How to run Spark locally - Summary

1. Download Scala Eclipse IDE
2. Download latest Spark and unzip it. All we need is the jar folder.
3. Create Scala project
4. Create package
5. Create Scala Object

Copy the program in the Scala Object

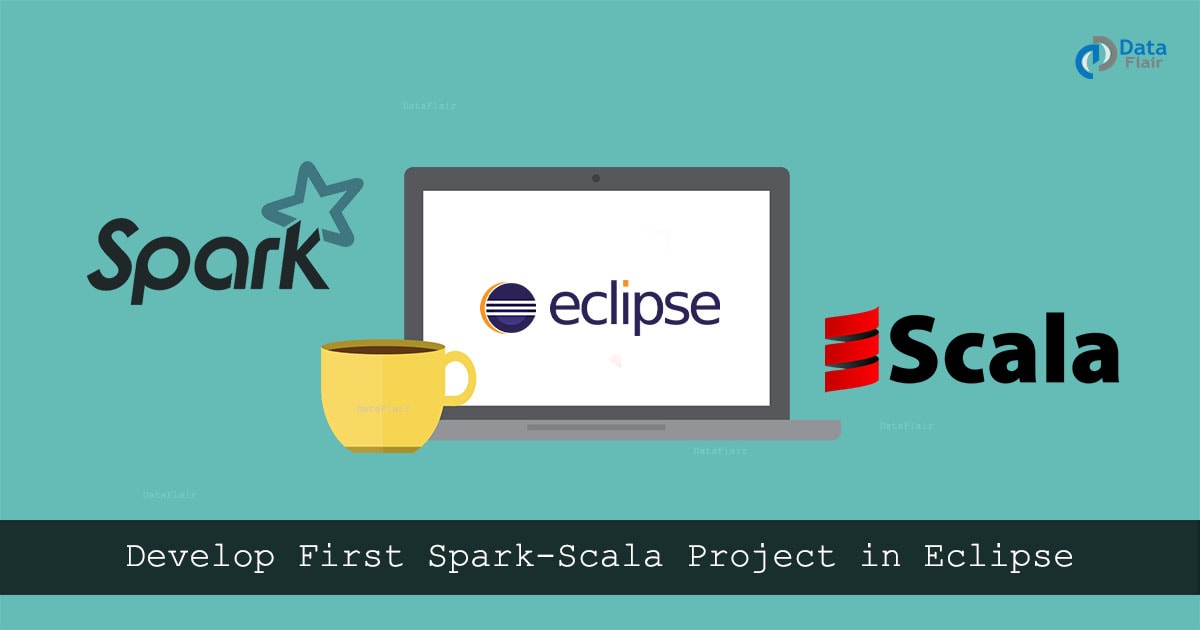
Make sure object has the same name as the file

6 Add external jars. Include all files in the Jar folder (2 above

Create Spark Project in Scala With Eclipse Without Maven

1. Objective – Spark Scala Project

This step by step tutorial will explain how to create a Spark project in Scala with Eclipse without Maven and how to submit the application after the creation of jar. This Guide also briefs about the installation of [**Scala**](http://data-flair.training/blogs/why-you-should-learn-scala-introductory-tutorial/)plugin in eclipse and setup spark environment in eclipse. Learn how to configure development environment for developing Spark applications in Scala in this tutorial.  
If you are completely new to Apache Spark, I recommend you to read this [**Apache Spark Introduction Guide.**](http://data-flair.training/blogs/apache-spark-tutorial/)

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/Create-First-Spark-Project-in-Scala-in-Eclipse-Tutorial-DataFlair.jpg)

*Create Spark project in Scala with Eclipse without Maven*

2. Steps to Create the Spark Project in Scala

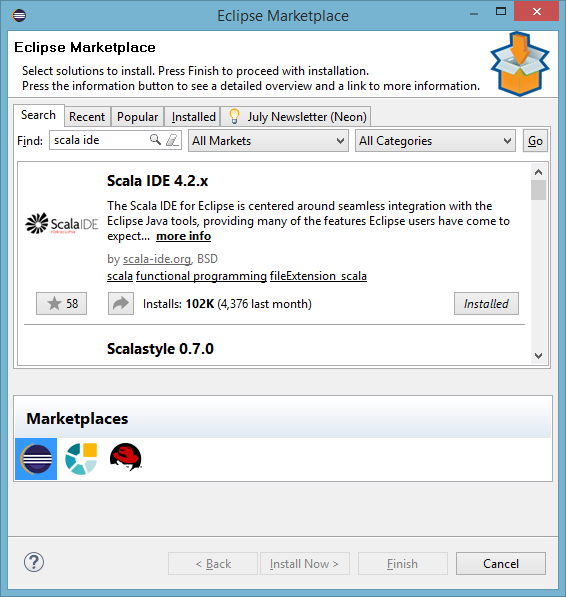
To create Spark Project in Scala with Eclipse without Maven follow the steps given below-

i. Platform Used / Required

* **Operating System:** Windows / Linux / Mac
* **Java:** Oracle Java 7
* **Scala:** 2.11
* **Eclipse:** Eclipse Luna, Mars or later

ii. Install Eclipse plugin for Scala

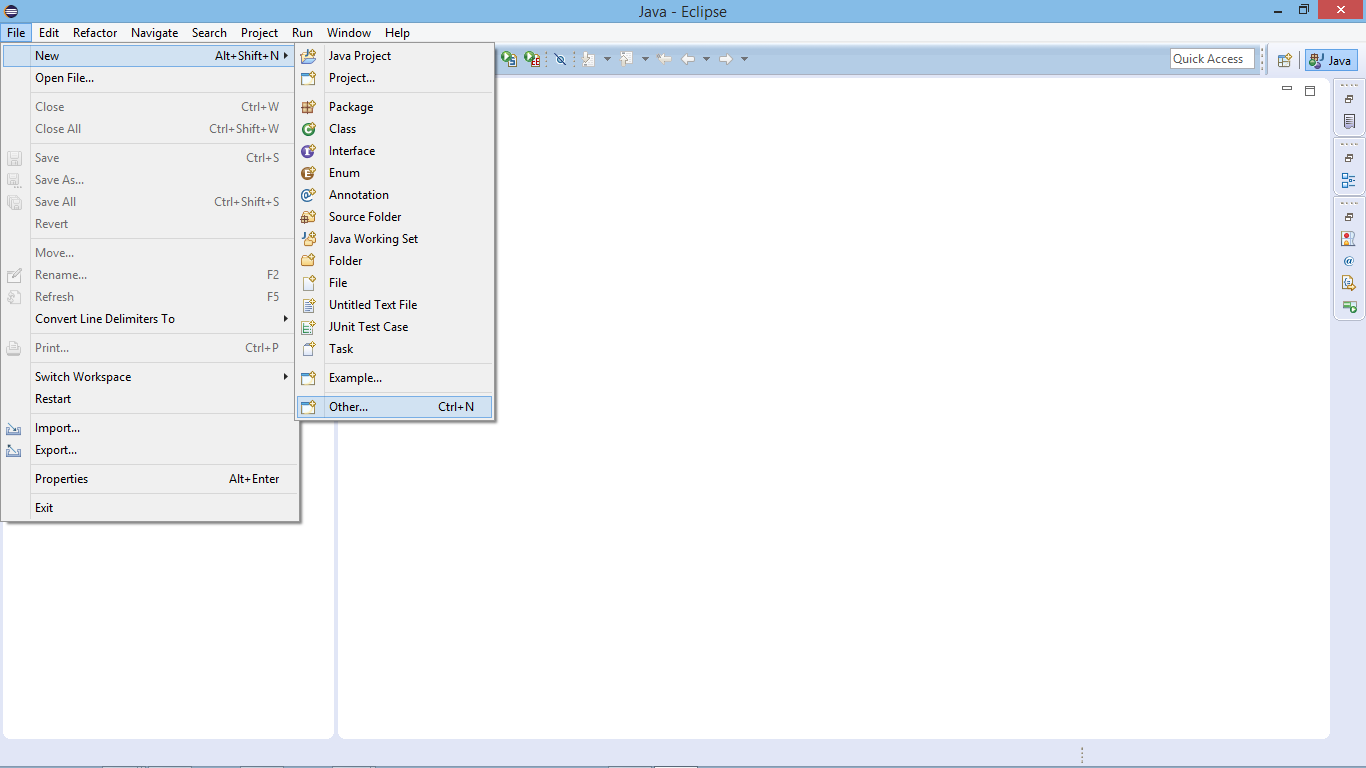
Open Eclipse Marketplace (**Help >> Eclipse Marketplace**) and search for “scala ide”. Now install the Scala IDE. Alternatively, you can download [Eclipse for Scala](http://scala-ide.org/).

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/install-scala-eclipse-plugin.png)

*Install Eclipse plugin for Scala*

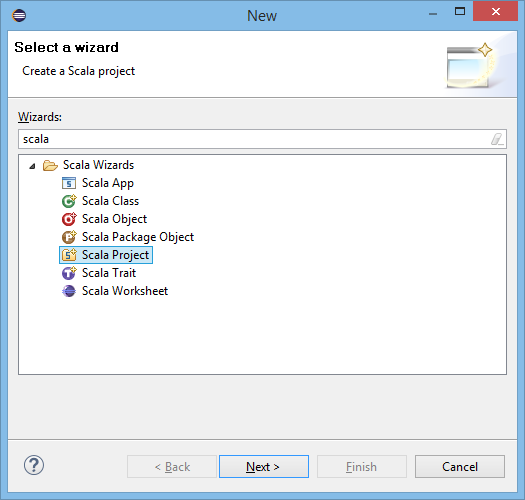
iii. Create a New Spark Scala Project

To create a new Spark Scala project, click on **File >> New >> Other**

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/create-new-spark-project-eclipse.png)

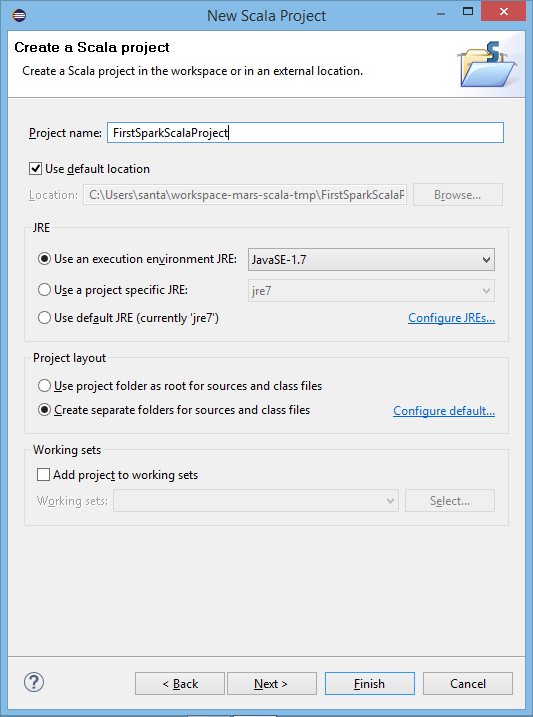
*Create a New Spark Scala Project*

Select Scala Project:

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/select-scala-project.png)

*Select Scala Project*

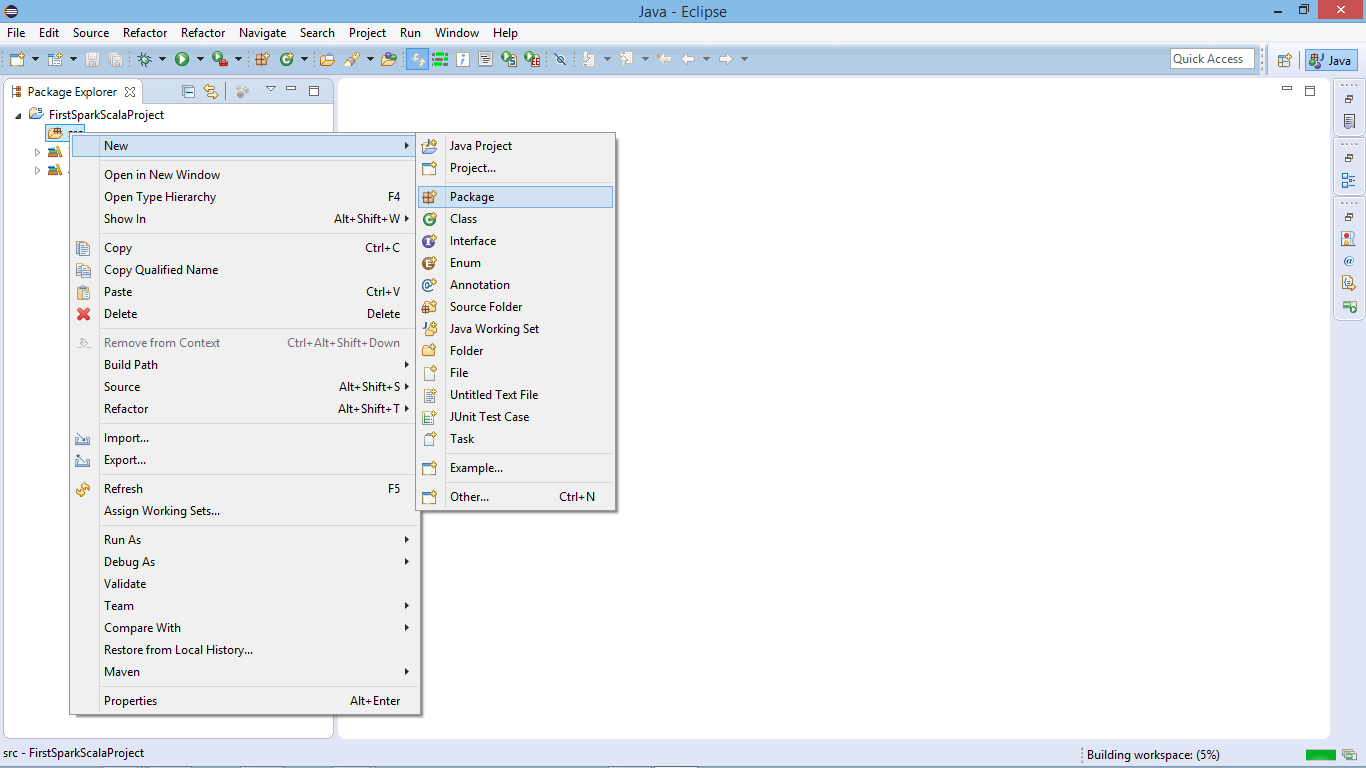
Supply Project Name:

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/supply-project-name-spark-scala.png)

*Supply Project Name*

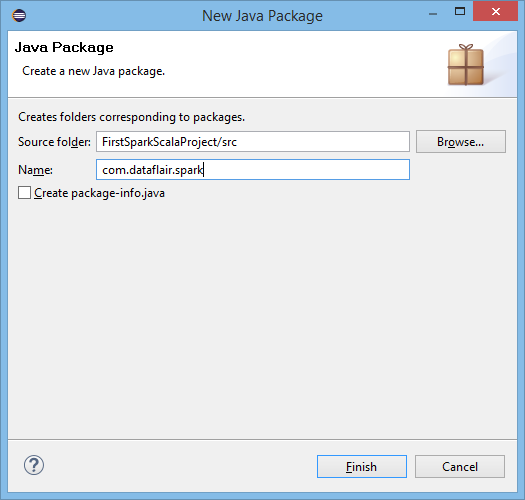
iv. Create New Package

After creating the project, now create a new package.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/create-new-package-spark-scala.png)

*Create New Package*

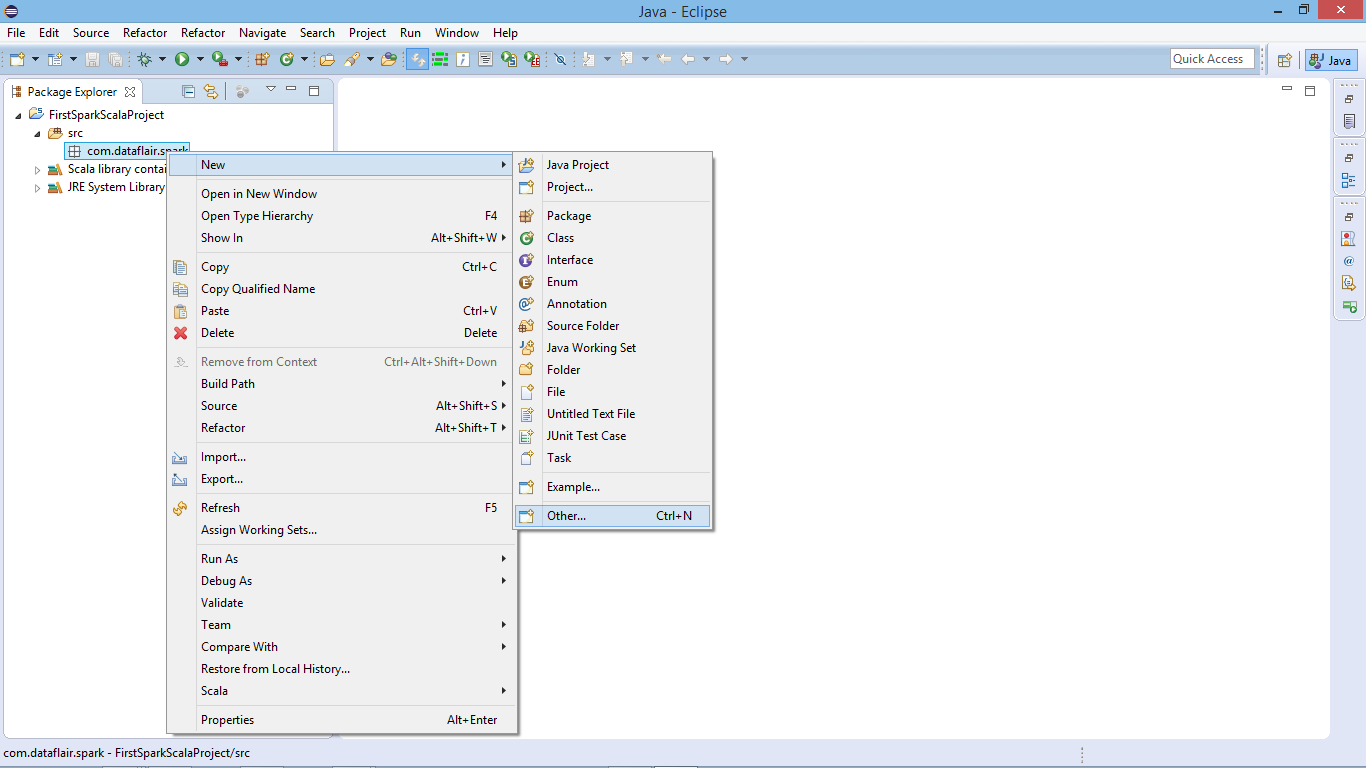
Supply Package Name:

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/supply-package-name-spark-scala.png)

*Supply Package Name*

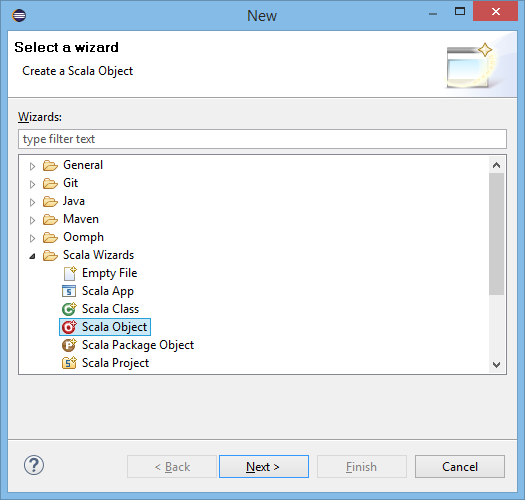
v. Create a New Scala Object

Now create a new Scala Object to develop Scala program for Spark application

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/create-new-scala-object-spark.png)

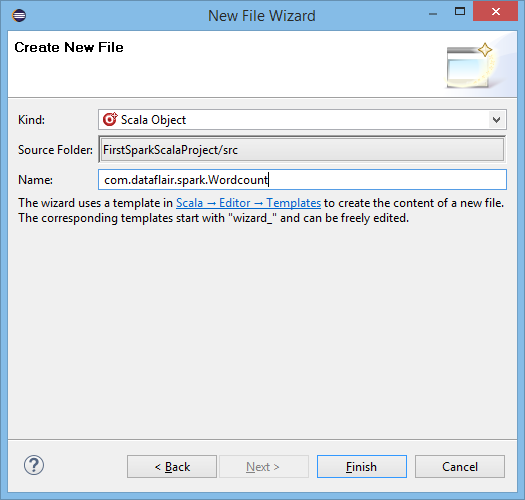
*Create a new Scala Object to develop Scala program for Spark application*

Select Scala Object:

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/select-scala-object.png)

*Select Scala Object*

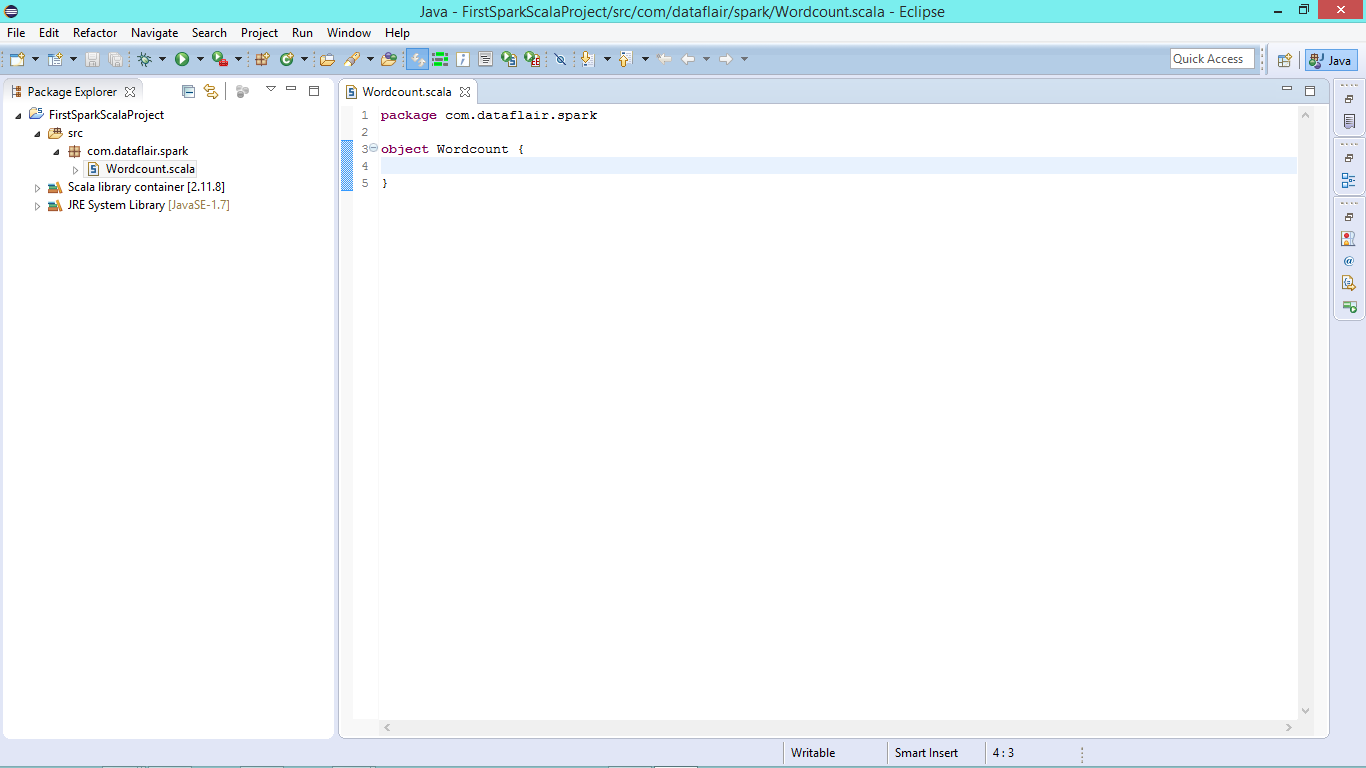
Supply Object Name:

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/supply-scala-object-name.png)

*Supply Object Name:*

vi. New Scala Object in Editor

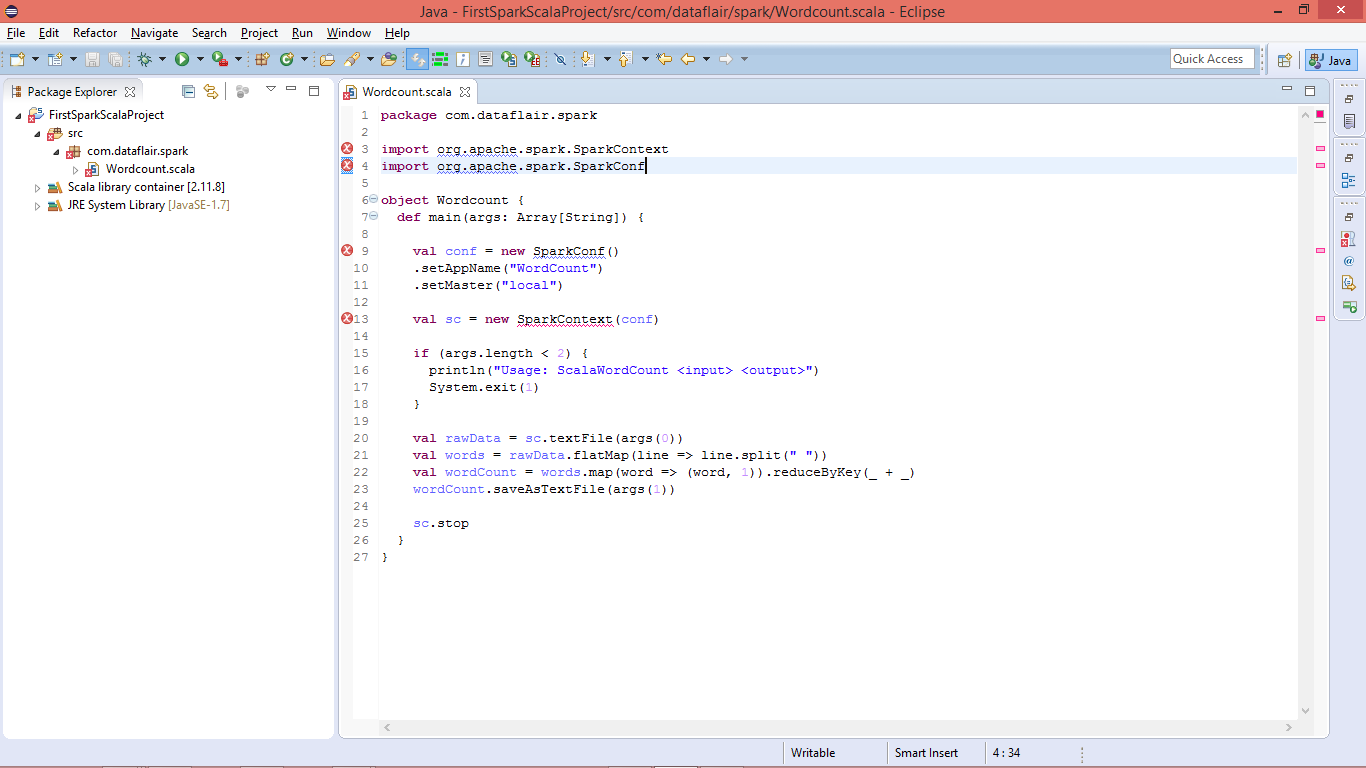
Scala object is ready now we can develop our Spark wordcount code in Scala-

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/new-scala-object.png)

*New Scala Object in Editor to create Spark Application*

vii. Copy below Spark Scala Wordcount Code in Editor

[php]  
package com.dataflair.spark  
import org.apache.spark.SparkContext  
import org.apache.spark.SparkConf  
object Wordcount {  
def main(args: Array[String]) {  
//Create conf object  
val conf = new SparkConf()  
.setAppName(“WordCount”)  
//create spark context object  
val sc = new SparkContext(conf)  
//Check whether sufficient params are supplied  
if (args.length < 2) {  
println(“Usage: ScalaWordCount <input> <output>”)  
System.exit(1)  
}  
//Read file and create RDD  
val rawData = sc.textFile(args(0))  
//convert the lines into words using flatMap operation  
val words = rawData.flatMap(line => line.split(” “))  
//count the individual words using map and reduceByKey operation  
val wordCount = words.map(word => (word, 1)).reduceByKey(\_ + \_)  
//Save the result  
wordCount.saveAsTextFile(args(1))  
//stop the spark context  
sc.stop  
}  
}[/php]

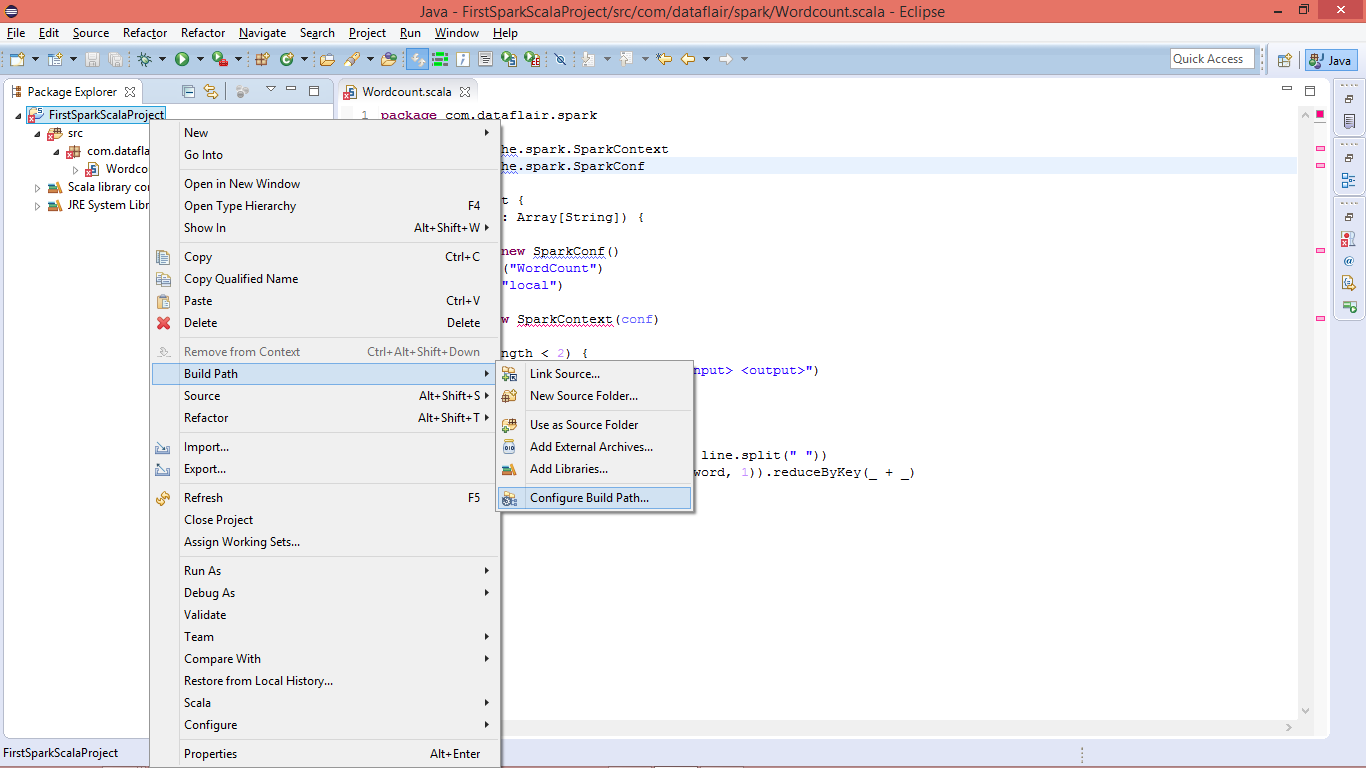
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/develop-spark-scala-code.png)

*Spark Scala WordCount Code in Editor*

*You will see lots of error due to missing libraries.*

viii. Add Spark Libraries

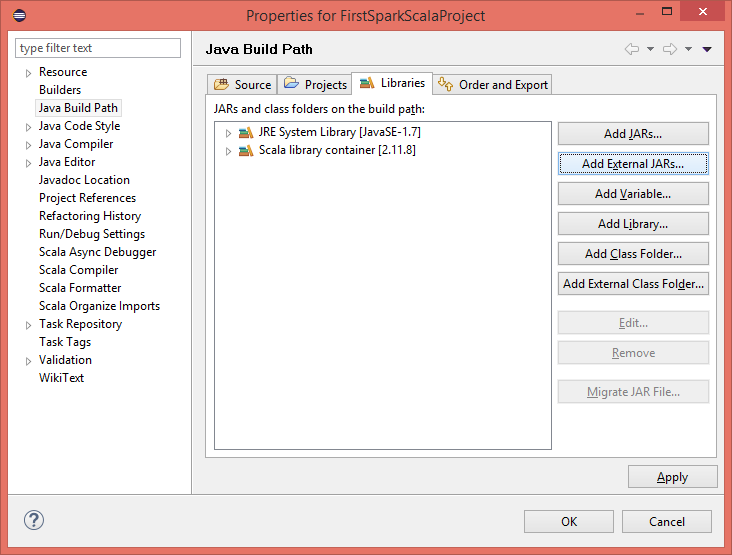
Configure Spark environment in Eclipse: Right click on project name >> build path >> Configure Build Path

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/add-spark-library-scala-project.png)

*Configure Spark environment in Eclipse*

Add the External Jars:

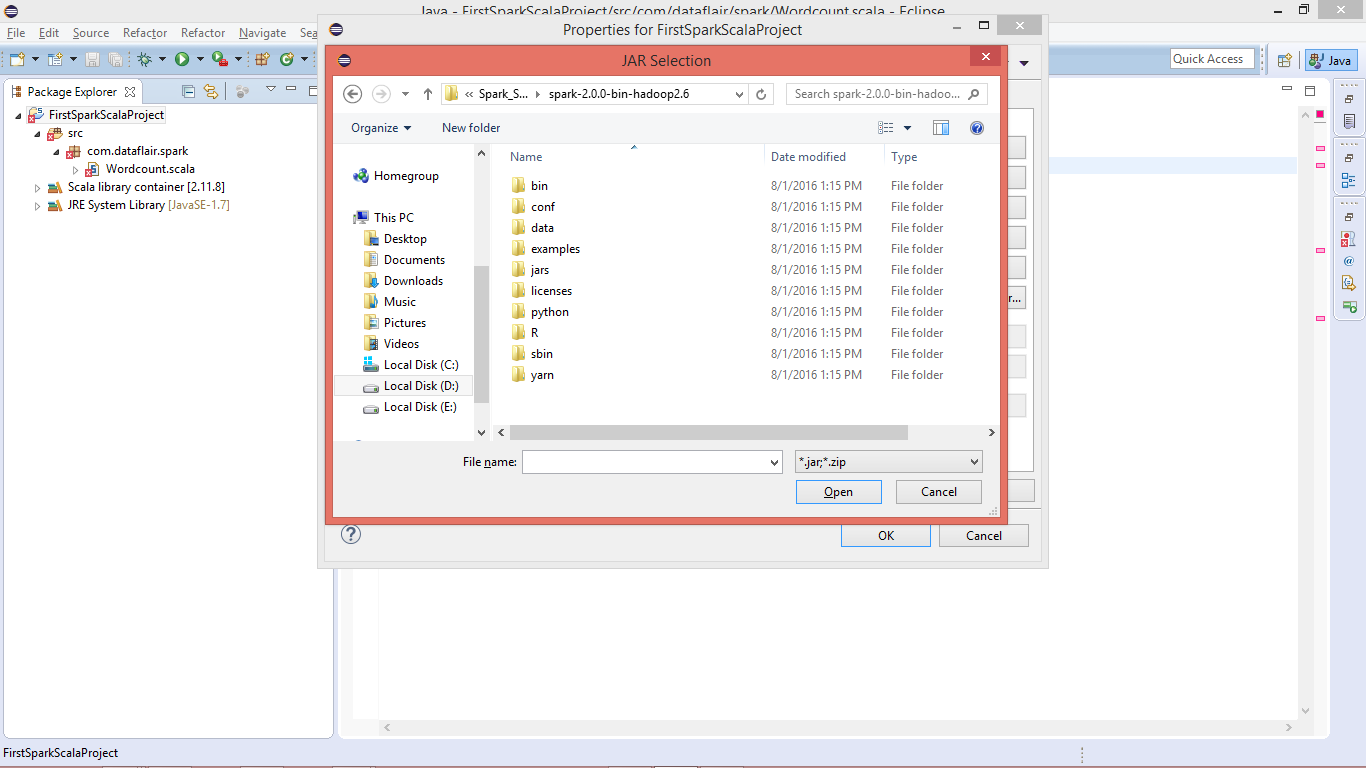
Skip Ad

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/library-add-external-jars.png)

*Add the External Jars*

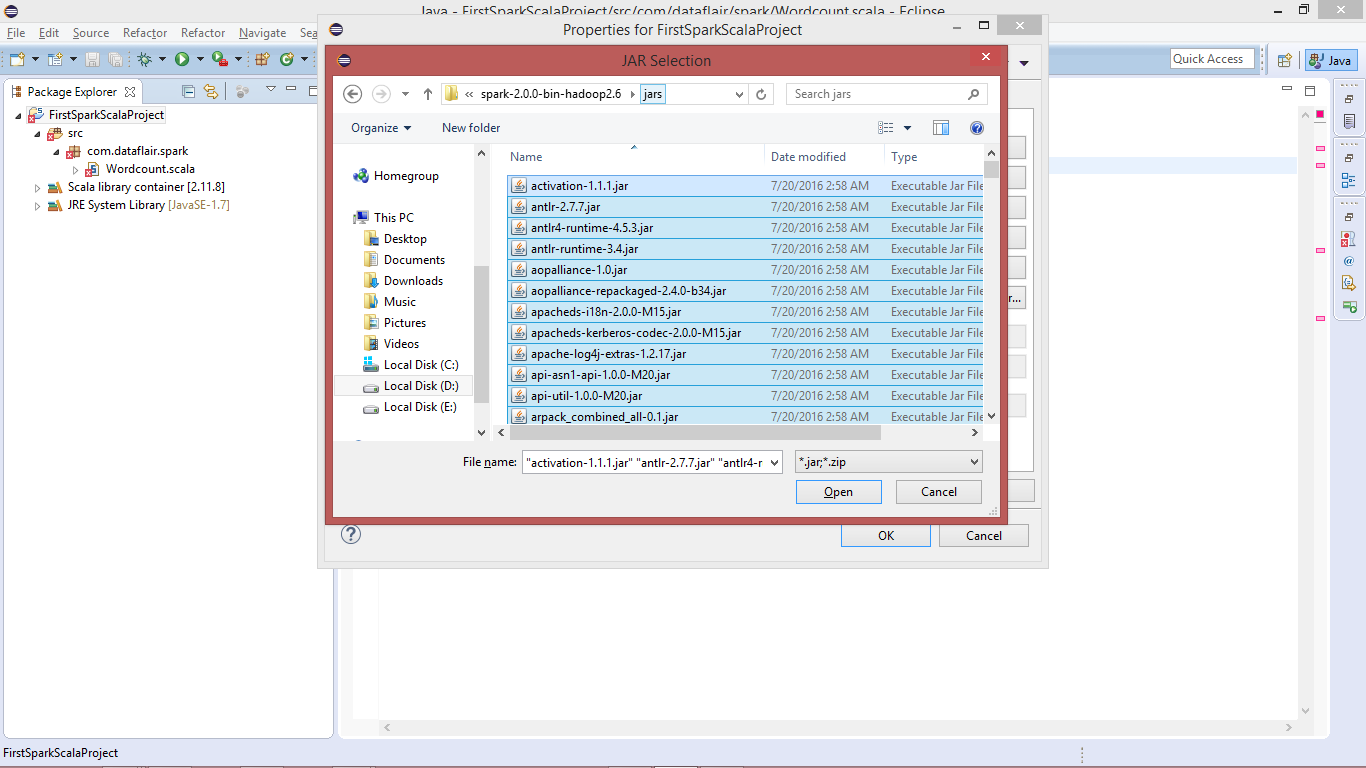
ix. Select Spark Jars and insert

You should have spark setup available in developing environment, it will be needed for spark libraries.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/select-spark-jars.png)

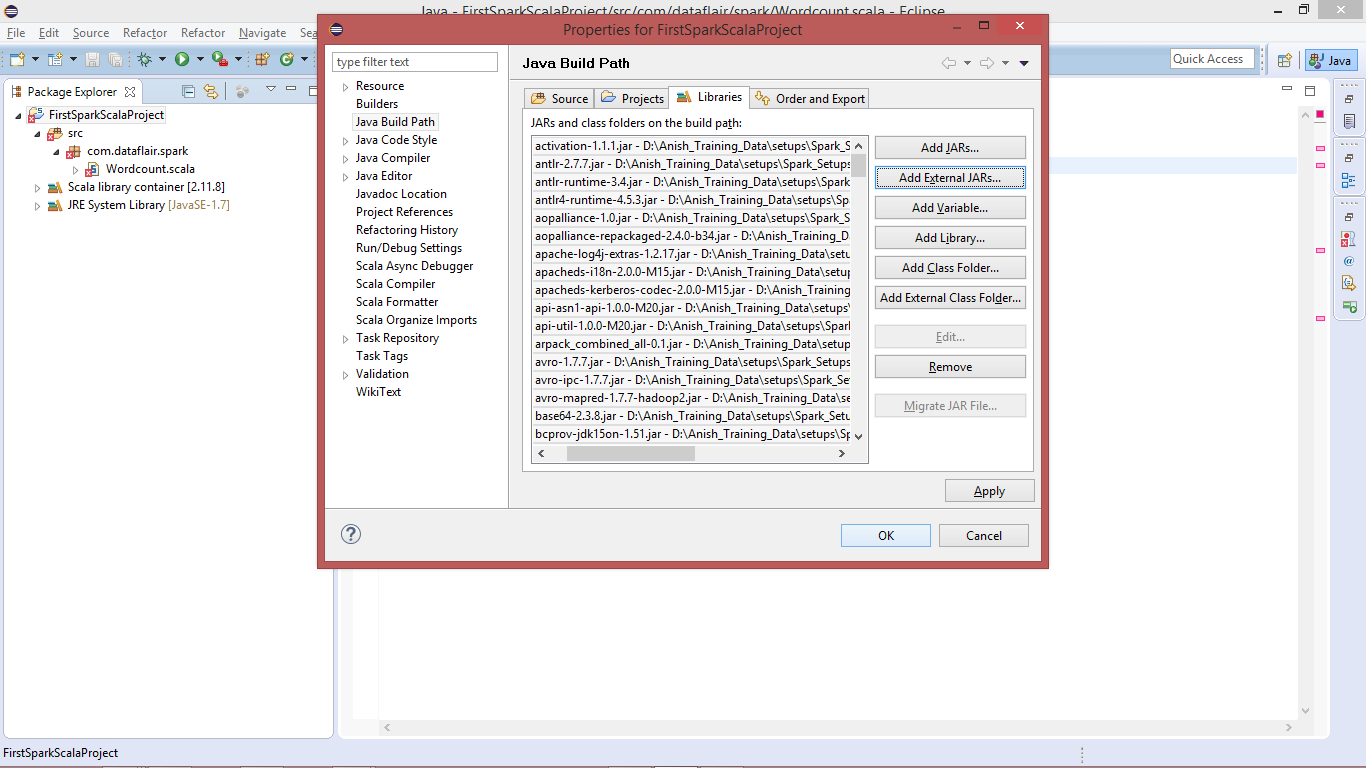
*Select Spark Jars and insert*

Go to “**Spark-Home >> jars**” and select all the jars:

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/select-spark-jars-select.png)

*select all the jars*

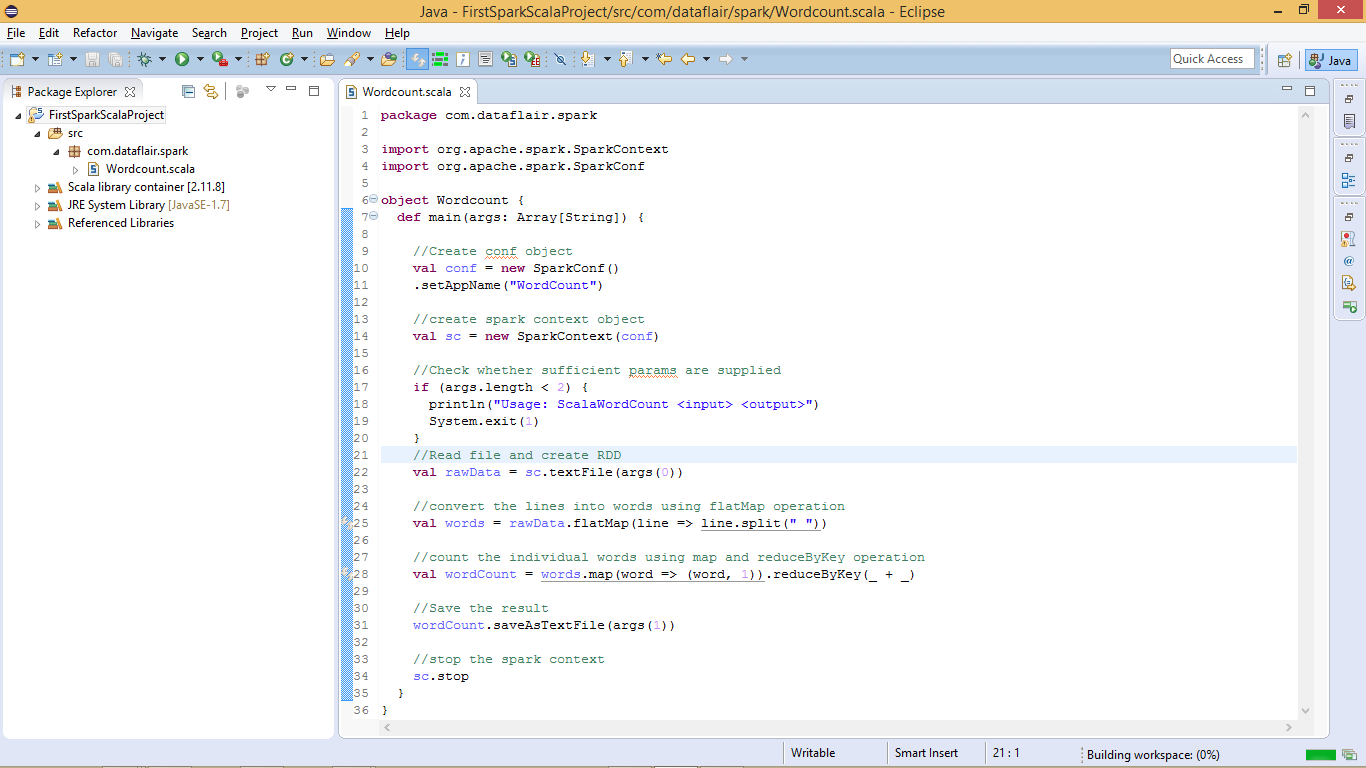
Import the selected jar:

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/select-spark-jars-add-build-path.png)

*Import the selected jar*

x. Spark Scala Word Count Program

After importing the libraries all the errors will be removed.

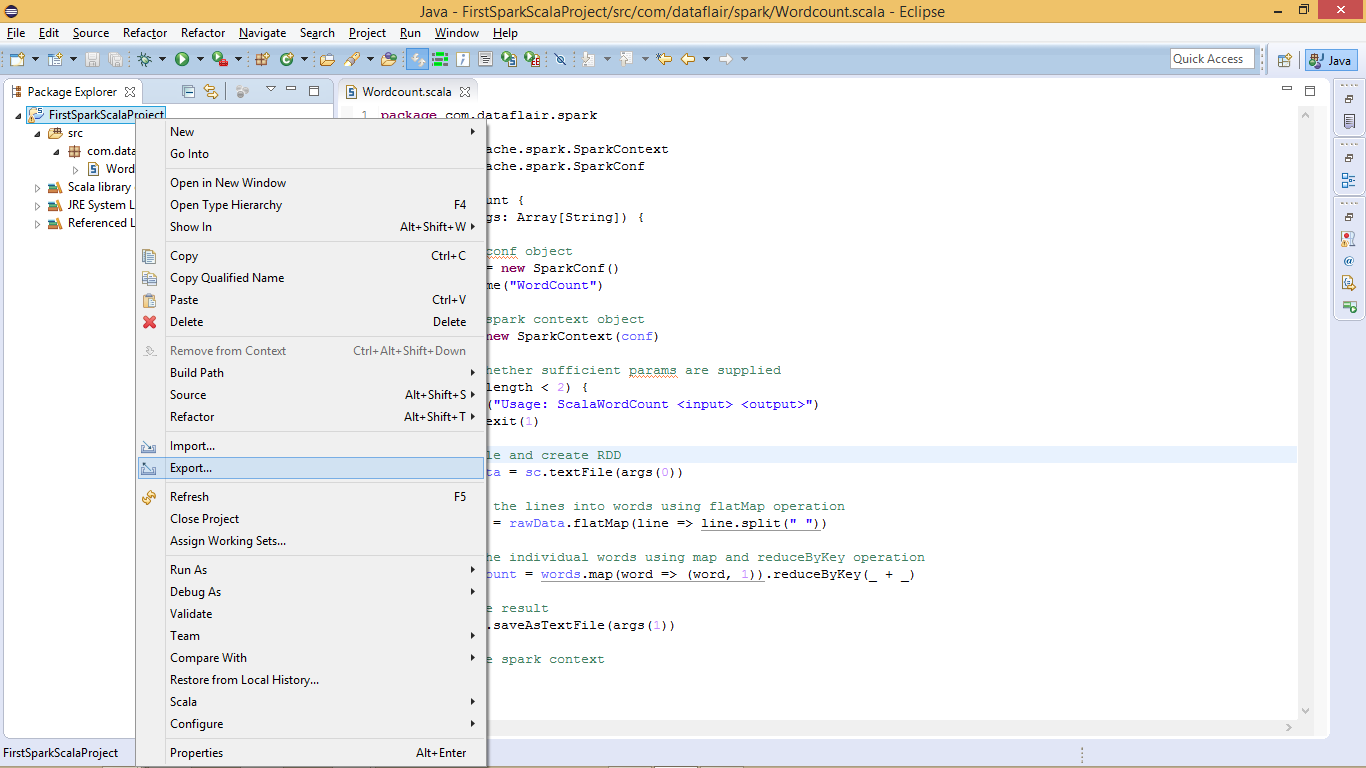
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/wordcount-program-spark-scala.png)

*Spark WordCount Program in Scala*

We have successfully created Spark environment in Eclipse and developed Spark Scala program. Now let’s deploy the Spark job on [Linux](http://data-flair.training/blogs/frequent-linux-commands-beginners-tutorial/), before deploying/running the application you must have Spark Installed.  
Follow this links to[install Apache Spark on single node cluster](http://data-flair.training/blogs/apache-spark-installation-in-standalone-mode/) or on the [multi-node cluster.](http://data-flair.training/blogs/apache-spark-installation-on-multi-node-cluster-step-by-step-guide/)

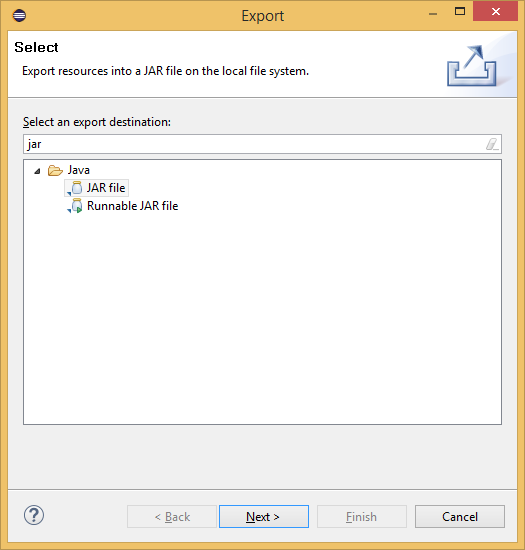
xi. Create the Spark Scala Program Jar File

Before running created **Spark word count application** we have to create a jar file. Right click on **project >> export**

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/export-project-spark-scala.png)

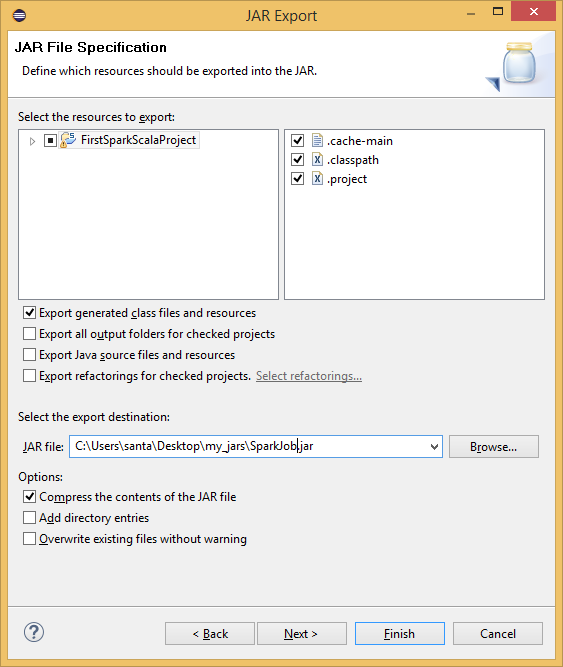
*Create the Spark Scala Program Jar File*

Select Jar-file Option to Export:

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/select-jar-file-to-export.png)

*Select Jar-file Option to Export*

Create the Jar file:

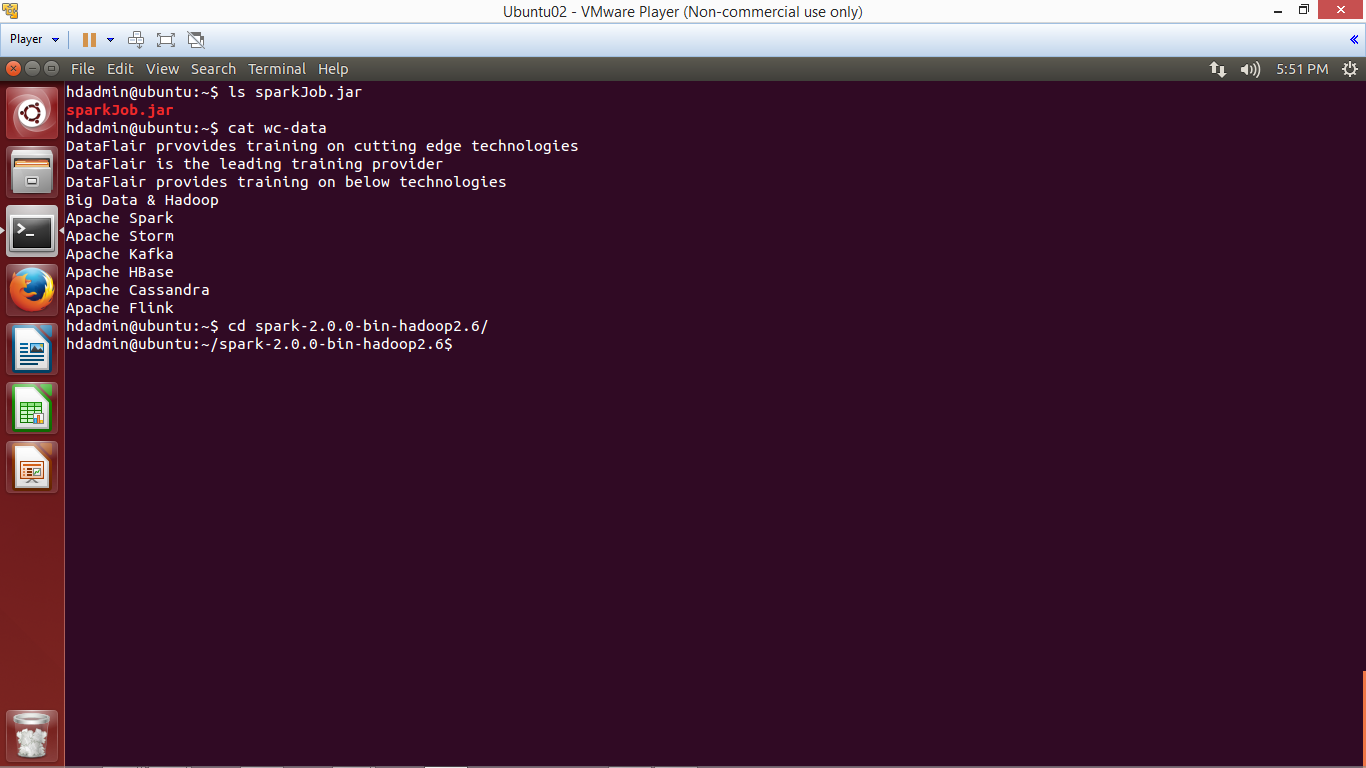
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/create-spark-wordcount-jar-scala.png)

*Create the Jar file*

The jar file for the Spark Scala application has been created, now we need to run it.

xii. Go to Spark Home Directory

Login to [Linux](http://data-flair.training/blogs/frequently-used-important-linux-commands-tutorial/) and open terminal. To run Spark Scala application we will be using Ubuntu Linux. Copy the jar file to Ubuntu and create one text file, which we will use as input for Spark Scala wordcount job.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/cd-spark-home-directory.png)

*cd spark home directory*

xiii. Submit Spark Application using spark-submit script

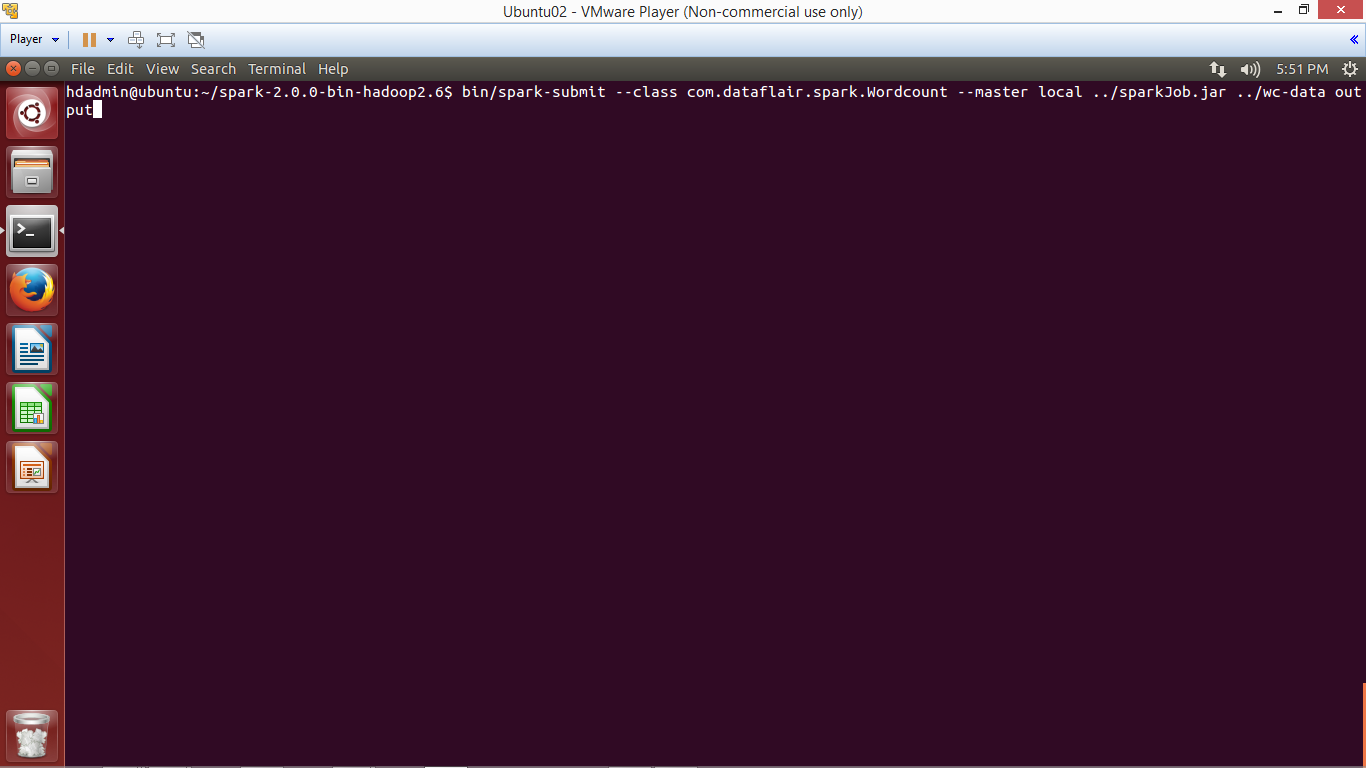
To submit the Spark application using below command:

bin/spark-submit --**class** <Qualified-**Class**-Name> --master <Master> <Path-**Of**-Jar-File> <Input-Path> <Output-Path>

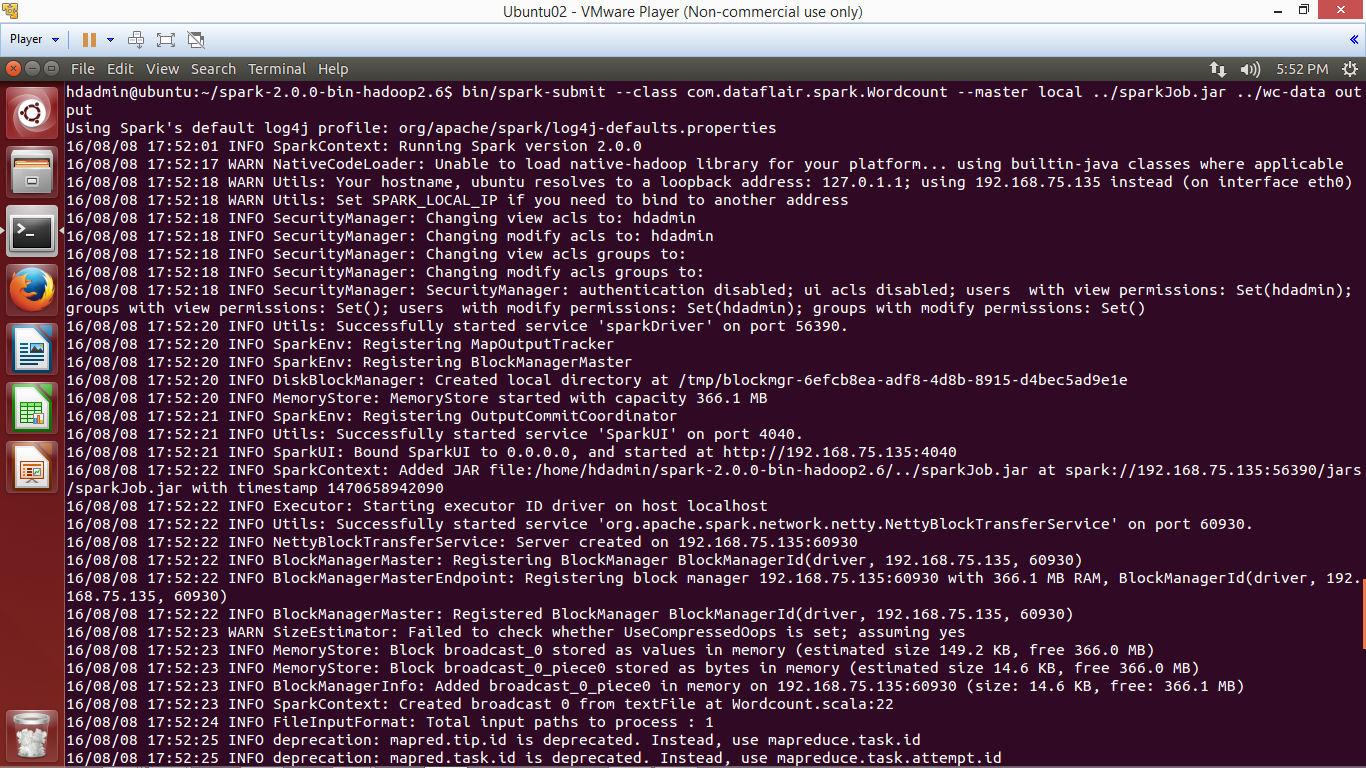
bin/spark-submit --**class** com.dataflair.spark.Wordcount --master local ../sparkJob.jar ../wc-data output

Let’s understand above command:

* **bin/spark-submit:** To submit Spark Application
* **–class:** To specify the class name to execute
* **–master:** Master (local / <Spark-URI> / yarn)
* **<Jar-Path>:** The jar file of application
* **<Input-Path>:** Location from where input data will be read
* **<Output-Path>:** Location where Spark application will write output

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/spark-scala-jar-submit-run-command.png)

*Submit Spark Application using spark-submit script*

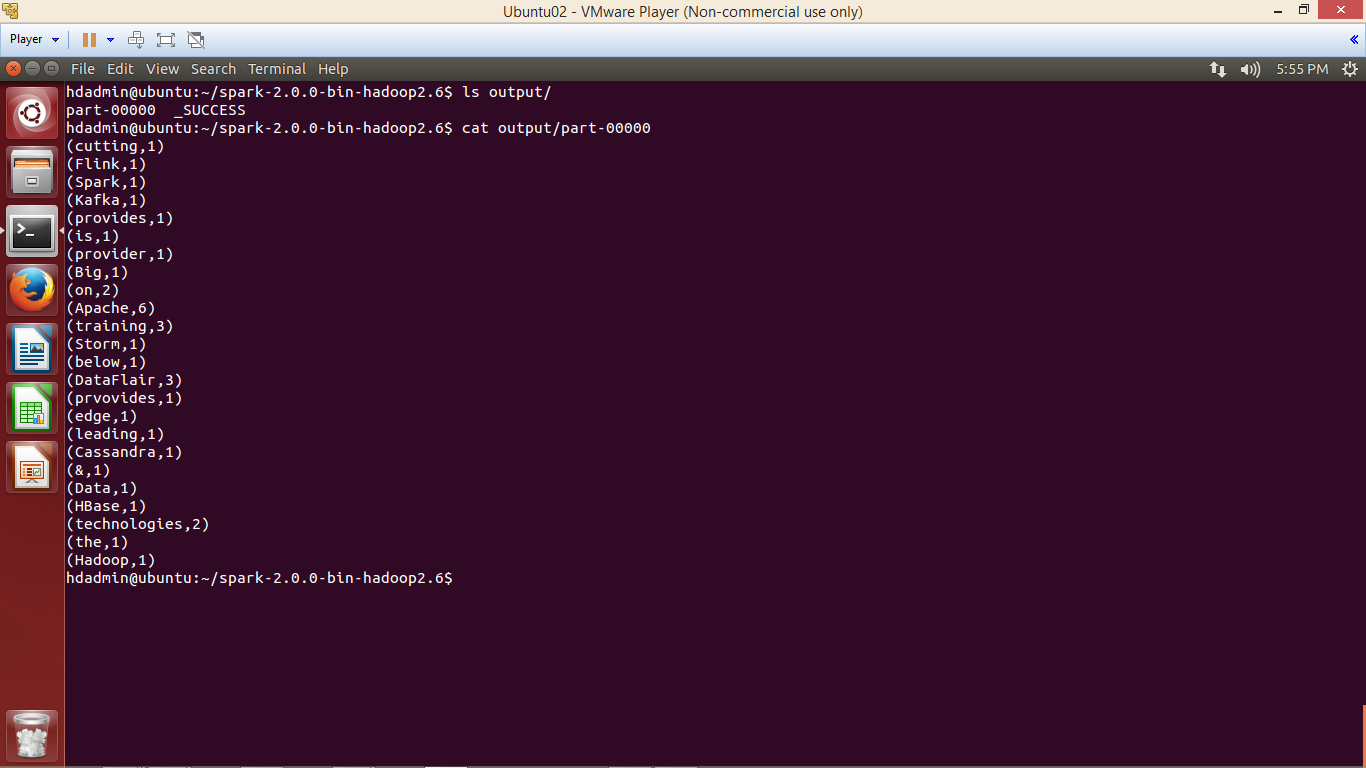
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/spark-wordcount-job-run.png)

*Submit Spark Application using spark-submit script*

*The application has been completed successfully, now browse the result.*

xiv. Browse the result

Browse the output directory and open the file with name part-xxxxx which contains the output of the application.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2016/08/spark-wordcount-job-success.png)

*spark wordcount job success*

We have successfully created Spark project in Scala and deployed on Ubuntu.  
To play with Spark First learn [**RDD**](http://data-flair.training/blogs/apache-spark-rdd-tutorial/),[**DataFrame**](http://data-flair.training/blogs/apache-spark-sql-dataframe-tutorial/), [**DataSet in Apache Spark**](http://data-flair.training/blogs/apache-spark-dataset-tutorial/)and then refer this [**Spark shell commands tutorial**](http://data-flair.training/blogs/apache-spark-shell-commands-beginners-tutorial/) to practically implements Spark functionalities.  
**See Also-**

* [Spark Transformation and Action APIs](http://data-flair.training/blogs/rdd-transformations-actions-apis-apache-spark/)
* [How does Apache Spark work?](http://data-flair.training/blogs/how-apache-spark-works-run-time-spark-architecture/)

**Did you like this article? If Yes, please give DataFlair 5 Stars on**[**Google**](https://g.page/DataFlair/review?kd)**|**[**Facebook**](https://www.facebook.com/DataFlairWS/reviews/)