CS522 – Big Data

Set-up Spark and Scala Manual

Maharishi University of Management

Department of Computer Science

Feb 2nd, 2017

Table of Contents

1. Host machine 3

2. Reference Guides sites 3

3. Verifying Spark installation 3

4. Verifying Java Installation 3

5. Install Scala IDE for Eclipse 4

6. Create Scala Project from Eclipse 6

7. Download the Sample Scala Apache Access Log Parser Source Code 12

8. Create Spark project and convert to Maven project 12

9. Download Apache Log File and then put into the input directory in HDFS 14

10. Export to JAR file 17

11. Deploy MySpark project to Spark 18

12. Compute the frequency of each response code 19

13. All IP Addresses that have accessed this server more than N times 20

14. The top endpoints requested by count 20

15. Generate a list of URLs, sorted by hit count 21

16. Calculate the average, max and min size of the content size 21

# Host machine

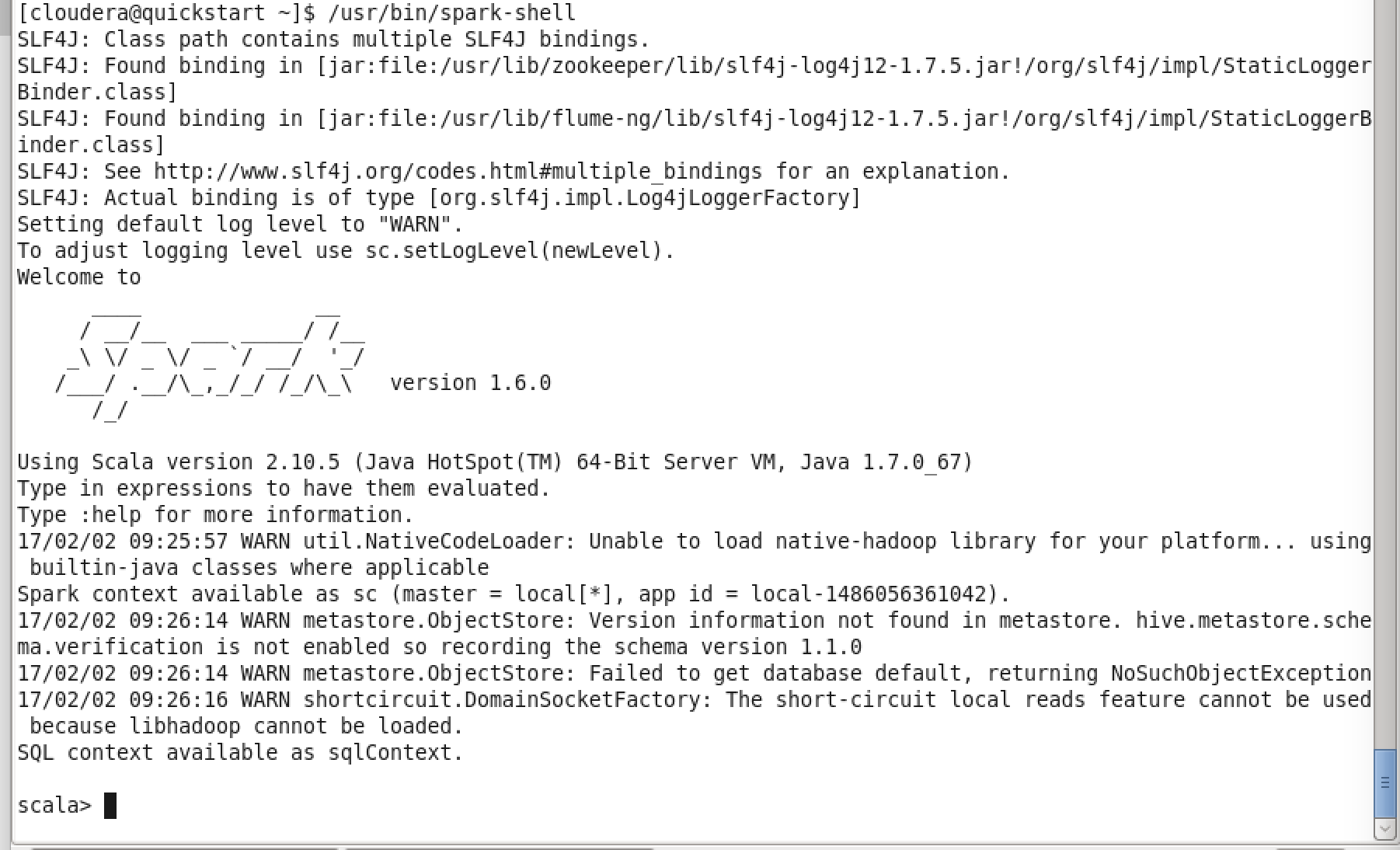
I installed Virtual Box and used Cloudera QuickStart VM on my MacBook Pro masOS Sierra v10.12.3

# Reference Guides sites

* Apache Spark Tutorial - <http://www.tutorialspoint.com//apache_spark/index.htm>
* Apache (Unix) Log Samples - <http://www.monitorware.com/en/logsamples/apache.php>
* Scala IDE - <http://scala-ide.org/docs/videos.html>

# Verifying Spark installation

$ /usr/bin/spark-shell



Spark version 1.6.0

Scala version 2.10.5

Configuration location

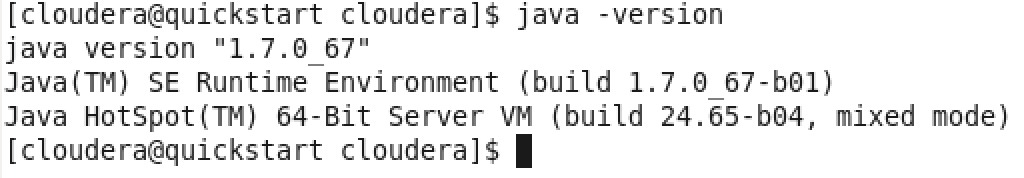
/etc/spark/conf

TO USE HDFS

/user/cloudera

# Verifying Java Installation

$ java -version

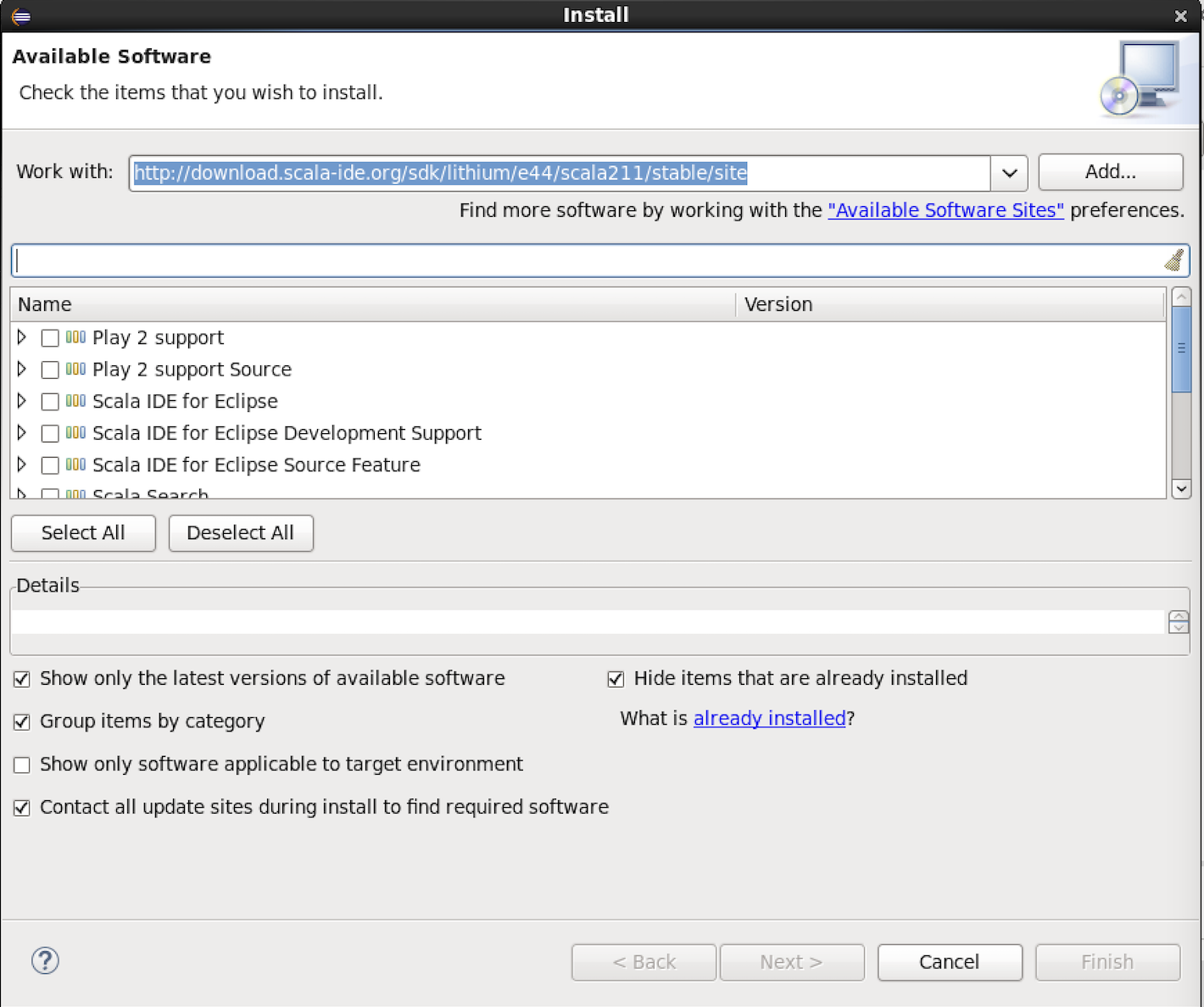


# Install Scala IDE for Eclipse

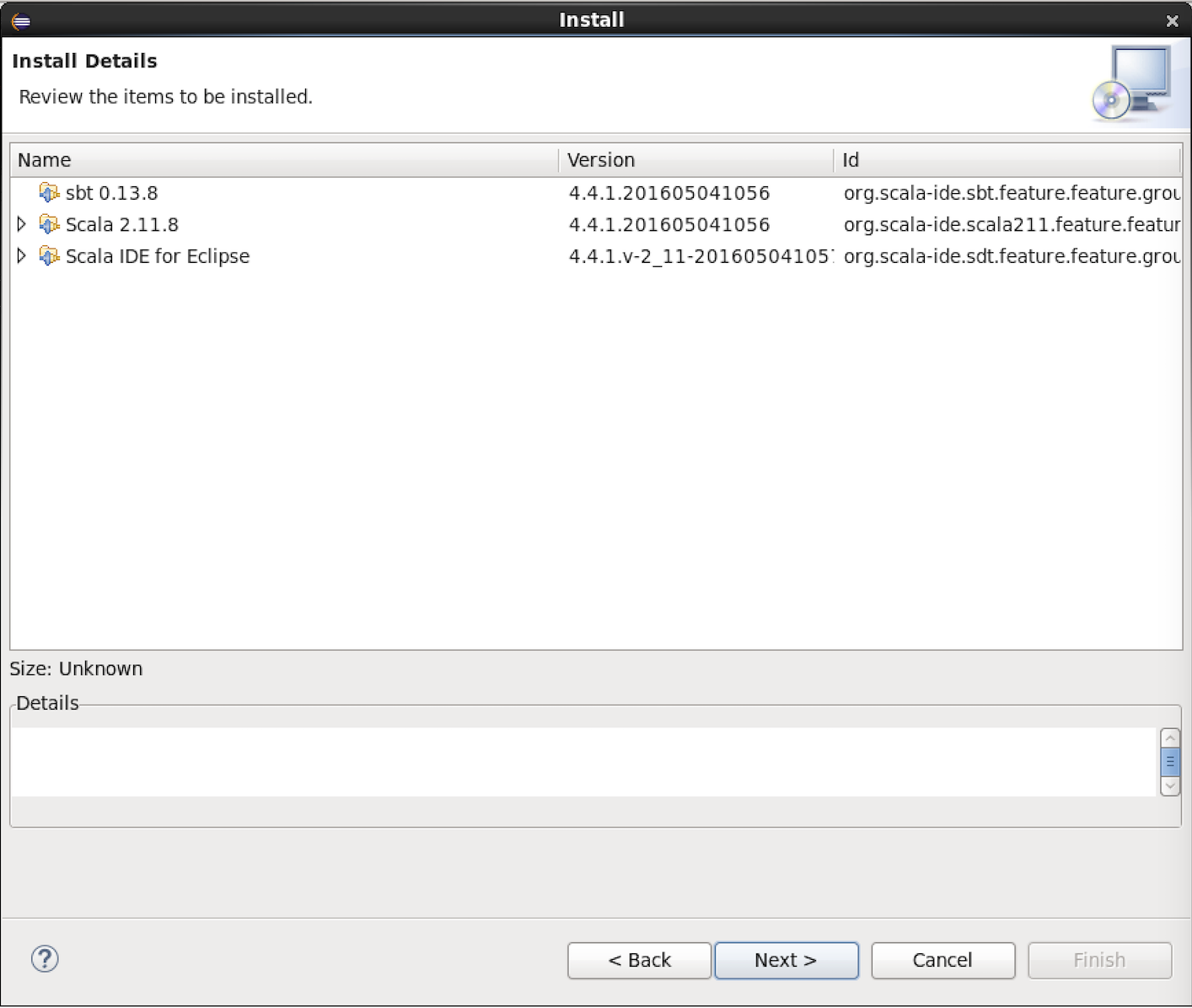
**Step 1**: Launch Eclipse, then select Help → Install New Software…

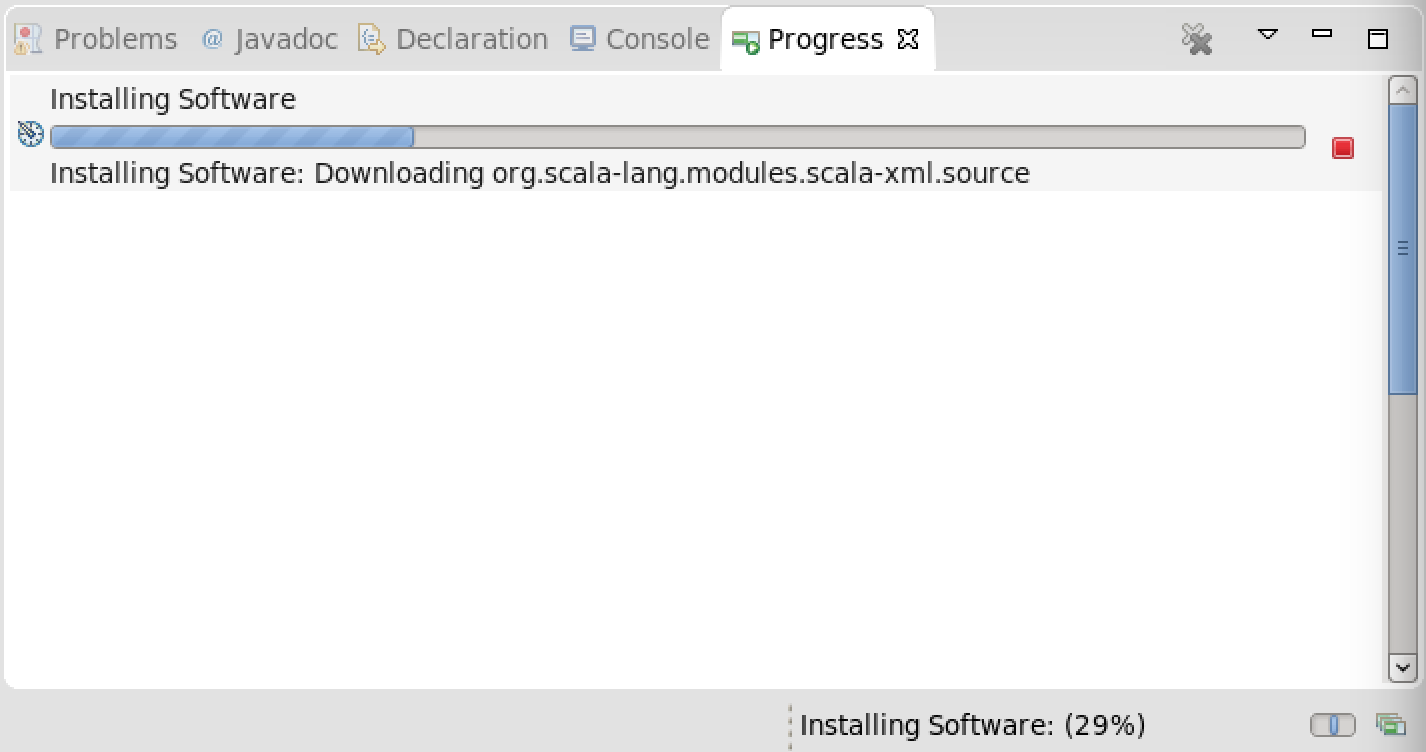
**Step 2**: Enter this URL to “Work with” dropdown list

<http://download.scala-ide.org/sdk/lithium/e44/scala211/stable/site>

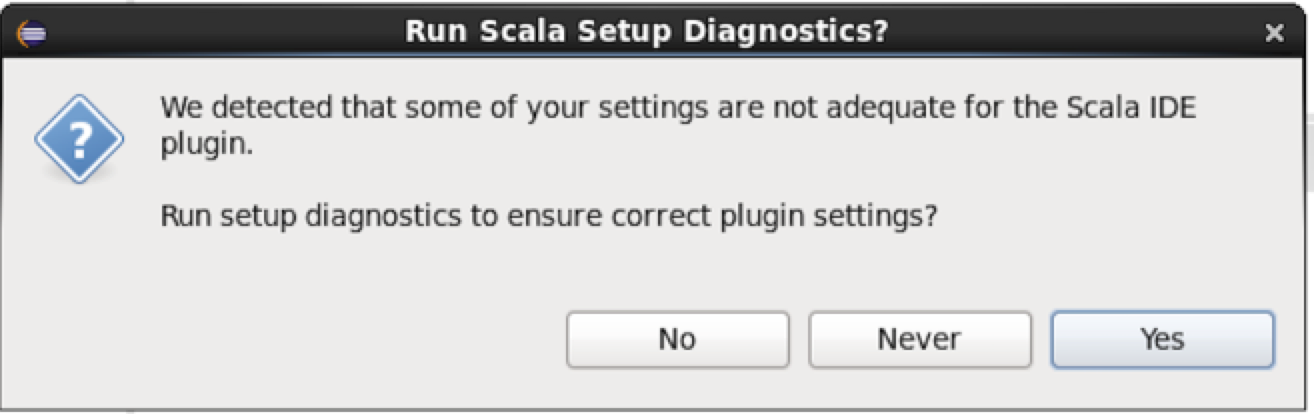


Select the checkbox “**Select Scala IDE for Eclipse**” to install.

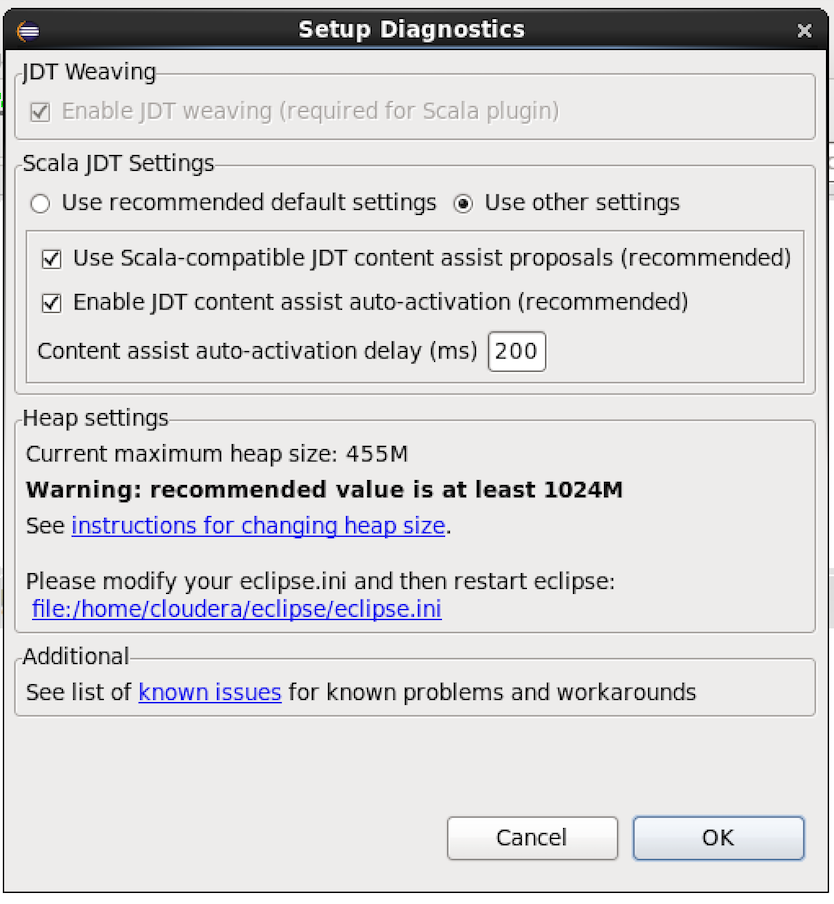




After restarted Eclipse, if it shows a message box like this:

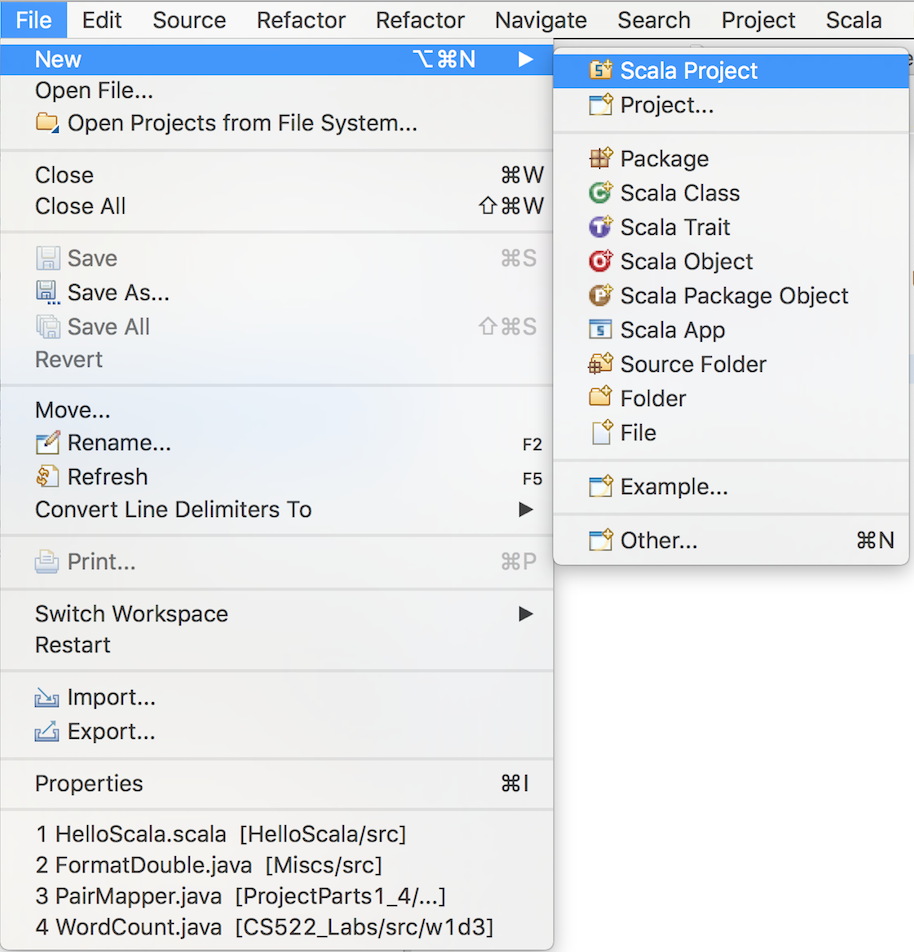


Click on Yes button and following as Wizard such as:



# Create Scala Project from Eclipse

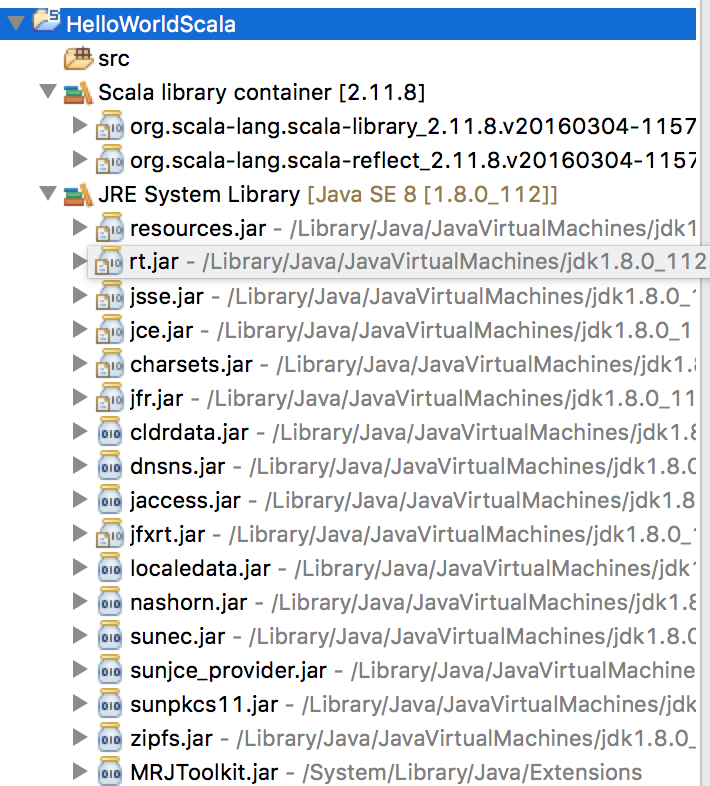
Launch Eclipse 🡪 click on File 🡪 New 🡪 Scala Project



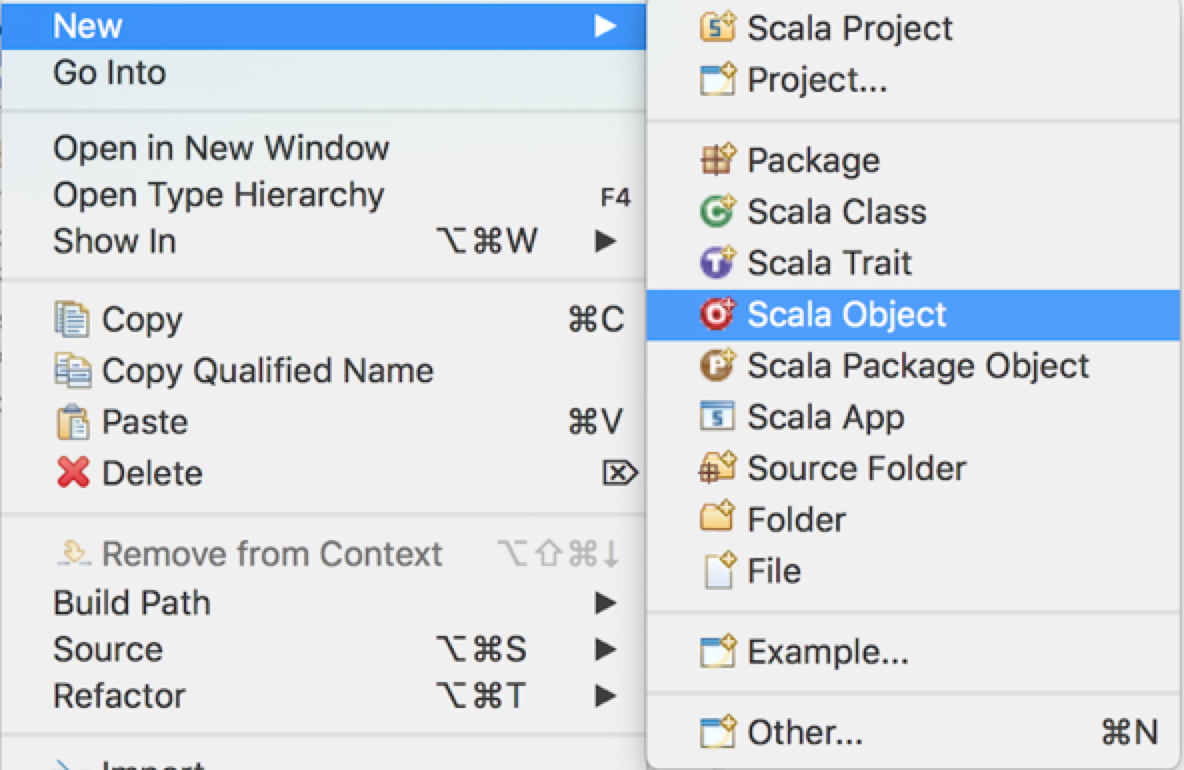
Enter your project name and the click Finish button



The project has been created like this screenshot:



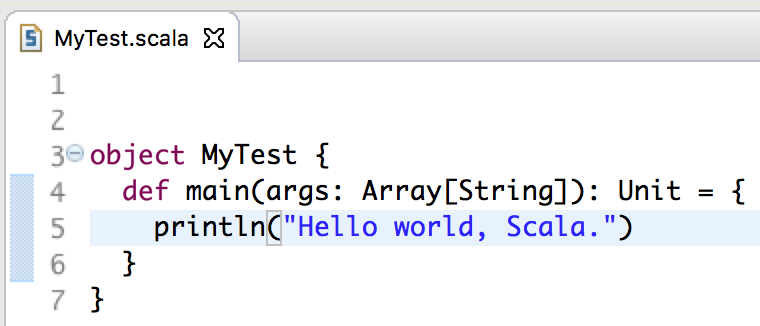
Add a new Scala Object: Right click on your newly created project 🡪 New 🡪 Scala Object



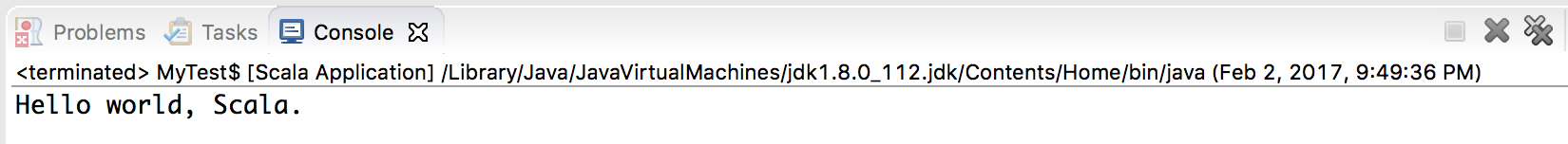
Enter your Object name and then click Finish button



Enter the main method and print out the text named “Hello World, Scala”



To run the this project, click on Run button (tool tip is Run MyTest.scala”) and see the output like this screenshot:



# Download the Sample Scala Apache Access Log Parser Source Code

**Step 1**: Access to this link: <https://github.com/alvinj/ScalaApacheAccessLogParser>



**Step 2**: Click on “Clone or download” button 🡪 save to file as “**ScalaApacheAccessLogParser-master.zip**”

**Step 3**: Extract to your favourite location

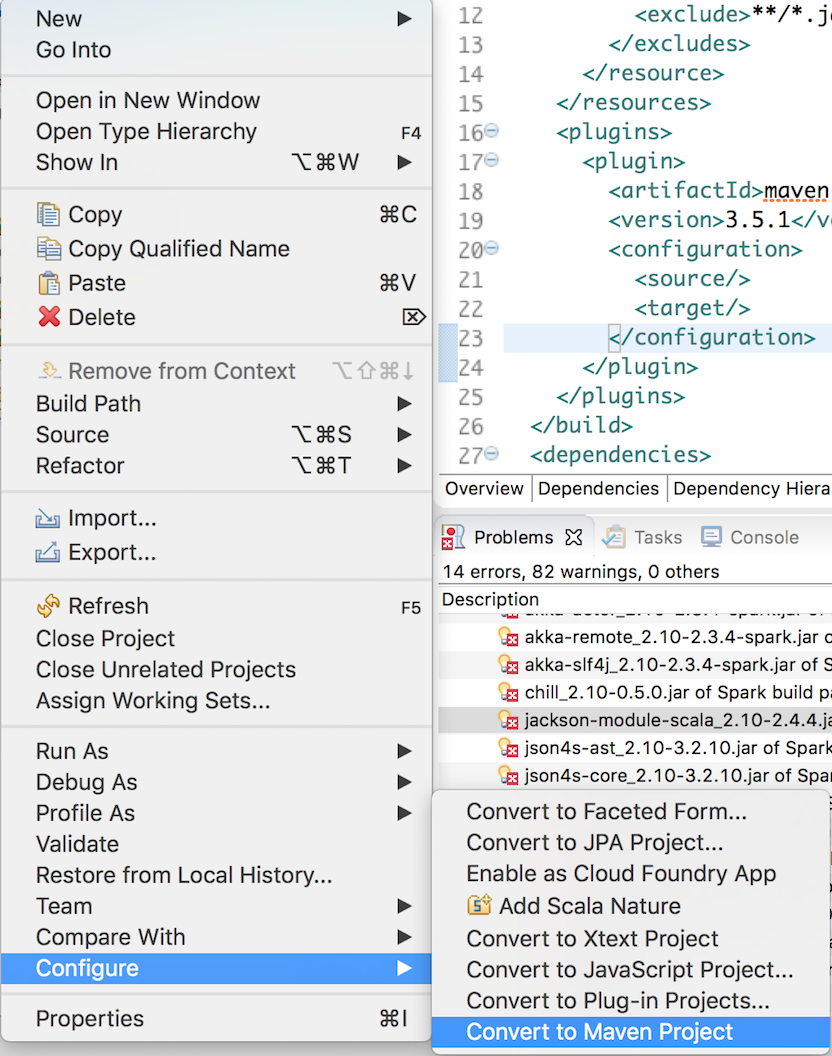
# Create Spark project and convert to Maven project

**Step 1**: Create a new Scala project named “MySpark” in Eclipse and then import these two files:

* AccessLogParser.scala
* AccessLogRecord.scala

**Step 2**: Convert to Maven project

Right click on this project, click on Configure 🡪 Convert to Maven project



Add dependency:

<dependency>

<groupId>org.apache.spark</groupId>

<artifactId>spark-core\_2.10</artifactId>

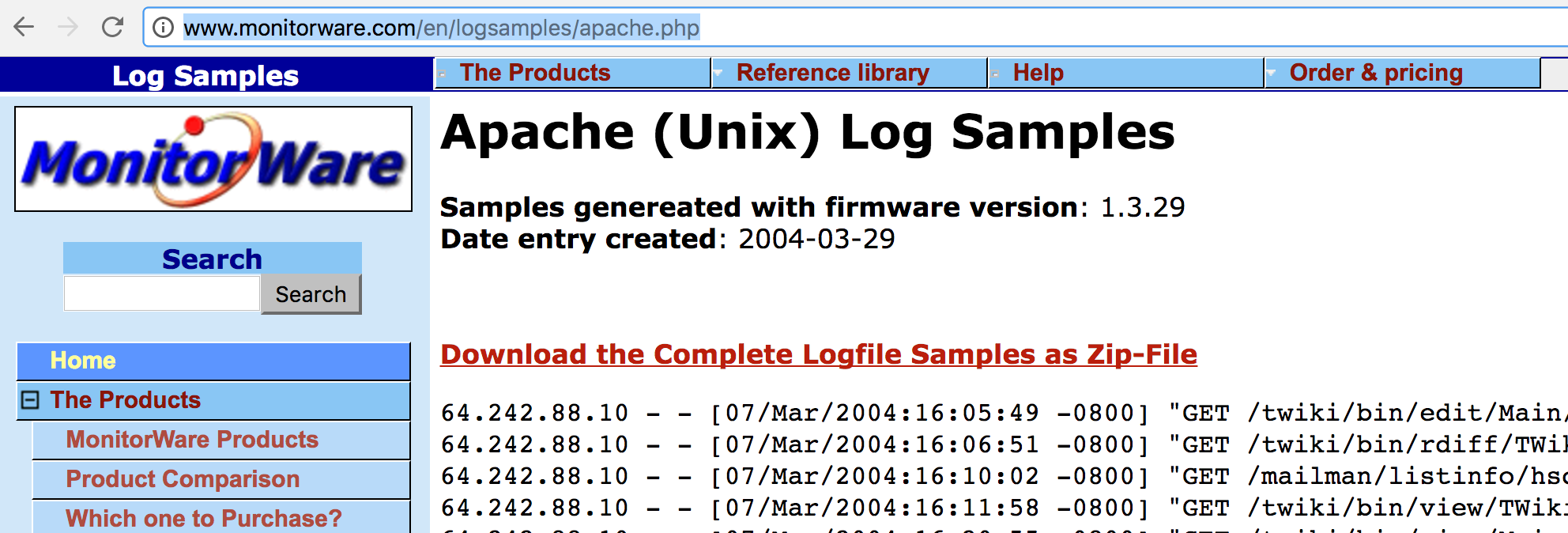
<version>2.1.0</version>

</dependency>

**Step 3**: Add a new Scala class named “ApacheLogFileAnalytics” to parse the Apache (Unix) log file

# Download Apache Log File and then put into the input directory in HDFS

**Step 1**: Access to this link: <http://www.monitorware.com/en/logsamples/apache.php>



**Step 2**: Click on “**Download the Complete Logfile Samples as Zip-File**” link and save to your favourite location as default file name “**apache-samples.jar**” and extract it.

For my case, I choose “access\_log” file.

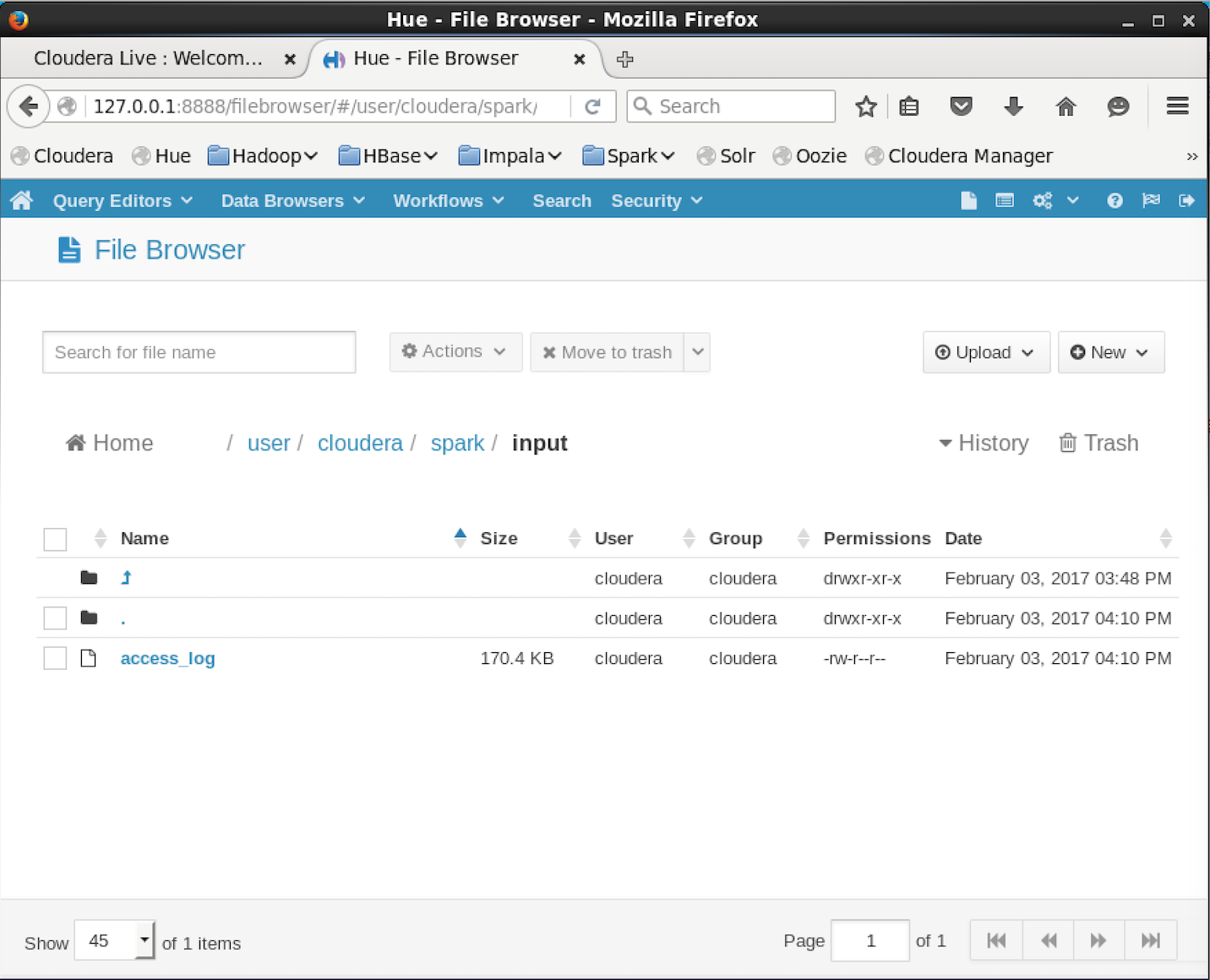
**Step 3**: Open Terminal and issue these commands below

**Step 3.1**: Create input directory in HDFS

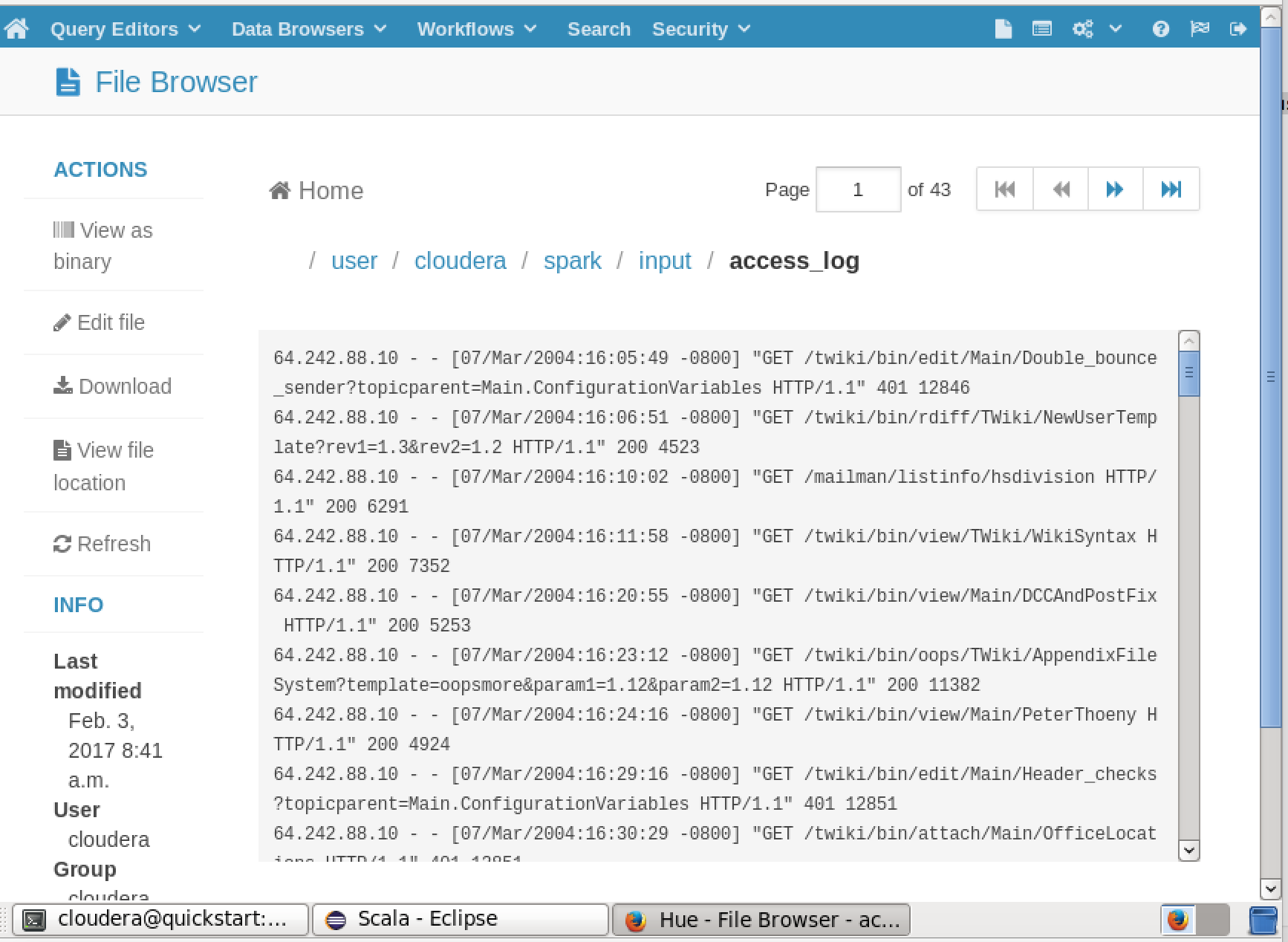
$ hadoop fs -mkdir /user/cloudera/spark /user/cloudera/spark/input

**Step 3.2**: Put the log file to the input directory in HDFS

$ hadoop fs -put /home/cloudera/gitlab/cloudera/myspark/src/input/access\_log /user/cloudera/spark/input

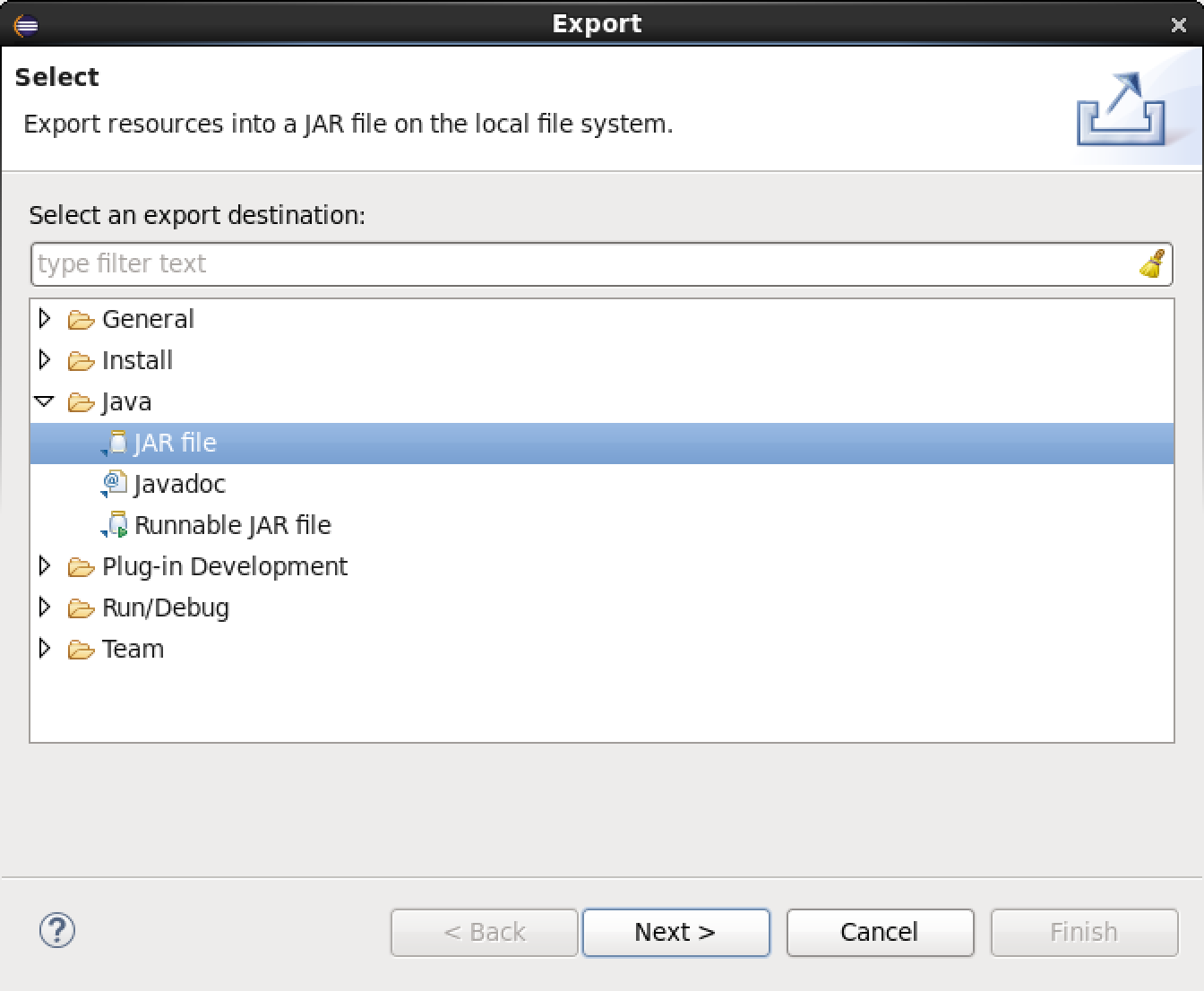


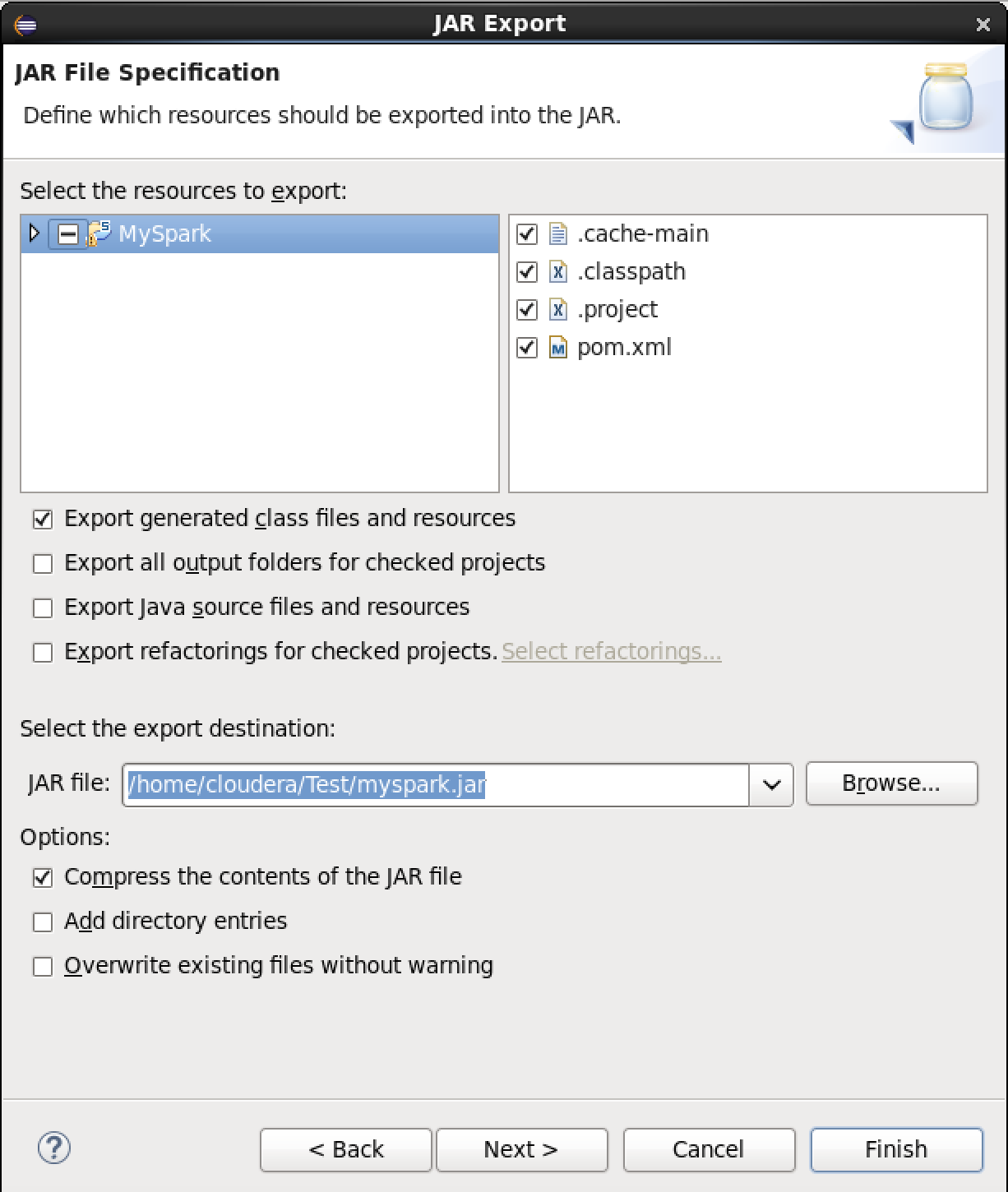
**Step 3.3**: View the input file



# Export to JAR file

Right click on the project 🡪 click on Export…



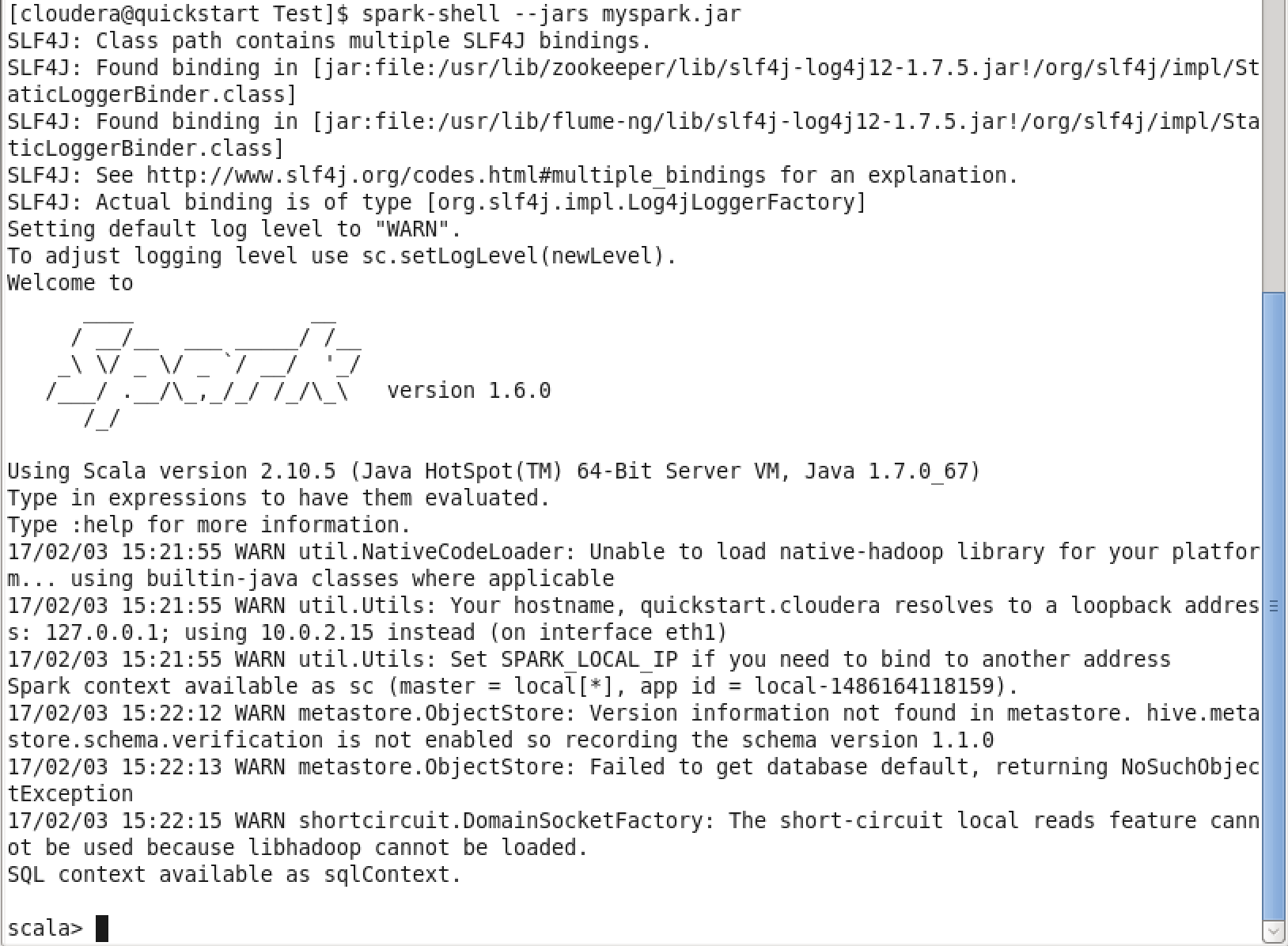


# Deploy MySpark project to Spark

**Step 1**: Deploy the “myspark.jar” file to Spark

$ spark-shell --jars myspark.jar

If deployed successfully, it will display like this screenshot:



**Step 2**: Load spark.apache.ApacheLogFileAnalytics Scala object

scala> import spark.apache.ApacheLogFileAnalytics

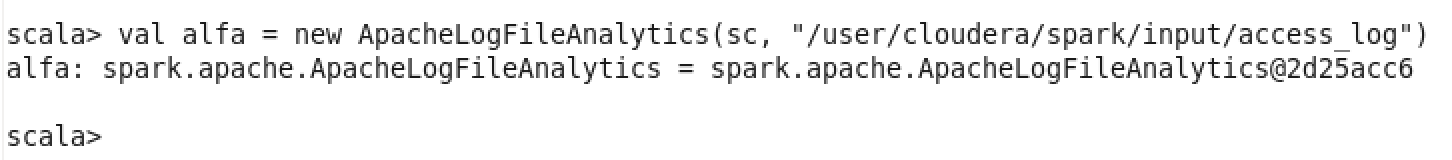
If imported successfully, it will display like this screenshot:



**Step 3**: Initialize the object

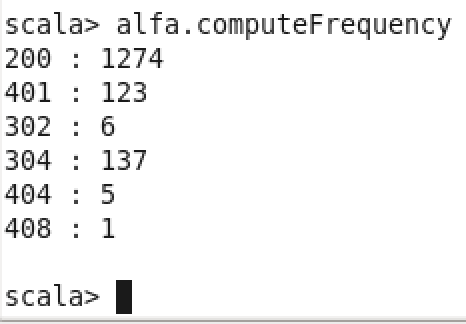
scala> val alfa = new ApacheLogFileAnalytics(sc, "/user/cloudera/spark/input/access\_log")

If initialized successfully, it will display like this screenshot:



# Compute the frequency of each response code

scala> alfa.computeFrequency



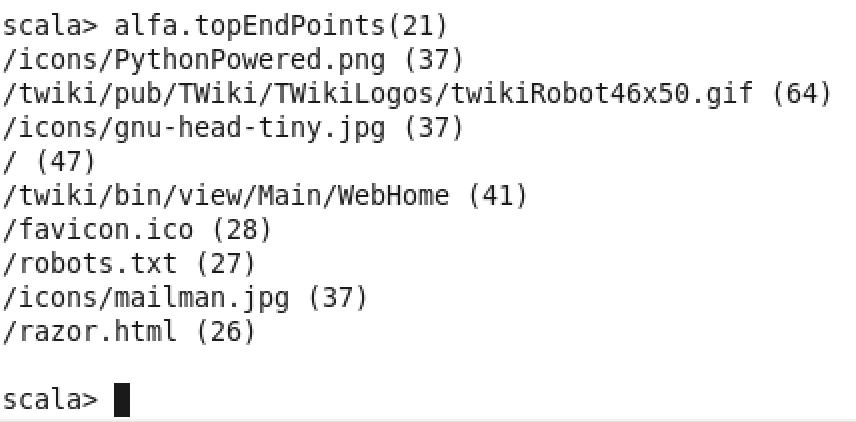
# All IP Addresses that have accessed this server more than N times

scala> alfa.accessMoreThan(21)



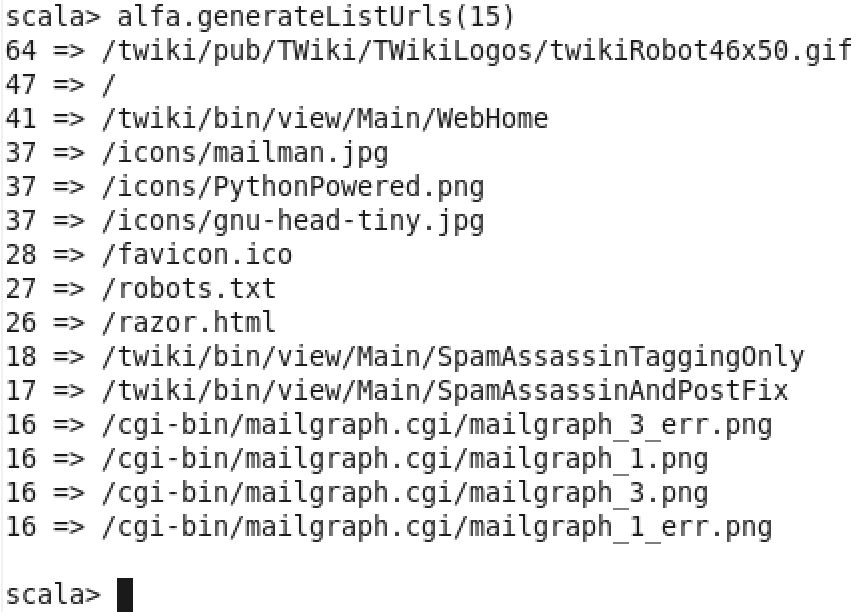
# The top endpoints requested by count

scala> alfa.topEndPoints(21)



# Generate a list of URLs, sorted by hit count

scala> alfa.generateListUrls(15)



# Calculate the average, max and min size of the content size

scala> alfa.calSize

