# Create Spark project in Scala with Eclipse without Maven

**Contents**

* [1. Objective](https://data-flair.training/blogs/create-spark-scala-project/#1_Objective)
* [2. Steps to create Spark project in Scala](https://data-flair.training/blogs/create-spark-scala-project/#2_Steps_to_createSpark_project_in_Scala)
  + [2.1. Platform Used / Required](https://data-flair.training/blogs/create-spark-scala-project/#21_Platform_Used_Required)
  + [2.2. Install Eclipse plugin for Scala](https://data-flair.training/blogs/create-spark-scala-project/#22_Install_Eclipse_plugin_for_Scala)
  + [2.3. Create a New Spark Scala Project](https://data-flair.training/blogs/create-spark-scala-project/#23_Create_a_New_Spark_Scala_Project)
  + [2.4. Create New Package](https://data-flair.training/blogs/create-spark-scala-project/#24_Create_New_Package)
  + [2.5. Create a New Scala Object](https://data-flair.training/blogs/create-spark-scala-project/#25_Create_a_New_Scala_Object)
  + [2.6. New Scala Object in Editor](https://data-flair.training/blogs/create-spark-scala-project/#26_New_Scala_Object_in_Editor)
  + [2.7. Copy below Spark Scala Wordcount Code in Editor](https://data-flair.training/blogs/create-spark-scala-project/#27_Copy_below_Spark_Scala_WordcountCode_in_Editor)
  + [2.8. Add Spark Libraries](https://data-flair.training/blogs/create-spark-scala-project/#28_Add_Spark_Libraries)
  + [2.9. Select Spark Jars and insert](https://data-flair.training/blogs/create-spark-scala-project/#29_Select_Spark_Jars_and_insert)
  + [2.10. Spark Scala Word Count Program](https://data-flair.training/blogs/create-spark-scala-project/#210_Spark_Scala_Word_Count_Program)
  + [2.11. Create the Spark Scala Program Jar File](https://data-flair.training/blogs/create-spark-scala-project/#211_Create_the_Spark_Scala_Program_Jar_File)
  + [2.12. Go to Spark Home Directory](https://data-flair.training/blogs/create-spark-scala-project/#212_Go_to_Spark_Home_Directory)
  + [2.13. Submit Spark Application using spark-submit script](https://data-flair.training/blogs/create-spark-scala-project/#213_Submit_Spark_Application_using_spark-submit_script)
  + [2.14. Browse the result](https://data-flair.training/blogs/create-spark-scala-project/#214_Browse_the_result)

**1. Objective**

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/)

**2. Steps to create Spark project in Scala**

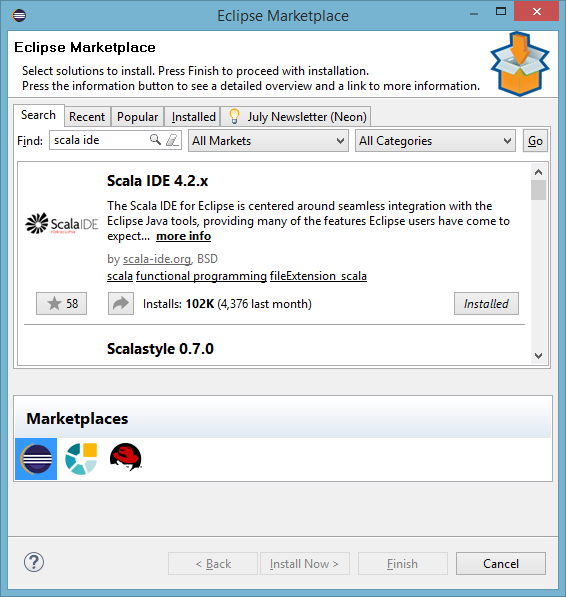
To create Spark Project in Scala with Eclipse without Maven follow the steps given below-

**2.1. Platform Used / Required**

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

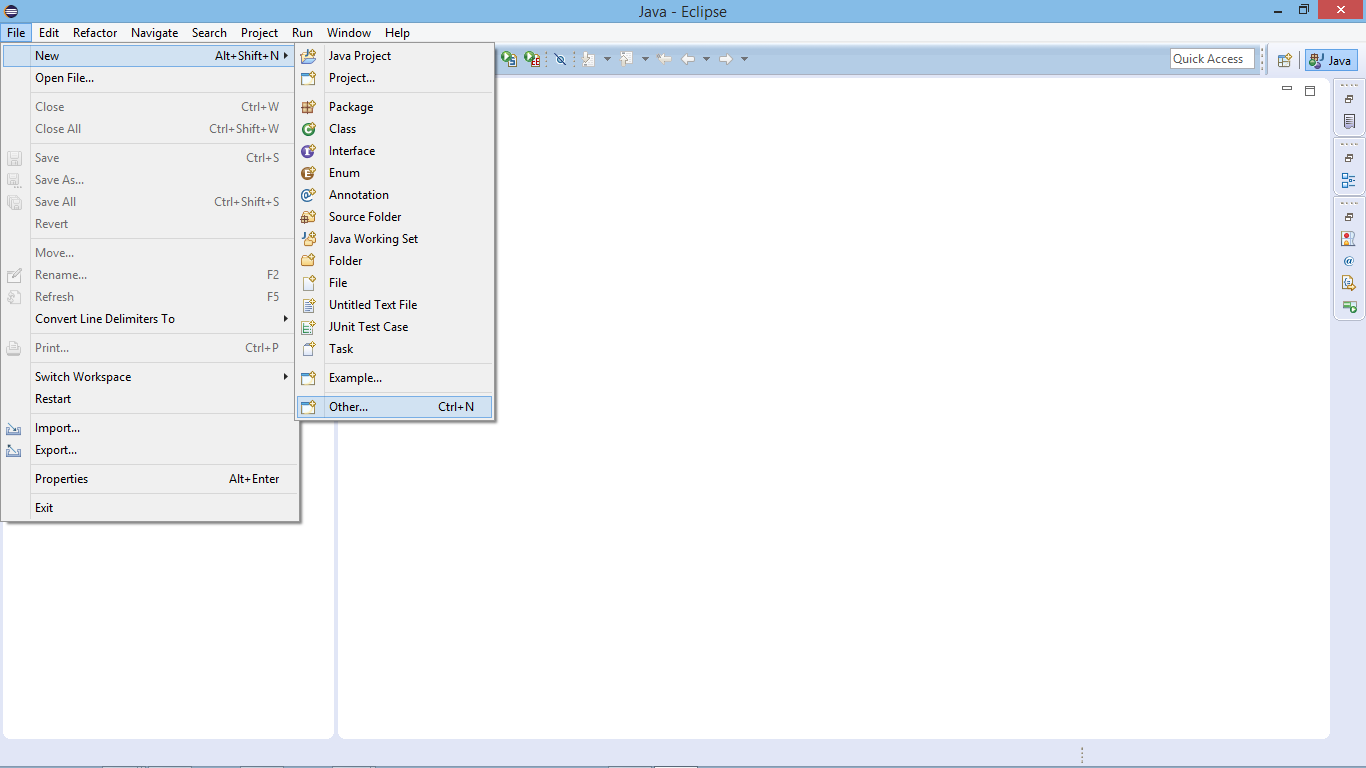
**2.2. Install Eclipse plugin for Scala**

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/).

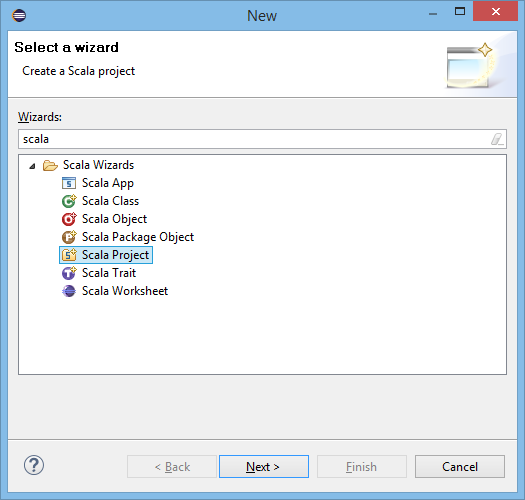
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/install-scala-eclipse-plugin.png)

**2.3. Create a New Spark Scala Project**

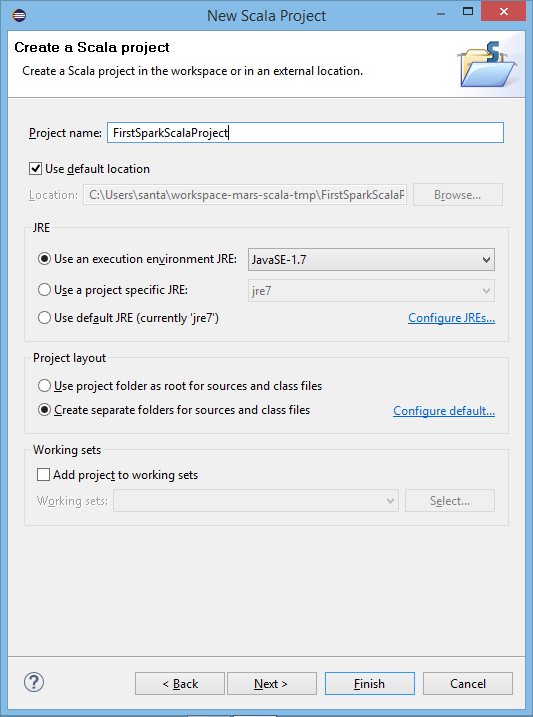
To create a new Spark Scala project, click on **File >> New >> Other**

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

Select Scala Project:

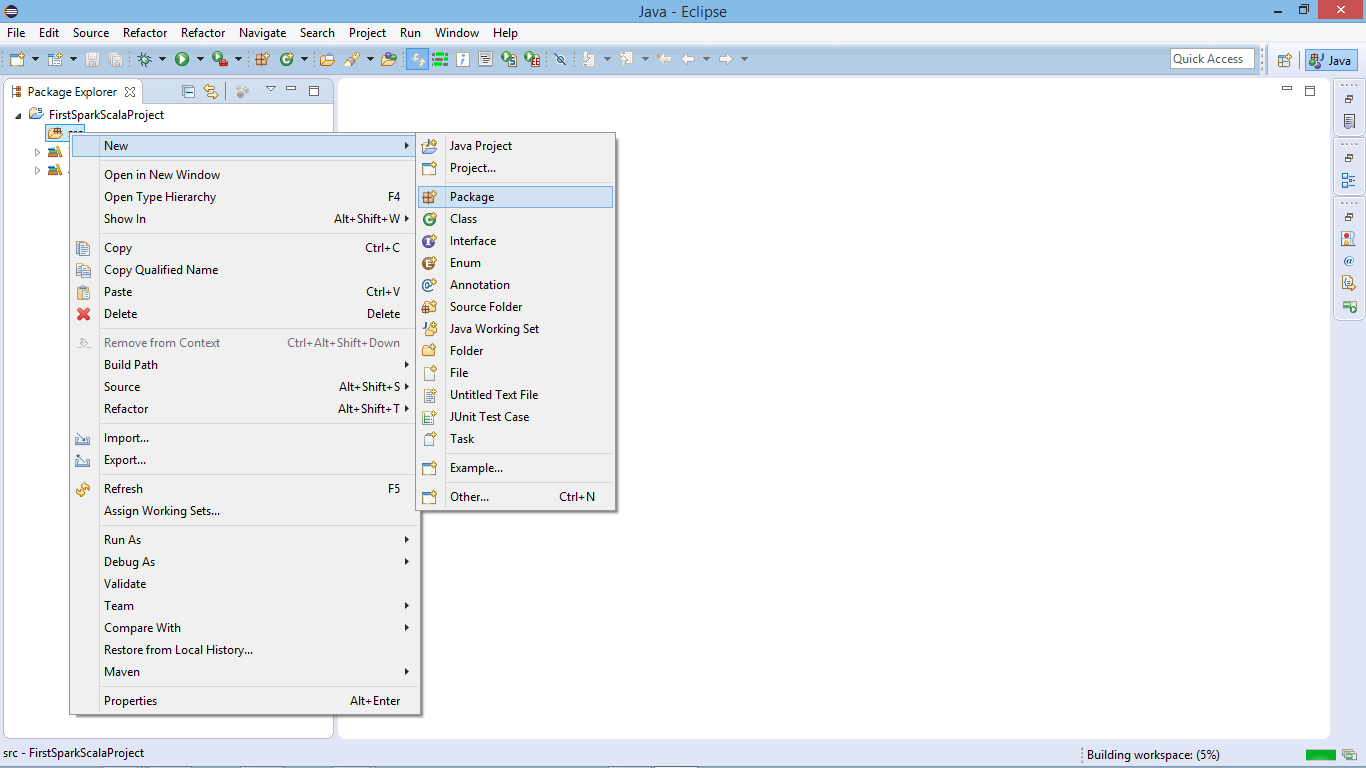
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/select-scala-project.png)

Supply Project Name:

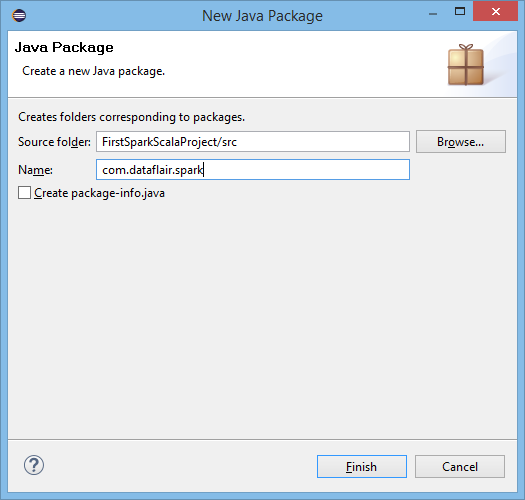
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/supply-project-name-spark-scala.png)

**2.4. Create New Package**

After creating the project, now create a new package.

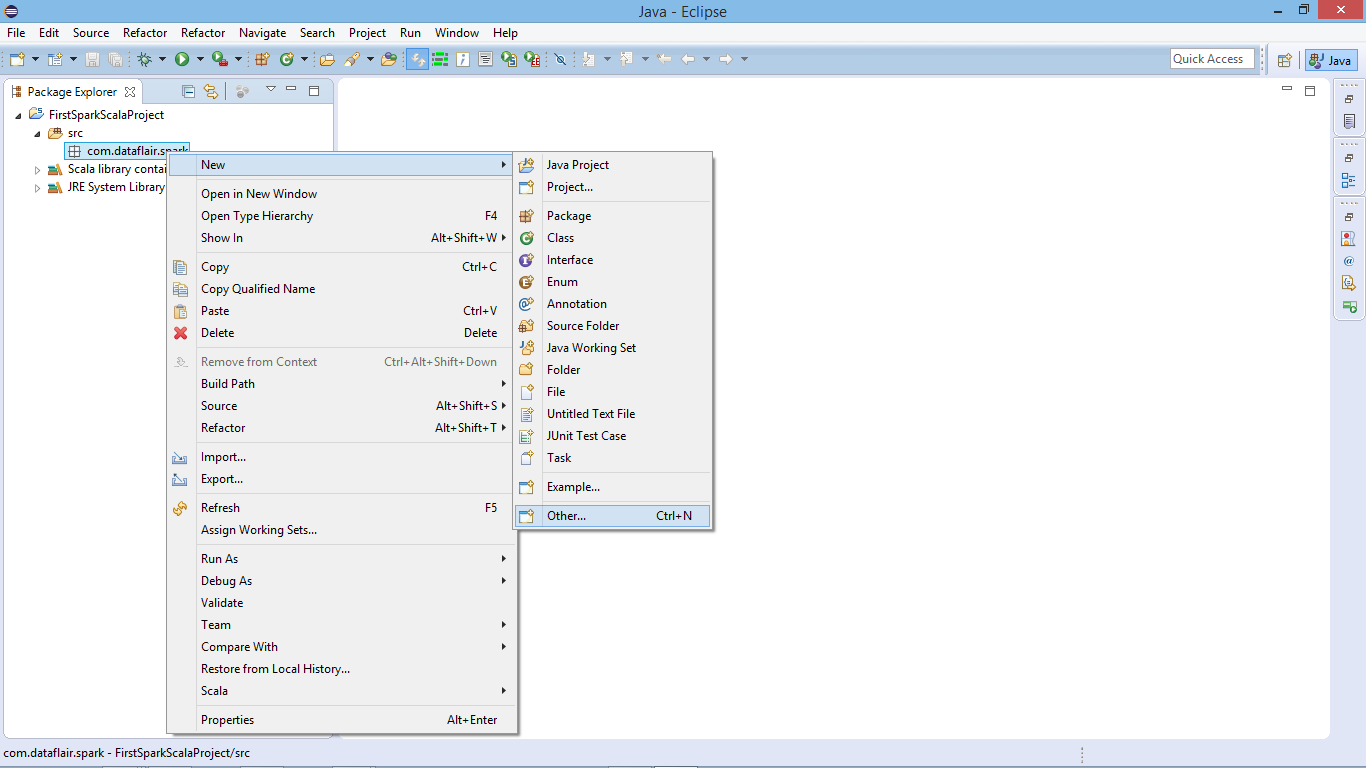
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/create-new-package-spark-scala.png)

Supply Package Name:

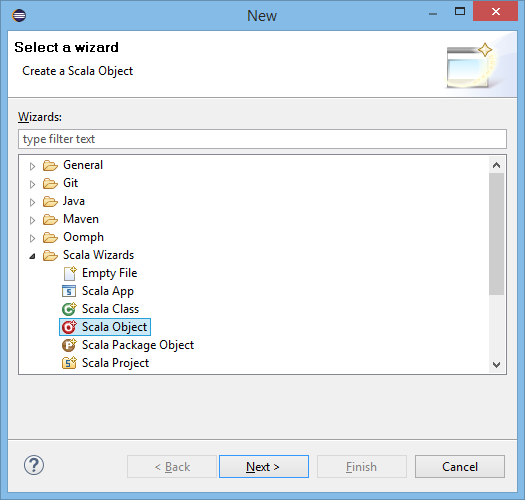
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/supply-package-name-spark-scala.png)

**2.5. Create a New Scala Object**

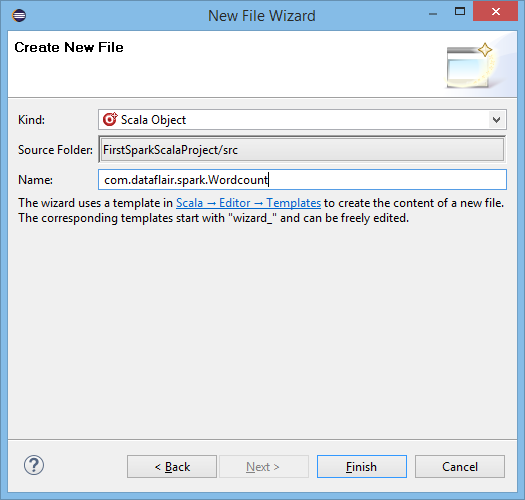
Now create a new Scala Object to develop Scala program for Spark application

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

Select Scala Object:

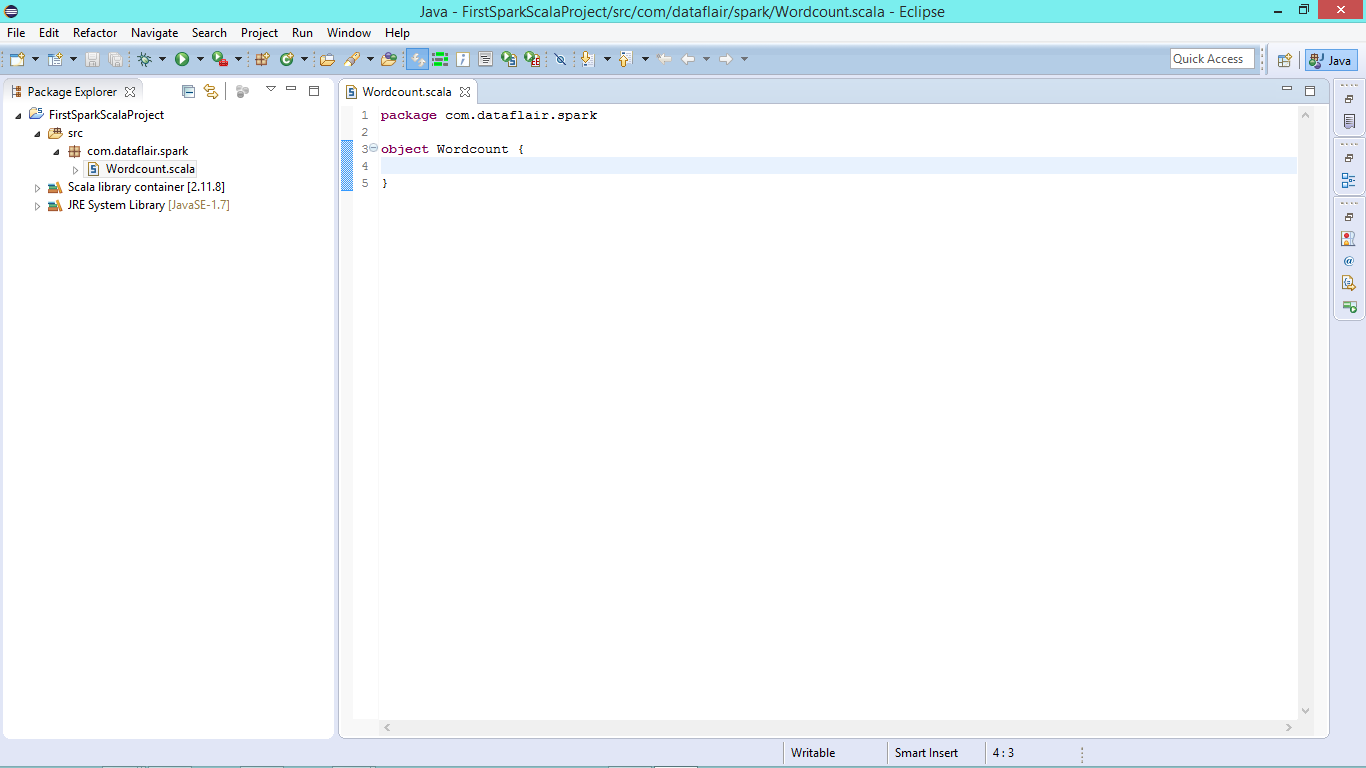
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/select-scala-object.png)

Supply Object Name:

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

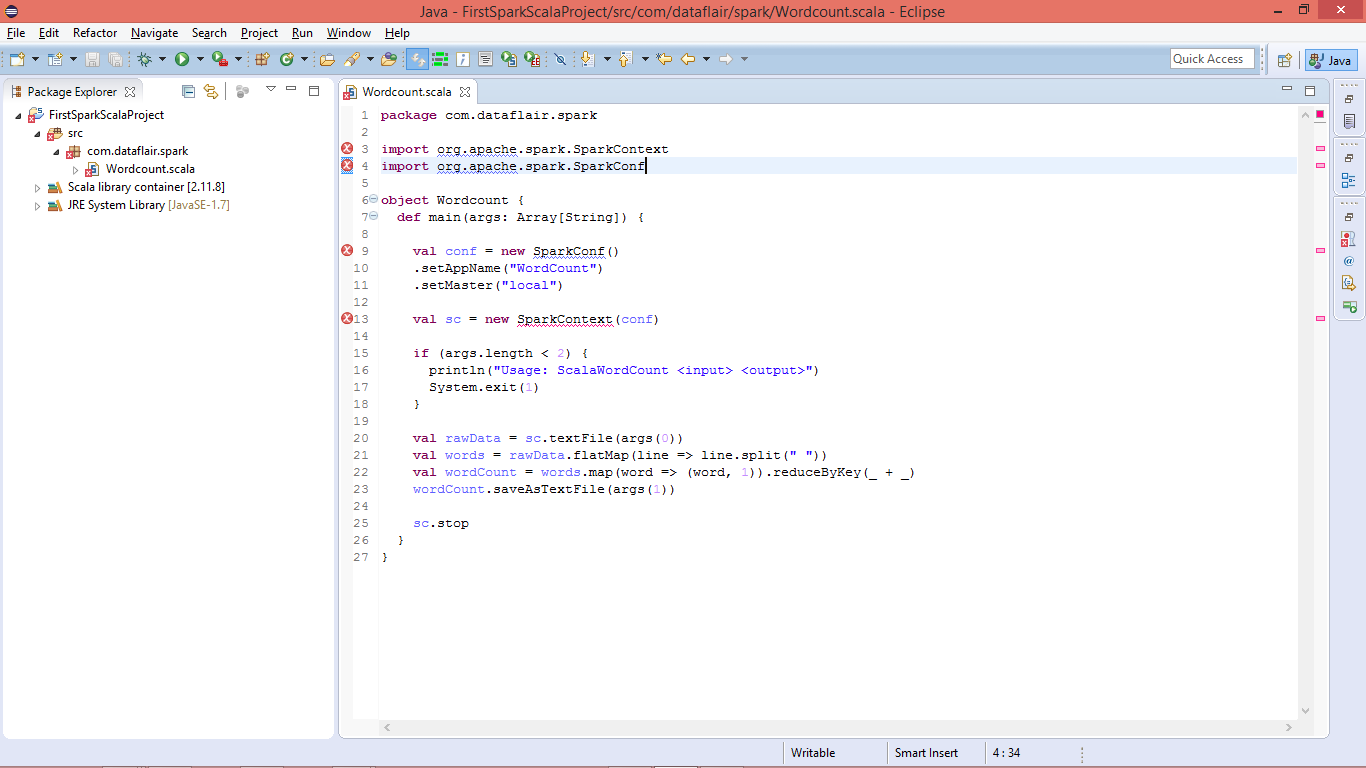
**2.6. New Scala Object in Editor**

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

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

**2.7. Copy below Spark Scala Wordcount Code in Editor**

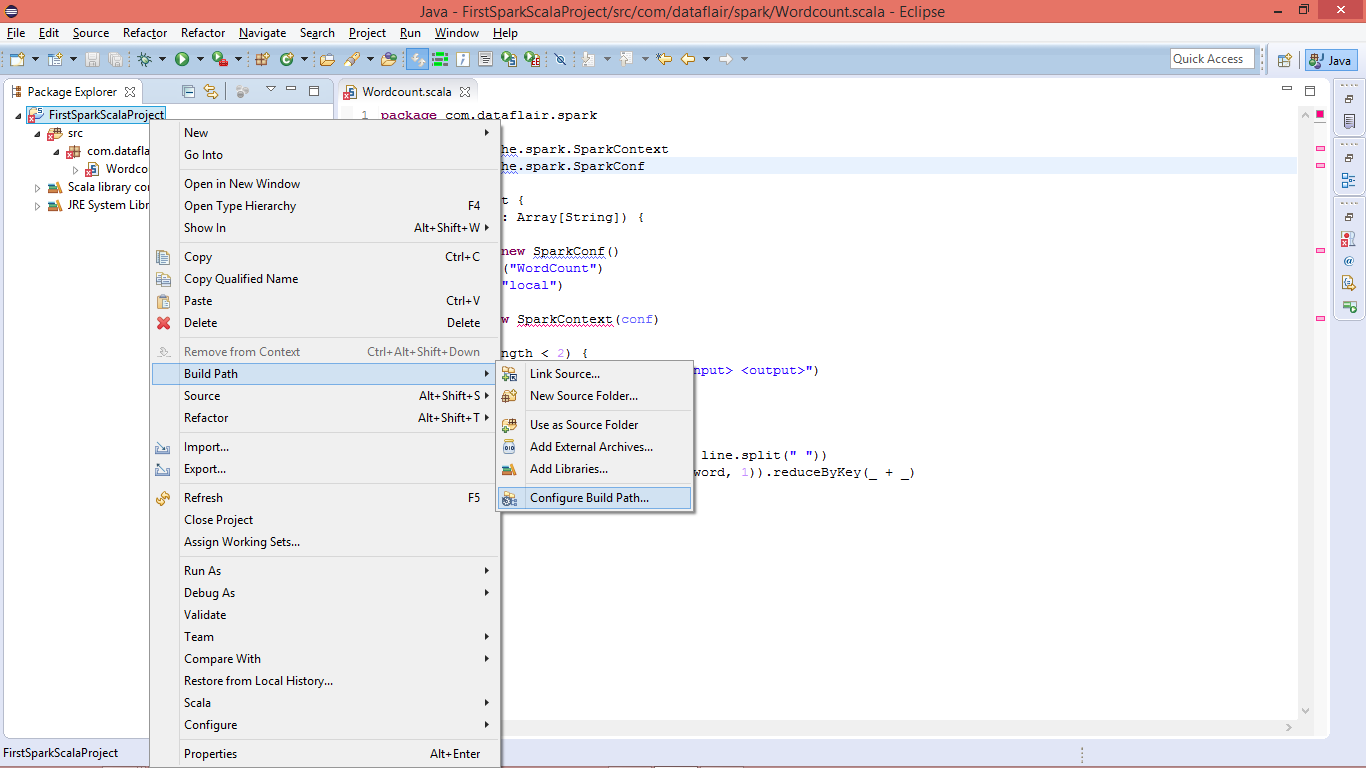
|  |
| --- |
| package com.dataflair.spark    import org.apache.spark.SparkContext  import org.apache.spark.SparkConf    object Wordcount {   def main(args: Array[String]) {     //Create conf object   val conf = new SparkConf()   .setAppName("WordCount")     //create spark context object   val sc = new 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   }  } |

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

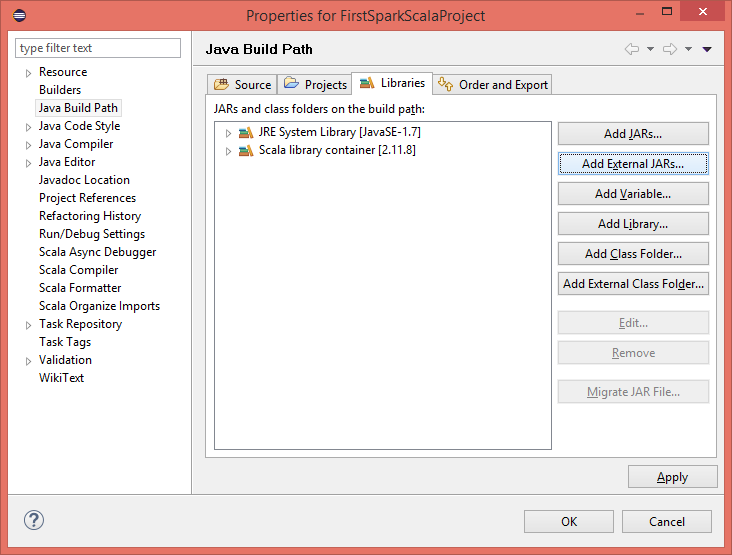
*You will see lots of error due to missing libraries.*

**2.8. Add Spark Libraries**

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

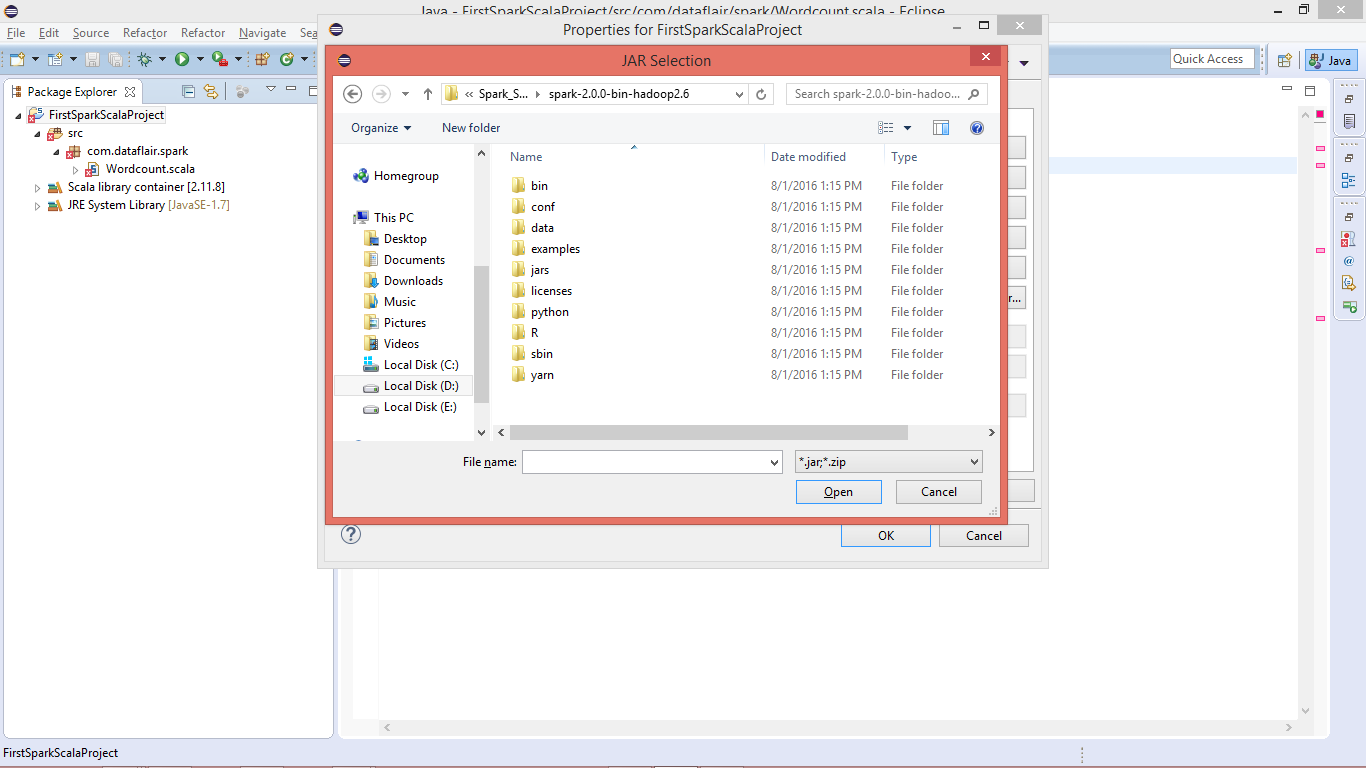
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/add-spark-library-scala-project.png)

Add the External Jars:

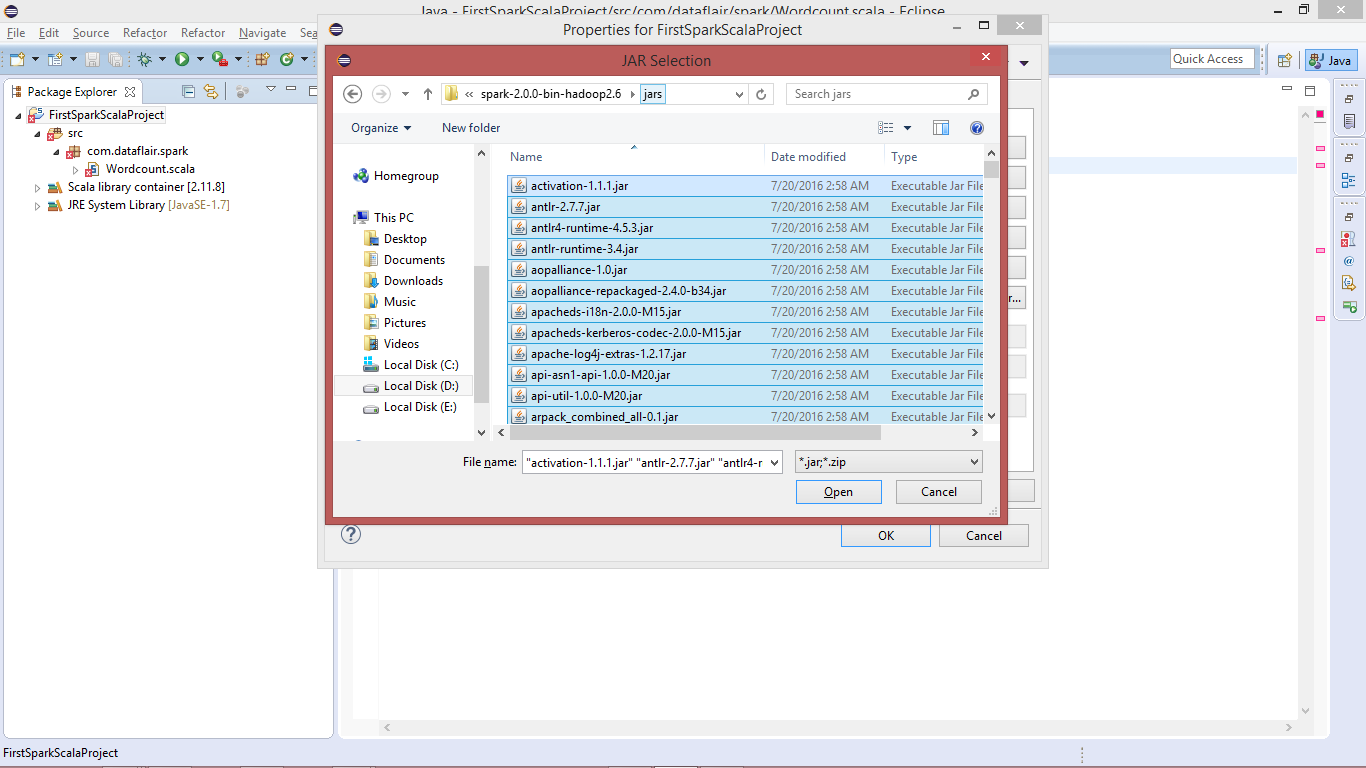
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/library-add-external-jars.png)

**2.9. Select Spark Jars and insert**

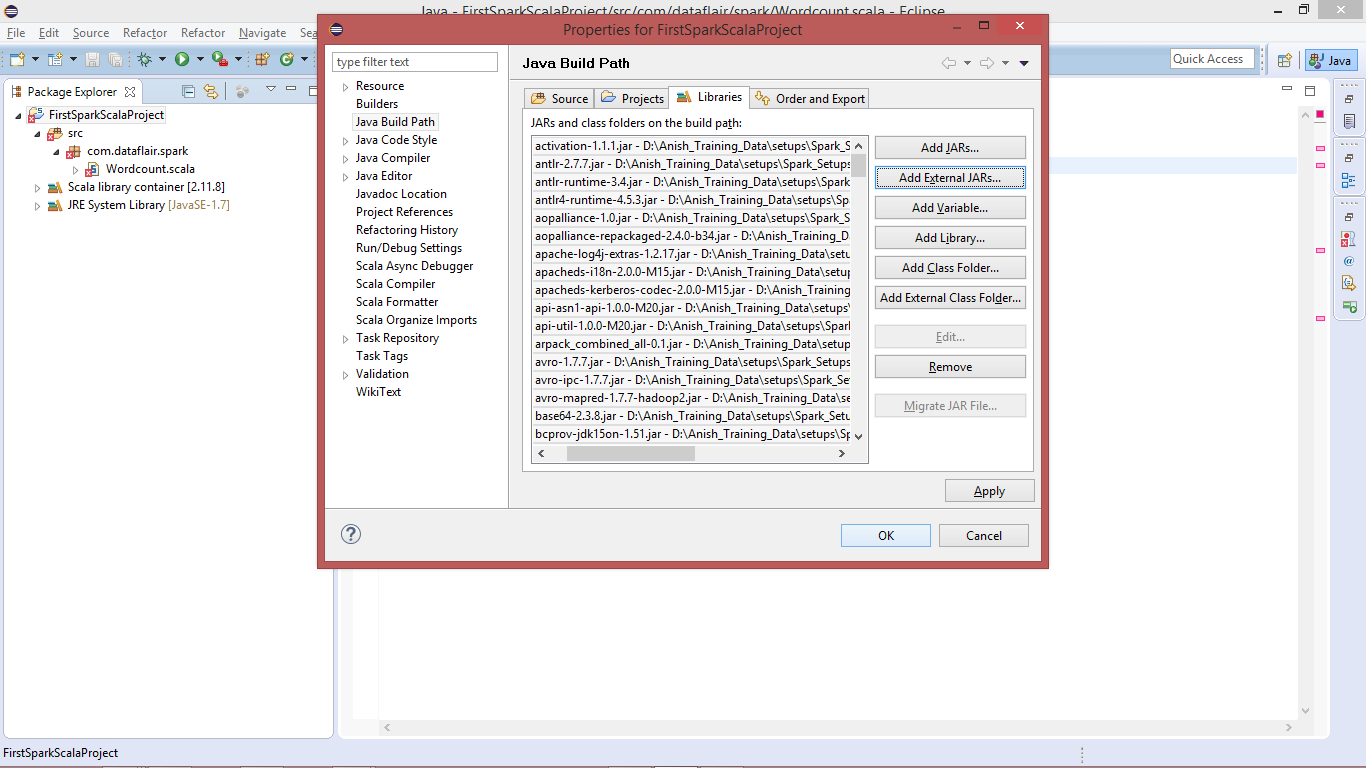
You should have spark setup available in developing environment, it will be needed for spark libraries.

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

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

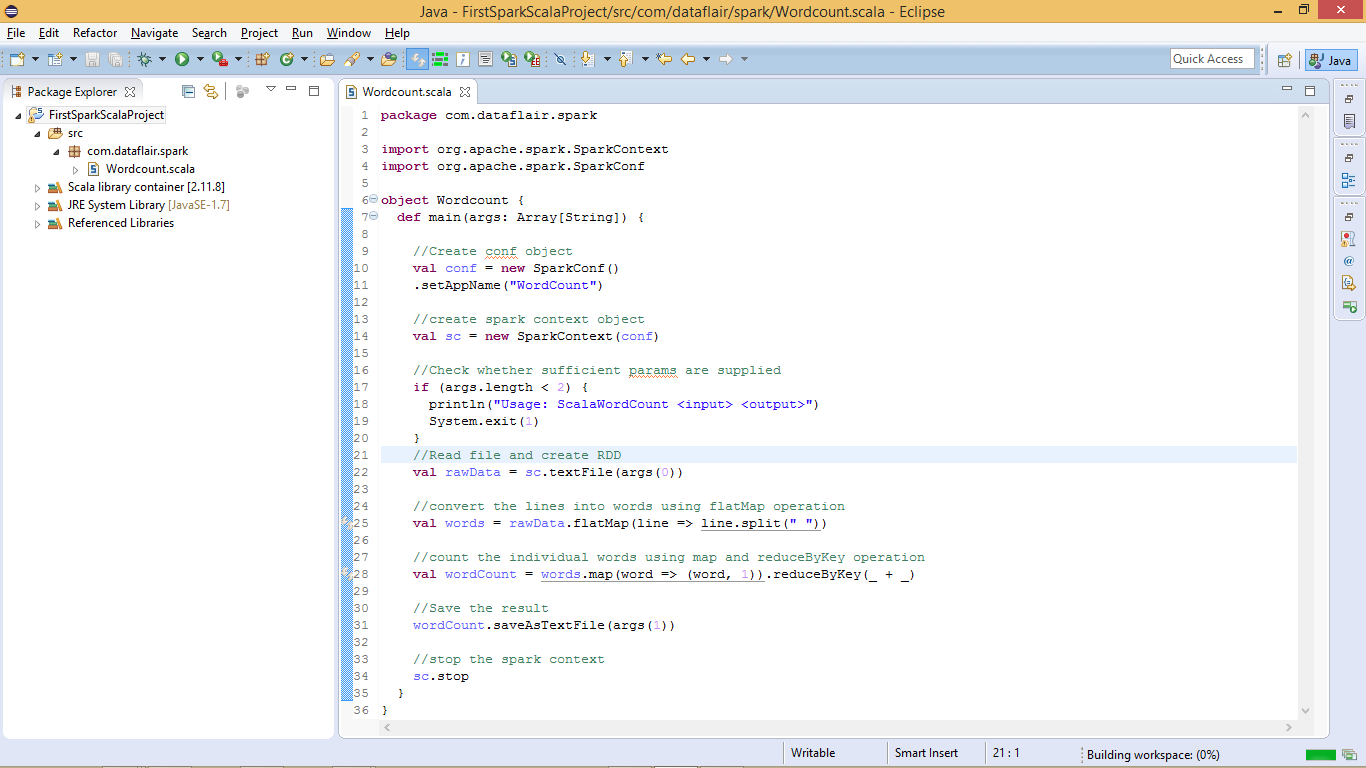
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/select-spark-jars-select.png)

Import the selected jar:

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

**2.10. Spark Scala Word Count Program**

After importing the libraries all the errors will be removed.

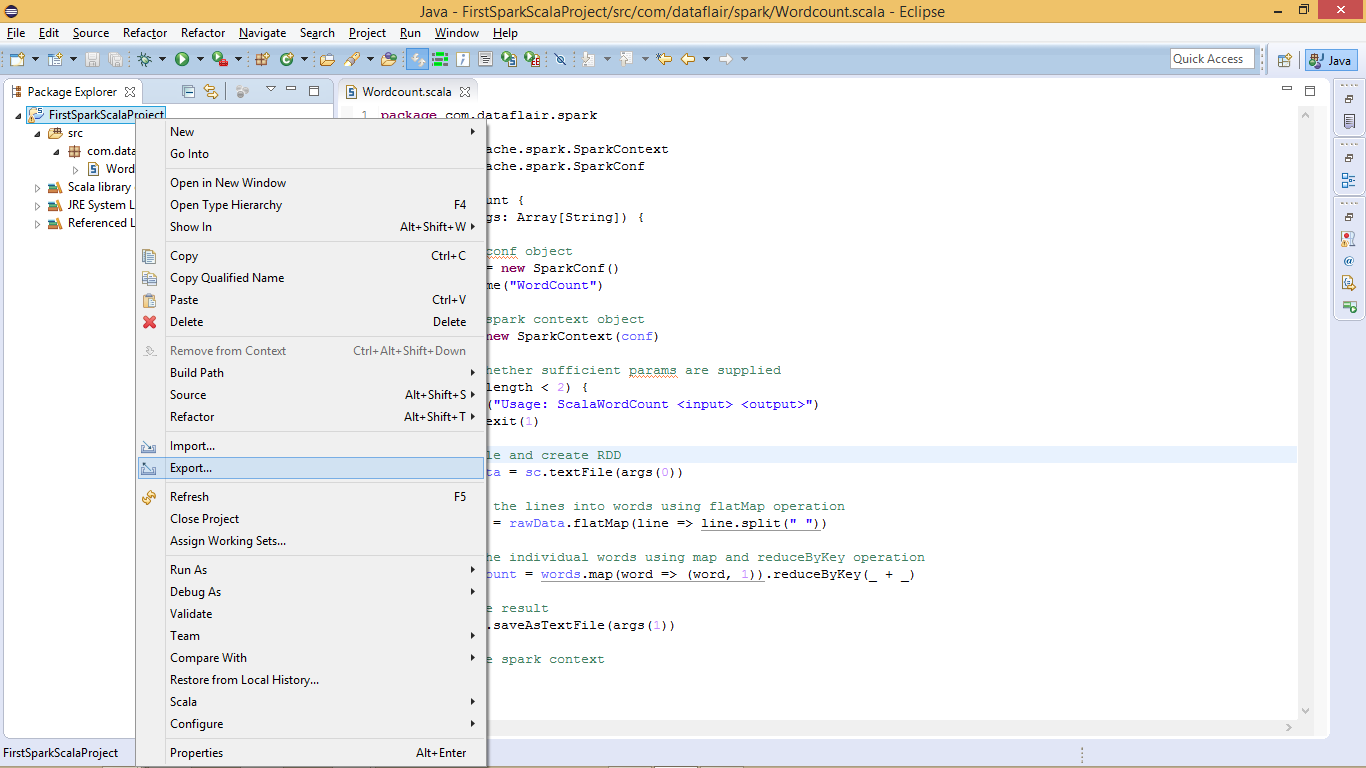
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/wordcount-program-spark-scala.png)

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.

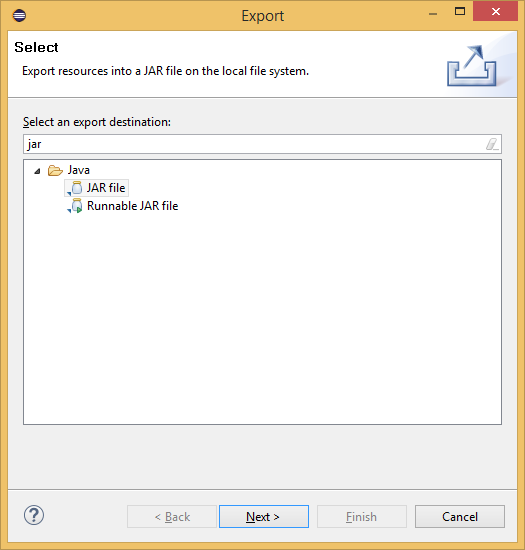
Follow this links to [install Apache Spark on single node cluster](http://data-flair.training/blogs/apache-spark-installation-in-standalone-mode/) or on the [multi-node cluster.](http://data-flair.training/blogs/apache-spark-installation-on-multi-node-cluster-step-by-step-guide/)

**2.11. Create the Spark Scala Program Jar File**

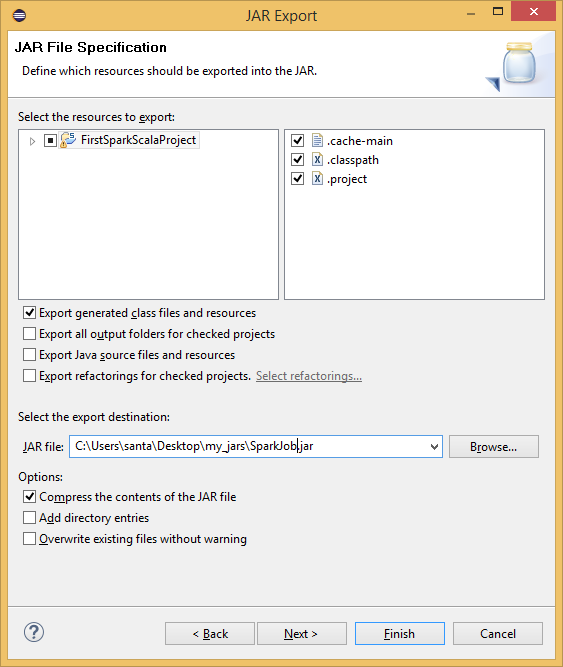
Before running created **Spark word count application** we have to create a jar file. Right click on **project >> export**

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

Select Jar-file Option to Export:

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

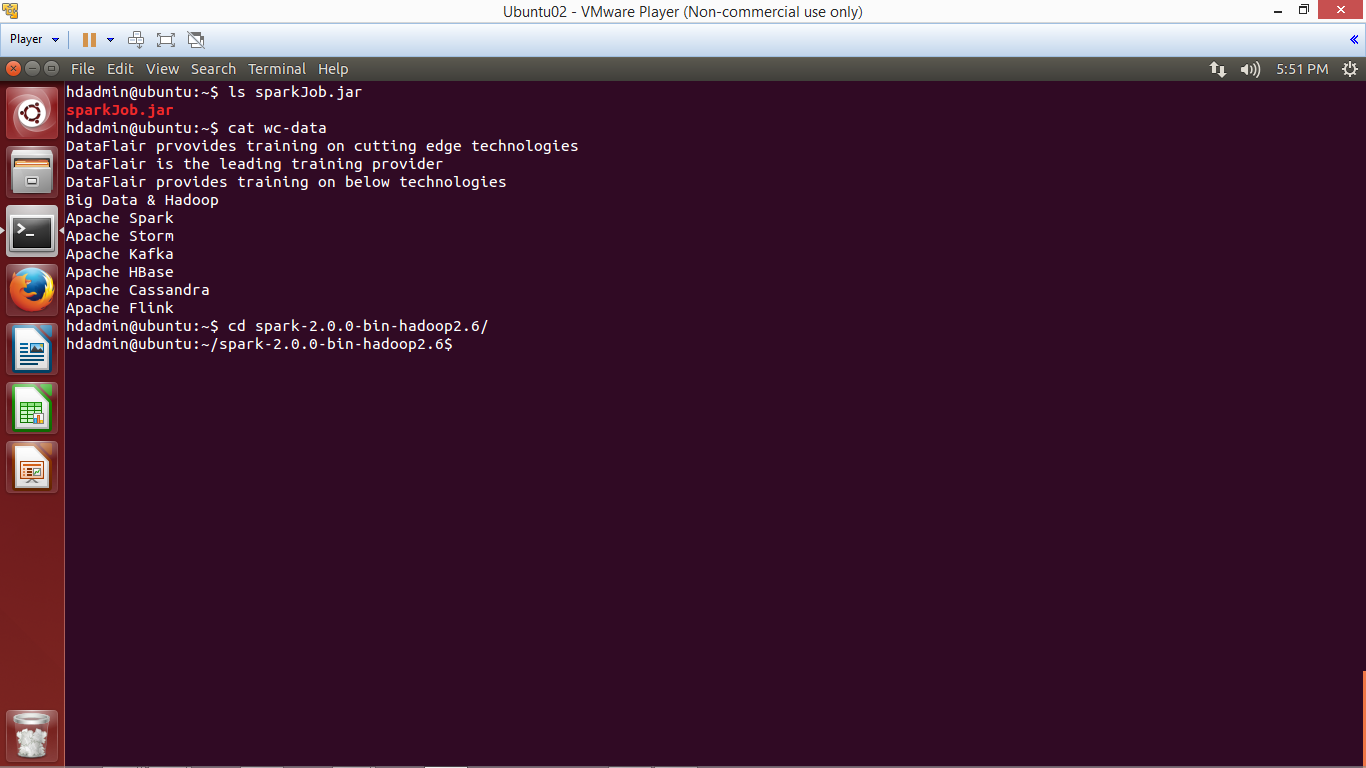
Create the Jar file:

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

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

**2.12. 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/cd-spark-home-directory.png)

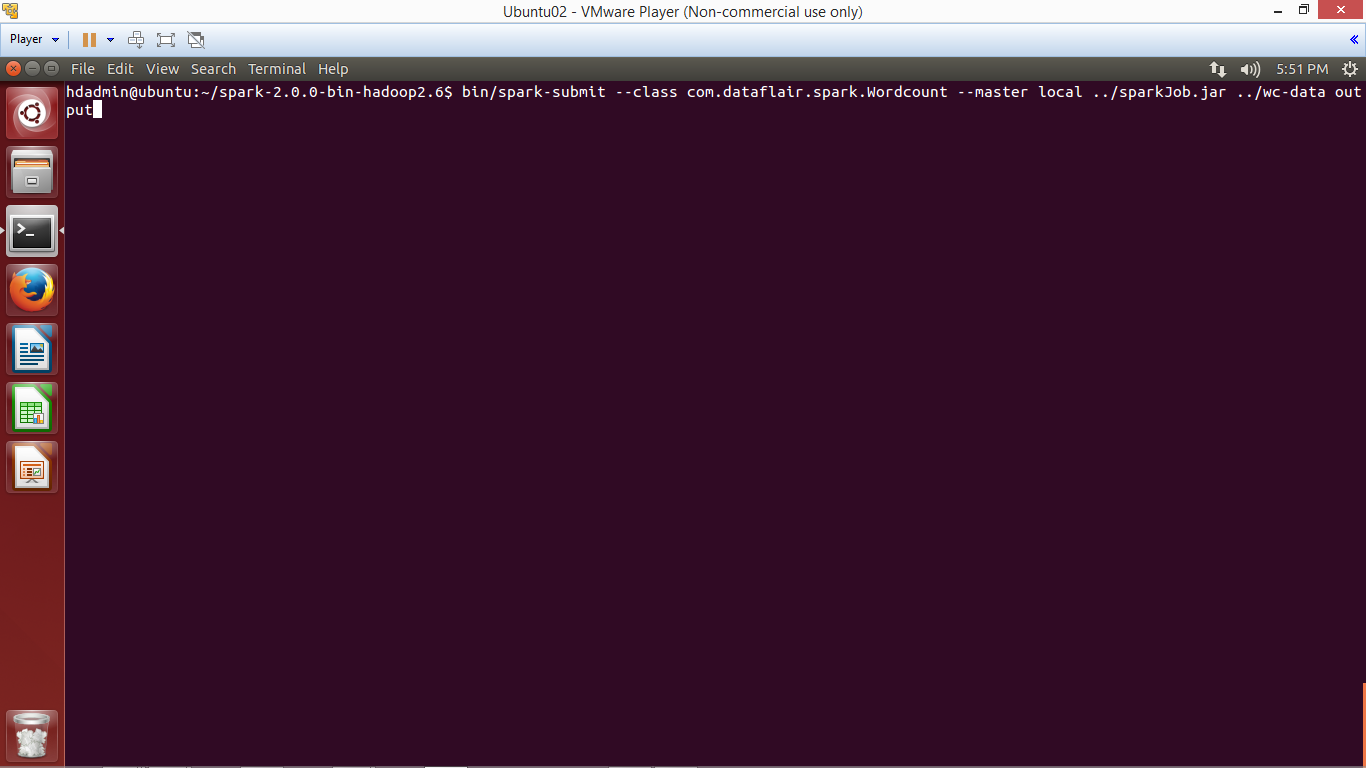
**2.13. Submit Spark Application using spark-submit script**

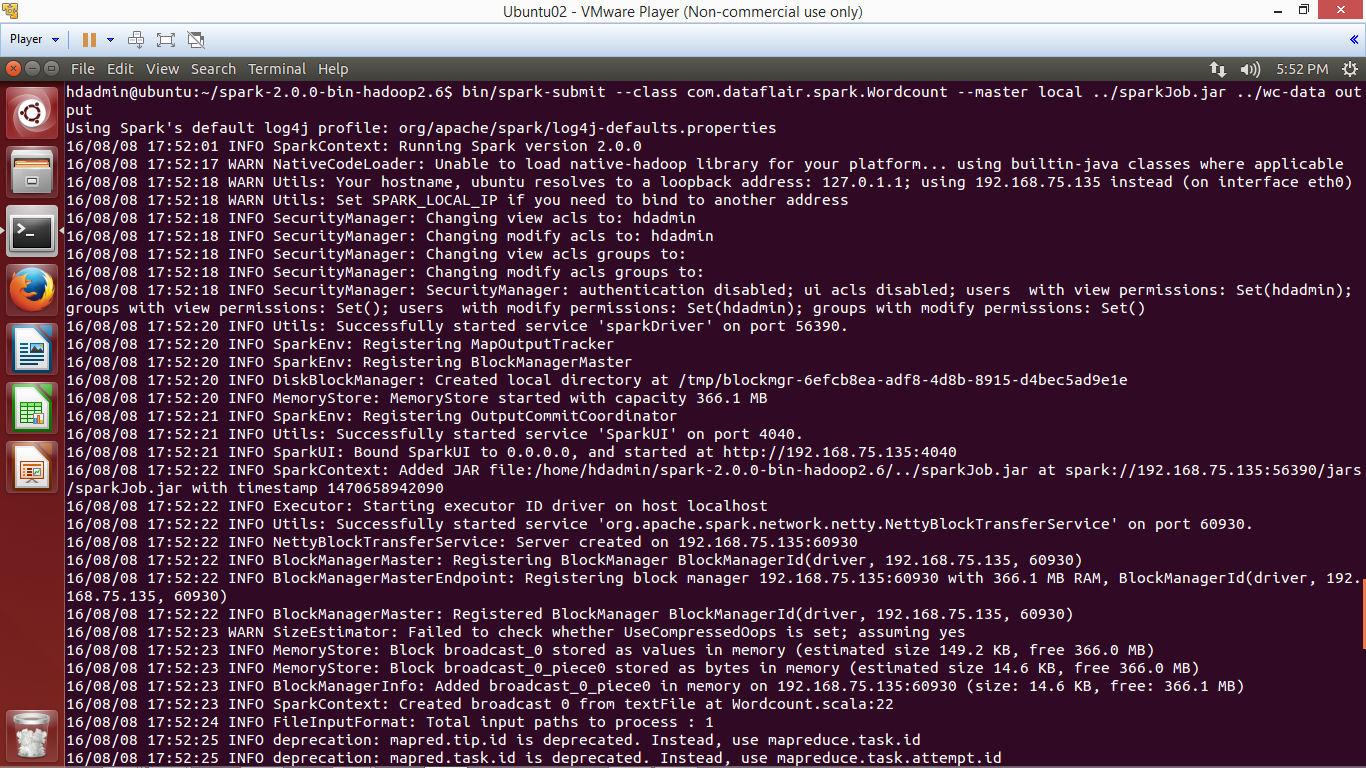
To submit the Spark application using below command:

|  |  |
| --- | --- |
| 1 | bin/spark-submit --class <Qualified-Class-Name> --master <Master> <Path-Of-Jar-File> <Input-Path> <Output-Path> |
| 1 | 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

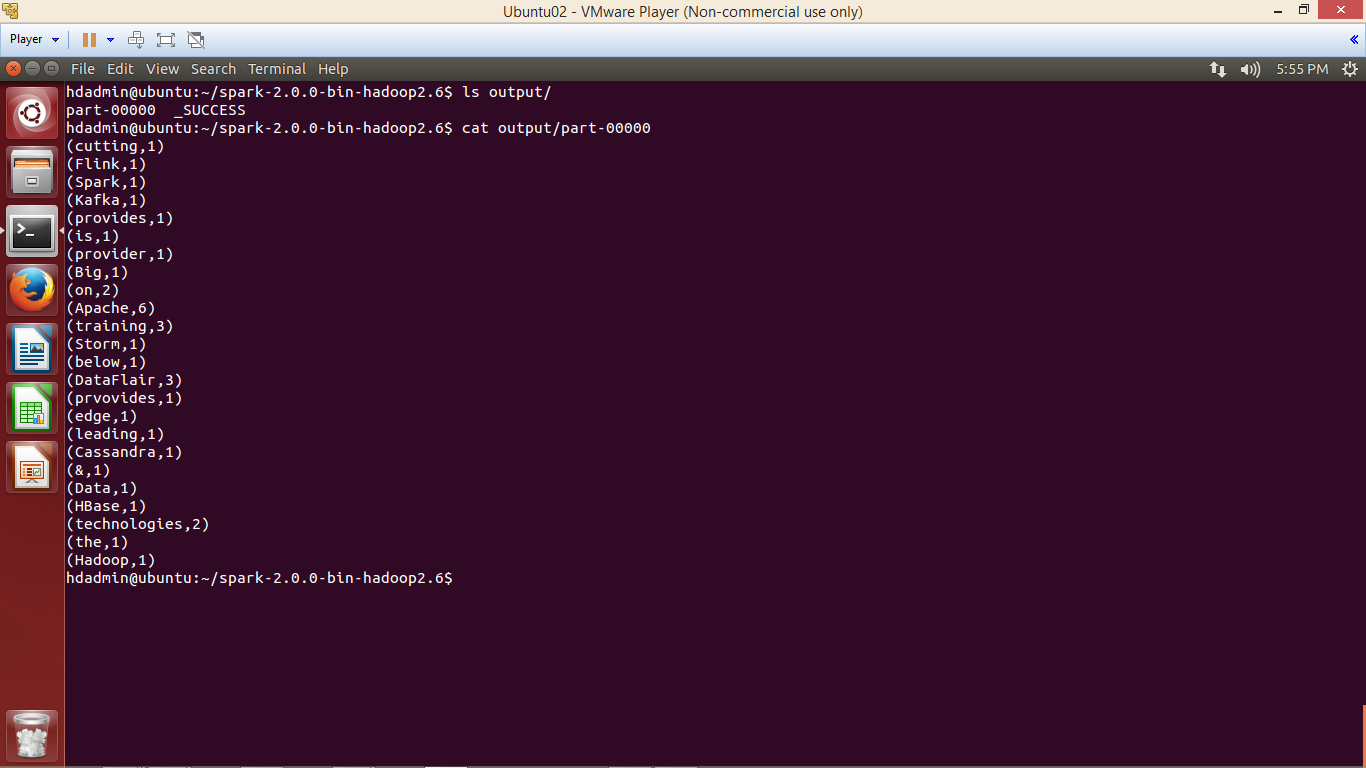
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/spark-scala-jar-submit-run-command.png)

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

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

**2.14. 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/spark-wordcount-job-success.png)

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.

### My Workout:

### 



