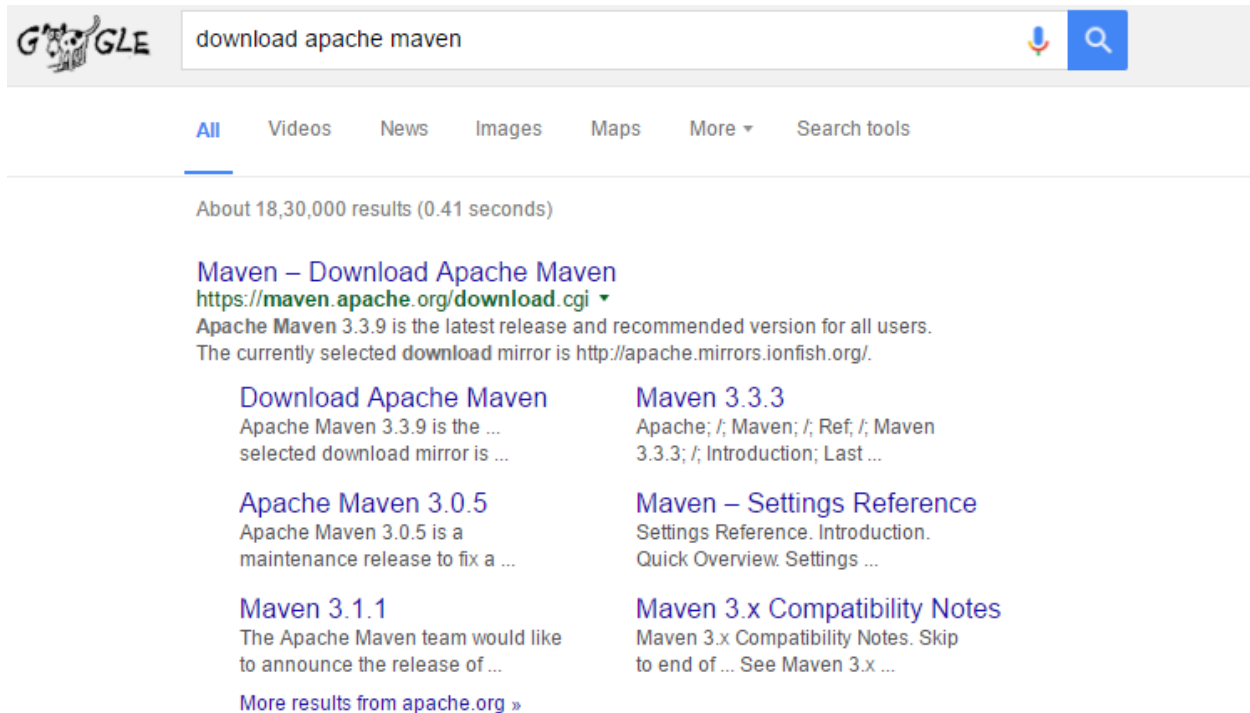


INTEGRATE SELENIUM WITH MAVEN

Download Maven Apache.



The screenshot shows a Google search interface. The search bar contains the text "download apache maven". Below the search bar, there are tabs for "All", "Videos", "News", "Images", "Maps", "More", and "Search tools". The "All" tab is selected. Below the tabs, it says "About 18,30,000 results (0.41 seconds)". The first search result is titled "Maven – Download Apache Maven" with a URL "https://maven.apache.org/download.cgi". Below the title, it says "Apache Maven 3.3.9 is the latest release and recommended version for all users. The currently selected download mirror is http://apache.mirrors.ionfish.org/". There are four more search results listed below: "Download Apache Maven", "Maven 3.3.3", "Apache Maven 3.0.5", and "Maven – Settings Reference". Each result has a brief description. At the bottom, there is a link "More results from apache.org »".

Download the latest version

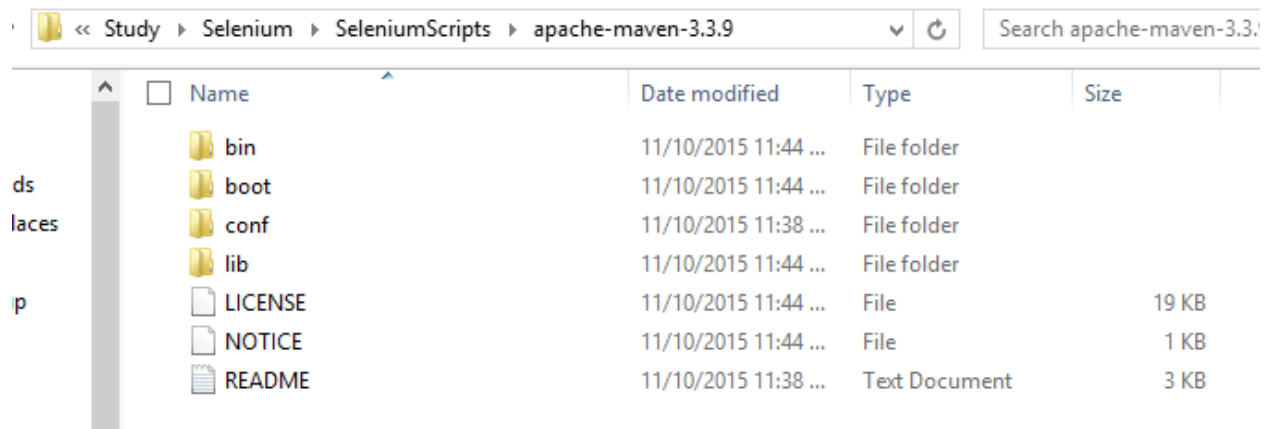
Files

Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the [installation instructions](#). Use a source archive if you intend to build Maven yourself.

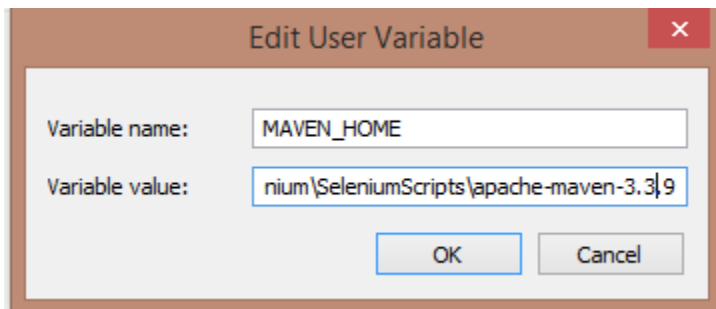
In order to guard against corrupted downloads/installations, it is highly recommended to [verify the signature](#) of the release bundles against the public [KEYS](#) used by the Apache Maven developers.

	Link	Checksum	Signature
Binary tar.gz archive	apache-maven-3.3.9-bin.tar.gz	apache-maven-3.3.9-bin.tar.gz.md5	apache-maven-3.3.9-bin.tar.gz.asc
Binary zip archive	apache-maven-3.3.9-bin.zip	apache-maven-3.3.9-bin.zip.md5	apache-maven-3.3.9-bin.zip.asc
Source tar.gz archive	apache-maven-3.3.9-src.tar.gz	apache-maven-3.3.9-src.tar.gz.md5	apache-maven-3.3.9-src.tar.gz.asc
Source zip archive	apache-maven-3.3.9-src.zip	apache-maven-3.3.9-src.zip.md5	apache-maven-3.3.9-src.zip.asc

Extract the file

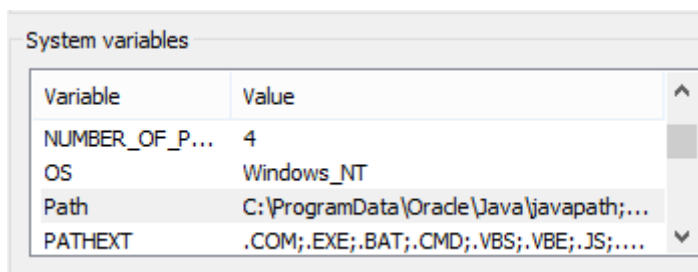


Now Declare Maven in Environment Variable.



Variable Name: MAVEN_HOME

Variable Value: M:\Study\Selenium\SeleniumScripts\apache-maven-3.3.9

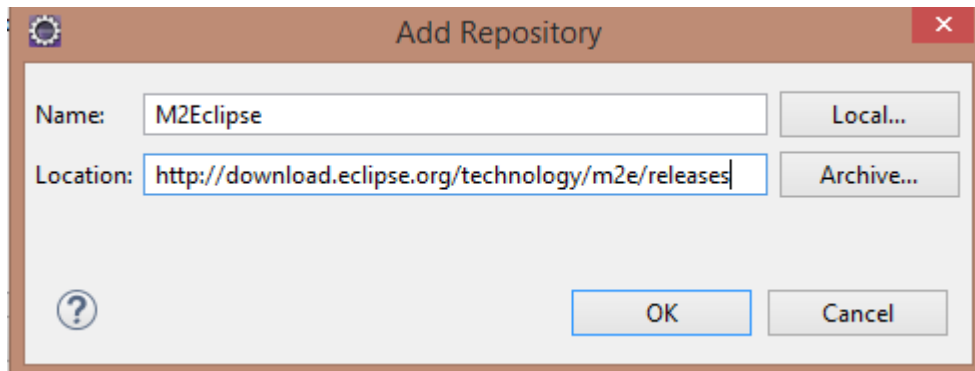


path Name: Path

Path value: M:\Study\Selenium\SeleniumScripts\apache-maven-3.3.9\bin;

Now Add the Maven Plugin in Eclipse . If you have Eclipse luna /higher version no need to add plugin because Maven is inbuild after Kepler.

*open Eclipse Kepler->help->Install new software->Add

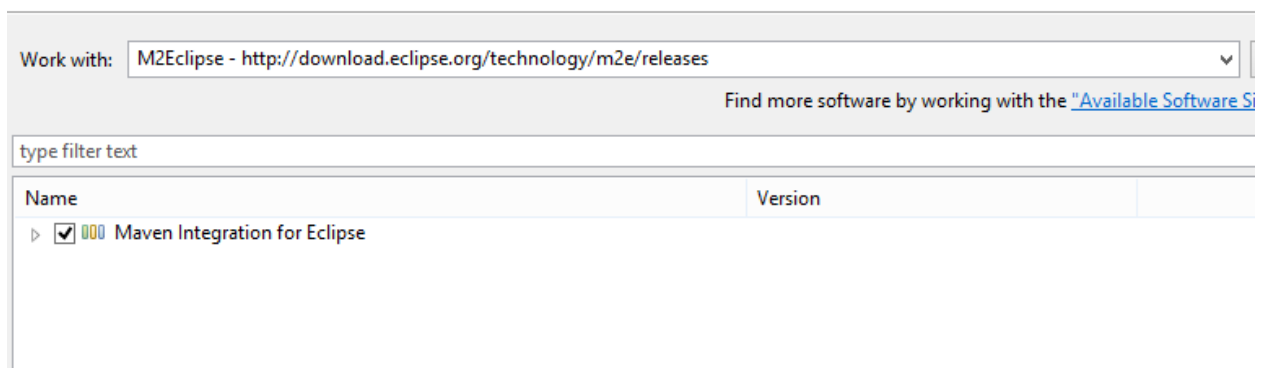


Press ok .

Check the maven integration for eclipse and proceed the installation.

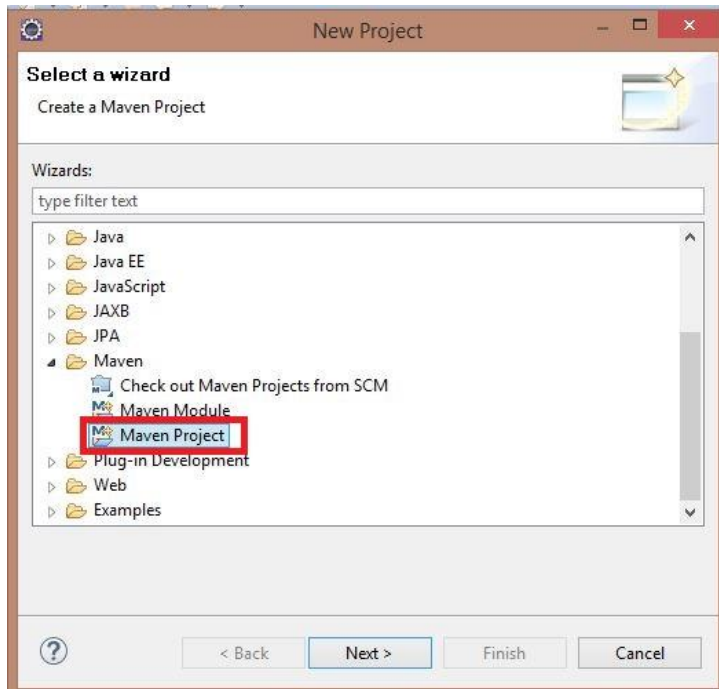
Available Software

Check the items that you wish to install.

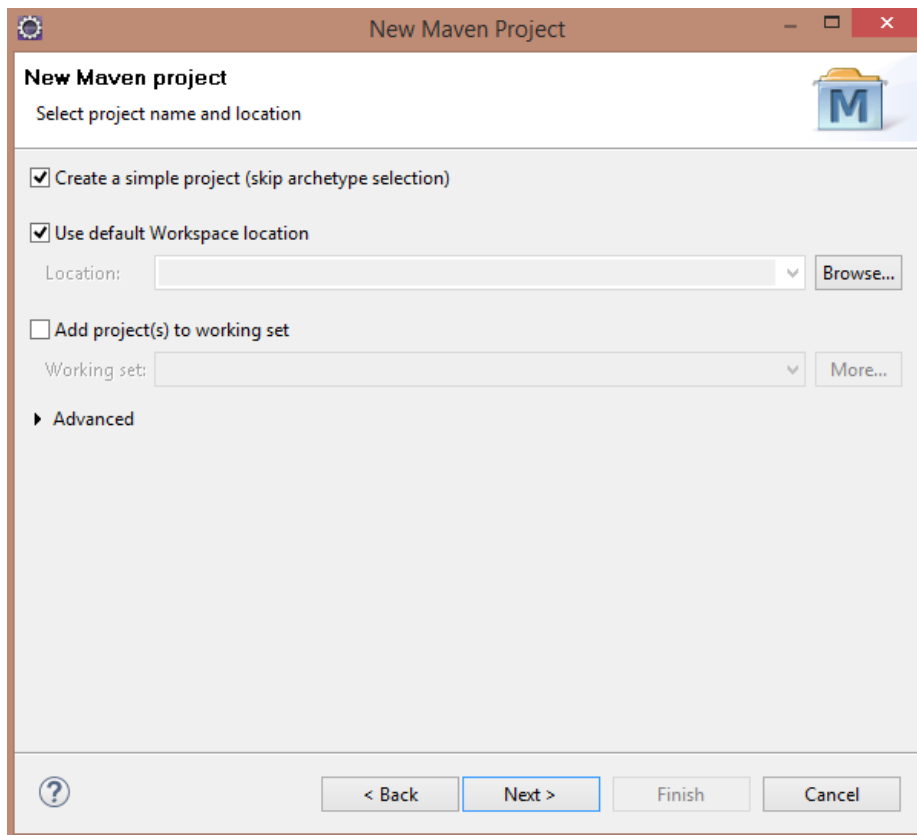


*Create New Maven Project.

File->Project->Maven project



*Check the **Create a simple project** and press next.



Enter Group ID, Artifact ID, Name and Description, it should be anything.->Finish.

New Maven Project

Project "com.dakinfotech.maventutorial" already exists.

Artifact

Group Id:

Artifact Id:

Version:

Packaging:

Name:

Description:

Parent Project

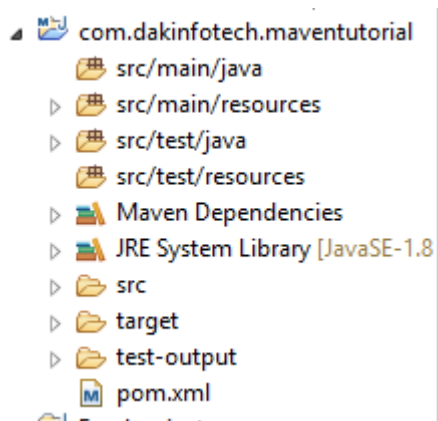
Group Id:

Artifact Id:

Version:

▶ **Advanced**

Will take some time for Project Creation. Check the progress bar right hand side bottom corner.



Edit the pom.xml file(Note : for maven, testng, selenium assign latest version).

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.dakinfotech</groupId>
  <artifactId>com.dakinfotech.maventutorial</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <name>Learn Maven Integration</name>
  <description>Learn Maven Integration</description>
  <packaging>jar</packaging>
  <properties>
<suiteXmlFile>src/main/resources/testng.xml</suiteXmlFile>
</properties>

  <dependencies>
    <dependency>
      <groupId>org.testng</groupId>
      <artifactId>testng</artifactId>
      <version>6.8</version>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>org.seleniumhq.selenium</groupId>
      <artifactId>selenium-java</artifactId>
      <version>2.53.0</version>
    </dependency>
  </dependencies>

  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.3</version>
        <configuration>
          <compilerVersion>1.8</compilerVersion>
          <source>1.8</source>
          <target>1.8</target>
        </configuration>
      </plugin>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-surefire-plugin</artifactId>
        <version>2.18.1</version>
        <configuration>
          <suiteXmlFiles>
            <suiteXmlFile>${suiteXmlFile}</suiteXmlFile>
          </suiteXmlFiles>
        </configuration>
      </plugin>
    </plugins>
  </build>
</project>
```

```

</suiteXmlFiles>
</configuration>
</plugin>
</plugins>
</build>
</project>

```

*Create a class in src/test/java

```

package com.dakinfotech.maventutorialtutor;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.testng.annotations.Test;

public class TutorialsPoint {
    @Test
    public void test1(){
        WebDriver driver=new FirefoxDriver();
        driver.get("http://newtours.demoaut.com/");
        driver.manage().window().maximize();
        driver.manage().timeouts().implicitlyWait(3, TimeUnit.SECONDS);

        driver.findElement(By.name("userName")).sendKeys("arund87@gmail.com");
        driver.findElement(By.name("password")).sendKeys("pass123");
        driver.findElement(By.name("login")).click();
        driver.manage().timeouts().implicitlyWait(3, TimeUnit.SECONDS);
        driver.quit();
    }
}

```

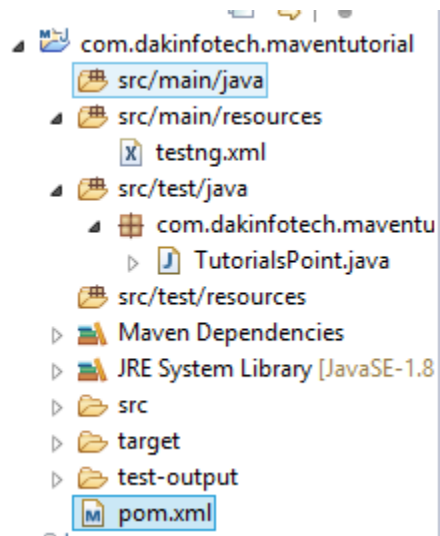
* Right click the class file and Convert to `testng.xml`.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
<suite name="Suite" parallel="none">
    <test name="Test">
        <classes>
            <class name="com.dakinfotech.maventutorialtutor.TutorialsPoint"/>
        </classes>
    </test> <!-- Test -->
</suite> <!-- Suite -->

```

*Move the testng.xml file to src/main/resource because in pom.xml we declared this path.



Now your maven setup is ready for execution.

*open the command prompt(cmd).

>Go to the path of the workspace.

>mvn clean =>This will clean the build.

>mvn install =>it will build the package and run the test.


```

M:\>cd M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial
M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial>mvn clean
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building Learn Maven Integration 0.0.1-SNAPSHOT
[INFO] -----
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ com.dakinfotech.maventutorial ---
[INFO] Deleting M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial\target
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.308 s
[INFO] Finished at: 2016-05-04T22:49:27+05:30
[INFO] Final Memory: 6M/123M
[INFO] -----
M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial>

```

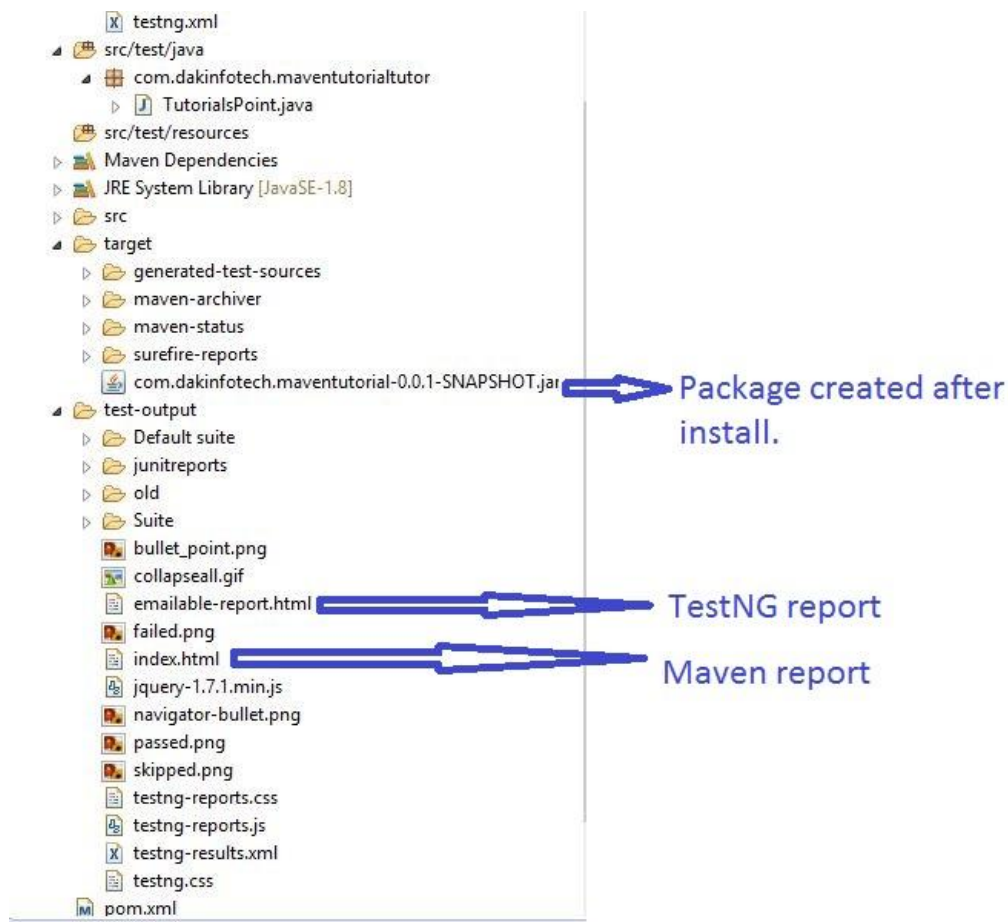
```

-----
T E S T S
-----
Running TestSuite
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 19.758 sec - in TestSuite
Results :
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ com.dakinfotech.maventutorial ---
[INFO] Building jar: M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial\target\com.dakinfotech.maventutorial-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- maven-install-plugin:2.4:install (default-install) @ com.dakinfotech.maventutorial ---
[INFO] Installing M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial\target\com.dakinfotech.maventutorial-0.0.1-SNAPSHOT.jar to C:\Users\arunkavi\.m2\repository\com\dakinfotech\com.dakinfotech.maventutorial\0.0.1-SNAPSHOT\com.dakinfotech.maventutorial-0.0.1-SNAPSHOT.jar
[INFO] Installing M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial\pom.xml to C:\Users\arunkavi\.m2\repository\com\dakinfotech\com.dakinfotech.maventutorial\0.0.1-SNAPSHOT\com.dakinfotech.maventutorial-0.0.1-SNAPSHOT.pom
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 23.407 s
[INFO] Finished at: 2016-05-04T22:50:38+05:30
[INFO] Final Memory: 17M/164M
[INFO] -----
M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial>

```

To rerun the build =>mvn test.

OUTPUTS:



TestNG Report:

file:///M:/Study/Selenium/SeleniumScripts/workspace/com.dakinfotech.maventutorial/test-output/emailable-report.html							
Test	Methods Passed	Scenarios Passed	# skipped	# failed	Total Time	Included Groups	Excluded Groups
Test	1	1	0	0	14.0 seconds		
Class			Method	# of Scenarios	Start	Time (ms)	
Test — passed							
com.dakinfotech.maventutorialtutor.TutorialsPoint			test1	1	1462379615044	14005	

Test

com.dakinfotech.maventutorialtutor.TutorialsPoint:test1

[back to summary](#)

Maven Report:

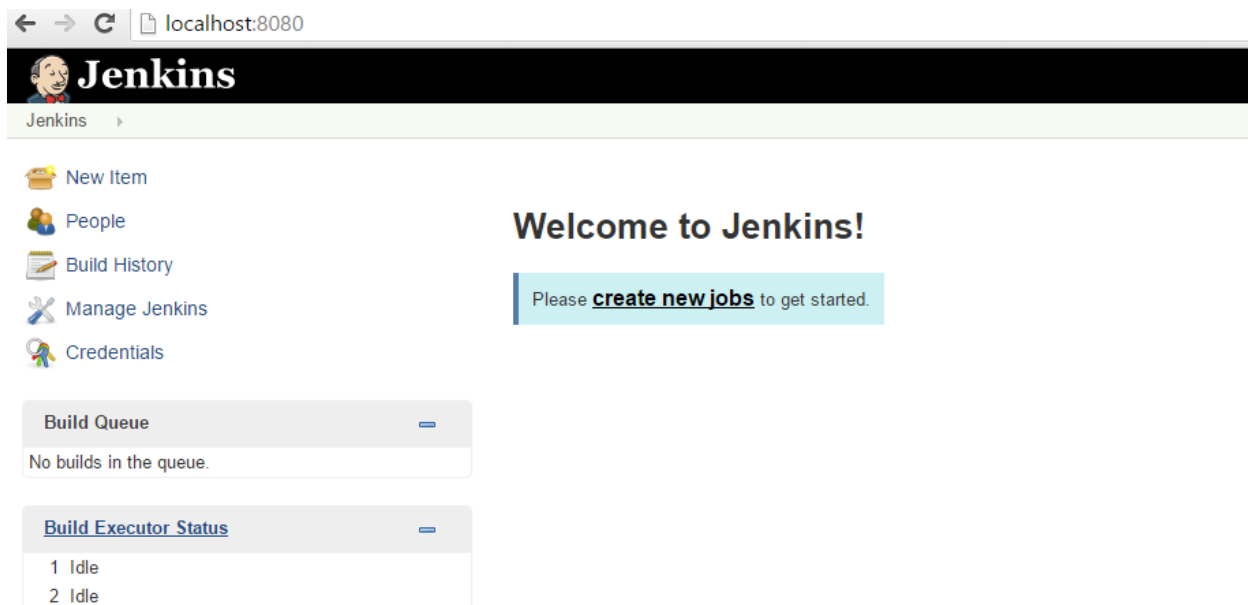


Integration of Selenium Webdriver with Maven and Jenkins

*Run the `java -jar jenkins.war(cmd)` file wait till jenkins is UP and RUNNING.

```
Command Prompt - java -jar jenkins.war
May 04, 2016 11:27:59 PM jenkins.InitReactorRunner$1 onAttained
INFO: Loaded all jobs
May 04, 2016 11:28:01 PM hudson.model.AsyncPeriodicWork$1 run
INFO: Started Download metadata
May 04, 2016 11:28:02 PM org.jenkinsci.main.modules.sshd.SSHD start
INFO: Started SSHD at port 58433
May 04, 2016 11:28:02 PM jenkins.InitReactorRunner$1 onAttained
INFO: Completed initialization
May 04, 2016 11:28:02 PM jenkins.InitReactorRunner$1 onAttained
INFO: Started initialization
May 04, 2016 11:28:02 PM jenkins.InitReactorRunner$1 onAttained
INFO: Listed all plugins
May 04, 2016 11:28:02 PM jenkins.InitReactorRunner$1 onAttained
INFO: Prepared all plugins
May 04, 2016 11:28:02 PM jenkins.InitReactorRunner$1 onAttained
INFO: Started all plugins
May 04, 2016 11:28:02 PM jenkins.InitReactorRunner$1 onAttained
INFO: Augmented all extensions
May 04, 2016 11:28:02 PM jenkins.InitReactorRunner$1 onAttained
INFO: Loaded all jobs
May 04, 2016 11:28:02 PM jenkins.InitReactorRunner$1 onAttained
INFO: Completed initialization
May 04, 2016 11:28:02 PM hudson.WebAppMain$3 run
INFO: Jenkins is fully up and running
```

*Now open the jenkins URL in browser. <http://localhost:8080/>



Go to Manage Jenkins->Configure System. Add the JDK Installation and Maven Installation then save it.

JDK

JDK installations

JDK Name

JAVA_HOME

☐ Install automatically ?

Delete JDK

Add JDK

List of JDK installations on this system

Ant

Ant installations

Add Ant

List of Ant installations on this system

Maven

Maven installations

Maven Name

MAVEN_HOME

☐ Install automatically ?

Save **Apply**

Refer Environment variable.

*New Item->Maven Project->Enter your project name and press OK.

Jenkins

?

Jenkins > All >

New Item

People

Build History

Manage Jenkins

Credentials

Build Queue -

No builds in the queue.

Build Executor Status -






1 Idle


2 Idle

Item name

- ☐ **Freestyle project**
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.
- ☒ **Maven project**
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
- ☐ **External Job**
This type of job allows you to record the execution of a process run outside Jenkins, even on a remote machine. This is designed so that you can use Jenkins as a dashboard of your existing automation system. See [the documentation for more details](#).
- ☐ **Multi-configuration project**
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.


OK

-  New Item
-  People
-  Build History
-  Manage Jenkins
-  Credentials

All		
S	W	Name ↓
		MavenJenkinsIntegration
Icon: S M L		

Build Queue 

No builds in the queue.

Build Executor Status 

1 Idle
2 Idle

Project is created.->open the project->configure the root pom and save it.

Pre Steps

Add pre-build step ▼

Build

Root POM

Goals and options

Post Steps

☐ Run only if build succeeds
 ☐ Run only if build succeeds or is unstable
 ☒ Run regardless

Should the post-build steps run only for successful builds, etc.

Add post-build step ▼

Build Settings

☐ E-mail Notification

Post-build Actions

Add post-build action ▼

Save

Apply

Now execute the program by Build Now. *Congrats your first integration test is successfully ran.

The screenshot shows the Jenkins web interface. At the top, there's a header with the Jenkins logo and the project name 'Maven project MavenJenkinsIntegration'. Below the header, there's a sidebar with navigation links: 'Back to Dashboard', 'Status', 'Changes', 'Workspace', 'Build Now', 'Delete Maven project', 'Configure', and 'Modules'. The main content area is titled 'Maven project MavenJenkinsIntegration' and contains a description: 'This is the sample project to understand the selenium-maven-jenkins integration.' Below this, there are links for 'Workspace', 'Recent Changes', 'Latest Test Result (no failures)', and 'Latest Test Result (no failures)'. On the left, there's a 'Build History' section with a search bar and a table showing build #1 on May 4, 2016 at 11:57 PM. Below the table are links for 'RSS for all' and 'RSS for failures'. At the bottom, there's a 'Permalinks' section with four links: 'Last build (#1). 1 min 1 sec ago', 'Last stable build (#1). 1 min 1 sec ago', 'Last successful build (#1). 1 min 1 sec ago', and 'Last completed build (#1). 1 min 1 sec ago'.

Jenkins

Jenkins > MavenJenkinsIntegration

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

[Delete Maven project](#)

[Configure](#)

[Modules](#)

Build History [trend](#)

find

#1	May 4, 2016 11:57 PM
----	----------------------

[RSS for all](#) [RSS for failures](#)

Maven project MavenJenkinsIntegration

This is the sample project to understand the selenium-maven-jenkins integration.

[Workspace](#)

[Recent Changes](#)

[Latest Test Result \(no failures\)](#)

[Latest Test Result \(no failures\)](#)

Permalinks

- [Last build \(#1\). 1 min 1 sec ago](#)
- [Last stable build \(#1\). 1 min 1 sec ago](#)
- [Last successful build \(#1\). 1 min 1 sec ago](#)
- [Last completed build \(#1\). 1 min 1 sec ago](#)

To

view console output. »Press #1 and select console output.

- [Back to Project](#)
- [Status](#)
- [Changes](#)
- [Console Output](#)**
- [View as plain text](#)
- [Edit Build Information](#)
- [Delete Build](#)
- [Test Result](#)
- [Redeploy Artifacts](#)
- [See Fingerprints](#)

Console Output

```
Started by user anonymous
Building in workspace C:\Users\arunkavi\.jenkins\jobs\MavenJenkinsIntegration\workspace
Parsing POMs
Discovered a new module com.dakinfotech:com.dakinfotech.maventutorial Learn Maven Integration
Modules changed, recalculating dependency graph
[com.dakinfotech.maventutorial] $ "C:\Program Files\Java\jdk1.8.0_73\bin/java" -cp C:\Users\arunkavi\.jenkins\plugins\maven-
plugin\WEB-INF\lib\maven3-agent-1.5.jar;M:\Study\Selenium\SeleniumScripts\apache-maven-3.3.9\boot\plexus-classworlds-
2.5.2.jar;M:\Study\Selenium\SeleniumScripts\apache-maven-3.3.9\conf\logging_jenkins.maven3.agent\Maven3Main
M:\Study\Selenium\SeleniumScripts\apache-maven-3.3.9 C:\Users\arunkavi\.jenkins\war\WEB-INF\lib\remoting-2.53.3.jar
C:\Users\arunkavi\.jenkins\plugins\maven-plugin\WEB-INF\lib\maven3-interceptor-commons-1.5.jar 58698
<==[JENKINS REMOTING CAPACITY]==>channel started
Executing Maven: -B -f M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial\pom.xml clean install
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building Learn Maven Integration 0.0.1-SNAPSHOT
[INFO] -----
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ com.dakinfotech.maventutorial ---
[INFO] Deleting M:\Study\Selenium\SeleniumScripts\workspace\com.dakinfotech.maventutorial\target
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ com.dakinfotech.maventutorial ---
[WARNING] Using platform encoding (Cp1252 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] Copying 1 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.3:compile (default-compile) @ com.dakinfotech.maventutorial ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ com.dakinfotech.maventutorial ---
[WARNING] Using platform encoding (Cp1252 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] Copying 0 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.3:testCompile (default-testCompile) @ com.dakinfotech.maventutorial ---
```

*****THANK YOU*****