06: Setting up Spark-shell on Mac & getting started

Posted on November 8, 2018

This tutorial outlines the basic steps to get started with Spark on Mac OS.

1. Install Xcode

Xcode can be installed via Apple appstore. Xcode is Apple's Integrated Development Environment (IDE). Xcode is a large suite of software development tools and libraries from Apple.

2. Install the Apple command line tools

300+ Java Interview FAQs

300+ Java FAQs 🔥

16+ Java Key Areas Q&As

150+ Java Architect FAQs

80+ Java Code Quality Q&As

150+ Java Coding Q&As

300+ Big Data Interview FAQs

300+ Big Data FAQs 🔥

Tutorials - Big Data

TUT - ⏯ Starting Big Data

TUT - Starting Spark & Scala

TOP

Once Xcode is installed, install the command line tools via “Xcode menu” –> “preferences” –> “command lines tools”, and click the install button. This may take a while to install. Once installed you can verify with a Terminal window

```
$ xcode-select -h
```

The Xcode Command Line Tools are part of XCode. The Xcode Command Line Tools include a GCC compiler, and many common Unix-based tools require the GCC compiler

3. Install homebrew

Homebrew is a package manager for OS. On a Terminal window type

```
$ ruby -e "$(curl -fsSL https://raw.github.com/mxc
```

Verify if brew is installed properly by typing the following on a Terminal window:

```
$ brew doctor
```

4. Install Java & Scala

```
$ brew cask install java
```

TUT - Starting with Python
TUT - Kafka
TUT - Pig
TUT - Apache Storm
TUT - Spark Scala on Zeppelin
TUT - Cloudera
TUT - Cloudera on Docker
TUT - File Formats
TUT - Spark on Docker
TUT - Flume
TUT - Hadoop (HDFS)
TUT - HBase (NoSQL)
TUT - Hive (SQL)
TUT - Hadoop & Spark
TUT - MapReduce
TUT - Spark and Scala
TUT - Spark & Java
TUT - PySpark on Databricks
TUT - Zookeeper

```
$ brew install scala
```

5. Install Apache Spark

```
$ brew install apache-spark
```

You can check where a package has been installed with:

```
$ brew info apache-spark
```

6. Spark shell to run Spark code interactively

```
$ spark-shell
```

```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.3.2
      /_/

Using Scala version 2.11.8 (Java HotSpot(TM) 64-B
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```

7. Spark example in Scala

Use "**:paste**" to paste the following code. After pasting the code press "**ctrl-D**" to run the Spark code.

```
scala> :paste
// Entering paste mode (ctrl-D to finish)

import org.apache.spark.sql.SparkSession;
import org.apache.spark.sql.Dataset;

val spark = SparkSession.builder.appName("Simple A
                                .config("spark.mast
                                .getOrCreate()

val list = List(1,2,3,4,5)
val rdd = spark.sparkContext.parallelize(list, 2)

val multipleBy2Rdd = rdd.map(_ * 2)
println(multipleBy2Rdd.collect().toList)

```

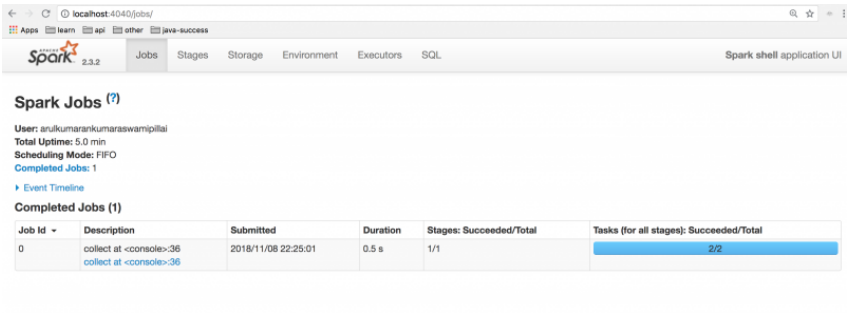
Use "**ctrl+c**" to quit the Spark REPL.

8. Spark UI

When the spark-shell is running, you can bring up the Spark UI in a browser.

```
http://localhost:4040/jobs/

```



Spark UI

The .bash_profile

This assumes that Hadoop has been installed as per one of the other tutorials so that you can do

$(hadoop classpath). Also, when you install it with "brew install" as opposed downloading the *.tar.gz and unzipping it.

```
//.....
export SPARK_HOME=/usr/local/Cellar/apache-spark/2
export SPARK_DIST_CLASSPATH=$(hadoop classpath)
export PATH=$SPARK_HOME/bin:${PATH}
//...

```

Activate the change:

```
$ source ~/.bash_profile
```

Starting the Spark master

```
sudo spark-class org.apache.spark.deploy.master.Ma

```

Note down the **host:port** where the Spark master is listening via "http://localhost:8080".

Spark worker joins the Spark master

In a separate shell window start the worker node, specify the master host:location to join.

```
sudo spark-class org.apache.spark.deploy.worker.Wo

```

You can submit a job as shown below. If you don't specify the URL of the master, it defaults to "local[*]".

```

```

```
2 $ spark-submit --master spark://192.168.0.18:7077
3
```

‹ 05: Setting up & getting started with Spark local mode with Sbt & Scala

07: Getting started with Spark-submit ›

Disclaimer

Top