Homework 7: Impala, HBase and HCatalog

# Learning Objectives

In this exercise, you will learn basic Hadoop Commands:

* Impala for Fast, SQL-like queries
* HBase for columnstore data
* HCatalog integration between Hive and HBase.

# 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. To be safe, let’s bring down the environment first:  
   docker-compose down
5. Bring up the Cloudera Hadoop environment, type:  
   docker-compose up -d
6. 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. When you need to connect to Impala:  
   **[cloudera@quickstart ~]** impala-shell
4. When you need to connect to HBase:  
   **[cloudera@quickstart ~]** hbase shell
5. When you need to connect to Hive:  
   **[cloudera@quickstart ~]** beeline -u jdbc:hive2://localhost:10000/default -n cloudera -p cloudera –-silent=true

# 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.

1. From Impala, use the two external tables **weblogs** created from **clickstream/logs\_noheader** and **iplookup** created from **clickstream/iplookup\_noheader** you created in the previous assignment to complete this question. Use the impala shell to answer the following questions, making sure to include the SELECT query you used to answer it.
   1. How many GET and POST requests are there in the weblogs?
   2. How many requests have Mac in the user agent?
   3. How many hosts (ip addresses) have Mac in the user agent?
2. From the HBase shell, include the commands required to complete the following.
   1. Create a table named **computers** with column family **info**.
   2. Issue HBase commands to write the following data to the table in the column family:

|  |  |  |  |
| --- | --- | --- | --- |
| Computer ID | Model | GB\_Ram | TB\_Disk |
| 1 | Dell | 16 | 1 |
| 2 | IBM | 32 | 1.5 |
| 3 | HP | 8 | 1 |
| 4 | Acer | 16 | 2 |

1. From the Hive shell, write an HQL statement to create an external Hive table from the HBase **computers** table. Then write a hive query to add up the total ram and disk across all computers. Your answer should include all HQL statements.
2. Use Hive to load the **iplookup** table you created from **clickstream/iplookup\_noheader** into and HBase table, with IP address as key. Include the HQL Queries you wrote to make the table and load the data as the answer to your question.
3. From the HBase shell, write an HBase query to retrieve the city and state columns for all rows in the **iplookup** table.

# 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