**FRAMEWORKS**

**Configure testing in eclipse**

Go to help in eclipse 🡪 Click on Eclipse Market place 🡪 Go to Find 🡪 Type TestNG 🡪 Click Install

**beforeclass, beforetest, test, dataprovider, afterclass, aftertest, beforesuite, aftersuite, parameters.**

package FW;

import org.testng.annotations.AfterClass;

import org.testng.annotations.AfterGroups;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.BeforeGroups;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Test;

public class Annotations {

@BeforeGroups("shopping")

public void beforeGroups() {

System.out.println("@BeforeGroups");

}

@AfterGroups("shopping")

public void afterGroups() {

System.out.println("@AfterGroups");

}

@BeforeClass

public void beforeClass() {

System.out.println("@BeforeClass");

}

@AfterClass

public void afterClass() {

System.out.println("@AfterClass");

}

@BeforeMethod

public void beforeMethod() {

System.out.println("@BeforeMethod");

}

@AfterMethod

public void afterMethod() {

System.out.println("@AfterMethod");

}

@Test(groups = "shopping")

public void runTest1() {

System.out.println("@Test - runTest1");

}

@Test

public void runTest2() {

System.out.println("@Test - runTest2");

}

}

**Configure testng.xml for creating test suites, specifying test classes, parameters**

**package FW;**

**import org.testng.annotations.Parameters;**

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

**public class ParameterTest {**

**/\*\***

**\* Following method takes one parameter as input. Value of the**

**\* said parameter is defined at suite level.**

**\*/**

**@Parameters({ "suite-param" })**

**@Test**

**public void prameterTestOne(String param) {**

**System.out.println("Test one suite param is: " + param);**

**}**

**/\*\***

**\* Following method takes one parameter as input. Value of the**

**\* said parameter is defined at test level.**

**\*/**

**@Parameters({ "test-two-param" })**

**@Test**

**public void prameterTestTwo(String param) {**

**System.out.println("Test two param is: " + param);**

**}**

**/\*\***

**\* Following method takes two parameters as input. Value of the**

**\* test parameter is defined at test level. The suite level**

**\* parameter is overridden at the test level.**

**\*/**

**@Parameters({ "suite-param", "test-three-param" })**

**@Test**

**public void prameterTestThree(String param, String paramTwo) {**

**System.out.println("Test three suite param is: " + param);**

**System.out.println("Test three param is: " + paramTwo);**

**}**

**}**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<suite name=*"Parameter test Suite"* verbose=*"1"*>

<!-- This parameter will be passed to every test in this suite -->

<parameter name=*"suite-param"* value=*"suite level parameter"* />

<test name=*"Parameter Test one"*>

<classes>

<class name=*"FW.ParameterTest"*>

<methods>

<include name=*"prameterTestOne"* />

</methods>

</class>

</classes>

</test>

<test name=*"Parameter Test two"*>

<!-- This parameter will be passed this test only -->

<parameter name=*"test-two-param"* value=*"Test two parameter"* />

<classes>

<class name=*"FW.ParameterTest"*>

<methods>

<include name=*"prameterTestTwo"* />

</methods>

</class>

</classes>

</test>

<test name=*"Parameter Test three"*>

<!-- Overriding suite level parameter -->

<parameter name=*"suite-param"* value=*"overiding suite parameter"* />

<!-- Test specific parameter -->

<parameter name=*"test-three-param"* value=*"test three parameter"* />

<classes>

<class name=*"FW.ParameterTest"*>

<methods>

<include name=*"prameterTestThree"* />

</methods>

</class>

</classes>

</test>

</suite>

**How to use include, exclude and parallel methods**

**package** FW;

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

**public** **class** Test1 {

@Test

**public** **void** addLocationTestCase() {

System.***out***.println("Im in add location test case");

}

@Test

**public** **void** addDepartmentTestCase() {

System.***out***.println("Im in add department test case");

}

@Test

**public** **void** addEmployeeTestCase() {

System.***out***.println("Im in add employee test case");

}

}

<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >

<suite name="Sample Test Suite" verbose="1" >

<test name="Method Test Cases" >

<classes>

<class name="frame.AddTestCase">

<methods>

<include name="addLocationTestCase" />

<include name="addDepartmentTestCase" />

<exclude name="addEmployeeTestCase" />

</methods>

</class>

</classes>

</test>

</suite>

<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >

<suite name="Sample Test Suite" parallel="methods" verbose="1" >

<test name="Method Test Cases" >

<classes>

<class name="frame.AddTestCase">

</class>

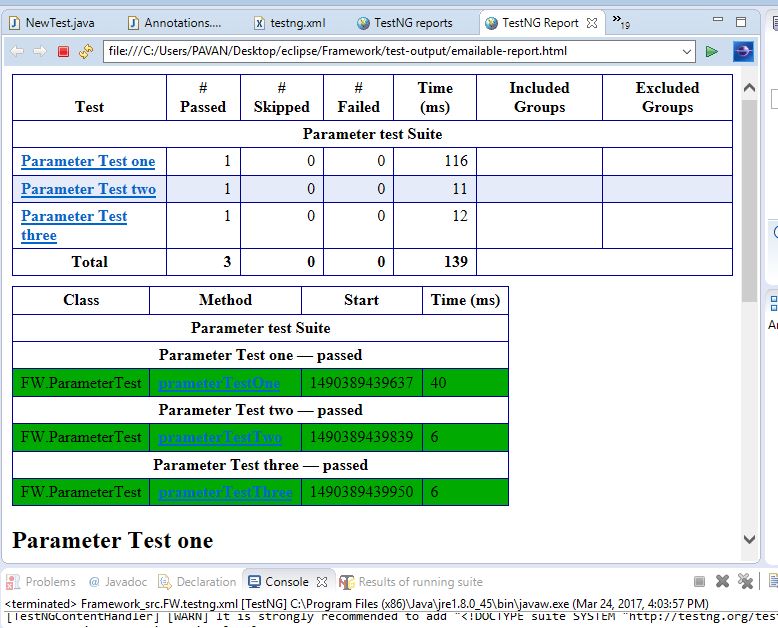
</classes>

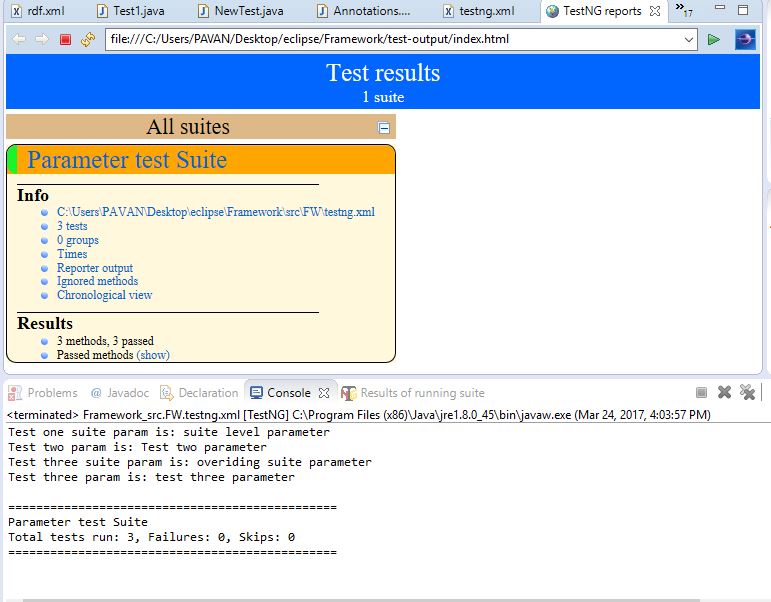
</test>

</suite>

**TestNG report generation**.

I have created test reports for the testing.xml. Below are console report and html reports.





**Using Listeners for report generation (itestlistener, isuitelistener)**

package frame;

import org.testng.IClass;

import org.testng.ITestResult;

import org.testng.TestListenerAdapter;

public class ListenerClass extends TestListenerAdapter {

@Override

public void onTestStart(ITestResult tr) {

log("Test Started....");

}

@Override

public void onTestSuccess(ITestResult tr) {

log("Test '" + tr.getName() + "' PASSED");

log(tr.getTestClass());

log("Priority of this method is " + tr.getMethod().getPriority());

System.out.println(".....");

}

@Override

public void onTestFailure(ITestResult tr) {

log("Test '" + tr.getName() + "' FAILED");

log("Priority of this method is " + tr.getMethod().getPriority());

System.out.println(".....");

}

@Override

public void onTestSkipped(ITestResult tr) {

log("Test '" + tr.getName() + "' SKIPPED");

System.out.println(".....");

}

private void log(String methodName) {

System.out.println(methodName);

}

private void log(IClass testClass) {

System.out.println(testClass);

}

}

<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >

<suite name=*"Log Suite Example"* verbose=*"1"*>

<listeners>

<listener class-name=*"frame.ListenerClass"* />

</listeners>

<test name=*"TestNG logs sample"* preserve-order=*"true"*>

<classes>

<class name=*"frame.LoggingClass"*>

<methods>

<include name=*"methodAddingNumbers"* />

<include name=*"dividedByZero"* />

<include name=*"methodSkip"* />

</methods>

</class>

</classes>

</test>

</suite>

**Create Maven Project**

Configuring Maven in Eclipse:

Open Eclipse 🡪 Go to Help 🡪 Click on Install new software 🡪 Available software window gets opened 🡪 Give the path 🡪 Add 🡪 Click on Finish.

Open Eclipse 🡪 Go to File 🡪 Open New🡪 Go to new project 🡪 Go to Other 🡪 Select New Maven Project 🡪 Click on Finish

**Include class for reading data from excel file under proper package**

import java.io.File;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.util.concurrent.TimeUnit;

import org.apache.poi.xssf.usermodel.XSSFSheet;

import org.apache.poi.xssf.usermodel.XSSFWorkbook;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class Testcase3 {

public static void main(String[] args) {

System.setProperty("webdriver.chrome.driver","C:\\Selenium\\chromedriver.exe");

WebDriver driver = new ChromeDriver();

driver.manage().window().maximize();

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

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

try{

FileInputStream file = new FileInputStream(new File("C:\\Users\\User\\Desktop\\input.xlsx"));

XSSFWorkbook workbook = new XSSFWorkbook(file);

XSSFSheet sheet = workbook.getSheetAt(0);

for(int i=0; i<= sheet.getLastRowNum(); i++){

String keyword = sheet.getRow(i).getCell(0).getStringCellValue();

searchbox.sendKeys(keyword);

searchbox.submit();

driver.manage().timeouts().implicitlyWait(10000, TimeUnit.MILLISECONDS);

}

workbook.close();

file.close();

}

catch(FileNotFoundException fnfe){

fnfe.printStackTrace();

}

catch (IOException ioe) {

ioe.printStackTrace();

}

}

}

**Page Factory**

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

**public** **class** Pagefactory {

**private** WebElement q;

**public** **void** searchFor(String text) {

q.sendKeys(text);

q.submit();

}

}

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

**import** org.openqa.selenium.chrome.ChromeDriver;

**import** org.openqa.selenium.support.PageFactory;

**public** **class** PagefactoryU {

**public** **static** **void** main(String[] args) {

// Create an instance of a driver

System.*setProperty*("webdriver.chrome.driver","C:\\selenium\\chromedriver.exe");

WebDriver driver = **new** ChromeDriver();

// Navigate to the right place

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

Pagefactory page = PageFactory.*initElements*(driver, Pagefactory.**class**);

// And now do the search.

page.searchFor("mahija");

}

}