

Spark Fundamentals - II

Using sbt, Eclipse, or IntelliJ for unit testing and debugging

Contents

USING SBT, ECLIPSE, OR INTELLIJ FOR UNIT TESTING AND DEBUGGING..... 3

1.1 USING SBT, ECLIPSE, OR INTELLIJ FOR UNIT TESTING AND DEBUGGING. 4

SUMMARY 6

Using sbt, Eclipse, or IntelliJ for unit testing and debugging

This lab will not use Zeppelin, but your choice of the listed tools in your own local workstation.

After completing this hands-on lab, you should be able to:

- Compile and package your application using SBT.
- (Optional) Use any of the listed IDEs for unit testing and debugging.

Allow 30 minutes to complete this section of lab.

1.1 Using sbt, Eclipse, or IntelliJ for unit testing and debugging.

Make sure your docker and the Zeppelin is running. If not, get it started before continuing with this lab. Use the instructions here:

<https://registry.hub.docker.com/u/bigdatauniversity/spark2/>

- ___1. Here I show how you would restart the container if you need to. Otherwise, continue from the previous instance of the boot2docker terminal from lab 3.

```

MINGW32/c/Users/IBM_ADMIN

setting environment variables ...
Writing C:\Users\IBM_ADMIN\.boot2docker\certs\boot2docker-vm\ca.pem
Writing C:\Users\IBM_ADMIN\.boot2docker\certs\boot2docker-vm\cert.pem
Writing C:\Users\IBM_ADMIN\.boot2docker\certs\boot2docker-vm\key.pem
export DOCKER_HOST=tcp://192.168.59.103:2376
export DOCKER_CERT_PATH='C:\\Users\\IBM_ADMIN\\.boot2docker\\certs\\boot2docker-vm'
export DOCKER_TLS_VERIFY=1

You can now use `docker` directly, or `boot2docker ssh` to log into the VM.
Welcome to Git (version 1.9.5-preview20150319)

Run 'git help git' to display the help index.
Run 'git help <command>' to display help for specific commands.

hlquach@ADMINIB-A620FF4 ~
$ docker start bdu_spark2
bdu_spark2

hlquach@ADMINIB-A620FF4 ~
$ docker attach bdu_spark2
starting namenode, logging to /var/log/hadoop-hdfs/hadoop-hdfs-namenode-b95f0ebb4856.out
Started Hadoop namenode:[ OK ]
starting datanode, logging to /var/log/hadoop-hdfs/hadoop-hdfs-datanode-b95f0ebb4856.out
Started Hadoop datanode (hadoop-hdfs-datanode):[ OK ]
starting resourcemanager, logging to /var/log/hadoop-yarn/yarn-yarn-resourcemanager-b95f0ebb4856.out
Started Hadoop resourcemanager:[ OK ]
starting nodemanager, logging to /var/log/hadoop-yarn/yarn-yarn-nodemanager-b95f0ebb4856.out
Started Hadoop nodemanager:[ OK ]
Starting sshd: [ OK ]
starting org.apache.spark.deploy.history.HistoryServer, logging to /usr/local/spark-1.2.1-bin-hadoop2.4/sbin/../logs/spark-student-org.apache.spark.deploy.history.HistoryServer-1-b95f0ebb4856.out
Zeppelin start [ OK ]
[root@b95f0ebb4856 /]# _

```

- __2. Open up **Lab 5** in the Zeppelin notebook. If you do not have SBT installed, do so now:
<http://www.scala-sbt.org/download.html>
- __3. Open up a terminal / command prompt within your on local machine (i.e. Linux / Windows).
- __4. Create a **lab5** directory.
- __5. Change into that directory.

- __6. Run the **sbt** command:

```
C:\spark2>mkdir lab5
C:\spark2>cd lab5
C:\spark2\lab5>sbt
Getting org.fusesource.jansi jansi 1.11 ...
downloading https://repo1.maven.org/maven2/org/fusesource/jansi/jansi/1.11/jansi-1.11.jar ...
    [SUCCESSFUL ] org.fusesource.jansi#jansi;1.11!jansi.jar (640ms)
:: retrieving :: org.scala-sbt#boot-jansi
   confs: [default]
   1 artifacts copied, 0 already retrieved (111kB/78ms)
Getting org.scala-sbt sbt 0.13.8 ...
downloading https://repo.typesafe.com/typesafe/ivy-releases/org.scala-sbt/sbt/0.13.8/jars/sbt.jar ...
    [SUCCESSFUL ] org.scala-sbt#sbt;0.13.8!sbt.jar (1956ms)
```

If this is your first time running *sbt*, it will download all the necessary files.

- __7. Follow the steps 3 - 10 inside the Zeppelin notebook. By the end, you should have a jar file in this directory:

```
target/scala-2.10/lab5_2.10-1.0.jar
```

- __8. You can submit this job to the Spark cluster to see it run. Use the spark-submit script to execute it.
 __9. For the remainder of the lab exercises, as an option, you can follow the instructions for your favorite IDE to set up sbt.

Summary

You should at least be able to compile and package your app using sbt. If you are using one of the IDEs listed in the lab, you should be able to use that IDE to develop your app. You have completed this course. Congratulations!

[illegible]

[illegible]



© Copyright IBM Corporation 2015.

The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. This information is based on current IBM product plans and strategy, which are subject to change by IBM without notice. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way.

IBM, the IBM logo and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.



Please Recycle
