Big Data bootcamp, Hive: hands on

# General tips

Use official documentation before googling: <https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DML>

<https://cwiki.apache.org/confluence/display/Hive/LanguageManual+UDF>

In console: beeline -u jdbc:hive2://localhost:10000

# Report

A file in sensible format (.docx and .pdf are fine) with all the steps you have taken and answers to questions. Include exact commands you used (text is OK for most points, when you should take screenshot it’s said so). If you had some problems, describe what the problem was, what you tried and what the solution was. Push report to git.

# Task

1. Get data from https://drive.google.com/file/d/1AOytyTA2MYYQOs6x9Q5M3n24gkbU8Jet/view

*These data are LV wikipipedia TOP 1000 articles by month, each file should contain one month and have header*

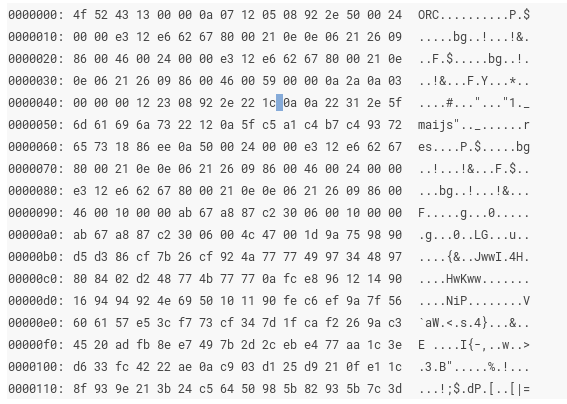
1. Clean data if necessary. Probably you will realize some cleaning is necessary few steps later. It’s ok. Come back and do some cleaning. It’s ok to do cleaning by hand. Extra points if you do it with script.
2. Put all data files on hdfs in directory /user/cloudera/wiki
3. Create database bootcamp
4. Create external table e\_wiki in database bootcamp, location: /user/cloudera/wiki
5. How can you be sure all the data from files are present in table? Think of some way how to do this check. Answer “it should be after my commands” is not enough, becasuse sometimes things break unexpectedly and sometimes people make a mistake. One of the simplest ways how to do it is check row count and number of items in new table.

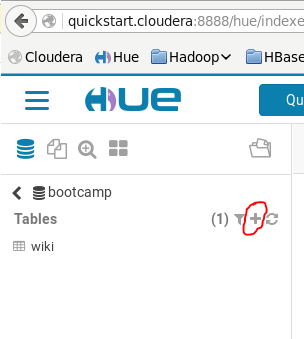
I wrote a command „SELECT \* FROM e\_wiki” in Hue and data was showed.

1. Create internal table t\_wiki and fill it with all the data from table e\_wiki and:
   1. Storage format ORC
   2. Schema:
      1. Year:int
      2. Month:int
      3. Article: string
      4. Views: int
      5. Rank: int
2. Open one of the files in the new table. Are there any differences in directory structure? Why? How the file content differs (screenshot for this point with file content) from data you have on your laptop?

Directory differences: „External table” uses data from external files, so the structure of the files stay the same – 7 .csv files in wiki folder and no additional files are created. But „Internal table” creates a directory with a table name and a file inside it with a default name, containing all the data.

File content differences: .csv is comma seperated values and is possible to read as a text in human language, but .orc is binary and not possible to understand the data.



1. Try creating a table trough the HUE – easy way!
   1. Click on + sign in Hue  
      
   2. Work your way trough GUI
   3. What are advantages and disadvantages of this approach? Can you think about one use case for each?

Advantages:

Header values are automatically created and the data type is also added.

Double quotes are automatically removed and it is easier to show that the value is INT

Disadvantages:

Errors while creating the table

1. Write a queries to get to know following. If you notice any weird answers, point them out and fix them if it can be done in step 2
   1. How many records are in table t\_wiki ?

select count(\*) from t\_wiki

5899

* 1. What’s the biggest view count in one month?
  2. What is the top article in all months? What’s the top real article in all months? (Sākumlapa isn’t a real article. Special:<something> isn’t a real article)

SELECT MAX(views) as max\_views FROM t\_wiki

92344

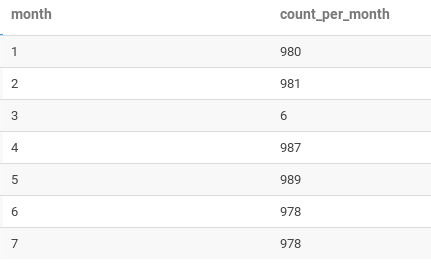
* 1. What’s the second biggest view count?

SELECT views FROM t\_wiki SORT BY views DESC LIMIT 1, 1

33803

* 1. How many records there are for each month?

SELECT month, COUNT(\*) as count\_per\_month FROM t\_wiki GROUP BY month;



* 1. How many real articles there are for each month?

As the data is cleaned there are no „not real” articles and the answer is the same as previous question.

* 1. Which article reached TOP 1000 in most months?
  2. What’s the average view count to reach TOP 1000?
  3. (for top achievers) Is there any correllation between page ranks between months? (hint: Hive udf exists for that, but you might rearrange data before. It’s OK to create a temporary table or a view for this)