

Maven Demo

1. Execute first maven command

Step 1: Log into your vagrant VM and navigate to folder devopsdemo which was created during git remote demo. This folder now contains the source code

```
cd gitremotedemo/devopsdemo
```

```
vagrant@precise32:~$ pwd
/home/vagrant
vagrant@precise32:~$ cd gitremotedemo/devopsdemo
vagrant@precise32:~/gitremotedemo/devopsdemo$ ls
pom.xml  src
vagrant@precise32:~/gitremotedemo/devopsdemo$
```

Step 2: execute maven compile

```
mvn compile
```

Maven then downloads a bunch of jars required by it to compile the source code. You should find a build success message at the end

```
[INFO] Compiling 1 source file to /home/vagrant/gitremotedemo/devopsdemo/target/classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1:01.257s
[INFO] Finished at: Sun Oct 09 07:15:24 UTC 2016
[INFO] Final Memory: 6M/15M
[INFO] -----
vagrant@precise32:~/gitremotedemo/devopsdemo$
```

Step 3: check point. A new folder called target would be created which contains the compiled source code. In this case App.class

```
vagrant@precise32:~/gitremotedemo/devopsdemo$ ls
pom.xml  src  target
vagrant@precise32:~/gitremotedemo/devopsdemo$ cd target
vagrant@precise32:~/gitremotedemo/devopsdemo/target$ ls
classes
vagrant@precise32:~/gitremotedemo/devopsdemo/target$ cd classes/
vagrant@precise32:~/gitremotedemo/devopsdemo/target/classes$ ls
App.class
vagrant@precise32:~/gitremotedemo/devopsdemo/target/classes$
```

Step 4: Execute maven test

```
mvn test
```

```
-----
T E S T S
-----
Running AppTest
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.124 sec

Results :

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 55.633s
[INFO] Finished at: Sun Oct 09 07:25:53 UTC 2016
[INFO] Final Memory: 7M/18M
[INFO] -----
vagrant@precise32:~/gitremotedemo/devopsdemo$
```

Note that this time it ran the testcases too and the folder target now has lot more folders too.

```
vagrant@precise32:~/gitremotedemo/devopsdemo$ cd target
vagrant@precise32:~/gitremotedemo/devopsdemo/target$ ls
classes  surefire  surefire-reports  test-classes
vagrant@precise32:~/gitremotedemo/devopsdemo/target$
```

Step 5: Execute maven install

```
mvn install
```

```
vagrant@precise32:~/gitremotedemo/devopsdemo$ cd target
vagrant@precise32:~/gitremotedemo/devopsdemo/target$ ls
classes          maven-devops-demo1-1.0.1-SNAPSHOT.jar  surefire-reports
maven-archiver  surefire                                test-classes
vagrant@precise32:~/gitremotedemo/devopsdemo/target$
```

Now check the creation of maven-devops-demo1-1.0.1-SNAPSHOT.jar