Hortonworks Tutorial 1

Working with Hive

In this tutorial we learn how to upload files to HDP, how to create tables in Hive and upload data or just part of the data into tables.

Task 1. Using HDP to upload files

1. In our tutorials we are going to use the basketball dataset made available to public by Open Source Sports. We will only be using (basketball_master.csv, basketball_players.csv and basketball_teams.csv). You can download the files from this link: `http://opensourcesports.com/files/basketball/BasketballDB-20130121.zip`
2. Start VirutalBox and then Hortonworks as specified in Installation Guide till you see this screen below. Select "Files View" from a drop down list from the Off-canvas menu at the top (right hand side) (see Fig. 1).

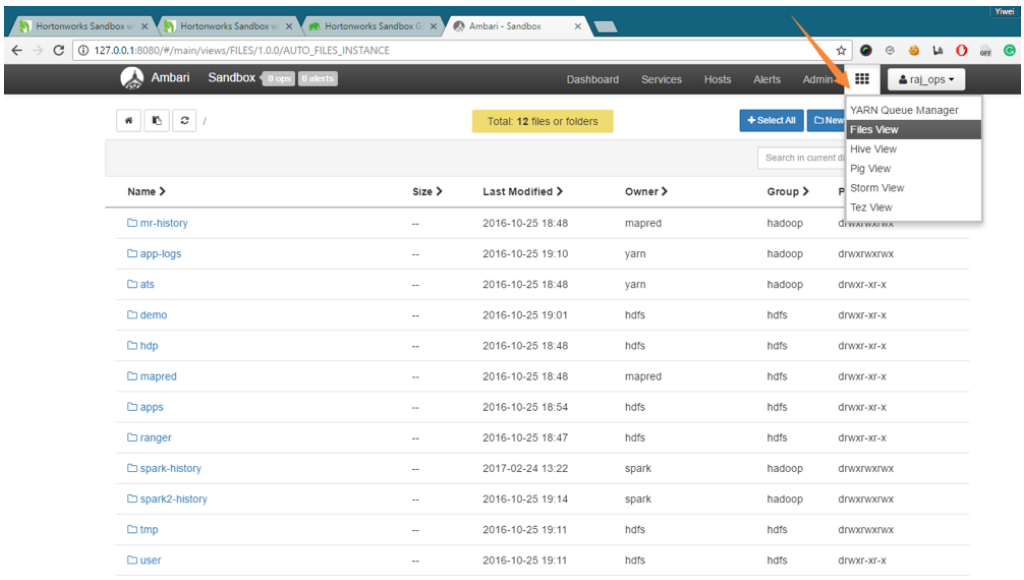

Figure 1: Files View

3. Click on the directory "tmp" at the bottom of the page. Then click on the "New Folder" button at the top right-hand side. Enter the folder name - "FIT5148", then click the "+Add" button.

4. You need to click on the column of "Permission" of the folder, "FIT5148" (see Fig. 2 (1)). Then, you see the menus on the top panel (see Fig. 2 (2)). Then, you can click on the "Permissions" menu and give "Write" permission to everyone (see Fig. 2 (3)).

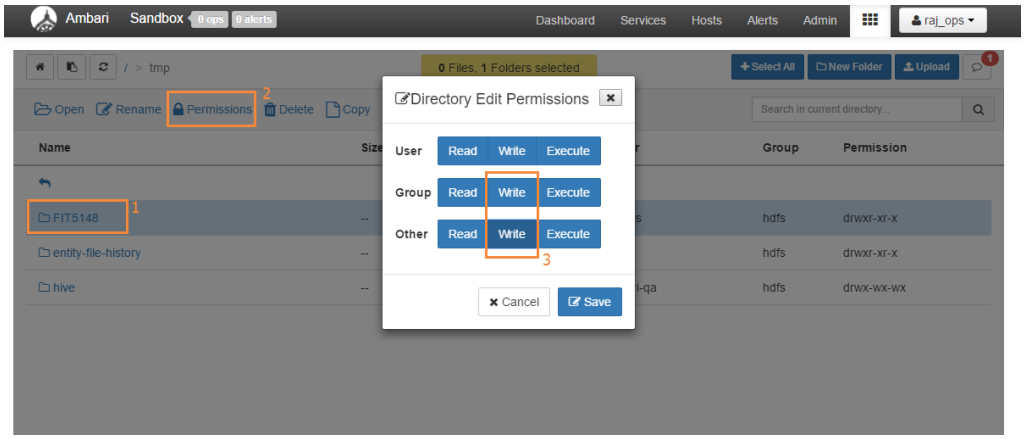

Figure 2: Change Permission

5. Move to the FIT5148 directory and upload 3 files that we mentioned earlier (first choose Browse and then click on Upload) (see Fig. 3).

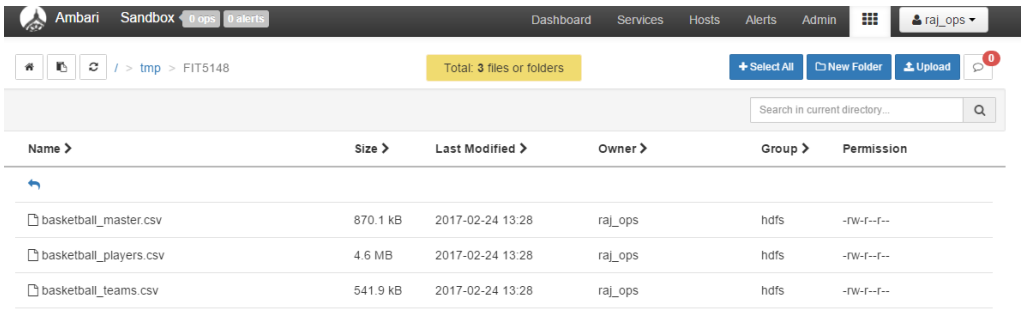

Figure 3: Upload files

6. From the Off-canvas menu at the top (right-hand side) select "Hive View" (see Fig. 4).

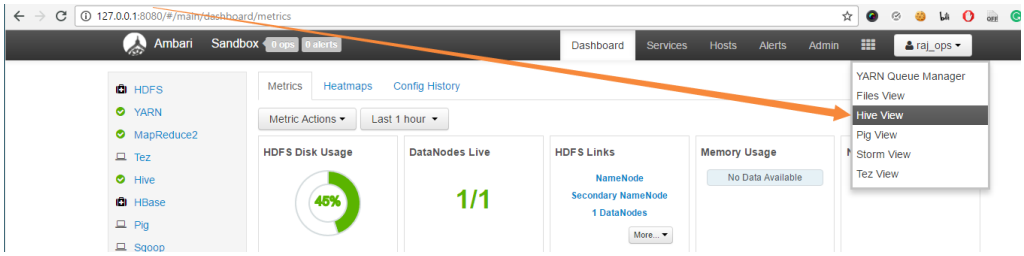

Figure 4: Hive view

7. In the Hive worksheet type: ”create table temp_master (col_value STRING);” Then, click on the ”Execute” button

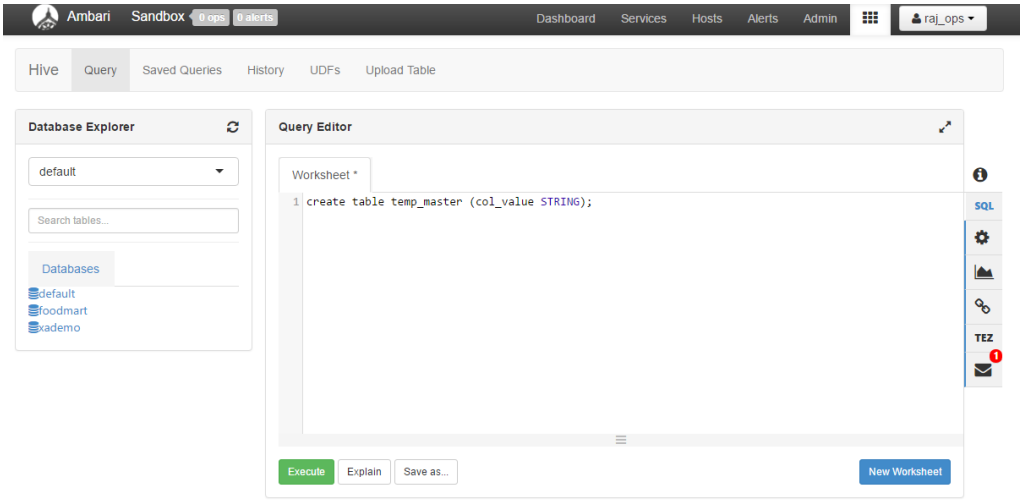

Figure 5: Create a table temp_master

8. If you refresh the page you will see that table is added under the default folder. If you click on the icon close to the table name a new worksheet opens up that loads data. You will see the table is empty. We havent yet uploaded data into the table. See Fig. 6.

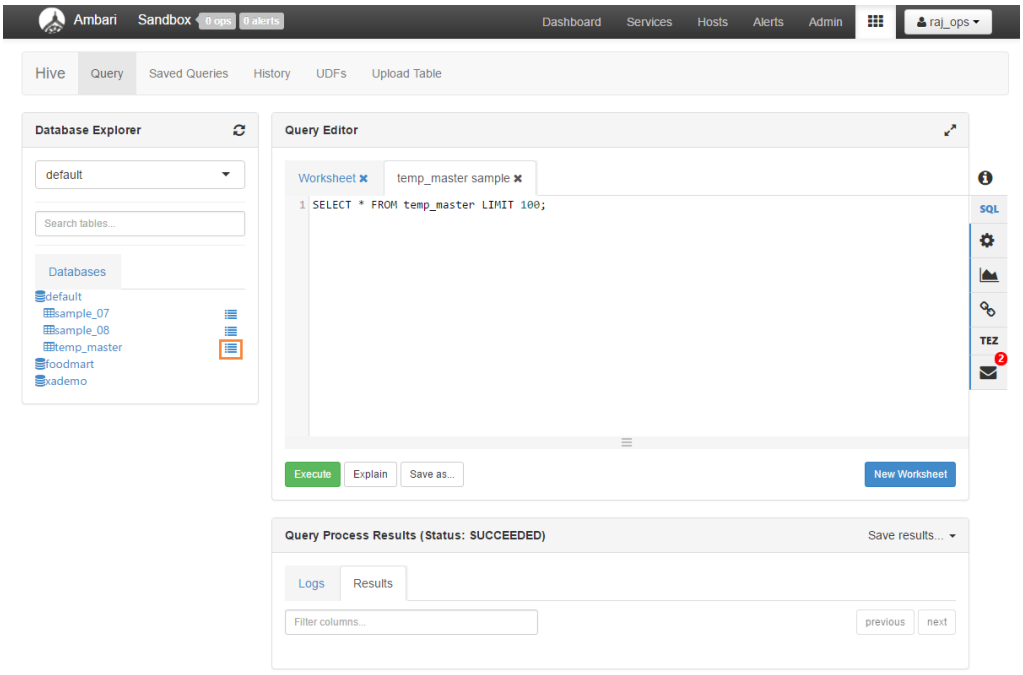

Figure 6: The temp_master table.

9. You need to load the data to this table by writing the following command: "LOAD DATA INPATH '/tmp/FIT5148/basketball_master.csv' OVERWRITE INTO TABLE temp_master;" (see Fig. 7).

   Note that, you need to clear the double quotation marks in the csv files, otherwise integer fields, e.g. height and weight, will be loaded as of String type thus cannot be manipulated properly. Open the csv file in a text editor and remove the double quotation marks. Then, upload the file again.

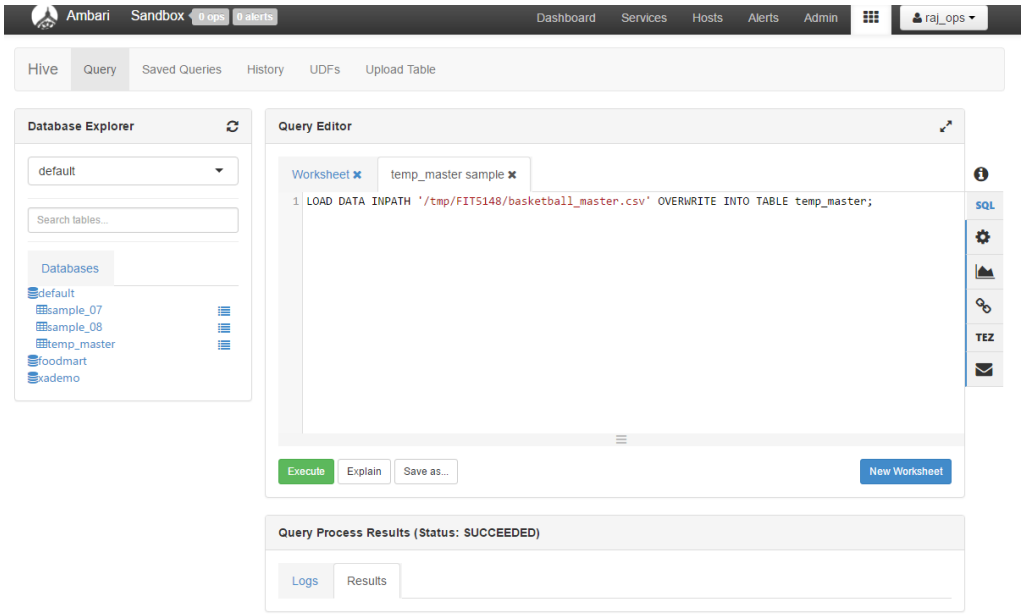

Figure 7: Load data into temp_master

10. Now you can execute the previous command to view records "(select *)".

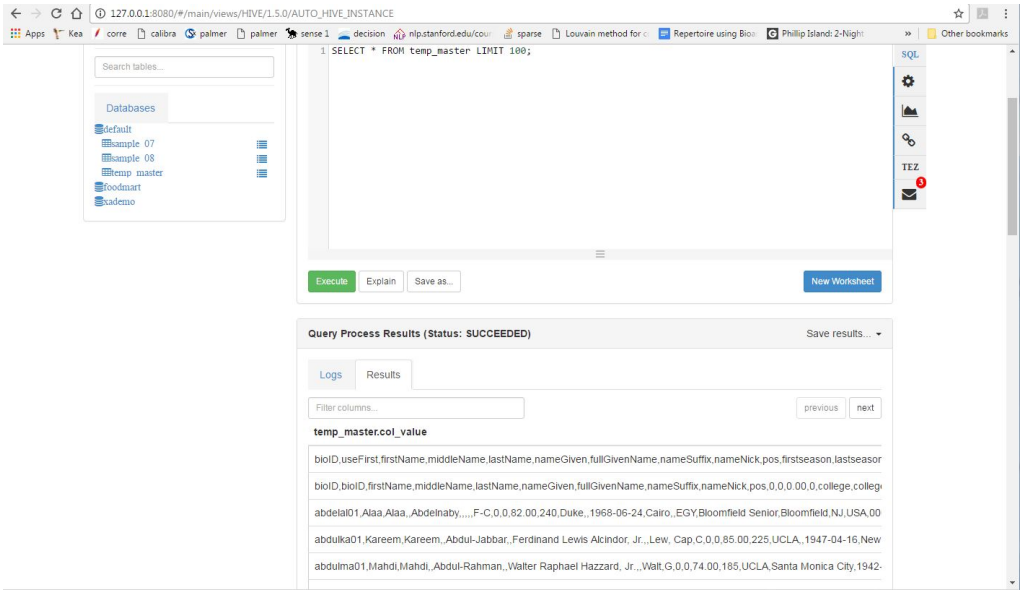

Figure 8: Select data from temp_master

Task 2. Your Task

1. Create the other two tables for teams (temp_teams) and players (temp_players) and upload data into them. View their contents.
2. In the previous steps we uploaded the data in all the columns into one column as String. We now need to change that. We will create new 3 tables and include the attributes/columns we will use later for data processing.
3. In a new worksheet write the code to create a table called basketball_teams. We will mainly consider 3 attributes for this table:"create table basketball_teams (year INT, tmID STRING, name STRING);"

   ```
   insert overwrite table basketball_teams
   SELECT
    regexp_extract(col_value, '^(?:([^,]*)\,?){1}', 1) year,
    regexp_extract(col_value, '^(?:([^,]*)\,?){3}', 1) tmID,
    regexp_extract(col_value, '^(?:([^,]*)\,?){10}', 1) name
   from temp_teams;
   ```

   In the above statement, regexp_extract was used. regexp_extract (string subject, string pattern, int index) returns the string extracted using the pattern. Some patterns are explained below:

   - ^: The beginning of a line/string
   - $*$: Zero or More of the preceding thing
   - 1: segment/column one
   - []: Range Operator, list of chars
   - [^,]*: capture any character except a comma, zero or many times
   - \, ?: Input ending with the comma character
   - (? :) {n}: Delimits a group of characters or patterns. It is a non-capturing group repeated n times. When the pattern repeats once in the first case the first element (year) on the comma separated array is captured. When the pattern repeats three times, the third element (tmID) is captured. When the pattern repeats 10 times then the tenth element (name) is captured.
4. View the results (select *).
5. Create the table basketball_master: "create table basketball_master (bioID STRING, fname STRING, lname STRING);"
6. Now we need to upload the data for only these columns from the temp_master.

   ```
   insert overwrite table basketball_master
   SELECT
    regexp_extract(col_value, '^(?:([^,]*)\,?){1}', 1) bioID,
    regexp_extract(col_value, '^(?:([^,]*)\,?){3}', 1) fname,
    regexp_extract(col_value, '^(?:([^,]*)\,?){5}', 1) lname
   from temp_master;
   ```
7. View the results using a "select" statement.
8. Create the table basketball_players: "create table basketball_players (playerID STRING, year int, tmID STRING, points int);""

   ```
   insert overwrite table basketball_players
   SELECT
   ```

```
 regexp_extract(col_value, '^(?:([^,]*)\,?){1}', 1) playerID,
 regexp_extract(col_value, '^(?:([^,]*)\,?){2}', 1) year,
 regexp_extract(col_value, '^(?:([^,]*)\,?){4}', 1) tmID,
 regexp_extract(col_value, '^(?:([^,]*)\,?){9}', 1) points
from temp_players;
```

9. View the results (select *).

Note: Prior to data processing and analysis (especially when you deal with big data), there is usually a need to do data cleaning and pre-processing. In our tutorials and assignment, this is kept to a minimum. You need to find out if the data requires any cleaning and address them (e.g. removing extra characters like , this will be necessary specially in Pig 2 tutorial).

If you need more practice or more information visit this link: `https://hortonworks.com/hadoop-tutorial/how-to-process-data-with-apache-hive`