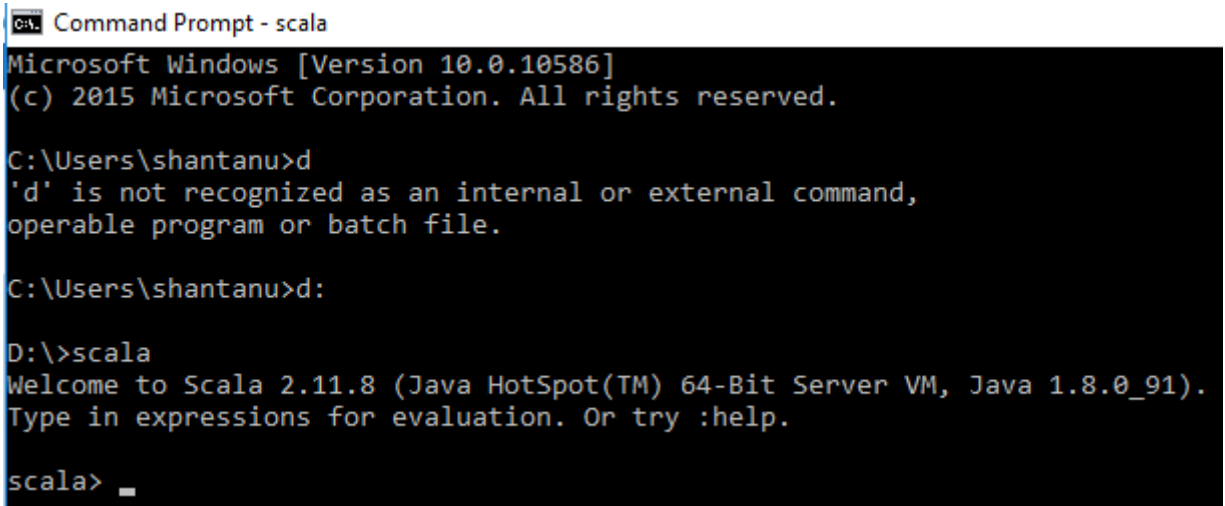


Installing Spark on Windows 10.

1. **Install Scala:** Download Scala from the link: <http://downloads.lightbend.com/scala/2.11.8/scala-2.11.8.msi>

- a. Set environmental variables:
 - i. User variable:
 - Variable: SCALA_HOME;
 - Value: C:\Program Files (x86)\scala
 - ii. System variable:
 - Variable: PATH
 - Value: C:\Program Files (x86)\scala\bin
- b. Check it on cmd, see below.



```
Command Prompt - scala
Microsoft Windows [Version 10.0.10586]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\shantanu>d
'd' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\shantanu>d:

D:\>scala
Welcome to Scala 2.11.8 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_91).
Type in expressions for evaluation. Or try :help.

scala> _
```

2. **Install Java 8:** Download Java 8 from the link:
<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

- a. Set environmental variables:
 - i. User variable:
 - Variable: JAVA_HOME
 - Value: C:\Program Files\Java\jdk1.8.0_91
 - ii. System variable:
 - Variable: PATH

- Value: C:\Program Files\Java\jdk1.8.0_91\bin

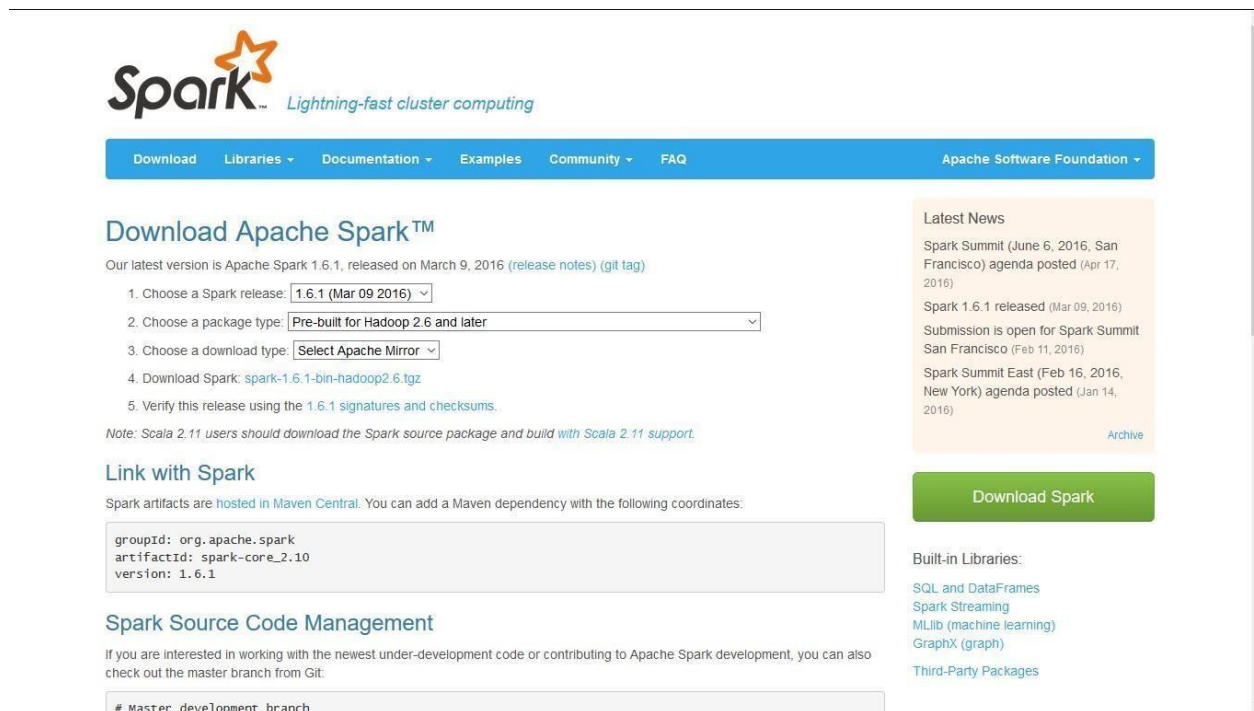
b. Check on cmd, see below:

```
C:\> Command Prompt

D:\> java -version
java version "1.8.0_91"
Java(TM) SE Runtime Environment (build 1.8.0_91-b14)
Java HotSpot(TM) 64-Bit Server VM (build 25.91-b14, mixed mode)

D:\> _
```

3. **Install Eclipse Mars.** Download it from the link: <https://eclipse.org/downloads/> and extract it into C drive.
 - a. Set environmental variables:
 - i. User variable:
 - Variable: ECLIPSE_HOME
 - Value: C:\eclipse
 - ii. System variable:
 - Variable: PATH
 - Value: C:\eclipse \bin
4. **Install Spark 1.6.1.** Download it from the following link: <http://spark.apache.org/downloads.html> and extract it into D drive, such as D:\Spark.



Download Apache Spark™

Our latest version is Apache Spark 1.6.1, released on March 9, 2016 ([release notes](#)) ([git tag](#))

- Choose a Spark release: **1.6.1 (Mar 09 2016)**
- Choose a package type: **Pre-built for Hadoop 2.6 and later**
- Choose a download type: **Select Apache Mirror**
- Download Spark: [spark-1.6.1-bin-hadoop2.6.tgz](#)
- Verify this release using the [1.6.1 signatures and checksums](#).

Note: Scala 2.11 users should download the Spark source package and build with Scala 2.11 support.

Link with Spark

Spark artifacts are [hosted in Maven Central](#). You can add a Maven dependency with the following coordinates:

```
groupId: org.apache.spark
artifactId: spark-core_2.10
version: 1.6.1
```

Spark Source Code Management

If you are interested in working with the newest under-development code or contributing to Apache Spark development, you can also check out the master branch from Git:

```
# Master development branch
```

Latest News

- Spark Summit (June 6, 2016, San Francisco) agenda posted (Apr 17, 2016)
- Spark 1.6.1 released (Mar 09, 2016)
- Submission is open for Spark Summit San Francisco (Feb 11, 2016)
- Spark Summit East (Feb 16, 2016, New York) agenda posted (Jan 14, 2016)

[Archive](#)

Download Spark

Built-in Libraries:

- [SQL and DataFrames](#)
- [Spark Streaming](#)
- [MLlib \(machine learning\)](#)
- [GraphX \(graph\)](#)

[Third-Party Packages](#)

- a. Set environmental variables:
 - i. User variable:
 - Variable: SPARK_HOME
 - Value: D:\spark\spark-1.6.1-bin-hadoop2.6
 - ii. System variable:
 - Variable: PATH

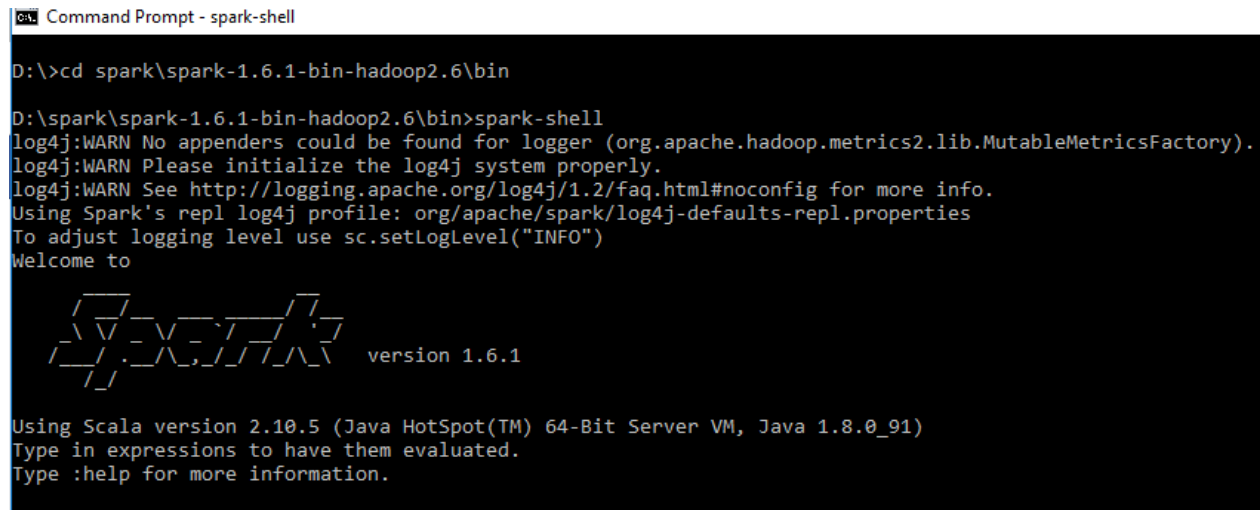
- Value: D:\spark\spark-1.6.1-bin-hadoop2.6\bin

5. **Download Windows Utilities:** Download it from the link:

<https://github.com/steveloughran/winutils/tree/master/hadoop-2.6.0/bin>

And paste it in D:\spark\spark-1.6.1-bin-hadoop2.6\bin

6. **Execute Spark** on cmd, see below:



```

Command Prompt - spark-shell

D:\>cd spark\spark-1.6.1-bin-hadoop2.6\bin

D:\spark\spark-1.6.1-bin-hadoop2.6\bin>spark-shell
log4j:WARN No appenders could be found for logger (org.apache.hadoop.metrics2.lib.MutableMetricsFactory).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
Using Spark's repl log4j profile: org/apache/spark/log4j-defaults-repl.properties
To adjust logging level use sc.setLogLevel("INFO")
Welcome to

  ____      _
 / ___|    / \
 \___ \  / _ \
  ___) / / ___\
 /____/_/_\___\

version 1.6.1

Using Scala version 2.10.5 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_91)
Type in expressions to have them evaluated.
Type :help for more information.

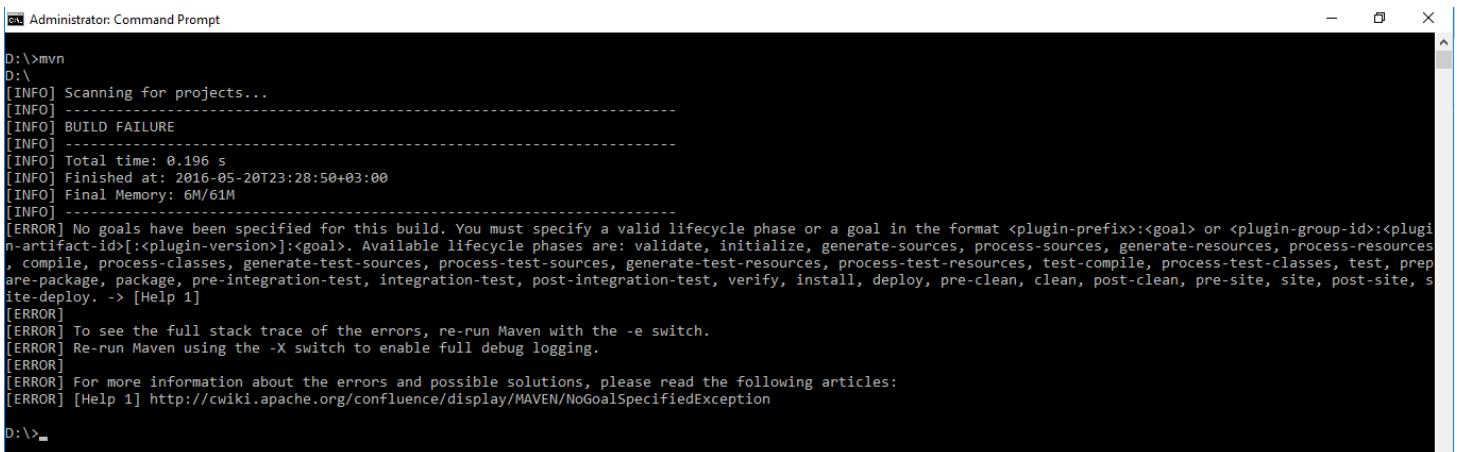
```

7. **Install Maven 3.3.** Download Apache-Maven-3.3.9 from the link:

<http://apache.mivzakim.net/maven/maven-3/3.3.9/binaries/apache-maven-3.3.9-bin.zip>

And extract it into D drive, such as D:\apache-maven-3.3.9

- Set Environmental variables:
 - User variable
 - Variable: MAVEN_HOME
 - Value: D:\apache-maven-3.3.9
 - System variable
 - Variable: Path
 - Value: D:\apache-maven-3.3.9\bin
- Check on cmd, see below



```

Administrator: Command Prompt

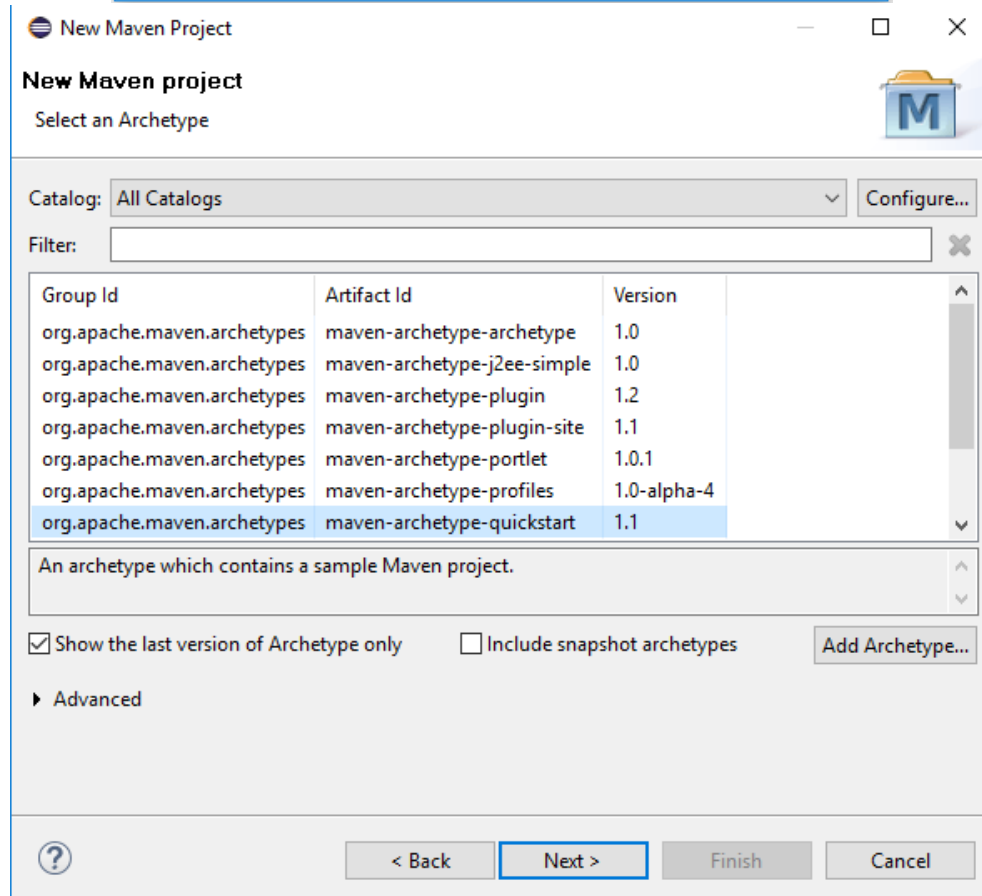
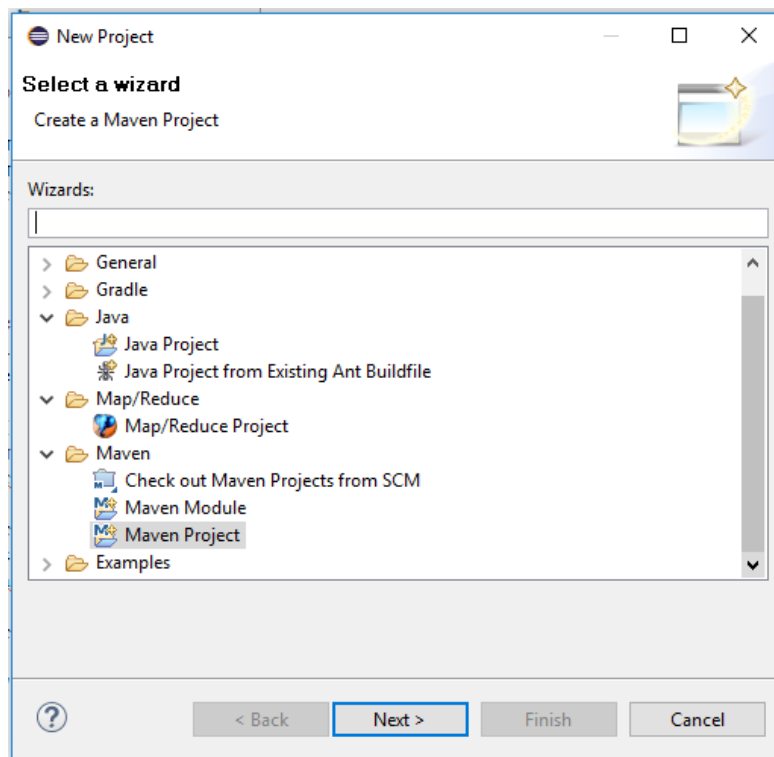
D:\>mvn
D:\>
[INFO] Scanning for projects...
[INFO] -----
[INFO] BUILD FAILURE
[INFO] -----
[INFO] Total time: 0.196 s
[INFO] Finished at: 2016-05-20T23:28:50+03:00
[INFO] Final Memory: 6M/61M
[INFO] -----
[ERROR] No goals have been specified for this build. You must specify a valid lifecycle phase or a goal in the format <plugin-prefix>:<goal> or <plugin-group-id>:<plugin-artifact-id>[:<plugin-version>]:<goal>. Available lifecycle phases are: validate, initialize, generate-sources, process-sources, generate-resources, process-resources, compile, process-classes, generate-test-sources, process-test-sources, generate-test-resources, process-test-resources, test-compile, process-test-classes, test, prepare-package, pre-integration-test, integration-test, post-integration-test, verify, install, deploy, pre-clean, clean, post-clean, pre-site, site, post-site, site-deploy. -> [Help 1]
[ERROR]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]
[ERROR] For more information about the errors and possible solutions, please read the following articles:
[ERROR] [Help 1] http://cwiki.apache.org/confluence/display/MAVEN/NoGoalSpecifiedException

D:\>

```

8. **Create first WordCount project.**

- Open Eclipse and do File → New → project → Select Maven Project; see below.



- b. Enter Group id, Artifact id, and click finish.

New Maven Project

New Maven project
Specify Archetype parameters

Group Id: sparkWCexample

Artifact Id: spWCexample

Version: 0.0.1-SNAPSHOT

Package: sparkWCexample.spWCexample

Properties available from archetype:

Name	Value

Advanced

Back Next Finish Cancel

c. **Edit pom.xml.** Paste the following code.

```
<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>

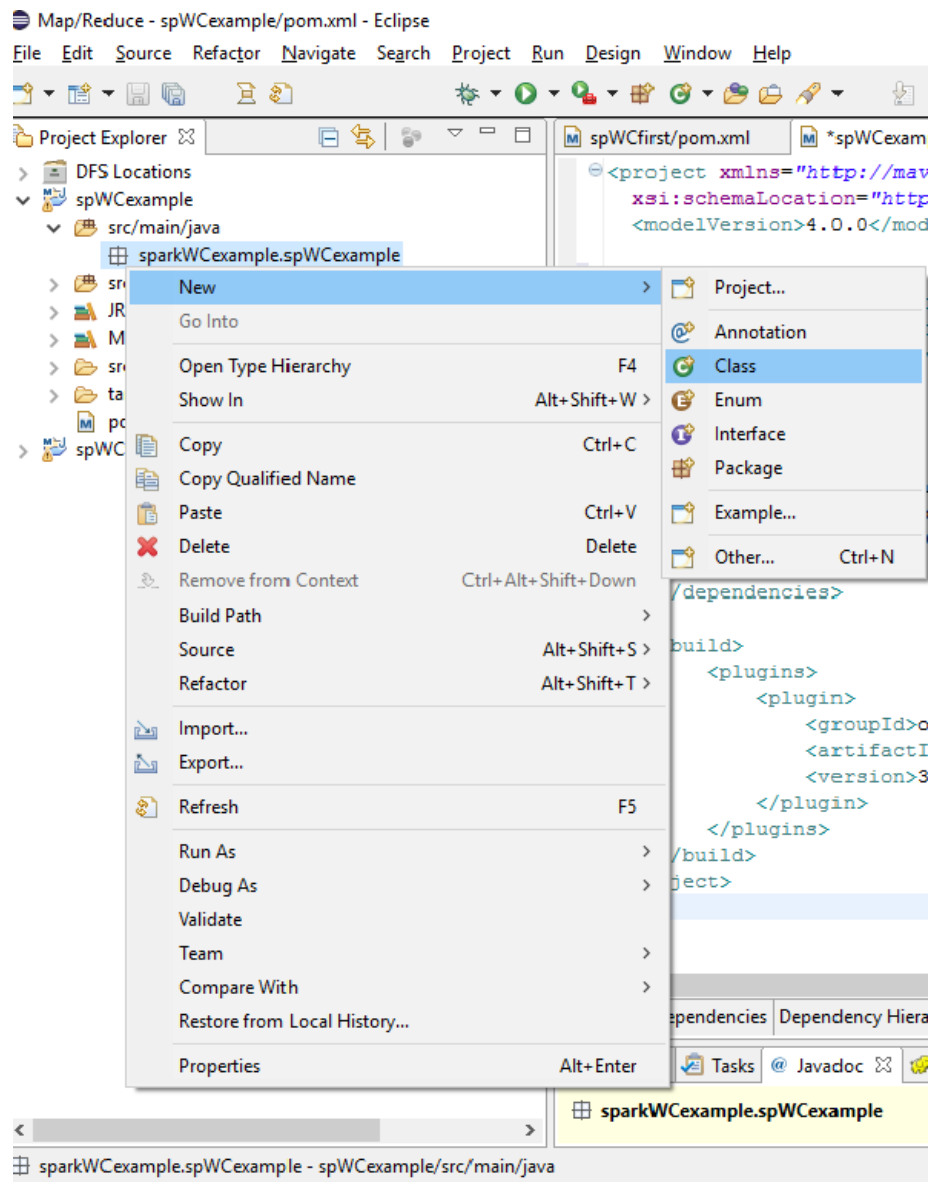
  <groupId>sparkWCexample</groupId>
  <artifactId>spWCexample</artifactId>
  <version>1.0-SNAPSHOT</version>

  <dependencies>
    <dependency>
      <groupId>org.apache.spark</groupId>
      <artifactId>spark-core_2.10</artifactId>
      <version>1.2.0</version>
    </dependency>
  </dependencies>

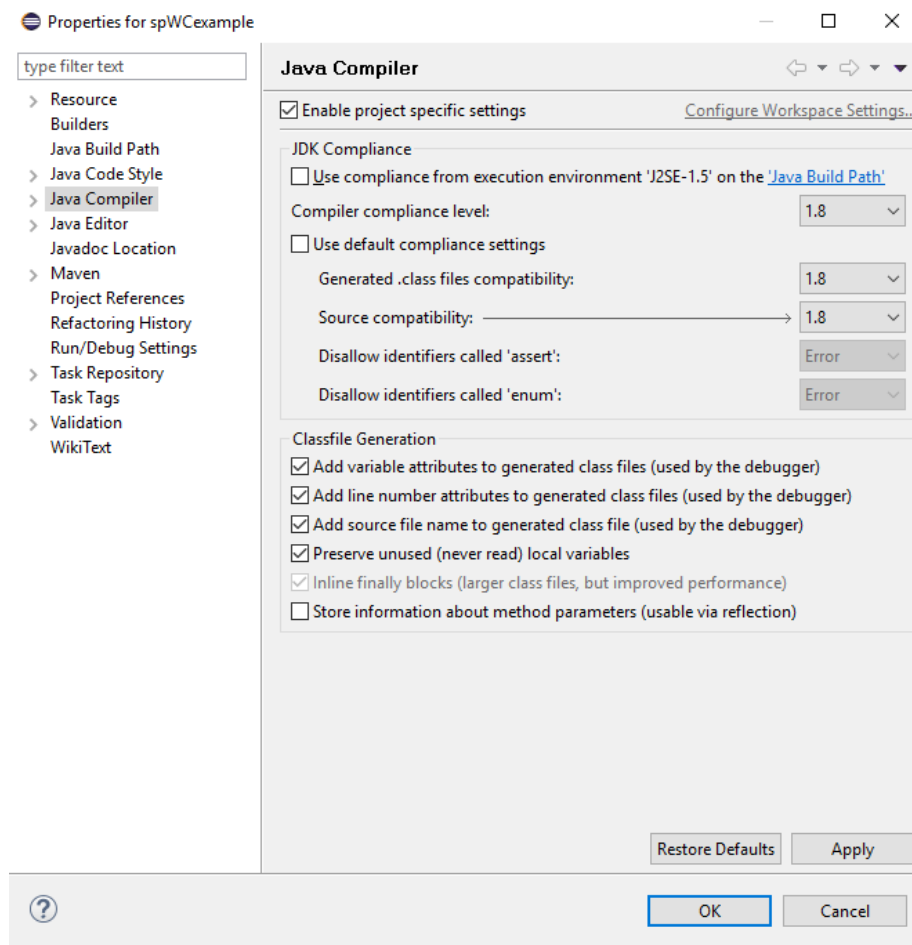
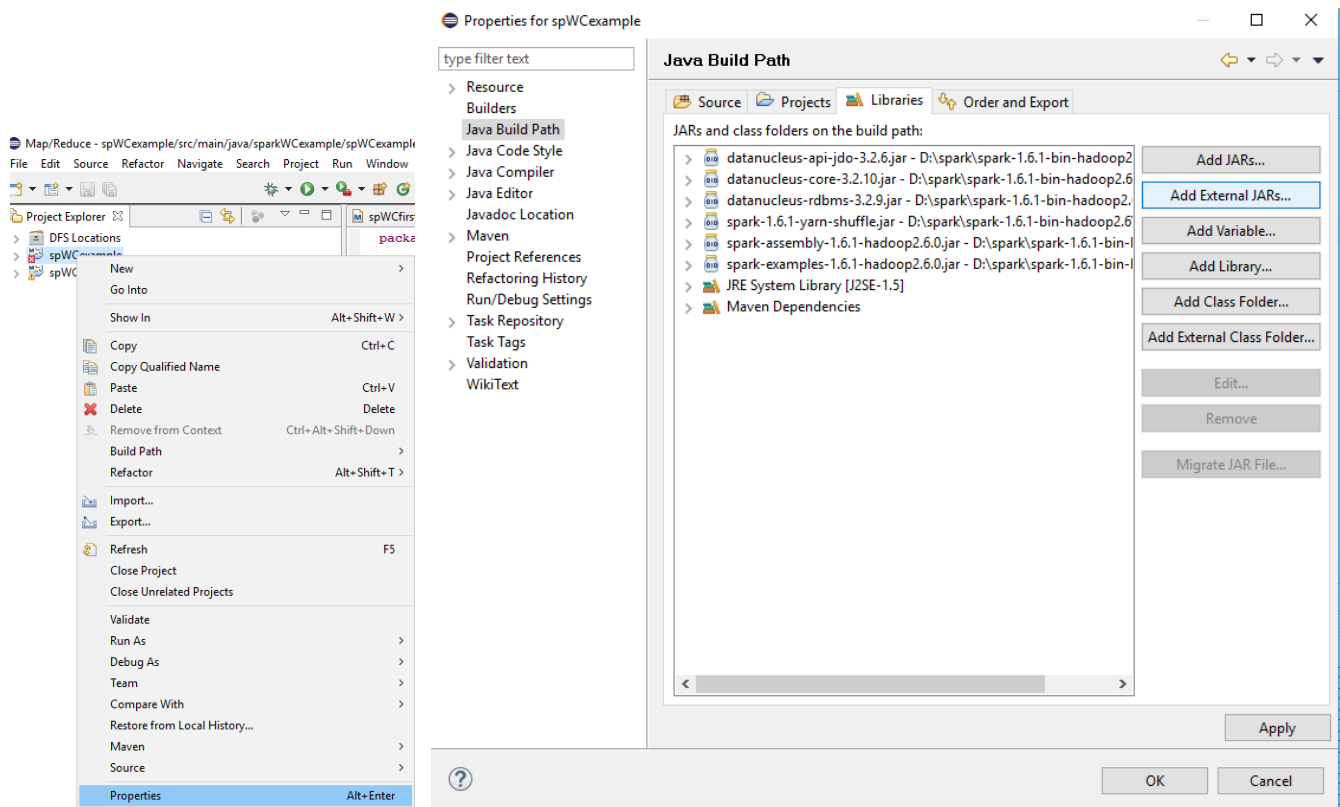
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.3</version>
      </plugin>
    </plugins>
  </build>
```

</project>

- d. Write your code or just copy given WordCount code from D:\spark\spark-1.6.1-bin-hadoop2.6\examples\src\main\java\org\apache\spark\examples



- e. Now, add external jar from the location D:\spark\spark-1.6.1-bin-hadoop2.6\lib and set Java 8 for compilation; see below.



- f. Build the project: Go to the following location (where we stored the project) on cmd:
D:\hadoop\examples\spWCexample
Write **mvn package** on cmd

```
Administrator: Command Prompt
D:\hadoop\examples\spWCexample>mvn package
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building spWCexample 1.0-SNAPSHOT
[INFO] -----
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ spWCexample ---
[WARNING] Using platform encoding (Cp1252 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory D:\hadoop\examples\spWCexample\src\main\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.3:compile (default-compile) @ spWCexample ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ spWCexample ---
[WARNING] Using platform encoding (Cp1252 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory D:\hadoop\examples\spWCexample\src\test\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.3:testCompile (default-testCompile) @ spWCexample ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ spWCexample ---
[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ spWCexample ---
[INFO] Building jar: D:\hadoop\examples\spWCexample\target\spWCexample-1.0-SNAPSHOT.jar
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----
[INFO] Total time: 8.499 s
[INFO] Finished at: 2016-05-20T23:48:48+03:00
[INFO] Final Memory: 21M/177M
[INFO] -----
D:\hadoop\examples\spWCexample>
```

- g. Execute the project: Go to the following location on cmd: D:\spark\spark-1.6.1-bin-hadoop2.6\bin
Write the following command

spark-submit --class groupid.artifactid.classname --master local[2] /path to the jar file created using maven /path to a demo test file /path to output directory

spark-submit --class sparkWCexample.spWCexample.WC --master local[2]
/hadoop/examples/spWCexample/target/spWCexample-1.0-SNAPSHOT.jar
/hadoop/examples/spWCexample/how.txt /hadoop/examples/spWCexample/answer.txt

```
Administrator: Command Prompt
D:\spark\spark-1.6.1-bin-hadoop2.6\bin>spark-submit --class sparkWCexample.spWCexample.WC --master local[2] /hadoop/examples/spWCexample/target/spWCexample-1.0-SNAPSHOT.jar /hadoop/examples/spWCexample/how.txt /hadoop/examples/spWCexample/answer.txt
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
16/05/21 00:01:47 INFO SparkContext: Running Spark version 1.6.1
16/05/21 00:01:48 INFO SecurityManager: Changing view acls to: shantanu
16/05/21 00:01:48 INFO SecurityManager: Changing modify acls to: shantanu
16/05/21 00:01:48 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with view permissions: Set(shantanu); users with modify permissions: Set(shantanu)
16/05/21 00:01:49 INFO Utils: Successfully started service 'sparkDriver' on port 58374.
16/05/21 00:01:50 INFO Slf4jLogger: Slf4jLogger started
16/05/21 00:01:50 INFO Remoting: Starting remoting
16/05/21 00:01:51 INFO Remoting: Remoting started; listening on addresses :[akka.tcp://sparkDriverActorSystem@132.72.225.79:58387]
16/05/21 00:01:51 INFO Utils: Successfully started service 'sparkDriverActorSystem' on port 58387.
16/05/21 00:01:51 INFO SparkEnv: Registering MapOutputTracker
16/05/21 00:01:51 INFO SparkEnv: Registering BlockManagerMaster
16/05/21 00:01:51 INFO DiskBlockManager: Created local directory at C:\Users\shantanu\AppData\Local\Temp\blockmgr-f08ed6e7-b8e6-4e81-b5be-72674397e4ba
16/05/21 00:01:51 INFO MemoryStore: MemoryStore started with capacity 511.1 MB
16/05/21 00:01:51 INFO SparkEnv: Registering OutputCommitCoordinator
16/05/21 00:01:52 WARN Utils: Service 'SparkUI' could not bind on port 4040. Attempting port 4041.
16/05/21 00:01:52 INFO Utils: Successfully started service 'SparkUI' on port 4041.
16/05/21 00:01:52 INFO SparkUI: Started SparkUI at http://132.72.225.79:4041
16/05/21 00:01:52 INFO HttpFileServer: HTTP File server directory is C:\Users\shantanu\AppData\Local\Temp\spark-d4f81ffc-b218-4bce-b3d9-877a4f81e1dc\httpd-673c260c-3ee6-4d8e-9627-ecd77a983c7e
16/05/21 00:01:52 INFO HttpServer: Starting HTTP Server
16/05/21 00:01:52 INFO Utils: Successfully started service 'HTTP file server' on port 58390.
16/05/21 00:01:52 INFO SparkContext: Added JAR file:/D:/hadoop/examples/spWCexample/target/spWCexample-1.0-SNAPSHOT.jar at http://132.72.225.79:58390/jars/spWCexample-1.0-SNAPSHOT.jar with timestamp 1463778112284
16/05/21 00:01:52 INFO Executor: Starting executor ID driver on host localhost
16/05/21 00:01:52 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferService' on port 58395.
16/05/21 00:01:52 INFO NettyBlockTransferService: Server created on 58395
16/05/21 00:01:52 INFO BlockManagerMaster: Trying to register BlockManager
16/05/21 00:01:52 INFO BlockManagerMasterEndpoint: Registering block manager localhost:58395 with 511.1 MB RAM, BlockManagerId(driver, localhost, 58395)
16/05/21 00:01:52 INFO BlockManagerMaster: Registered BlockManager
16/05/21 00:01:53 INFO MemoryStore: Block broadcast_0 stored as values in memory (estimated size 127.4 KB, free 127.4 KB)
16/05/21 00:01:53 INFO MemoryStore: Block broadcast_0_piece0 stored as bytes in memory (estimated size 13.9 KB, free 141.3 KB)
16/05/21 00:01:53 INFO BlockManagerInfo: Added broadcast_0_piece0 in memory on localhost:58395 (size: 13.9 KB, free: 511.1 MB)
16/05/21 00:01:53 INFO SparkContext: Created broadcast 0 from textFile at WC.java:66
16/05/21 00:01:54 INFO MemoryStore: Block broadcast_1 stored as values in memory (estimated size 127.4 KB, free 268.8 KB)
16/05/21 00:01:54 INFO MemoryStore: Block broadcast_1_piece0 stored as bytes in memory (estimated size 13.9 KB, free 282.7 KB)
16/05/21 00:01:54 INFO BlockManagerInfo: Added broadcast_1_piece0 in memory on localhost:58395 (size: 13.9 KB, free: 511.1 MB)
16/05/21 00:01:54 INFO SparkContext: Created broadcast 1 from textFile at WC.java:68
16/05/21 00:01:54 INFO FileInputFormat: Total input paths to process : 1
Exception in thread "main" org.apache.hadoop.mapred.FileAlreadyExistsException: Output directory file:/hadoop/examples/spWCexample/answer.txt already exists
    at org.apache.hadoop.mapred.FileOutputFormat.checkOutputSpecs(FileOutputFormat.java:132)
    at org.apache.spark.rdd.PairRDDFunctions$$anonfun$saveAsHadoopDataset$1.apply$mcV$sp(PairRDDFunctions.scala:1179)
```

- h. You can also check the progress of the project at: <http://localhost:4040/jobs/>
- i. Finally get the answers; see below.

