Homework 5: HDFS, MapReduce and YARN

# Learning Objectives

In this exercise, you will learn basic Hadoop Commands:

* HDFS commands to upload and download files
* Use Sqoop and Flume to import / export data
* Run simple map-reduce jobs

# Setup

To complete this homework or follow along with the instructor’s class demos from your coursework, you must first complete the steps in this setup section. For this exercise we require the Cloudera Hadoop instance in the Docker environment. For more details on the how’s and why’s of managing the database environments used in this course, consult Lab 1. Instructions for this assignment:

1. Open the PowerShell prompt on Windows or the Terminal on MacOS.
2. Type:  
   cd adv-db-labs  
   to change the working directory to the repository folder. If you are in the correct spot, your command prompt should have adv-db-labs in it, for instance: PS adv-db-labs>
3. Change into the hadoop folder, type:  
   cd hadoop  
   if you are in the correct folder, your command prompt should have hadoop in it.
4. Bring up the Cloudera Hadoop environment, type:  
   docker-compose up -d
5. Check to make sure the environment is running, type:  
   docker-compose ps

The state of the **cloudera** container should be **Up** on **ports 7180, 8080,** and **80**.

1. You are now ready to connect to the running instance, and login as the **cloudera** user:   
   docker-compose exec cloudera bash -c "su -l cloudera"
2. Your prompt should now say: [cloudera@quickstart ~] where you are ready to execute lab commands!
3. Next, we need to install the demo files. Open a browser to <http://localhost:8888> and login to Hue as user: cloudera password: cloudera
4. Complete the Quick Start Wizard, installing examples for:
   1. Hive
   2. Impala
   3. Solr Search
   4. HBase Browser
   5. Pig Editor

# Exercises

Complete each of the following exercises. If you are unsure how to accomplish the task, please consult the coursework videos where there are explanations and demos. For each of the following write down the commands required to complete the task.

1. Upload all the documents in **datasets/text** into a folder called **text** in HDFS. What HDFS command must you run to verify the files are there after they are uploaded? Your answer should consist of the command you typed to complete the task.
2. In this part you will upload the **clickstream** dataset to HDFS. Specifically, create a **clickstream** folder in HDFS, then create a **logs** and **iplookup** folder inside the clickstream folder. Upload all of the **\*.log** files from the **datasets/clickstream** local folder into **clickstream/logs** in HDFS. Upload the **ip\_lookup.csv** file from the same folder into **clickstream/iplookup** on HDFS. Verify the files are there. Your answer should consist of the commands you typed to complete the task.
3. Use the MapReduce examples:  
   export MREX=/usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar   
   to perform a wordcount on the 2016 State of the Union address, saving the output to the HDFS folder **sotu2016**. Write down the commands to complete the task. How many times does the word **are** appear in the 2016 State of the Union address? Describe a process which could be done to make the wordcount more useful?
4. Type the following command to import the **fudgemart\_v3** database into the local **mysql** instance on the Hadoop client:  
   mysql -u root -p < ~/datasets/fudgemart/mysql.sql   
   The password is **cloudera**. Write down the commands you used to complete these tasks:  
   Use the **sqoop** utility to verify there are tables in the database by listing them from the **fudgemart\_v3** database. Next write a sqoop command to import Fudgemart products in the ‘Clothing’ department into a HDFS folder **/user/cloudera/fudgemart-clothing**
5. Let’s import HDFS data into MySQL. Record each command you type as your solution:
   1. Load **datasets/tweets/tweets.psv** into the HDFS folder **tweets**
   2. Login to MySQL: mysql -u root -p The password is **cloudera**. Create a database **twitter**
   3. Create a table called **tweets** inside the database **twitter** the table should have columns for id, timestamp, date time, username, and tweet\_text.
   4. Export the data from HDFS into the MySQL table.  
      **TIPS:** If your SQOOP job fails it is likely due to the table constraints such as selecting a data type too small for the imported data. Try to insert a row in the table using a sample from the HDFS data. This will help you to ensure your chosen data types will work.

# Turning it in

Take your copy and paste each of the solutions to the exercises into the submission template file included with this assignment. Make sure your name and SU email are at the top and turn in your work through the course learning management system.

# Tear-Down / Troubleshooting Common Issues

**IMPORTANT NOTE:** Hadoop is a complex environment with many interworking parts. Since there are many services running in the container, it is suggested that you bring down the container completely whenever:

1. You are finished with the lab, or
2. Whenever your computer goes to sleep. When that happens, the running Hadoop container is going to lose connectivity among its services.

When you are **finished with the homework or stopping work for an extended period of time** you should bring down the environment like this:

1. From the terminal window where you typed docker-compose up -d type in the following:  
   docker-compose down

If your **Hadoop container seems to have broken services**. Tear down the container and bring it back up again:

1. From the terminal window where you typed docker-compose up -d type in the following:  
   docker-compose down and then docker-compose up -d