cloudera®

Ask Bigger Questions

CAP - Developing with Spark and Hadoop:

Homework Assignment Guide for Students

Homework: Create and Populate Tables in Impala or Hive

Files and Directories Used in this Homework

Exercise directory: `$DEV1/exercises/impala`

MySQL Database: `loudacre`

MySQL Tables: `device`

Data files (HDFS): `/loudacre/webpage`

In these exercises you will define Impala/Hive tables to model and view data in HDFS.

Note: The accounts data will not be used in this exercise but will in a subsequent exercise.

You may perform this and subsequent exercises in either Impala or Hive. Most of the instructions are the same whichever tool you choose; where the instructions differ, the difference is noted. Following whichever set of instructions you prefer; if you have no preference, we suggest Impala because it is faster.

Create and Query a Table in Impala or Hive

In this section you will use Impala or Hive to query the webpage data you imported in the previous exercise.

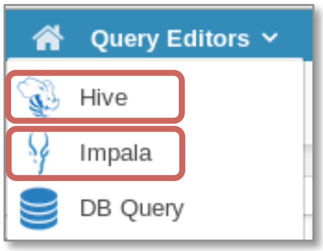
There are a number of ways to interact with Impala and Hive. In this exercise you will use the Impala or Hive Query Editor in Hue.

1. If you plan to use Hive rather than Impala for this or subsequent exercises, start the Hive server, which is not started by default, by entering the following two commands in a terminal window:

```
$ sudo service zookeeper-server start
$ sudo service hive-server2 start
```

2. Visit the Hue page in Firefox, as described earlier in the "Using HDFS" exercise.

3. Open either the Impala query editor or Hive query editor, by selecting the editor of your choice from the **Query Editors** menu.

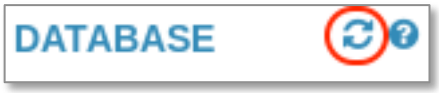


4. In the query editor pane on the right, enter a SQL command to create a table for the webpage data imported in the previous exercises:

```
CREATE EXTERNAL TABLE webpage
    (page_id SMALLINT,
     name STRING,
     assoc_files STRING)
   ROW FORMAT DELIMITED
      FIELDS TERMINATED BY '\t'
      LOCATION '/loudacre/webpage'
```

5. Click the **Execute** button to execute the command.

6. To see the table you just created, refresh the table list on the left.



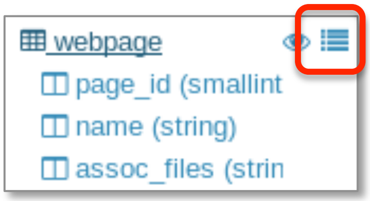
7. Click on the **webpage** table to see the column definitions.

8. Click the **New Query** button, then enter and execute a test query such as:

```
SELECT * FROM webpage WHERE name LIKE "ifruit%"
```

The resulting data is shown in the **Results** tab of the pane below the query.

9. Click on the Preview Sample Data icon to view a sampling of the table data.



webpage
page_id (smallint
name (string)
assoc_files (strin

Use Sqoop to Import Directly into Hive and Impala

In this section you will use Sqoop to import data from MySQL into HDFS and also automatically create the corresponding table in the Hive Metastore.

10. In a terminal window, import the `device` table directly into the Hive Metastore.

```
$ sqoop import \
  --connect jdbc:mysql://localhost/loudacre \
  --username training --password training \
  --fields-terminated-by '\t' \
  --table device \
  --hive-import
```

Use `--hive-import` for either Impala or Hive; this adds metadata to the Metastore, which both tools use.

Note: You may get a warning message that `chgrp` is unable to change the ownership of the generated files; you can disregard the warning, it does not affect the import.

11. Using Hue or the HDFS command line, review the imported data files. The Hive import copies the data to the default Hive warehouse location: `/user/hive/warehouse/device`

12. If you are using Impala, refresh the Impala metadata cache by entering the command in the Hue Impala Query Editor:

```
INVALIDATE METADATA
```

13. As in the previous exercise, view the columns and execute a test query:

```
SELECT * FROM device LIMIT 10;
```

This is the end of the Homework