1. Know the order of execution of TestNG annotations.

import org.testng.annotations.AfterClass;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.AfterSuite;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.BeforeSuite;

import org.testng.annotations.BeforeTest;

import org.testng.annotations.Test;

public class TestClass1

{

public void beforeSuite()

{

System.out.println(“Inside Before Suite Method.”);

}

public void beforeClass()

{

System.out.println(“Inside Before Class of TestClass1.”);

}

public void beforeTest()

{

System.out.println(“Inside Before Test Method of TestClass1.”);

}

public void beforeMethod1()

{

System.out.println(“Inside Before Method 1 of TestClass1.”);

}

public void beforeMethod2()

{

System.out.println(“Inside Before Method 2 of TestClass1.”);

}

public void testMethod1()

{

System.out.println(“Inside Test Method 1 of TestClass1.”);

}

public void testMethod2()

{

System.out.println(“Inside Test Method 2 of TestClass1.”);

}

public void afterTest()

{

System.out.println(“Inside After Test Method of TestClass1.”);

}

public void afterMethod1()

{

System.out.println(“Inside After Method 1 of TestClass1.”);

}

public void afterMethod2()

{

System.out.println(“Inside After Method 2 of TestClass1.”);

}

public void afterClass()

{

System.out.println(“Inside After Class Method of TestClass1.”);

}

public void afterSuite()

{

System.out.println(“Inside After Suite Method.”);

}

}

2. Create testing.xml file to run the test cases in a class file.

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

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

<suite name="Practice">

<test name="Test1">

<parameter name="Username" value="user123@gmail.com"/>

<parameter name="password" value="test@123"/>

<parameter name="mobile\_number" value="1234567899"/>

<classes>

<class name="Practice.Test1"/>

</classes>

</test><!-- Test -->

</suite><!-- Test -->

3. Create a test suite and test groups.

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

<suite name="Test-Suite" >

<test name="ToolsQA" >

<classes>

<class name="TestNG" />

</classes>

</test>

</suite>

<groups>

<run>

<include name="bonding" />

</run>

</groups>

4. List down the assertions and use them in a test case.

public class LearnAssertions

{

WebDriver driver;

String path = System.getProperty("user.dir");

public void SetDriver()

{

System.setProperty("webdriver. chrome.driver",path+"\\Drivers\\chromedriver.exe");

driver = new ChromeDriver();// Object is created- Chrome browser is opened

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

}

public void verifyTitle()

{

driver.get(https://www.amazon.com);

String ActualTitle = driver.getTitle();

String ExpectedTitle = “Welcome to Amazon”;

Assert.assertEquals(ActualTitle, ExpectedTitle);

System.out.println(“Assert passed”);

}

public void closedriver()

{

driver.close();

}

public class LearnAssertionsSoft

{

WebDriver driver;

SoftAssert softassert = new SoftAssert();

SoftAssert softassert2 = new SoftAssert();

String path = System.getProperty("user.dir");

public void SetDriver()

{

System.setProperty("webdriver. chrome.driver",path+"\\Drivers\\chromedriver.exe");

driver = new ChromeDriver();// Object is created - Chrome browser is opened

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

}

public void verifyTitle()

{

driver.get("https://amazon.in");

String ActualTitle = driver.getTitle();

System.out.println("Actual Title :"+ActualTitle);

String ExpectedTitle = "cameras, books, watches, apparel, shoes and e-Gift

Cards. Free Shipping &amp; Cash on Delivery Available.";

softassert.assertEquals(ActualTitle, ExpectedTitle);

System.out.println("Assertion 1 is executed”);

softassert.assertAll();

}

public void verifyElement()

{

WebElement AmazonIcon =

driver. findElement(By.Xpath(“//div[contains(@id,’amazon\_icon’)]);

softassert2.assertEquals (true, AmazonIcon.isDisplayed());

softassert2.assertAll();

System.out.println("Icon is displayed");

System.out.println("Assertion 2 is executed”);

}

public void closedriver()

{

driver.close();

}

}

5. Disable or ignore a test from running.

@Test(enabled = true)

public class DisableTestDemo

{

@Test(enabled = true)

public void testMethodOne()

{

System.out.println("Test method one.");

}

@Test(enabled = false)

public void testMethodTwo()

{

System.out.println("Test method two.");

}

public void testMethodThree()

{

System.out.println("Test method three.");

}

}

6. Make one test script dependent on the other and run both of them.

public class DependentTestExamples

{

@Test (dependsOnMethods = { "testTwo", "testThree" })

public void testOne() {

System.out.println("Test method one");

}

public void testTwo()

{

System.out.println("Test method two");

}

public void testThree()

{

System.out.println("Test method three");

}

}

7. Set priority to all the tests, execute and observe the order of execution.

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.testng.Assert;

import org.testng.annotations.Test;

public class Priority\_In\_testNG

{

WebDriver driver;

@Test (priority=1)

public void openBrowser()

{

driver = new FirefoxDriver();

}

@Test (priority=2)

public void launchGoogle()

{

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

}

@Test (priority=3)

public void peformSeachAndClick1stLink()

{

driver.findElement(By.xpath(".//\*[@title='Search']")).sendKeys("Facebook");

}

@Test (priority=4)

public void FaceBookPageTitleVerification() throws Exception

{

driver.findElement(By.xpath(".//\*[@value='Search']")).click();

Thread.sleep(3000);

Assert.assertEquals(driver.getTitle().contains("Facebook - Google Search"), true);

}

}

8. How to run the test multiple times using invocationCount.

import org.testng.annotations.Test;

public class MultiTimeRunner

{

@Test(invocationCount=5)

public void testRunner()

{

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

}

}

9. Pass parameters to test script.

import org.testng.annotations.Parameters;

import org.testng.annotations.Test;

public class Params

{

@Test

@Parameters ({"val1", "val2"})

public void Sum(int v1, int v2) {

int finalsum = v1 + v2;

System.out.println("The final sum of the given values is " + finalsum);

}

}

10. How to group test cases.

public class Test1

{

@Test(groups = { "group1", "group2" })

public void test\_method1()

{

}

@Test(groups = {"group2"} )

public void test\_method2()

{

}

@Test(groups = {"group1"})

public void test\_method3()

{

}

}

12. Running test cases in parallel.

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

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

import org.testng.annotations.Test;

public class ParallelTest

{

public WebDriver driver;

public void FirefoxTest()

{

driver = new FirefoxDriver();

driver.get("https://www.demoqa.com");

driver.findElement(By.xpath("//\*[@id=\"app\"]/div/div/div[2]/div/div[1]/div/div[1]")).click();

driver.quit();

}

public void ChromeTest()

{

driver = new ChromeDriver();

driver.get("https://www.demoqa.com");

driver.findElement(By.xpath("//\*[@id=\"app\"]/div/div/div[2]/div/div[1]/div/div[1]")).click();

driver.quit();

}

}