

TESTNG - ENVIRONMENT

http://www.tutorialspoint.com/testng/testng_environment.htm

Copyright © tutorialspoint.com

TestNG is a framework for Java, so the very first requirement is to have JDK installed in your machine.

System Requirement

JDK	1.5 or above.
Memory	No minimum requirement.
Disk Space	No minimum requirement.
Operating System	No minimum requirement.

Step 1 - Verify Java Installation in Your Machine

Open the console and execute a **java** command based on the operating system you have installed on your system.

OS	Task	Command
Windows	Open Command Console	c:\> java -version
Linux	Open Command Terminal	\$ java -version
Mac	Open Terminal	machine:~ joseph\$ java -version

Let's verify the output for all the operating systems:

OS	Output
Windows	java version "1.7.0_25" Java™ SE Runtime Environment build1.7.0_25 - b15 Java HotSpot™ 64-Bit Server VM build23.25 - b01, mixedmode
Linux	java version "1.7.0_25" Java™ SE Runtime Environment build1.7.0_25 - b15 Java HotSpot™ 64-Bit Server VM build23.25 - b01, mixedmode
Mac	java version "1.7.0_25" Java™ SE Runtime Environment build1.7.0_25 - b15 Java HotSpot™ 64-Bit Server VM build23.25 - b01, mixedmode

If you do not have Java, install the Java Software Development Kit *SDK* from <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. We are assuming Java 1.7.0_25, as the installed version for this tutorial.

Step 2: Set JAVA Environment

Set the **JAVA_HOME** environment variable to point to the base directory location, where Java is installed on your machine. For example,

OS	Output
Windows	Set the environment variable JAVA_HOME to C:\Program Files\Java\jdk1.7.0_25.
Linux	Export JAVA_HOME=/usr/local/java-current.
Mac	Export JAVA_HOME=/Library/Java/Home.

Append Java compiler location to System Path.

OS	Output
Windows	Append the string C:\Program Files\Java\jdk1.7.0_25\bin at the end of the system variable, Path.
Linux	Export PATH=PATH:JAVA_HOME/bin/
Mac	Not required

Verify Java Installation using the command **java -version** as explained above.

Step 3: Download TestNG Archive

Download the latest version of TestNG jar file from <http://www.testng.org>. At the time of writing this tutorial, we have downloaded *testng-6.8.jar* and copied it onto C:\>TestNG folder.

OS	Archive name
Windows	testng-6.8.jar
Linux	testng-6.8.jar
Mac	testng-6.8.jar

Step 4: Set TestNG Environment

Set the **TESTNG_HOME** environment variable to point to the base directory location, where TestNG jar is stored on your machine. The following table shows how to set the environment variable in Windows, Linux, and Mac, assuming that, we've stored testng-6.8.jar at the location C:\> TestNG.

OS	Description
Windows	Set the environment variable TESTNG_HOME to C:\TESTNG.
Linux	Export TESTNG_HOME=/usr/local/TESTNG
Mac	Export TESTNG_HOME=/Library/TESTNG

Step 5: Set CLASSPATH Variable

Set the **CLASSPATH** environment variable to point to the TestNG jar location.

OS	Description
Windows	Set the environment variable CLASSPATH to %CLASSPATH%;%TESTNG_HOME%\testng-6.8.jar.
Linux	Export CLASSPATH=CLASSPATH:TESTNG_HOME/testng-6.8.jar.
Mac	Export CLASSPATH=CLASSPATH:TESTNG_HOME/testng-6.8.jar.

Step 6: Test TestNG Setup

Create a java class file named TestNGSimpleTest at **C:\>TestNG_WORKSPACE**.

```
import org.testng.annotations.Test;
import static org.testng.Assert.assertEquals;

public class TestNGSimpleTest {
    @Test
    public void testAdd() {
        String str = "TestNG is working fine";
        assertEquals("TestNG is working fine", str);
    }
}
```

TestNG can be invoked in several different ways:

- With a testng.xml file.
- With ANT.
- From the command line.

Let us invoke using the testng.xml file. Create an xml file with the name testng.xml in **C:\>TestNG_WORKSPACE** to execute Test cases.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >

<suite name="Suite1">
  <test name="test1">
    <classes>
      <class name="TestNGSimpleTest"/>
    </classes>
  </test>
</suite>
```

Step 7: Verify the Result

Compile the class using **javac** compiler as follows:

```
C:\TestNG_WORKSPACE>javac TestNGSimpleTest.java
```

Now, invoke the testng.xml to see the result:

```
C:\TestNG_WORKSPACE>java -cp "C:\TestNG_WORKSPACE" org.testng.TestNG testng.xml
```

Verify the output.

```
=====
Suite1
```

Total tests run: 1, Failures: 0, Skips: 0

=====

Loading [MathJax]/jax/output/HTML-CSS/jax.js