Please review the attached Class Project document after reading this document.  Please post any questions about the assignment in a discussion post under week 6. It is okay to work in small groups of 2 – 3 students on the project. 50,000 lines of data is fine for the size of the dataset. In addition, several small videos will be provided in weeks 6 and 7 to give insight into the project and to provide some demonstrations that may help you along the way.

Here are some helpful links that will guide you to complete the PySpark commands for the Class Project.

[**https://www.analyticsvidhya.com/blog/2016/09/comprehensive-introduction-to-apache-spark-rdds-dataframes-using-pyspark/**](https://www.analyticsvidhya.com/blog/2016/09/comprehensive-introduction-to-apache-spark-rdds-dataframes-using-pyspark/)

[**https://www.analyticsvidhya.com/blog/2016/10/using-pyspark-to-perform-transformations-and-actions-on-rdd/**](https://www.analyticsvidhya.com/blog/2016/10/using-pyspark-to-perform-transformations-and-actions-on-rdd/)

[**https://github.com/MingChen0919/learning-apache-spark/blob/master/notebooks/01-data-strcture/1.3-conversion-between-rdd-and-dataframe.ipynb**](https://github.com/MingChen0919/learning-apache-spark/blob/master/notebooks/01-data-strcture/1.3-conversion-between-rdd-and-dataframe.ipynb)

[**https://s3.amazonaws.com/assets.datacamp.com/blog\_assets/PySpark\_SQL\_Cheat\_Sheet\_Python.pdf**](https://s3.amazonaws.com/assets.datacamp.com/blog_assets/PySpark_SQL_Cheat_Sheet_Python.pdf)

**Hint from the Instructor:**  Probably 60 – 70% of what you need is in the spark-WordCount example.

**Hints from a prior student:** For this project, I created an inverted index of Stackoverflow posts according to tag. Then, I used native Spark and Spark SQL to query the data to find various metrics. I found that Spark code using pyspark looks like normal Python code.  Many of the operations in Spark have equivalents in Python. For instance, map and filter.

My biggest challenge involved using Spark SQL to split the tags for a post. This is not a task that is generally suited to SQL. This assignment required creating a row for each tag/post combination. **I found that creating an RDD with the tags and IDs, then converting it back to a DataFrame allowed me to accomplish what was needed.** In this case, working with RDDs seemed appropriate for the work of generating the inverted index. SQL was more appropriate for asking questions of the data.

Another challenge involved calculating the average score. I could not do this with the standard map and reduceByKey commands. I found combineByKey(), which is in itself a map/reduce algorithm. With this function, we pass in lambdas for a mapper, combiner and reducer. I can see many uses for combineByKey.

**Week 6**

1. Introduction to project – review all aspects of the project and resources provided above, and focus on getting required dataset from StackExchange through stack query
2. **No Deliverable**

**Week 7**

1. Continue configuring environment (confirm anaconda install, install findspark on vm, verify jupyter nb works)
2. Get all required files into HDFS/Jupyter notebook
3. Work through word count problem example
4. Work through the Java example within the project document.
5. **No Deliverable**

**Week 8**

1. "Word Count" our stack exchange file
2. Use Inverted Index Program to create our index with Stack exchange data
3. **Essential and Required: Provide write up in a MS Word document regarding the entire project and any data files from project. To earn full credit for the project, you must explain each line of code in your solution code to create the Inverted Index.**

**Misc Project Notes:**

**You will have to complete Lab 5 - installing Anaconda for this to work.  If you have not completed lab 5, then do so before starting the project.**

You'll have to**pip** install findspark at least on all VMs prior to starting the project.

[vagrant@c7401 ~]$ pip install findspark (remember to do this as the Vagrant/centos user on all VMs)

**The spark-wordcount.ipynb file is provided via WorldClass as an example. Walk through the notebook, run the cells, and ensure you understand what is happening at each step. This will give you an “idea” of how to go about starting your project.**

**Another example that you should review is the Wikipedia Inverted Index example that is listed in the Project document. The Java code used in the inverted index example is provided, however, note that for our project you will be using Spark rather than Java, but if you pay special attention to the Java “steps” you will have an idea of the steps required for the project inverted index…hint, hint.**

**Note that the lab instructions are vague intentionally, so some amount of struggle will be normal and part of the learning process! ☺**