[1]/a")).click();

log.debug("Click On register");

driver.findElement(By.*xpath*(".//\*[@id='FirstName']")).sendKeys("test");

log.debug("Enter First Name");

driver.findElement(By.*xpath*(".//\*[@id='LastName']")).sendKeys("test");

log.debug("Enter Last Name");

driver.findElement(By.*xpath*(".//\*[@id='Email']")).sendKeys("test123@gmail.com");

log.debug("Enter Email");

driver.findElement(By.*xpath*(".//\*[@id='Password']")).sendKeys("test@123");

log.debug("Enter Password");

driver.findElement(By.*xpath*(".//\*[@id='ConfirmPassword']")).sendKeys("test@123");

log.debug("Confirm Password");

driver.findElement(By.*xpath*(".//\*[@id='register-button']")).click();

log.debug("Click On register Button Password");

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[1]/a")).click();

log.debug("Go to user profile");

**boolean** is\_exist = driver.findElements(By.*xpath*("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[1]/a")).size() > 0;

~~Assert~~.*assertEquals*(**true**,is\_exist);

}

}

@AfterMethod

**public** **void** afterMethod() {

log.debug("@AfterTest");

//Logout User

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[1]/a")).click();

log.debug("Go to user profile");

driver.findElement(By.*xpath*("html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a")).click();

log.debug("Click on LogOut");

}

}

**Step 6)**It will generate log file of execution as below (manual.log)

26/10/2016 17:16:18 FirstTestNGFile @BeforeClass

26/10/2016 17:17:04 FirstTestNGFile new FirefoxDriver()

26/10/2016 17:17:21 FirstTestNGFile Open 'http://demo.nopcommerce.com/'

26/10/2016 17:17:22 FirstTestNGFile @BeforeTest

26/10/2016 17:17:25 FirstTestNGFile driver.findElement(By.xpath('html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a')).click();

26/10/2016 17:17:29 FirstTestNGFile driver.findElement(By.xpath('.//\*[@id='Email']')).sendKeys('test123@gmail.com');

26/10/2016 17:17:30 FirstTestNGFile driver.findElement(By.xpath('.//\*[@id='Password']')).sendKeys('test@123');

26/10/2016 17:17:34 FirstTestNGFile Click on Login Button

26/10/2016 17:17:34 FirstTestNGFile Check for login status

26/10/2016 17:17:49 FirstTestNGFile Login Successfully, User is Avilable

26/10/2016 17:17:49 FirstTestNGFile driver.findElement(By.xpath('html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a'));

26/10/2016 17:17:49 FirstTestNGFile @Test:- Test adidasShoes

26/10/2016 17:17:51 FirstTestNGFile Go to Home

26/10/2016 17:17:55 FirstTestNGFile Click on Apparel -> Shoes

26/10/2016 17:17:58 FirstTestNGFile Open adidas Consortium Campus 80s Running Shoes

26/10/2016 17:17:58 FirstTestNGFile @AfterTest

26/10/2016 17:18:04 FirstTestNGFile Go to user profile

26/10/2016 17:18:06 FirstTestNGFile Click on LogOut

26/10/2016 17:18:06 FirstTestNGFile @BeforeTest

26/10/2016 17:18:10 FirstTestNGFile driver.findElement(By.xpath('html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a')).click();

26/10/2016 17:18:11 FirstTestNGFile driver.findElement(By.xpath('.//\*[@id='Email']')).sendKeys('test123@gmail.com');

26/10/2016 17:18:11 FirstTestNGFile driver.findElement(By.xpath('.//\*[@id='Password']')).sendKeys('test@123');

26/10/2016 17:18:14 FirstTestNGFile Click on Login Button

26/10/2016 17:18:14 FirstTestNGFile Check for login status

26/10/2016 17:18:29 FirstTestNGFile Login Successfully, User is Avilable

26/10/2016 17:18:30 FirstTestNGFile driver.findElement(By.xpath('html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a'));

26/10/2016 17:18:30 FirstTestNGFile @Test:- Test cell-phones

26/10/2016 17:18:31 FirstTestNGFile Go to Home

26/10/2016 17:18:36 FirstTestNGFile Click on Electronics -> Cell phones

26/10/2016 17:18:39 FirstTestNGFile Open Mobile Nokia Lumia 1020

26/10/2016 17:18:39 FirstTestNGFile @AfterTest

26/10/2016 17:18:41 FirstTestNGFile Go to user profile

26/10/2016 17:18:43 FirstTestNGFile Click on LogOut

26/10/2016 17:18:43 FirstTestNGFile @BeforeTest

26/10/2016 17:18:46 FirstTestNGFile driver.findElement(By.xpath('html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a')).click();

26/10/2016 17:18:48 FirstTestNGFile driver.findElement(By.xpath('.//\*[@id='Email']')).sendKeys('test123@gmail.com');

26/10/2016 17:18:48 FirstTestNGFile driver.findElement(By.xpath('.//\*[@id='Password']')).sendKeys('test@123');

26/10/2016 17:18:51 FirstTestNGFile Click on Login Button

26/10/2016 17:18:51 FirstTestNGFile Check for login status

26/10/2016 17:19:07 FirstTestNGFile Login Successfully, User is Avilable

26/10/2016 17:19:07 FirstTestNGFile driver.findElement(By.xpath('html/body/div[5]/div[1]/div[1]/div[2]/div[1]/ul/li[2]/a'));

26/10/2016 17:19:07 FirstTestNGFile @Test:- Test OpenSoftware

26/10/2016 17:19:08 FirstTestNGFile Go to Home

26/10/2016 17:19:11 FirstTestNGFile Click on Computers -> Software

26/10/2016 17:19:14 FirstTestNGFile Open Windows 8 Pro Software item

26/10/2016 17:19:14 FirstTestNGFile @AfterTest

26/10/2016 17:19:17 FirstTestNGFile Go to user profile

26/10/2016 17:19:20 FirstTestNGFile Click on LogOut

26/10/2016 17:19:20 FirstTestNGFile @AfterClass

26/10/2016 17:19:22 FirstTestNGFile Close Firefox