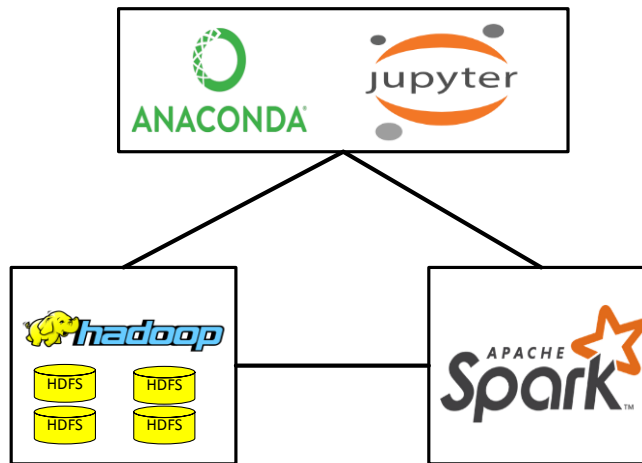


Build Machine learning Lab

Data science Lab



- Java (OpenJDK 8)
- Hadoop (using version: 3.1.2)
- Spark (using version: 2.7.0 to 3.2.0 preview)
- Anaconda (Jupyter). Windows 7: 2018 and Windows 10: 2019

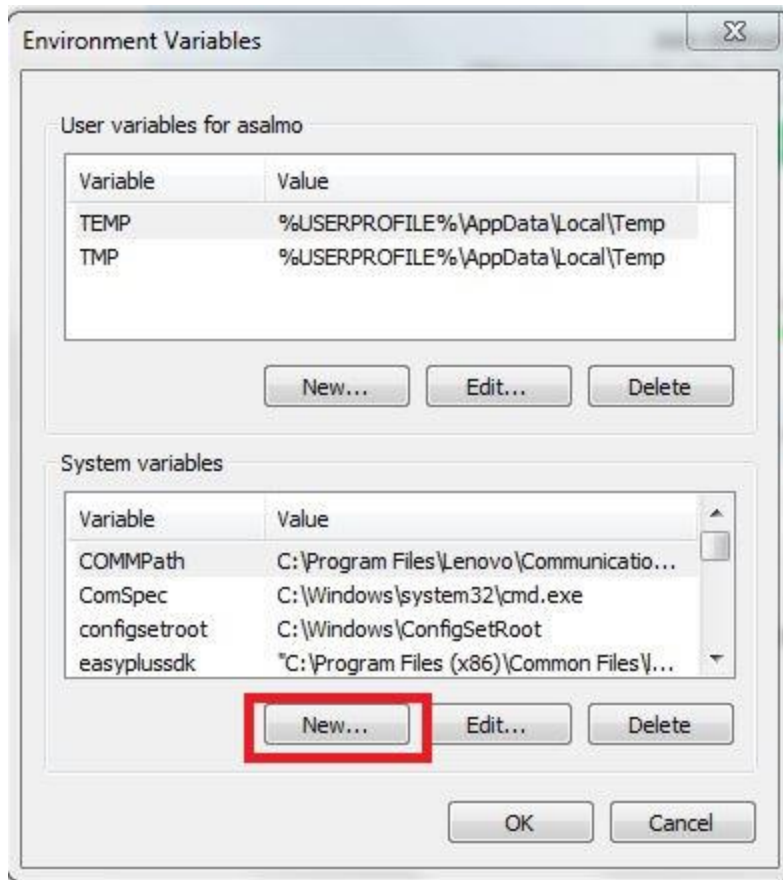
- 1- If you don't have Java 8, you will need to implement step 1 & 2. If you already have Java 8, you can pass step # 1. Hadoop 3.1.2 works with Java 8.

Install OpenJDK-8 Java

Download and install

https://developers.redhat.com/products/openjdk/download?extIdCarryOver=true&sc_cid=701f2000001OH7JAAW

- 2- Add JAVA_HOME to environment variables.



JAVA_HOME= C:\alaa\AdoptOpenJDK\jdk-8.0.232.09-hotspot
 Add JAVA_HOME to path %JAVA_HOME%\bin

- 3- Build Hadoop on Windows (one node)
 - A- Download and install <https://www.7-zip.org/> (to unzip Linux)
 - B- Download <http://archive.apache.org/dist/hadoop/common/hadoop-3.1.2/hadoop-3.1.2.tar.gz>
 - C- Make directory "bigdata" on c drive
 C:\bigdata
 - D- Unzip hadoop-3.1.2.tar.gz
 C:\bigdata\hadoop-3.1.2
 - E- Download Hadoop windows patch <https://github.com/cdarlint/winutils>

Dismiss

Join GitHub today

GitHub is home to over 40 million developers working together to host and review code, manage projects, and build software together.

[Sign up](#)

winutils.exe hadoop.dll and hdfs.dll binaries for hadoop windows

[hadoop](#) [winutils](#) [binaries](#)

5 commits

1 branch

0 packages

0 releases

1 contributor

Branch: master

New pull request

Find file

Clone or download

cdarlint add 321 winutils

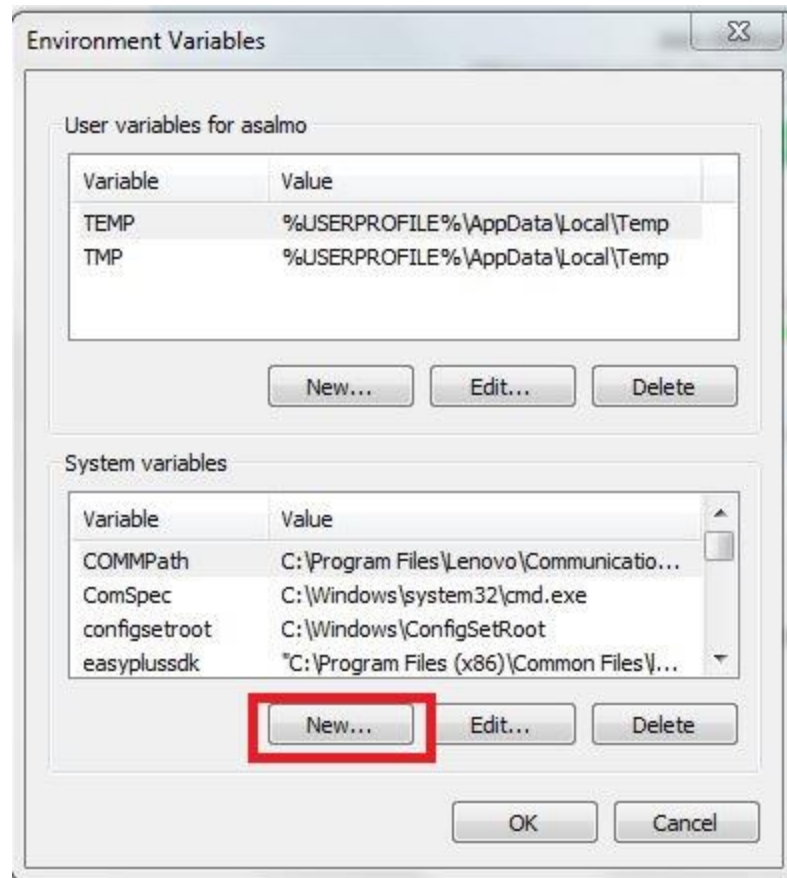
Latest commit ec63c2d on Oct 8, 2019

hadoop-2.6.5/bin	fixed exe and lib 265-312	10 months ago
hadoop-2.7.3/bin	fixed exe and lib 265-312	10 months ago
hadoop-2.7.4/bin	fixed exe and lib 265-312	10 months ago

F- Download all patches, unzip the folder and copy Hadoop-3.1.2/bin to bin folder
(C:\bigdata\hadoop-3.1.2\bin)

G- Build Hadoop variables:

Go "System properties" → Choose "Environment variables"



H- Press "New" to add:

- HADOOP_HOME = C:\bigdata\hadoop-3.1.2
- HADOOP_BIN = C:\bigdata\hadoop-3.1.2\bin
- HADOOP_SBIN= C:\bigdata\hadoop-3.1.2\sbin
- Add % HADOOP_HOME %; % HADOOP_BIN %;% HADOOP_SBIN% to **path** variable

I- Configure Hadoop to run on single machine

We will need to change the following files (C:\bigdata\hadoop-2.9.1\etc\hadoop):

hadoop-env.cmd

core-site.xml

hdfs-site.xml

mapred-site.xml

1- hadoop-env.cmd:

Change JAVA_HOME variable

From:

set JAVA_HOME=%JAVA_HOME%

To:

set JAVA_HOME=C:\AdoptOpenJDK\jdk-8.0.232.09-hotspot

2- core-site.xml

Open this file to add:

```
<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://localhost:9000</value>
  </property>
</configuration>
```

Note: Replace the empty < configuration></ configuration>

3- hdfs-site.xml

Open this file to add:

```
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:///C:/bigdata/hadoop-3.1.2/data/namenode</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:///C:/bigdata/hadoop-3.1.2/data/datanode</value>
  </property>
</configuration>
```

Note: Replace the empty < configuration></ configuration>

In this case, we will need to make directory for

C:\BigData\ hadoop-3.1.2\data

C:\BigData\ hadoop-3.1.2\data\namenode

C:\BigData\ hadoop-3.1.2\data\datanode

- 4- mapred-site.xml
Open this file to add:

```
<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
</configuration>
```

Note:

- 1- Replace the empty < configuration></ configuration>

- J- Format NameNode:
- Open cmd
 - Type "hadoop namenode -format"

```
Command Prompt
2019-12-18 10:39:29,190 INFO blockmanagement.BlockManagerSafeMode: dfs.namenode.safemode.min.datanodes = 0
2019-12-18 10:39:29,190 INFO blockmanagement.BlockManagerSafeMode: dfs.namenode.safemode.extension = 30000
2019-12-18 10:39:29,190 INFO blockmanagement.BlockManager: defaultReplication = 1
2019-12-18 10:39:29,190 INFO blockmanagement.BlockManager: maxReplication = 512
2019-12-18 10:39:29,190 INFO blockmanagement.BlockManager: minReplication = 1
2019-12-18 10:39:29,190 INFO blockmanagement.BlockManager: maxReplicationStreams = 2
2019-12-18 10:39:29,190 INFO blockmanagement.BlockManager: redundancyRecheckInterval = 3000ms
2019-12-18 10:39:29,190 INFO blockmanagement.BlockManager: encryptDataTransfer = false
2019-12-18 10:39:29,190 INFO blockmanagement.BlockManager: maxNumBlocksToLog = 1000
2019-12-18 10:39:29,253 INFO namenode.FSDirectory: GLOBAL serial map: bits=24 maxEntries=16777215
2019-12-18 10:39:29,300 INFO util.GSet: Computing capacity for map INodeMap
2019-12-18 10:39:29,300 INFO util.GSet: VM type = 64-bit
2019-12-18 10:39:29,300 INFO util.GSet: 1.0% max memory 889 MB = 8.9 MB
2019-12-18 10:39:29,300 INFO util.GSet: capacity = 2^20 = 1048576 entries
2019-12-18 10:39:29,300 INFO namenode.FSDirectory: ACLs enabled? false
2019-12-18 10:39:29,300 INFO namenode.FSDirectory: POSIX ACL inheritance enabled? true
2019-12-18 10:39:29,300 INFO namenode.FSDirectory: XAttrs enabled? true
2019-12-18 10:39:29,300 INFO namenode.NameNode: Caching file names occurring more than 10 times
2019-12-18 10:39:29,315 INFO snapshot.SnapshotManager: Loaded config captureOpenFiles: false, skipCaptureAcc
2019-12-18 10:39:29,315 INFO snapshot.SnapshotManager: SkipList is disabled
2019-12-18 10:39:29,331 INFO util.GSet: Computing capacity for map cachedBlocks
2019-12-18 10:39:29,331 INFO util.GSet: VM type = 64-bit
2019-12-18 10:39:29,331 INFO util.GSet: 0.25% max memory 889 MB = 2.2 MB
2019-12-18 10:39:29,331 INFO util.GSet: capacity = 2^18 = 262144 entries
2019-12-18 10:39:29,346 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
2019-12-18 10:39:29,346 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
2019-12-18 10:39:29,346 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
2019-12-18 10:39:29,362 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
2019-12-18 10:39:29,362 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache
2019-12-18 10:39:29,378 INFO util.GSet: Computing capacity for map NameNodeRetryCache
2019-12-18 10:39:29,378 INFO util.GSet: VM type = 64-bit
2019-12-18 10:39:29,378 INFO util.GSet: 0.029999999329447746% max memory 889 MB = 273.1 KB
2019-12-18 10:39:29,378 INFO util.GSet: capacity = 2^15 = 32768 entries
2019-12-18 10:39:29,471 INFO namenode.FSImage: Allocated new BlockPoolId: BP-503097948-137.43.75.180-1576665
2019-12-18 10:39:29,487 INFO common.Storage: Storage directory C:\Hadoop\hadoop-3.1.3\namenode has been succ
2019-12-18 10:39:29,596 INFO namenode.FSImageFormatProtobuf: Saving image file C:\Hadoop\hadoop-3.1.3\namenode
2019-12-18 10:39:29,800 INFO namenode.FSImageFormatProtobuf: Image file C:\Hadoop\hadoop-3.1.3\namenode\curr
2019-12-18 10:39:29,815 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2019-12-18 10:39:29,831 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid = 0 when meet shutdown.
2019-12-18 10:39:29,831 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at Pres006/137.43.75.180
*****/

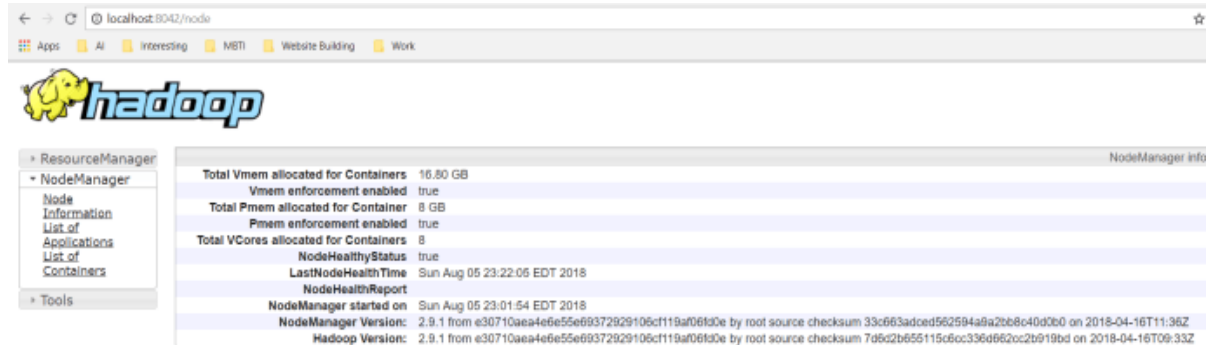
C:\Users>
```

- K- One more thing to do: copy **hadoop-yarn-server-timelineservice-3.1.2** from C:\bigdata\hadoop-3.1.2\share\hadoop\yarn\timelineservice to C:\bigdata\hadoop-3.1.2\share\hadoop\yarn
- L- We need to type **start-all.cmd** to start all nodes on one machine

You should have:

1- Node Manager

<http://localhost:8042>



NodeManager info	
Total Vmem allocated for Containers	16.80 GB
Vmem enforcement enabled	true
Total Pmem allocated for Container	8 GB
Pmem enforcement enabled	true
Total VCores allocated for Containers	8
NodeHealthyStatus	true
LastNodeHealthTime	Sun Aug 05 23:22:05 EDT 2018
NodeHealthReport	
NodeManager started on	Sun Aug 05 23:01:54 EDT 2018
NodeManager Version:	2.9.1 from e30710aea4e6e55e69372929106c119af06f9de by root source checksum 33c663adced562594a9a2bb8c40d0b0 on 2018-04-16T11:36Z
Hadoop Version:	2.9.1 from e30710aea4e6e55e69372929106c119af06f9de by root source checksum 7d6c2b655115c5cc336d662cc2b919bd on 2018-04-16T09:33Z

K- Working with HDFS

>notepad Sample.txt

Write anything and save the file

```
hdfs dfs -ls /  
hdfs dfs -mkdir /test  
hdfs dfs -copyFromLocal Sample.txt /test  
hdfs dfs -ls /test  
hdfs dfs -cat /test/Sample.txt
```

- 1- Hadoop reference: <https://dev.to/awwsmm/installing-and-running-hadoop-and-spark-on-windows-33kc#comments>
- 2- Hadoop Patch reference <https://github.com/cdarlint/winutils>

Install Spark

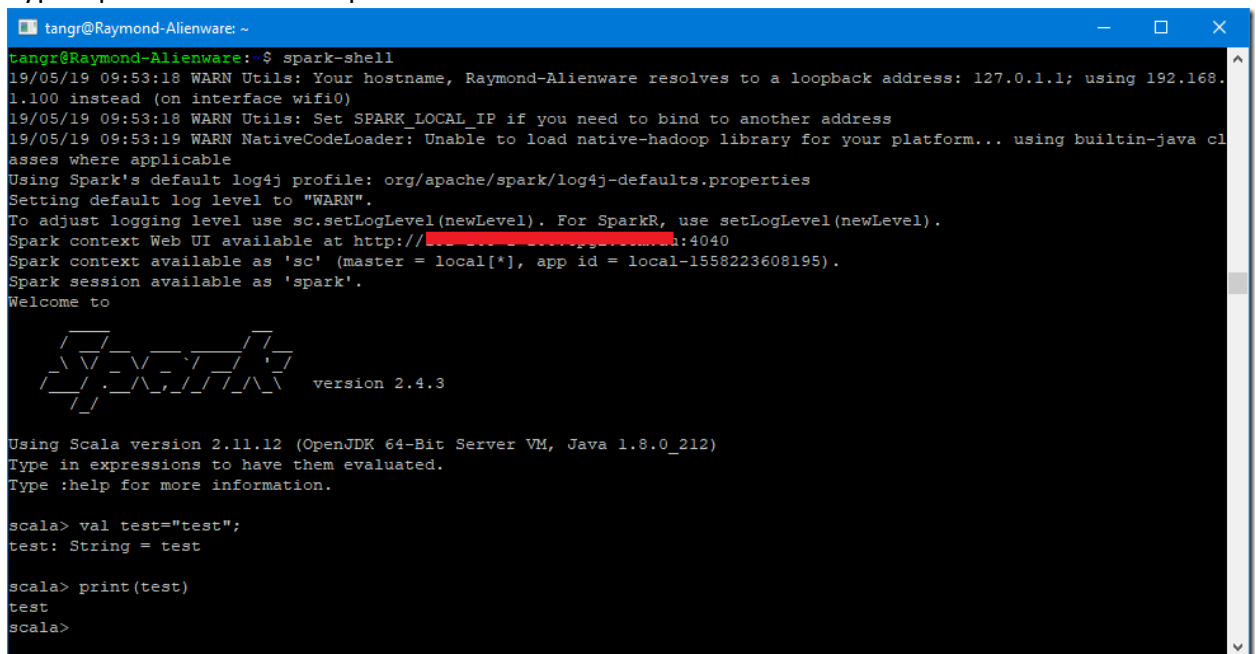
- A. Download Spark version 2.3 or 2.4 (<https://spark.apache.org/downloads.html>)

Download Apache Spark™

1. Choose a Spark release:
2. Choose a package type:
3. Download Spark: [spark-2.4.4-bin-hadoop2.7.tgz](#)
4. Verify this release using the 2.4.4 [signatures](#), [checksums](#) and [project release KEYS](#).

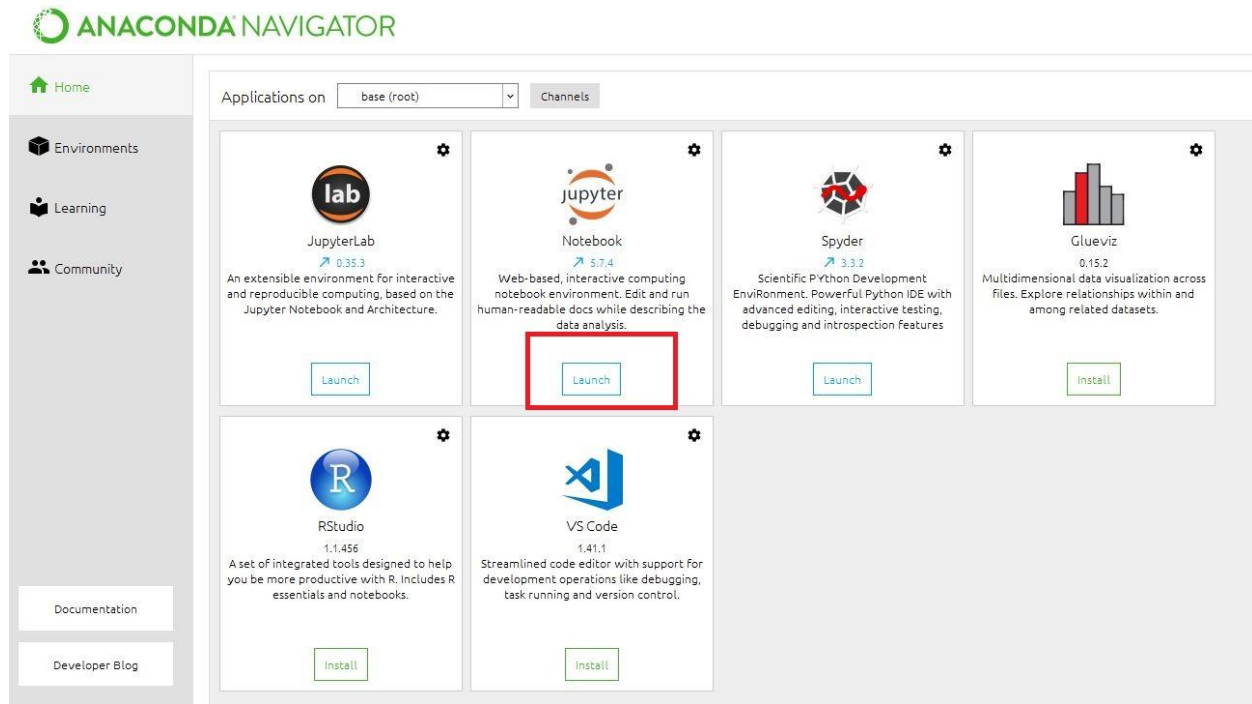
Note that, Spark is pre-built with Scala 2.11 except version 2.4.2, which is pre-built with Scala 2.12.

- B. Unzip spark-2.4.4-bin-hadoop2.7.tgz
- C. Put the folder in C:\bigdata\spark-2.4.4-bin-hadoop2.7
- D. Add to system variables
SPARK_HOME=C:\alaa\bigdata\spark-2.4.4-bin-hadoop2.7
- E. Add SPARK_HOME to path % SPARK_HOME %\bin
- F. Open cmd
- G. Type: spark-shell to start spark scala



```
tangr@Raymond-Alienware: ~  
tangr@Raymond-Alienware:~$ spark-shell  
19/05/19 09:53:18 WARN Utils: Your hostname, Raymond-Alienware resolves to a loopback address: 127.0.1.1; using 192.168.1.100 instead (on interface wifi0)  
19/05/19 09:53:18 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address  
19/05/19 09:53:19 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties  
Setting default log level to "WARN".  
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).  
Spark context Web UI available at http://192.168.1.100:4040  
Spark context available as 'sc' (master = local[*], app id = local-1558223608195).  
Spark session available as 'spark'.  
Welcome to  
  
          _ _ _ _ _  
         / _ _ _ _ \  
        / _ _ _ _ \  
       / _ _ _ _ \  
      / _ _ _ _ \  
     / _ _ _ _ \  
    / _ _ _ _ \  
   / _ _ _ _ \  
  / _ _ _ _ \  
 / _ _ _ _ \  
/_ _ _ _ _ \  
version 2.4.3  
  
Using Scala version 2.11.12 (OpenJDK 64-Bit Server VM, Java 1.8.0_212)  
Type in expressions to have them evaluated.  
Type :help for more information.  
  
scala> val test="test";  
test: String = test  
  
scala> print(test)  
test  
scala>
```

- H. Exit type :q
- I. You need to install Python (download: <https://www.python.org/downloads/windows/>)
- J. Add the variable: PYSPARK_DRIVER_PYTHON
PYSPARK_DRIVER_PYTHON= C:\Users\XXXXXXX
(Username)\AppData\Local\Programs\Python\Python37\python.exe
- K. Open cmd and type pyspark



Press Jupyter to start the web notebook.

3- Install findspark

Go to "search program and files" write "anaconda"

Choose "Anaconda Prompt"

Write "conda install -c conda-forge findspark"