

Selenium Training

TestNG (Testing Next Generation)

Overview

01	Introduction to TestNG
02	Benefits of TestNG
03	Annotations in TestNG
04	Installation of TestNG Using Eclipse

Cont....

05	Create TestNG Program and Execute
06	TestNG Data Providers
07	TestNG Test Suite
08	Check reports generated by TestNG

Introduction to TestNG



- **TestNG** is a testing framework inspired from JUnit but introducing some new functionality that make it more powerful and easier to use.
- It is an open source automated testing framework; where NG of TestNG means Next Generation. TestNG is similar to JUnit but it is much more powerful than JUnit but still it's inspired by JUnit. It is designed to be better than JUnit, especially when testing integrated classes.

Benefits of TestNG



- It gives the ability to produce HTML Reports of execution.
- Annotations made testers life easy.
- Test cases can be Prioritized more easily.
- Data Parameterization is possible.
- Assertion.



Annotations in TestNG

@BeforeSuite: The annotated method will be run before all tests in this suite have run.

@AfterSuite: The annotated method will be run after all tests in this suite have run.

@BeforeTest: The annotated method will be run before any test method belonging to the classes inside the tag is run.

@AfterTest: The annotated method will be run after all the test methods belonging to the classes inside the tag have run.

@BeforeMethod: The annotated method will be run before each test method.

@AfterMethod: The annotated method will be run after each test method.

@Test: The annotated method is a part of a test case.

Example



```
package Download_Code;

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

public class Annotation_Example
{
    @BeforeSuite
    public void beforeSuite()
    {
        System.out.println("This will execute before the Test Suite");
    }

    @BeforeMethod
    public void beforeMethod()
    {
        System.out.println("This will execute before every Method");
    }

    @Test
    public void TC1() {
        System.out.println("This is the Test Case 1");
    }
}
```

Cont...

```
@Test
public void TC2()
{
    System.out.println("This is the Test Case 2");
}

@BeforeTest
public void beforeTest()
{
    System.out.println("This will execute before the Test");
}

@AfterTest
public void afterTest()
{
    System.out.println("This will execute after the Test");
}
```


Cont...

```
@BeforeTest
public void beforeTest()
{
    System.out.println("This will execute before the Test");
}
```

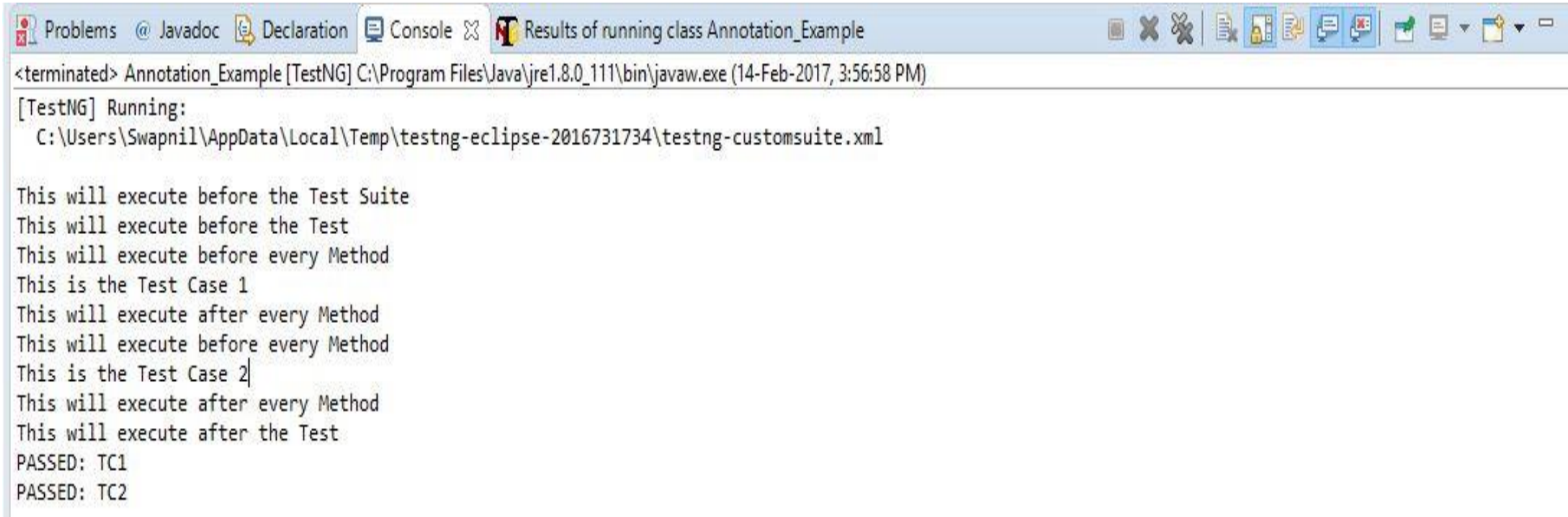
```
@AfterTest
public void afterTest()
{
    System.out.println("This will execute after the Test");
}
```

```
@AfterMethod
public void afterMethod()
{
    System.out.println("This will execute after every Method");
}
```

```
@AfterSuite
public void afterSuite() {
    System.out.println("This will execute after the Test Suite");
}
```

```
}
```

Output



The screenshot shows the Eclipse IDE's console window with the title 'Results of running class Annotation_Example'. The console output displays the TestNG execution process, including the running of a test suite, execution of before and after methods for the test suite and individual test cases, and the final status of two test cases (TC1 and TC2) as 'PASSED'.

```
<terminated> Annotation_Example [TestNG] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (14-Feb-2017, 3:56:58 PM)
[TestNG] Running:
  C:\Users\Swapnil\AppData\Local\Temp\testng-eclipse-2016731734\testng-customsuite.xml

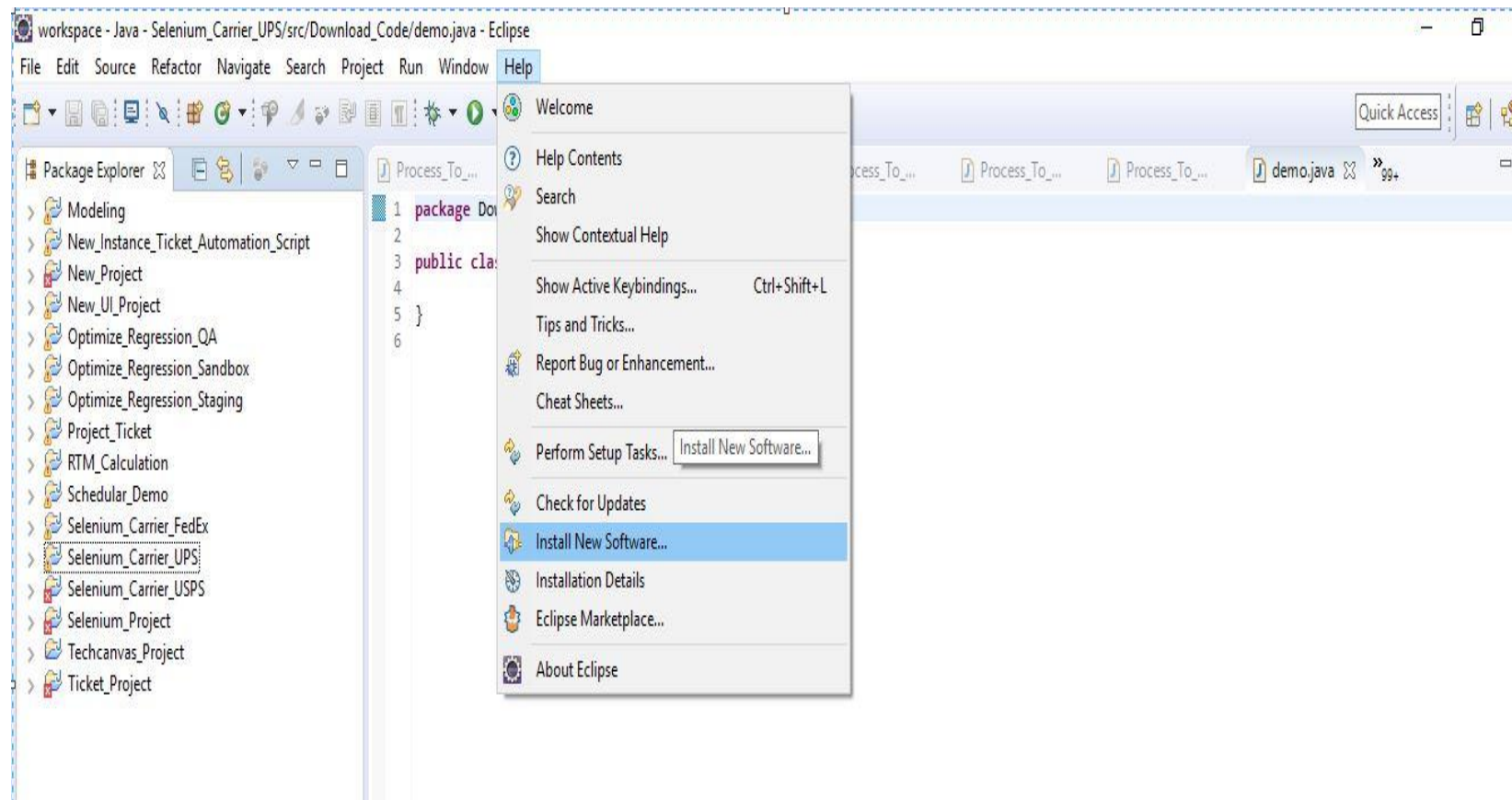
This will execute before the Test Suite
This will execute before the Test
This will execute before every Method
This is the Test Case 1
This will execute after every Method
This will execute before every Method
This is the Test Case 2
This will execute after every Method
This will execute after the Test
PASSED: TC1
PASSED: TC2
```

Installation of TestNG Using Eclipse



Steps to follow :

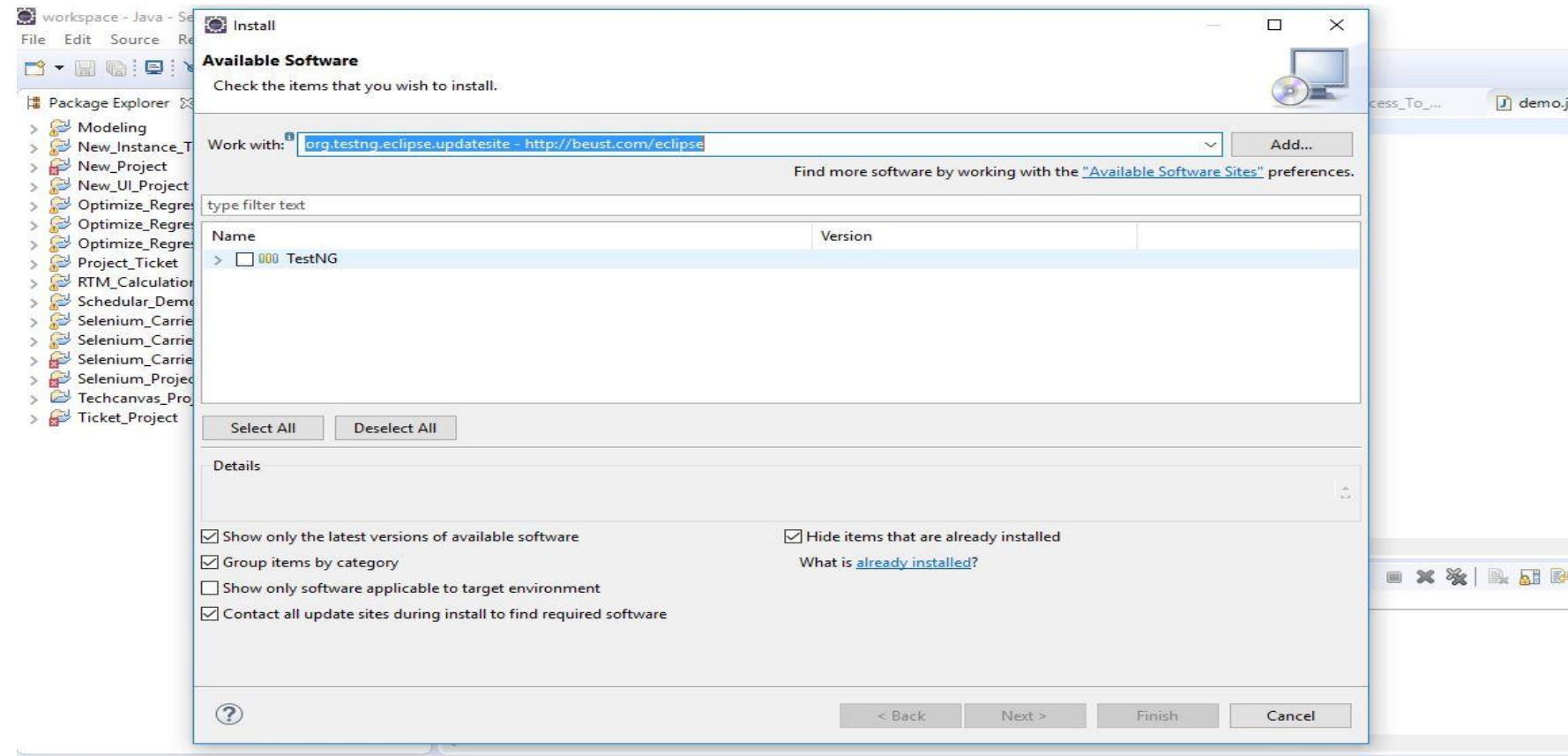
Step 1 : Open the Eclipse IDE and from Help menu, click “**Install New Software**”.



Cont.....



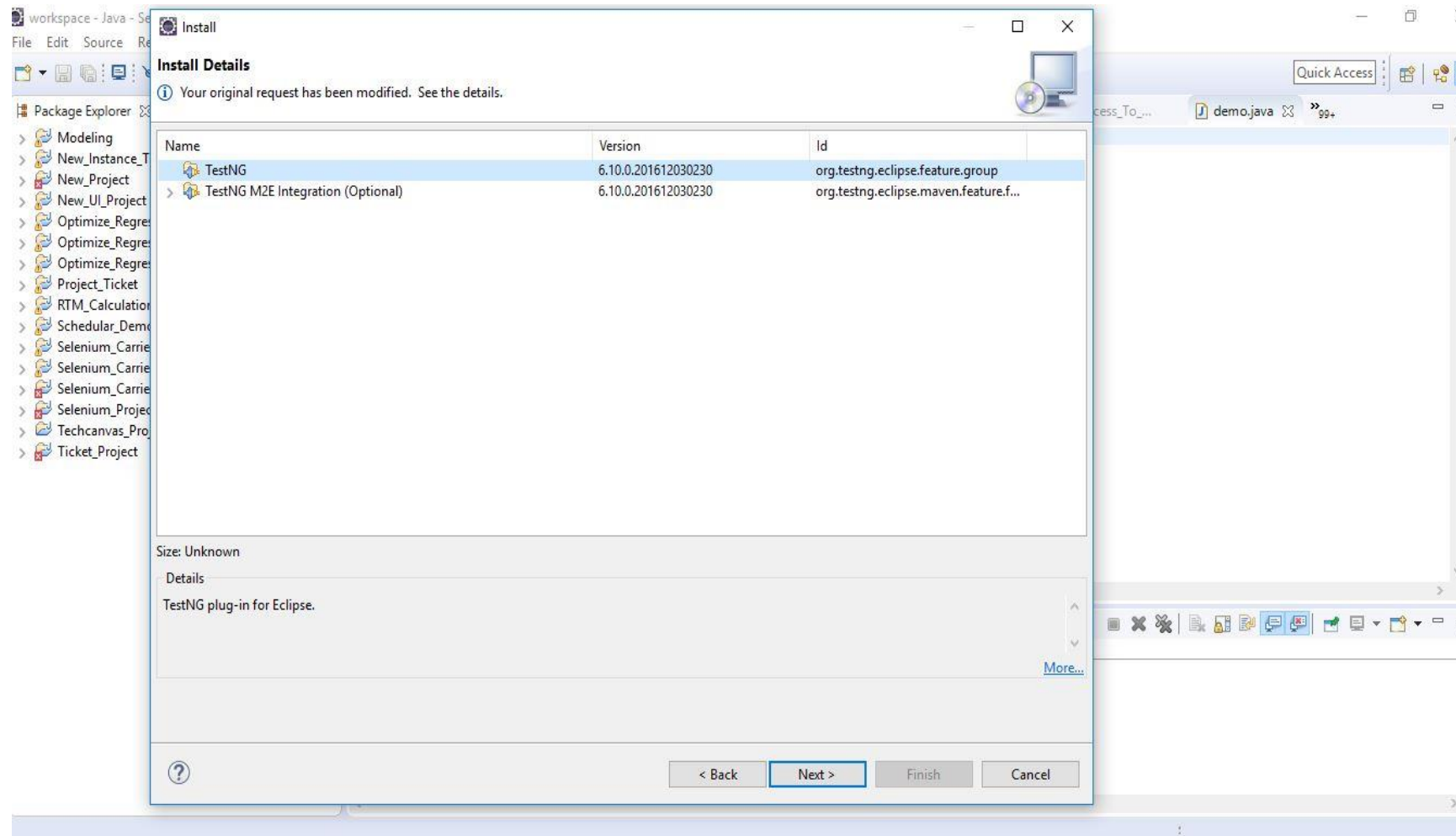
Step 2: Install window will get open, type <http://beust.com/eclipse/> in work with field and press Enter.



Cont...



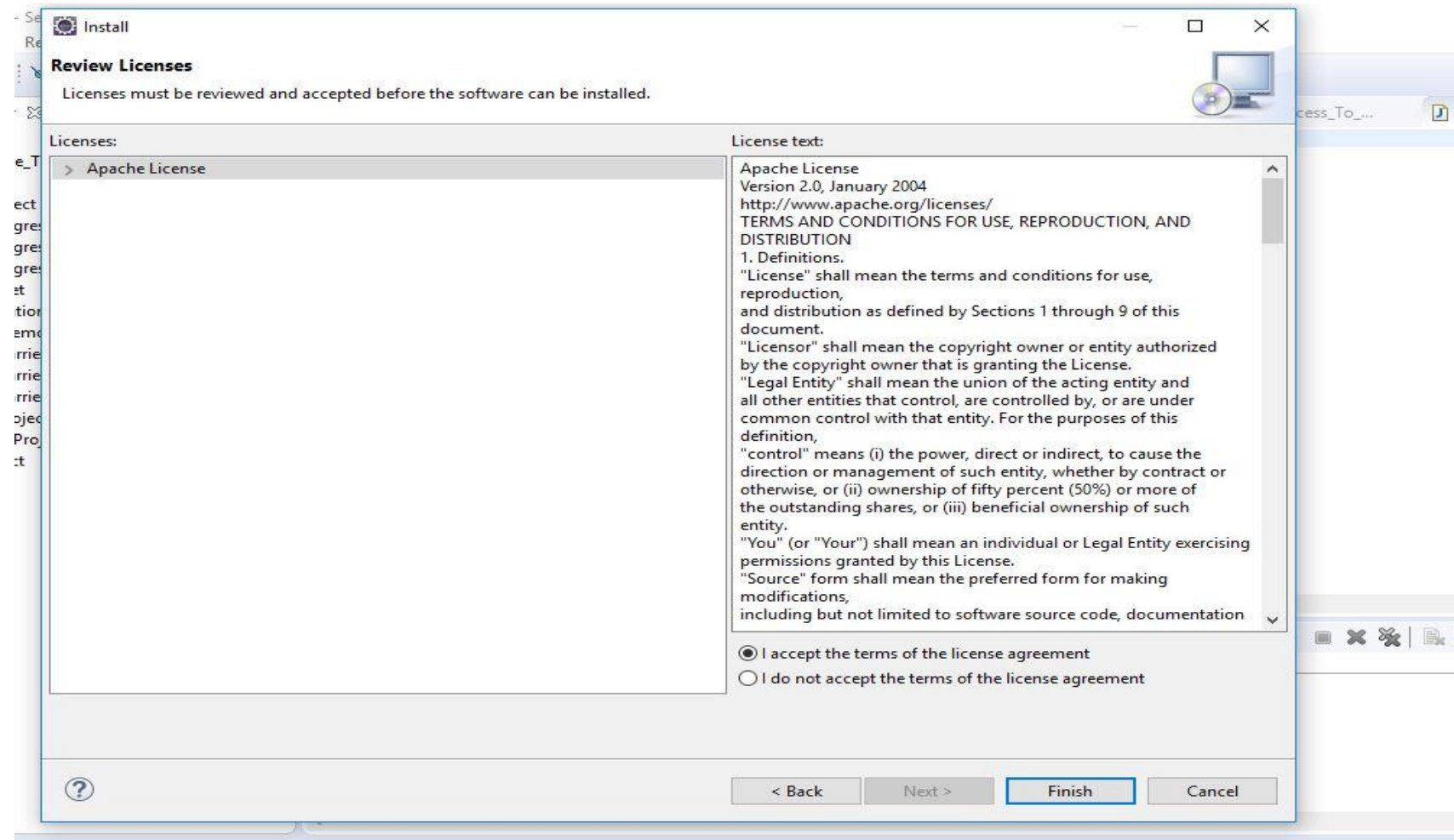
Step 3: Select “TestNG” as from Install Details window and click Next button



Cont...



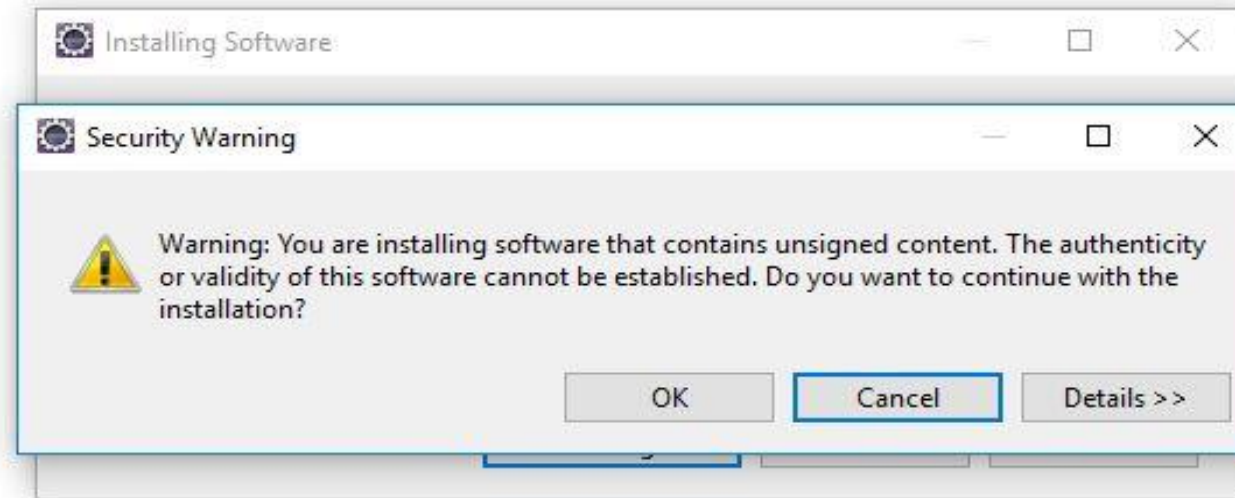
Step 4: Select “TestNG” checkbox and Click on “Next” button. Agreement Page will get display “Accept” it. Click “Finish”.



Cont...



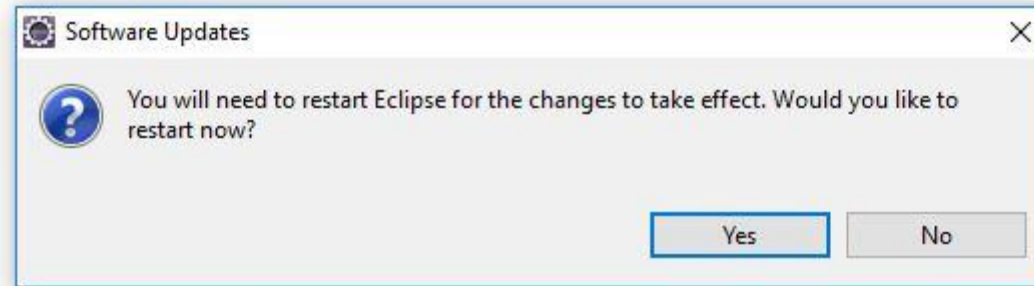
Step 5: “Window Security ” warning will get display. Click “OK”.



Cont...



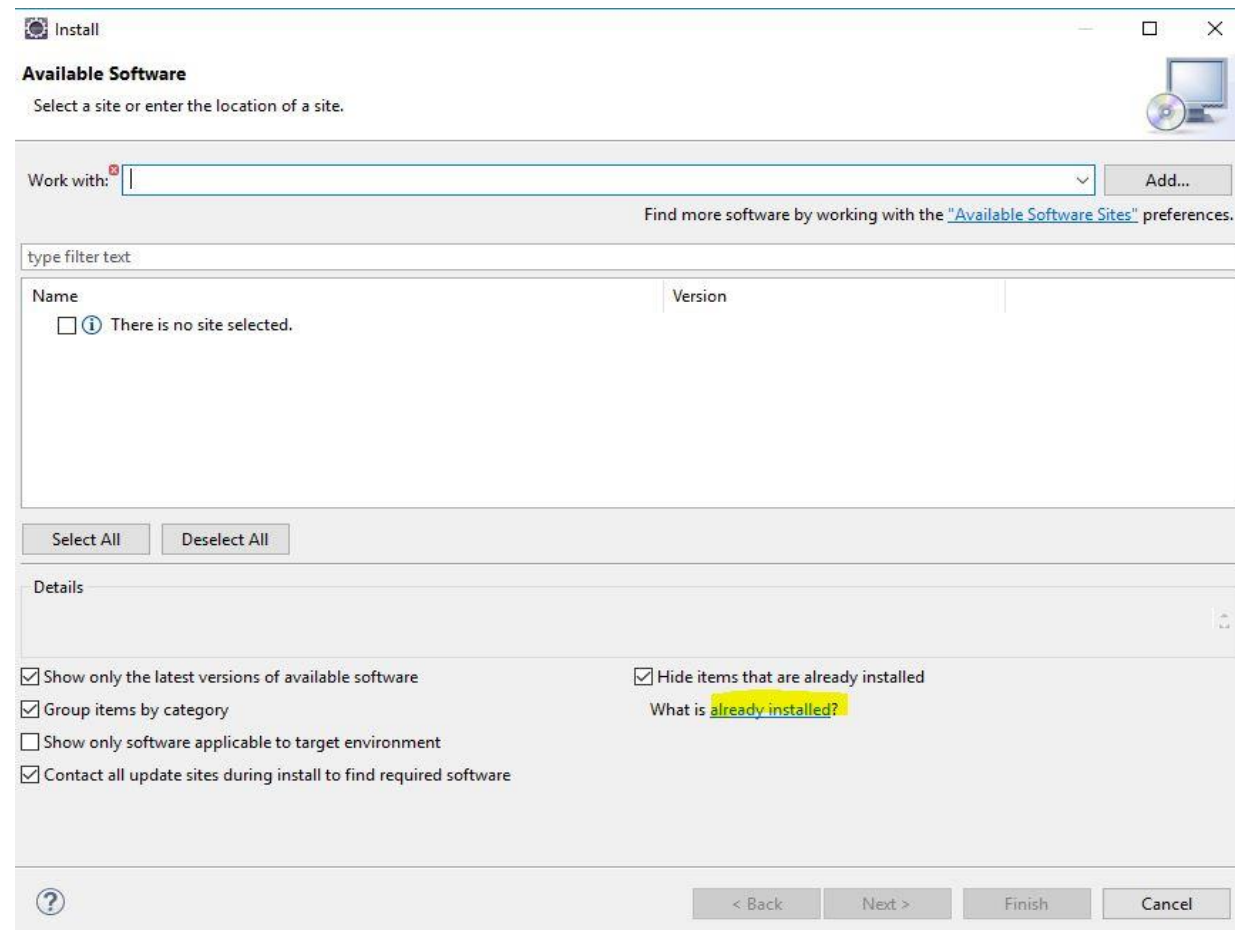
Step 6: After Installation get completed, it will ask to restart eclipse. Click “Yes”



Cont...



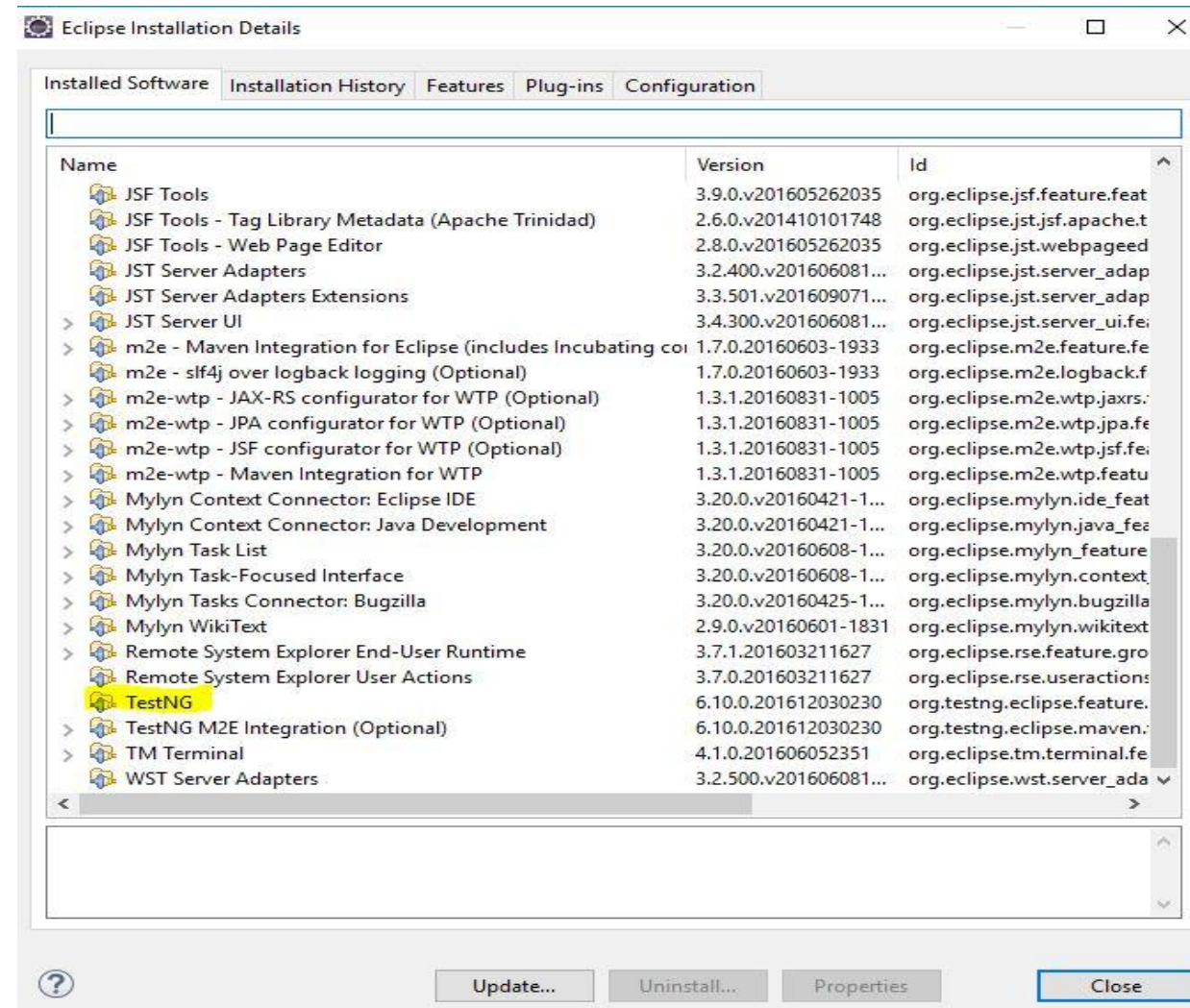
Step 7: After restart, to verify TestNG is successfully installed.
Click on Help -> click **“Install New Software”** and click on **“already installed”**? Link.



Cont....



Step 8: On “Eclipse Installation Details” window it will show “TestNG”.

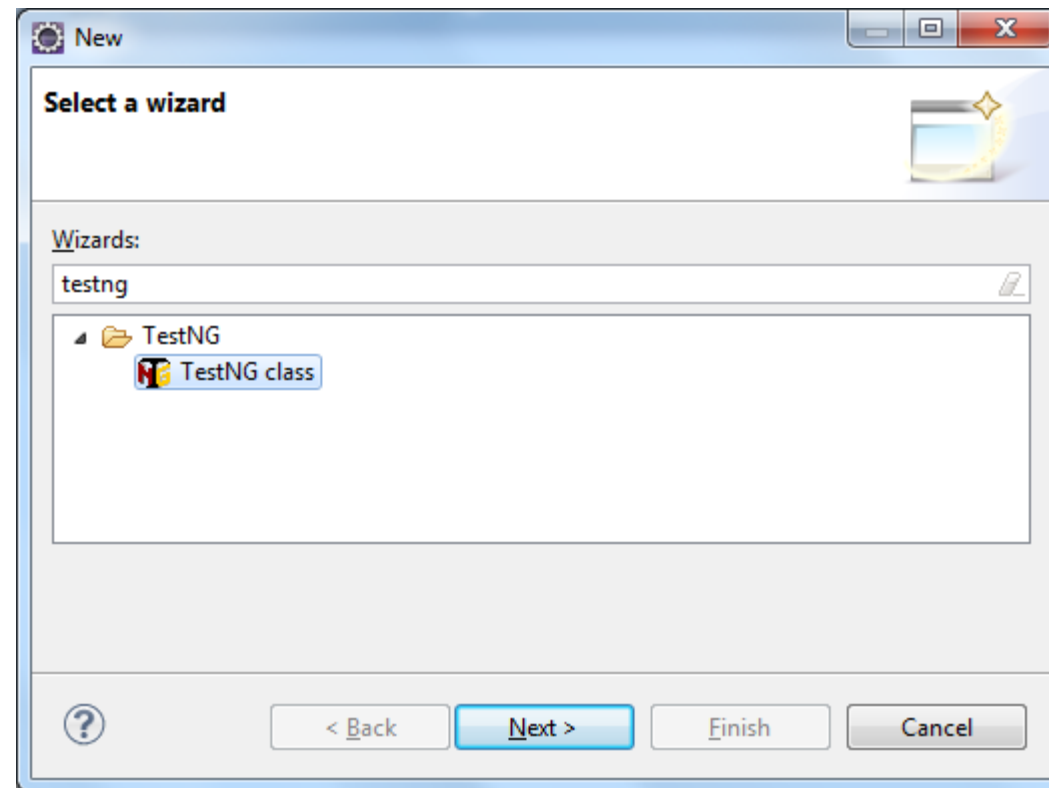


Create TestNG Program and Execute



Steps to create TestNG class:

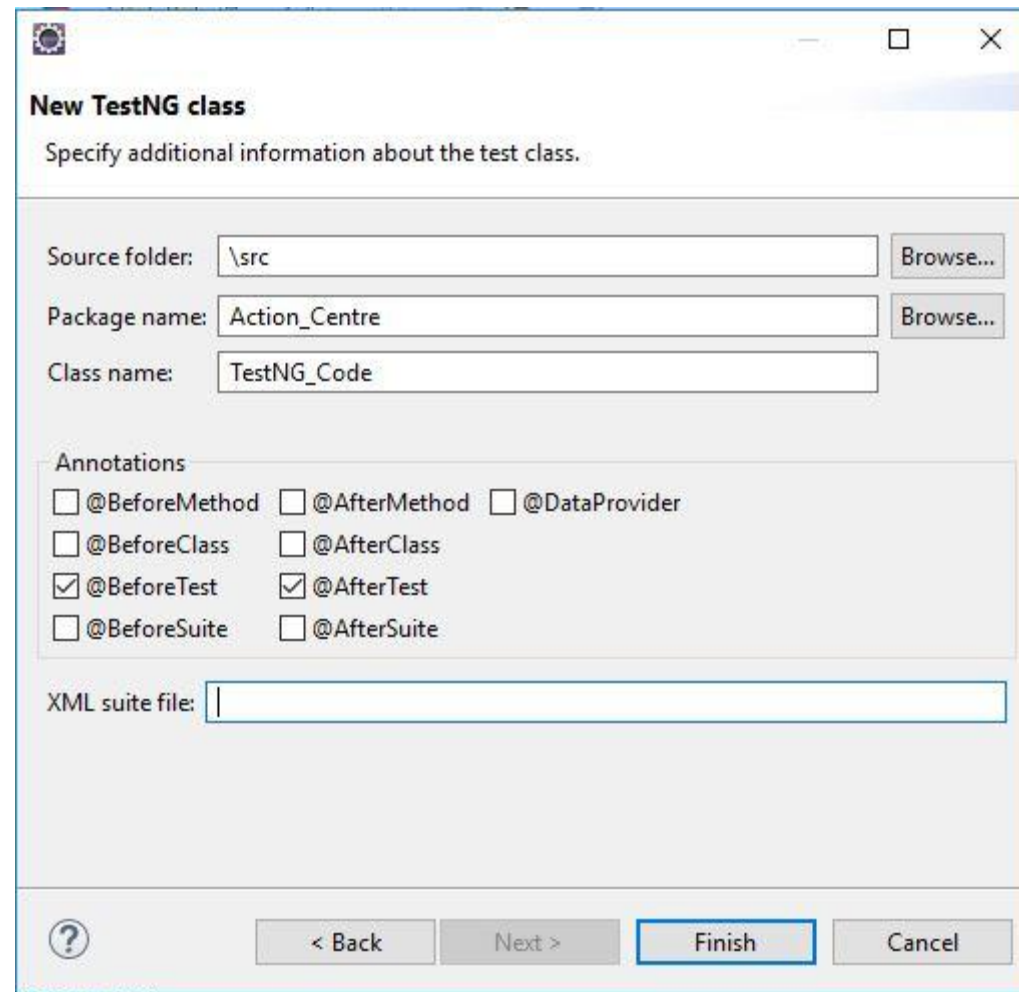
Step 1: Right Click on Package => New => Other. This will open a new Add wizard window in Eclipse.



Cont...



Step 2: Enter the Class name , select annotations and click on Finish.



New TestNG class
Specify additional information about the test class.

Source folder: \src Browse...

Package name: Action_Centre Browse...

Class name: TestNG_Code

Annotations

<input type="checkbox"/> @BeforeMethod	<input type="checkbox"/> @AfterMethod	<input type="checkbox"/> @DataProvider
<input type="checkbox"/> @BeforeClass	<input type="checkbox"/> @AfterClass	
<input checked="" type="checkbox"/> @BeforeTest	<input checked="" type="checkbox"/> @AfterTest	
<input type="checkbox"/> @BeforeSuite	<input type="checkbox"/> @AfterSuite	

XML suite file:

? < Back Next > **Finish** Cancel

Cont..



On Eclipse Code will get display in this manner.

```
1 package Action_Centre;
2
3 import org.testng.annotations.Test;
4
5
6
7 public class TestNG_Code {
8     @Test
9     public void f() {
10    }
11    @BeforeTest
12    public void beforeTest() {
13    }
14
15    @AfterTest
16    public void afterTest() {
17    }
18
19 }
20
```

Cont...



Modify the code.

```
1 package Action_Centre;
2
3+ import org.testng.annotations.Test;
4
5
6
7 public class TestNG_Code
8 {
9     @Test
10    public void f()
11    {
12        System.out.println("TestNG Code...");
13    }
14    @BeforeTest
15    public void beforeTest()
16    {
17        System.out.println("Pre-Condition...");
18    }
19
20    @AfterTest
21    public void afterTest()
22    {
23        System.out.println("Post-Condition...");
24    }
25
26 }
27
```

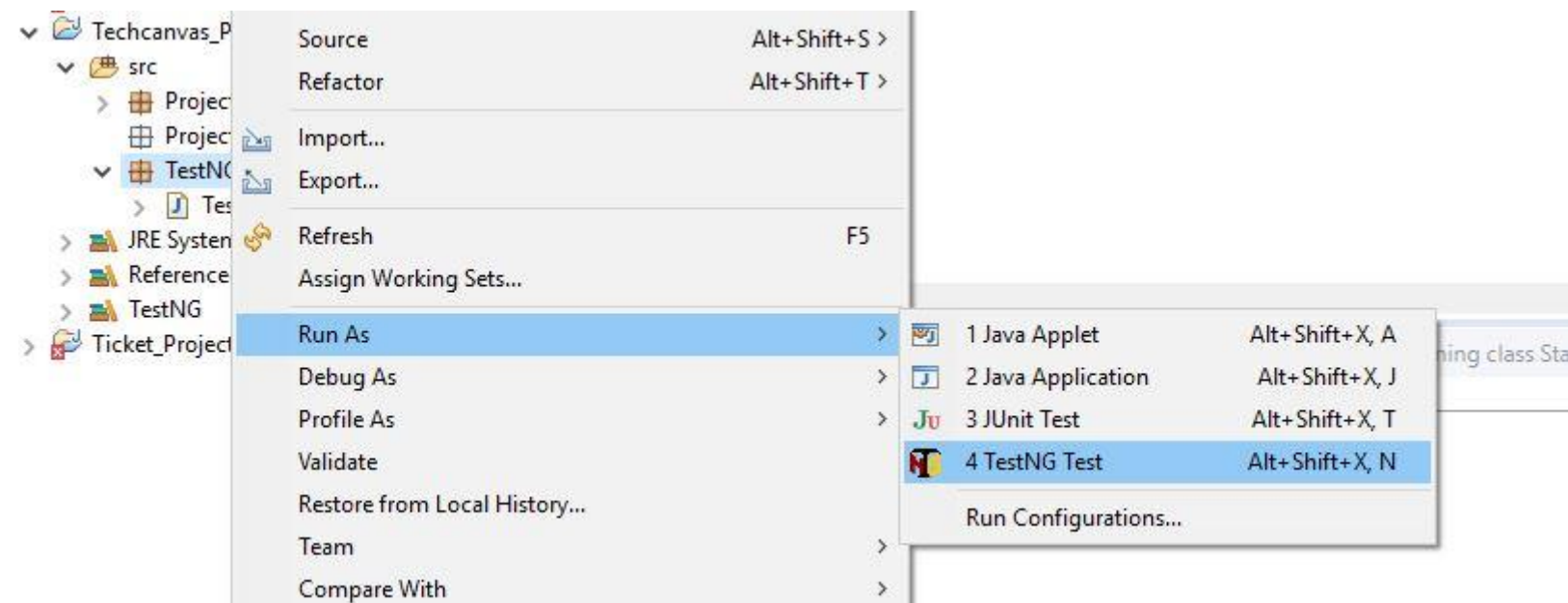
Cont....



Execute the code.

Steps to run tests through Eclipse:

Right Click on class file => Run As => TestNG Test



Cont...



Code and Result window :

```
1 package Action_Centre;
2
3 import org.testng.annotations.Test;
4
5
6
7 public class TestNG_Code
8 {
9     @Test
10    public void f()
11    {
12        System.out.println("TestNG Code....");
13    }
14    @BeforeTest
15    public void beforeTest()
16    {
17        System.out.println("Pre-Condition...");
18    }
19
20    @AfterTest
21    public void afterTest()
22    {
23        System.out.println("Post-Condition...");
24    }
25
26 }
```

Problems @ Javadoc Declaration Console Results of running class TestNG_Code

Search: Passed: 1 Failed: 0 Skipped: 0 Tests: 1/1 Methods: 1 (856 ms)

All Tests Failed Tests Summary

- Default suite (1/0/0/0) (0.032 s)
 - Default test (0.032 s)
 - Action_Centre.TestNG_Code
 - f (0.032 s)

Failure Exception

TestNG Data Provider



- **DataProvider** feature helps you to write data-driven tests, which essentially means that same test method can be run multiple times with different data-sets.
- A Data Provider is a method on your class that returns an array of array of objects.
- This method is annotated with `@DataProvider`.
- To use the DataProvider feature in your tests you have to declare a method annotated by `@DataProvider` and then use the said method in the test method using the 'dataProvider' attribute in the Test annotation.

Create a class for DataProvider




Steps:

- Right Click on Package => New =>Other.
- Select TestNG class => Next
- Enter Class name => Select annotation “@BeforeTest”, "AfterTest” and “@DataProvider”.

TestNG Class Window



 **New TestNG class**
Specify additional information about the test class.

Source folder:


Package name:

Class name:

Annotations

<input type="checkbox"/> @BeforeMethod	<input type="checkbox"/> @AfterMethod	<input checked="" type="checkbox"/> @DataProvider
<input type="checkbox"/> @BeforeClass	<input type="checkbox"/> @AfterClass	
<input checked="" type="checkbox"/> @BeforeTest	<input checked="" type="checkbox"/> @AfterTest	
<input type="checkbox"/> @BeforeSuite	<input type="checkbox"/> @AfterSuite	

XML suite file:



Cont...



Example below has dp() method in data provider. We will send 3 rows and 1 columns i.e. we will pass three different data.

```
1 package TestNG;
2
3 import org.testng.annotations.Test;
4
5
6
7
8 public class Test_DataProvider {
9     @Test(dataProvider = "dp")
10    public void f(String s)
11    {
12        System.out.println("Data Selected is : "+s);
13    }
14
15    @DataProvider
16    public Object[][] dp() {
17        return new Object[][] {
18            new Object[] { "Data1" },
19            new Object[] { "Data2" },
20            new Object[] { "Data3" }
21        };
22    }
23    @BeforeTest
24    public void beforeTest()
25    {
26        System.out.println("Pre..condition\n");
27    }
28
29    @AfterTest
30    public void afterTest()
31    {
32        System.out.println("\nPost..condition");
33    }
34
35 }
36
```

Output:

```
Problems @ Javadoc Declaration Console Results of running class Test_DataProvider
<terminated> Test_DataProvider [TestNG] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe (08-Feb-2017, 2:38:42 PM)
Pre..condition

Data Selected is : Data1
Data Selected is : Data2
Data Selected is : Data3

Post..condition
PASSED: f("Data1")
PASSED: f("Data2")
PASSED: f("Data3")

=====
Default test
Tests run: 3, Failures: 0, Skips: 0
=====
```

Test Suite in TestNG

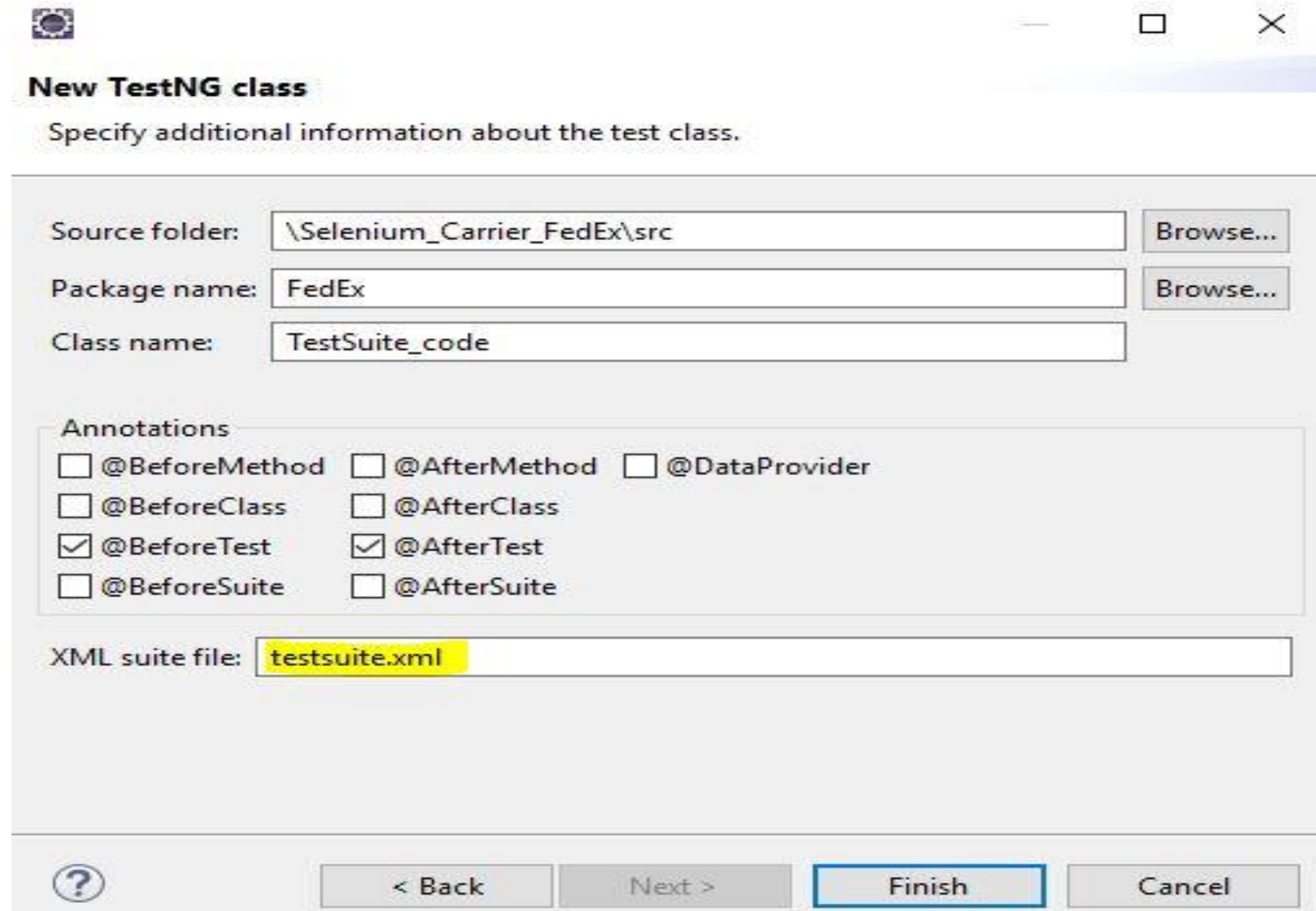
What is a Test Suite?

- In any project, you will have a bunch of test cases, So TestNG Provide a feature to execute suites.
- **Test Suite** in TestNG is a XML file that includes the set of test cases which need to be executed.
- TestNG provide a way to manage test execution. Those test cases may be dependent on each other or follow a certain execution order. You will create one **testng.xml** file to include all the test classes, packages, methods, parameters if required etc.

Step-By-Step working of Test Suite

- Right Click on Package => New =>Other.
- Select TestNG class => Next
- Enter Class name => Select annotation “@BeforeTest“ and "AfterTest
- Enter suite file name in “XML suite file” field.

Create XML Suite file



New TestNG class

Specify additional information about the test class.

Source folder:

Package name:

Class name:

Annotations

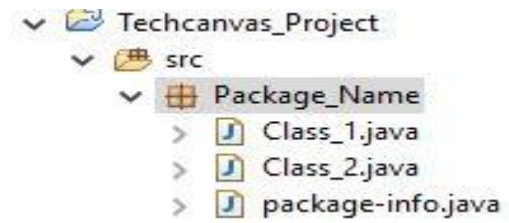
☐ @BeforeMethod ☐ @AfterMethod ☐ @DataProvider

☐ @BeforeClass ☐ @AfterClass

☒ @BeforeTest ☒ @AfterTest

☐ @BeforeSuite ☐ @AfterSuite

XML suite file:



```

1 package Package_Name;
2
3 import org.testng.annotations.Test;
4
5 public class Class_1 {
6     @Test
7     public void class_1()
8     {
9         System.out.println("Code for Class 1");
10    }
11 }
12
```

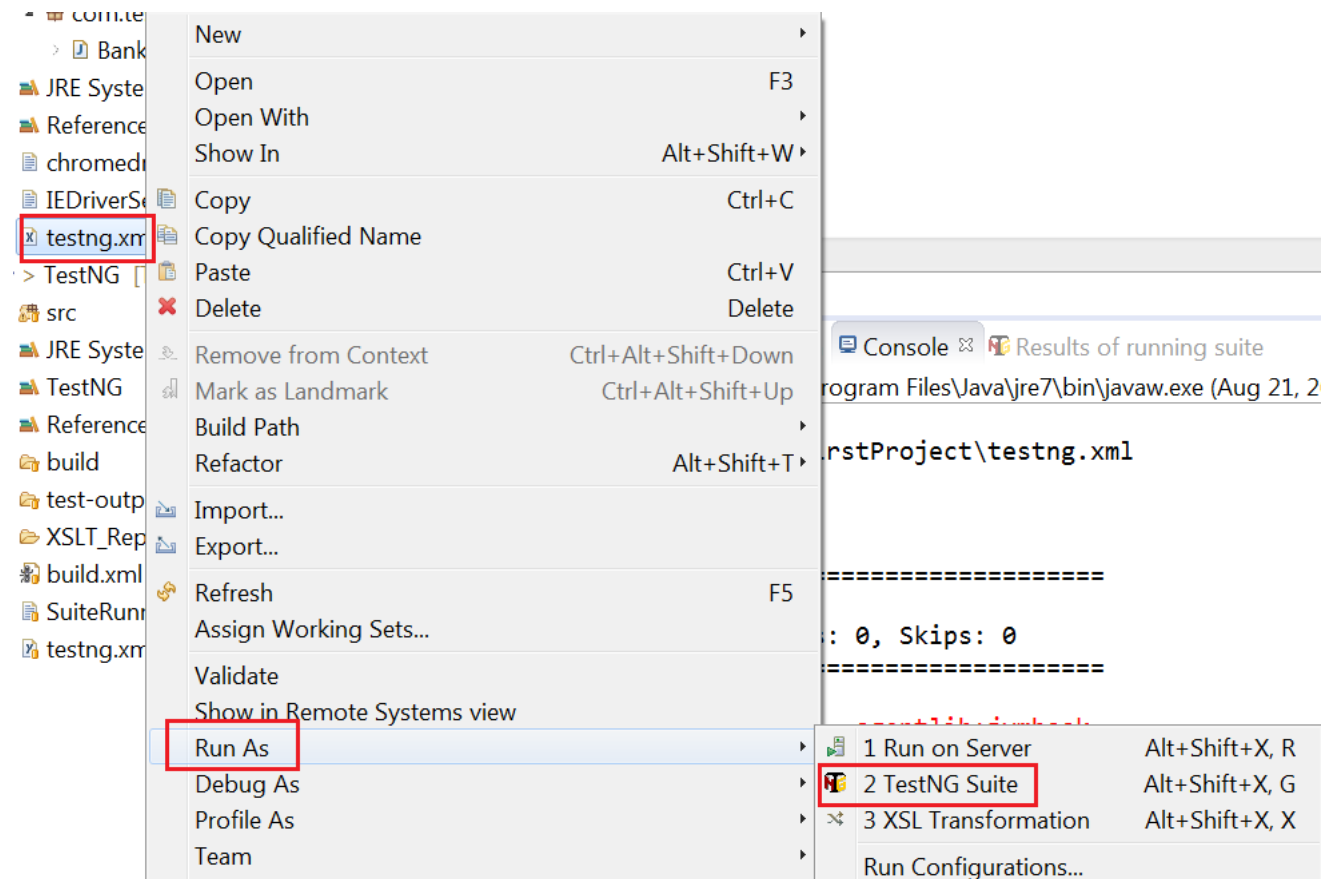
```

1 package Package_Name;
2
3 import org.testng.annotations.Test;
4
5 public class Class_2 {
6     @Test
7     public void class_2()
8     {
9         System.out.println("Code for Class 2");
10    }
11 }
12
```

Testng.xml file structure

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <suite name="TestSuite" parallel="false">
3
4      <test name="Test1">
5          <classes>
6              <class name="Package_Name.Class_1" />
7          </classes>
8      </test> <!-- Test -->
9
10
11
12      <test name="Test2">
13          <classes>
14              <class name="Package_Name.Class_2" />
15          </classes>
16      </test> <!-- Test -->
17
18
19
20
21 </suite> <!-- Suite -->
22
```

- To run the Test Suite , right click on **.xml** → Run As → TestNG Suite.



Output:

```
[TestNG] Running:  
C:\Users\Swapnil\workspace\Techcanvas_Project\src\t1.xml
```

```
Code for Class 1
```

```
Code for Class 2
```

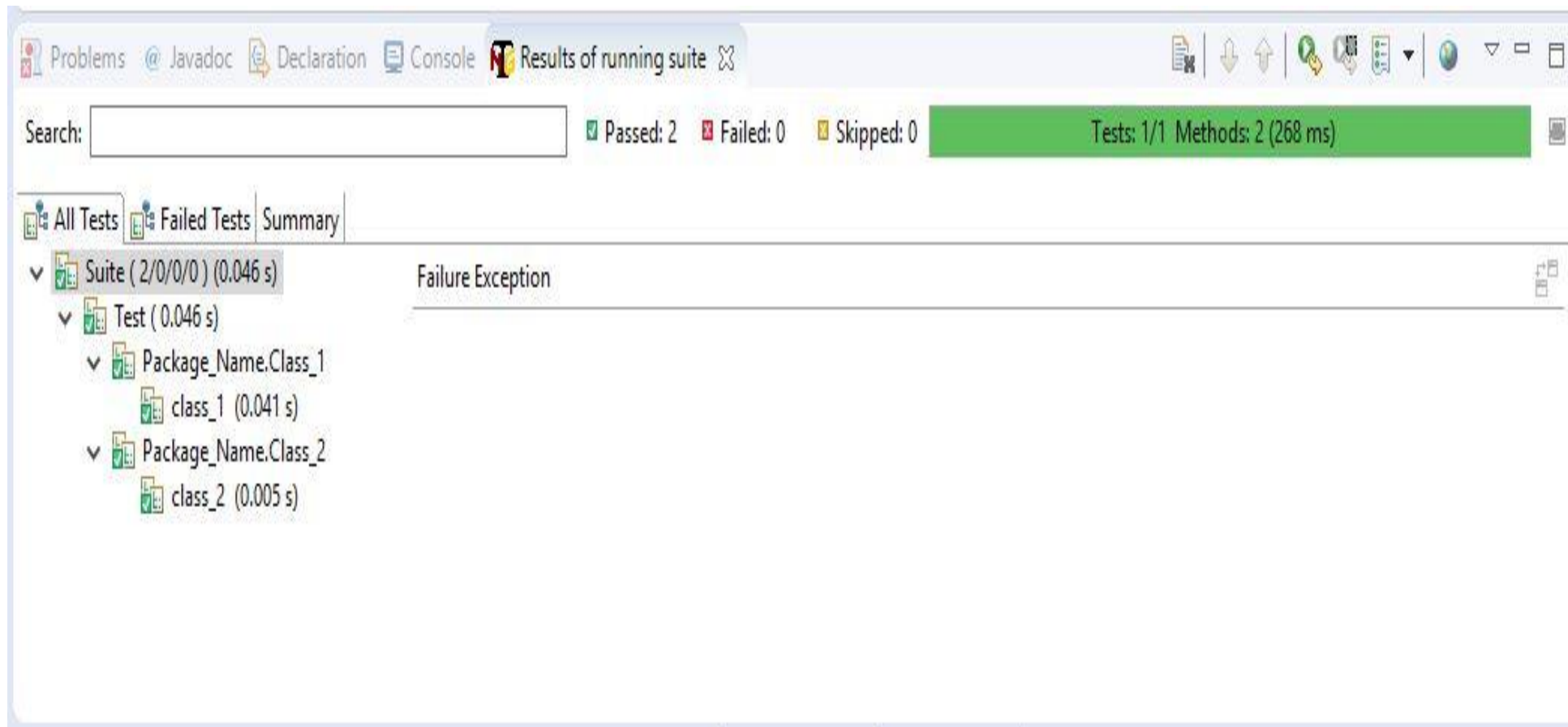
```
=====  
Suite  
Total tests run: 2, Failures: 0, Skips: 0  
=====
```

Reports generated by TestNG

List of reports are below:

- TestNG Console.
- E-mailable Report.
- Default report .
- HTML Format Report.

TestNG Console.



E-mailable Report

Test	# Passed	# Skipped	# Failed	Time (ms)	Included Groups	Excluded Groups
Suite						
Test	2	0	0	91		

Class	Method	Start	Time (ms)
Suite			
Test — passed			
Package_Name.Class_1	class_1	1486551850387	41
Package_Name.Class_2	class_2	1486551850437	5

Test

Package_Name.Class_1#class_1

[back to summary](#)

Package_Name.Class_2#class_2

[back to summary](#)

Default report

Default test

Tests passed/Failed/Skipped:	3/0/0
Started on:	Wed Feb 08 14:38:43 IST 2017
Total time:	0 seconds (123 ms)
Included groups:	
Excluded groups:	

(Hover the method name to see the test class name)

PASSED TESTS			
Test method	Exception	Time (seconds)	Instance
f Test class: TestNG.Test_DataProvider Parameters: Data1		0	TestNG.Test_DataProvider@61832929
f Test class: TestNG.Test_DataProvider Parameters: Data2		0	TestNG.Test_DataProvider@61832929
f Test class: TestNG.Test_DataProvider Parameters: Data3		0	TestNG.Test_DataProvider@61832929

HTML Format Report

Test results

1 suite

All suites

Suite

Info

- C:\Users\Swapnil\workspace\Techcanvas_Project\src\1.xml
- 1 test
- 0 groups
- Times
- Reporter output
- Ignored methods
- Chronological view

Results

- 2 methods, 2 passed
- Passed methods (hide)
 - class 1
 - class 2

✓ Package_Name.Class_2

class_2

✓ Package_Name.Class_1

class_1

Thanks You..