

## Execute Testng.xml from command line

Reference - <http://www.seleniumeasy.com/testng-tutorials/how-to-run-testng-xml-webdriver-tests-from-command-line>

[Home](#) >> [TestNG Tutorials](#) >> [Execute Testng.xml from command line](#)

We can execute testng.xml file in different ways.

1. Eclipse
2. IntelliJ
3. Ant / Maven
4. Command Line
5. Batch File.

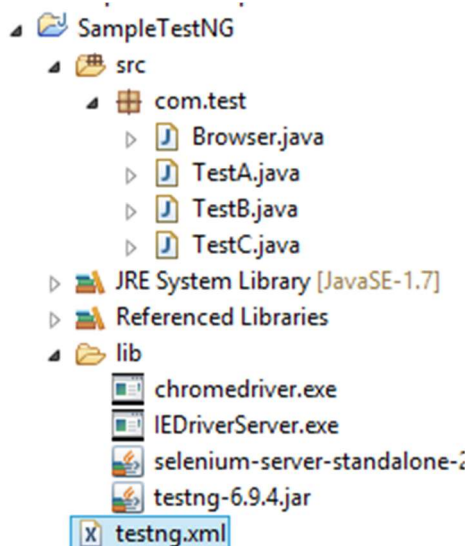
In IDE you can just right click on testng.xml file and click on Run as 'TestNG' will invoke your tests. And we can also [invoke TestNG.xml with Ant build.xml file](#) and we also by doing some simple [configuration to maven pom.xml](#) which we have see examples earlier

Now let us see how to invoke testng.xml from command line step by step:

**Step 1:** Create new project, In example we have created project 'SampleTestNG'

**Step 2:** Create new package. 'com.test' as example package created here.

**Step 3:** Create sample classes which has @Test methods. In the example below we have created four classes as below :



Class - Browser.java, which has getBrowser(String browserType) method which accepts browser Type and returns driver based on the string passed by the user.

```
package com.test;
```

```
import org.openqa.selenium.WebDriver;
```

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

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

```
import org.openqa.selenium.ie.InternetExplorerDriver;
```

```
public class Browser {
```

```
    static String driverPath = "F:/Selenium/SampleTestNG/lib/";
```

```
    public static WebDriver driver;
```

```
    public static WebDriver getBrowser(String browserType) {  
        switch (browserType) {  
            case "firefox":  
                return driver = new FirefoxDriver();  
            case "chrome":
```

```

        System.setProperty("webdriver.chrome.driver", driverPath+"chromedriver.exe");
    }
    case "IE":
        System.setProperty("webdriver.ie.driver", driverPath+"IEDriverServer.exe");
        return driver = new InternetExplorerDriver();
    default:
        System.out.println("browser : " + browserType + " is invalid, Launching Firefox as browser of choice..");
        return driver = new FirefoxDriver();
    }
}

```

We will now try to create three different classes each passing different browser type (Firefox, chrome and InternetExplorer) and create test method in each class.

First create a class and get browser firefox :

```

package com.test;

import org.openqa.selenium.WebDriver;
import org.testng.Assert;
import org.testng.annotations.AfterClass;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.Test;

public class TestA {

    WebDriver driver;

    @BeforeClass
    public void setUp() {
        System.out.println("*****");
        System.out.println("launching firefox browser");
        driver = Browser.getBrowser("firefox");
        driver.manage().window().maximize();
    }

    @Test
    public void testGooglePageTitleInFirefox() {
        driver.navigate().to("http://www.google.com");
        String strPageTitle = driver.getTitle();
        Assert.assertTrue(strPageTitle.equalsIgnoreCase("Google"), "Page title doesn't match");
    }

    @AfterClass
    public void tearDown() {
        if(driver!=null) {
            System.out.println("Closing firefox browser");
            driver.quit();
        }
    }
}

```

Create another class 'TestB.java' and get browser chrome driver to [execute tests in chrome browser](#) :

```

package com.test;

```

```

import org.openqa.selenium.WebDriver;
import org.testng.Assert;
import org.testng.annotations.AfterClass;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.Test;

public class TestB {

    WebDriver driver;

    @BeforeClass
    public void setUp() {
        System.out.println("*****");
        System.out.println("launching chrome browser");
        driver = Browser.getBrowser("chrome");
        driver.manage().window().maximize();
    }

    @Test
    public void testGooglePageTitleInChrome() {
        driver.navigate().to("http://www.google.com");
        String strPageTitle = driver.getTitle();
        Assert.assertTrue(strPageTitle.equalsIgnoreCase("Google"), "Page title doesn't match");
    }

    @AfterClass
    public void tearDown() {
        if(driver!=null) {
            System.out.println("Closing chrome browser");
            driver.quit();
        }
    }
}

```

Create another class 'TestC.java' and get browser chrome driver to [execute tests in IE browser](#):

```

package com.test;

import org.openqa.selenium.WebDriver;
import org.testng.Assert;
import org.testng.annotations.AfterClass;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.Test;

public class TestC {

    WebDriver driver;

    @BeforeClass
    public void setUp() {
        System.out.println("*****");
        System.out.println("launching IE browser");
        driver = Browser.getBrowser("IE");
        driver.manage().window().maximize();
    }

    @Test
    public void testGooglePageTitleInIEBrowser() {

```

```

        driver.navigate().to("http://www.google.com");
        String strPageTitle = driver.getTitle();
        System.out.println("Page title: - "+strPageTitle);
        Assert.assertTrue(strPageTitle.equalsIgnoreCase("Google"), "Page title doesn't match");
    }

    @AfterClass
    public void tearDown() {
        if(driver!=null) {
            System.out.println("Closing IE browser");
            driver.quit();
        }
    }
}

```

The below is the testng.xml file, which will execute all the tests that are available under 'com.test' package. In the above classes, each class has a test method with @Before and @After class which will open browser, validate Google Home page title and close browser. In IDE, it is very simple to execute to click on Run as TestNG.

The below is the testng.xml file

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >
<suite name="Main Test Suite" verbose="2">

    <test name="TestNG Test Group">
        <!-- <classes>
        <class name="com.test.TestA"/>
        </classes> -->
        <packages>
        <package name="com.test"/>
        </packages>
    </test>

</suite>

```

Here is below commands to execute 'testng.xml' file from command line

```

cd F:\Selenium\SampleTestNG
java -cp F:\Selenium\SampleTestNG\lib\*;F:\Selenium\SampleTestNG\bin org.testng.TestNG testng.xml

```

After executing above command, it should execute the tests that we have specified in testng.xml file. And the below is the screen shot after executing the tests.

```
CA. C:\Windows\system32\cmd.exe
F:\Selenium\SampleTestNG>java -cp F:\Selenium\SampleTestNG\lib\*;F:\Selenium\SampleTes
...
... TestNG 6.9.6 by Cedric Beust (cedric@beust.com)
...

[TestNG] Running:
  F:\Selenium\SampleTestNG\testng.xml

*****
launching firefox browser
Closing firefox browser
*****
launching chrome browser
Starting ChromeDriver (v2.9.248315) on port 39703
Closing chrome browser
*****
launching IE browser
Started InternetExplorerDriver server (32-bit)
2.48.0.0
Listening on port 11152
Page title: - Google
Closing IE browser
PASSED: testGooglePageTitleInFirefox
PASSED: testGooglePageTitleInChrome
PASSED: testGooglePageTitleInIEBrowser

=====
TestNG Test Group
Tests run: 3, Failures: 0, Skips: 0
=====

Main Test Suite
Total tests run: 3, Failures: 0, Skips: 0
=====

[TestNG] Time taken by org.testng.reporters.XMLReporter@16181be: 5 ms
[TestNG] Time taken by org.testng.reporters.SuiteHTMLReporter@7691c0: 30 ms
[TestNG] Time taken by org.testng.reporters.JUnitReportReporter@193b604: 5 ms
[TestNG] Time taken by org.testng.reporters.jq.Main@1a896a4: 10 ms
[TestNG] Time taken by org.testng.reporters.EmailableReporter2@1bhea67: 0 ms
[TestNG] Time taken by [FailedReporter passed=0 failed=0 skipped=0]: 0 ms

F:\Selenium\SampleTestNG>
```

In the above command, for 'java -cp' argument, we provide the classpath i.e. path to classes / libraries that our program requires to run. You can check out [Oracle Docs](#) for more information on class path.