Big Data Exam – S&A Big Data Academy

**Rules of the road:**

* At the end of exam time, you will send across screenshots of your answers (preferably in a similar format to how you did your presentations in Jupyter)
* Include your process (all the commands that you ran) so that I can award partial points if there is a wrong answer with good process. Please include applicable returns as well (example: show me the result of an “hdfs dfs -ls /your/path” so that I know a file is in place)
* Email of your exam responses **should be sent prior to 4:30pm**
* **Labeling must be included in your screenshots to let the grader know where one question ends and another begins**

As part of this practice test, you should be able to develop Apache Spark based solutions to process New York Stock Exchange (NYSE) data as per the requirements.

* Local Data Sets: **nyse\_all**
* It contains 2 directories **nyse\_stocks** and **nyse\_data**
* **nyse\_stocks** have metadata of the stocks. We will be using the data to get stock name.
* **nyse\_data** have end of day stock trade data.

#### Questions for this assignment

**Problem Statement**

Perform the count of records of end of day stock data of NYSE.

**Data Description**

\* Input Location: **/public/nyse\_all/nyse\_data**

\* Get count of all the records that are traded with out any filtering.

**Output Requirements**

Place the imported data in the HDFS directory  
/user/`whoami`/spark\_practice/problem1/data/nyse\_count

Replace **`whoami`** with **your OS user name**

\* Make sure output is saved only in one file

**Hint or Additional Information**

count does not return Data Frame. Either you can copy manually or convert the count into Data Frame to save back to HDFS.

**End of Problem**

**Problem Statement**

Get unique stock tickers that are traded in NYSE for the year 2010. Data should be sorted in ascending order by stock ticker.

**Data Description**

\* Input Location: **/public/nyse\_all/nyse\_data**

\* Field Names: **stockticker, tradedate, openprice, highprice, lowprice, closeprice, volume**

**Output Requirements**

Place the imported data in the HDFS directory  
/user/`whoami`/spark\_practice/problem2/data/unique\_stocks

Replace **`whoami`** with **your OS user name**

Make sure data is saved to only **one** text file. It should not contain header.

Data should be alphabetically sorted by stock ticker.

**End of Problem**

**Problem Statement**

Convert NYSE End of day Stock Data from csv file format to JSON and store it back to HDFS.

**Data Description**

\* Input Location: **/public/nyse\_all/nyse\_data**

\* Field Names: **stockticker, tradedate, openprice, highprice, lowprice, closeprice, volume**

\* stockticker should be represented as string and tradedate as integer or long or bigint.

\* prices are float/double, while volume is long or bigint.

**Output Requirements**

Place the imported data in the HDFS directory  
/user/`whoami`/spark\_practice/problem3/data/nyse\_data\_json

Replace **`whoami`** with **your OS user name**

Make sure **data** is saved to 8 files

Files should not be compressed

Column Names should match the names provided

Data Types should be relevant to the data.

Data should be sorted in Ascending order by tradedate and then stockticker.

**End of Problem**

**Problem Statement**

Get all the stocks which do not have corresponding stock names in ascending order by stock ticker.

**Data Description**

\* Input Location for stock transaction data: **/public/nyse\_all/nyse\_data**

\* Field Names for stock transaction data: **stockticker, tradedate, openprice, highprice, lowprice, closeprice, volume**

\* Input Location for stock metadata such as names: **/public/nyse\_all/nyse\_stocks**

\* First 2 fields are nothing but **stockticker** and **stockname**, you can ignore rest of the fields are leave them to default names and data types. Data type of both the fields is string.

**Output Requirements**

Place the imported data in the HDFS directory  
/user/`whoami`/spark\_practice/problem4/data/no\_stock\_names

Replace **`whoami`** with **your OS user name**

Make sure data is saved to only **one** text file. It should not contain header.

Data should contain only stock tickers and should be alphabetically sorted.

**End of Problem**

**Problem Statement**

Get all the stock names which are never traded. Stock Names should be in ascending order.

**Data Description**

\* Input Location for stock transaction data: **/public/nyse\_all/nyse\_data**

\* Field Names: **stockticker, tradedate, openprice, highprice, lowprice, closeprice, volume**

\* Input Location for stock metadata such as names: **/public/nyse\_all/nyse\_stocks**

**Output Requirements**

Place the imported data in the HDFS directory  
/user/`whoami`/spark\_practice/problem5/data/untraded\_stocks

Replace **`whoami`** with **your OS user name**

Make sure data is saved to only **one** text file. It should not contain header.

Data should be alphabetically sorted by stock ticker.

**End of Problem**

**Problem Statement**

Get stock name by using nyse\_stocks along with all the fields from nyse\_data. If there is no stock name then replace with a string "Stock Name Not Available"

**Data Description**

\* Input Location for stock transaction data: **/public/nyse\_all/nyse\_data**

\* Field Names for stock transaction data: **stockticker, tradedate, openprice, highprice, lowprice, closeprice, volume**

\* Input Location for stock metadata such as names: **/public/nyse\_all/nyse\_stocks**

**Output Requirements**

Place the imported data in the HDFS directory  
/user/`whoami`/spark\_practice/problem6/data/stock\_data\_with\_names

Replace **`whoami`** with **your OS user name**

Make sure **data** is saved to 8 files

Files should not be compressed

Files should be in text file format with ; as delimiter.

Fields should be in this order - **stockticker, tradedate, openprice, highprice, lowprice, closeprice, volume, stockname**

There should be no header for the data.

Data should be sorted in Ascending order by tradedate and then stockticker.

**End of Problem**

**Problem Statement**

Sort the data in ascending order by date and descending order by volume and save it in Avro format into Hive Table. Make sure data is saved into 4 files only while being loaded into Hive Table.

**Data Description**

\* Input Location: **/public/nyse\_all/nyse\_data**

\* Field Names: **stockticker, tradedate, openprice, highprice, lowprice, closeprice, volume**

\* stockticker should be represented as string and tradedate as integer or long or bigint.

\* prices are float/double, while volume is long or bigint.

**Output Requirements**

Create Hive table **nyse\_data\_avro** pointing to your Database **`whoami`**

Replace **`whoami`** with **your OS user name**

Make sure **data** is saved to 4 files

Column Names should match the names provided

Data Types should be relevant to the data.

Data should be sorted in Ascending order by tradedate and then in descending order by volume.

**End of Problem**