**[Setup Apache Spark in eclipse(Scala IDE) : Word count example using Apache spark in Scala IDE](http://www.devinline.com/2016/01/apache-spark-setup-in-eclipse-scala-ide.html)**

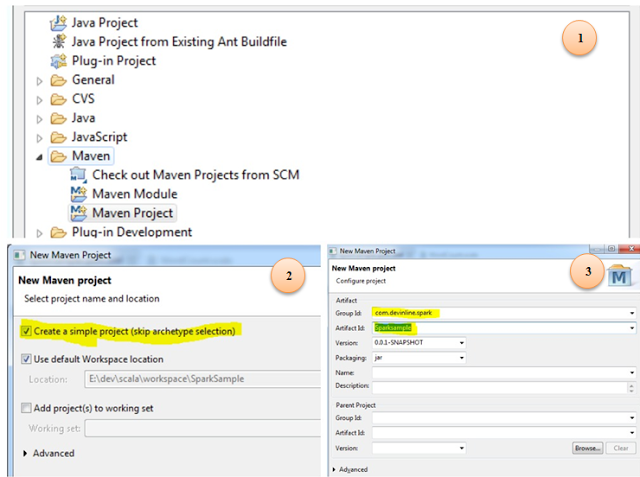
**Apache spark -** a very known in memory computing engine to process big data workloads. Scala IDE(an eclipse project) can be used to develop spark application. The main agenda of this post is to setup development environment for spark application in scala IDE and run word count example.

**Download Scala IDE**:-

Scala IDE is an eclipse project which provides a very intuitive development environment for Scala and Spark application. [Download Scala IDE](http://scala-ide.org/) and install it.

**Create a Maven project**:-  
Maven is a popular package management tool for Java-based languages that allows us to link libraries present in public repositories.We can use Maven itself to build our project, or use other tools like Scala’s sbt tool or Gradle.

**1**. Go to: **File-> New -> Project -> Maven project**  and create a maven project.Fill Group Id and Artifact Id & click finish.   
Group Id = com.devinline.spark and Artifact Id = SparkSample

[](http://1.bp.blogspot.com/-z4iPl0QKLFM/Vo4yhE6mw2I/AAAAAAAAJ3o/A51_1FEJ8zw/s1600/11111.PNG)

**2.** Update pom.xml:- [Download pom.xml](https://drive.google.com/file/d/0B-ur4R5mlgGLRGNETk5nX1JUNVE/view?usp=sharing) sample and update it in above maven project. It has spark dependency jar entry which will be downloaded while building.

**3.** Add Scala Nature to this project :-

Right click on project -> configure - > Add Scala Nature.

**4.** Update Scala compiler version for Spark:-

Scala IDE by default uses latest version(2.11) of Scala compiler, however Spark uses version 2.10.So we need to update appropriate version for IDE.

Right click on project- > Go to **properties -> Scala compiler -> update Scala installation** version to 2.10.5 

**5.** Remove Scala Library Container from build path :- (Optional)

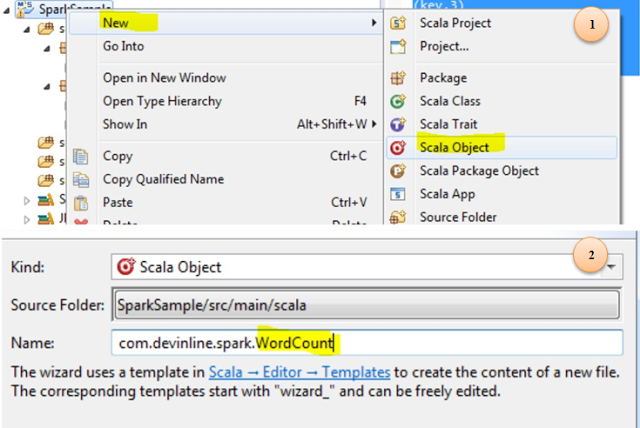
Jars required in already added via spark core(via pom.xml), so multiple jars is not required.

Right click on the project -> Build path -> Configure build path  and remove Scala Library Container.

**6.** Update source folder **src/main/java to src/main/scala** (Right click -> Refactor -> Rename  to scala).Now create a package under this name it as com.devinline.spark.

**7.**Create a Scala object under package created above name it as WordCount.scala

Right click on package -> New -> Scala Object  and add WordCount at the end of Name.

[](http://4.bp.blogspot.com/-3FQPyNgywv8/Vo4z6FR1r9I/AAAAAAAAJ4A/FPL2QnGyEzE/s1600/111111_345.PNG)

**8.** Update WordCount.scala with following code lines

**package** **com.devinline.spark**

**import** **org.apache.spark.SparkConf**

**import** **org.apache.spark.SparkContext**

**import** **org.apache.spark.rdd.RDD.rddToPairRDDFunctions**

**object** **WordCount** {

**def** main(args**:** **Array**[**String**]) **=** {

//Start the Spark context

**val** conf **=** **new** **SparkConf**()

.setAppName("WordCount")

.setMaster("local")

**val** sc **=** **new** **SparkContext**(conf)

//Read some example file to a test RDD

**val** test **=** sc.textFile("input.txt")

test.flatMap { line **=>** //for each line

line.split(" ") //split the line in word by word.

}

.map { word **=>** //for each word

(word, **1**) //Return a key/value tuple, with the word as key and 1 as value

}

.reduceByKey(**\_** + **\_**) //Sum all of the value with same key

.saveAsTextFile("output.txt") //Save to a text file

//Stop the Spark context

sc.stop

}

}