Big Data Platforms - Home Assignment 1

Due to 11.11.2021

Required software and packages

The following software and packages required to be installed. You can either use Windows, Linux (Ubuntu|), Mac OS. (Alternative, there is an option to install Oracle Virtual Box and setup local VM. Virtual Box instructions can be provided separately. This option is less advised)

1. PyEnv environment for Python
2. Git
3. Python 3.8.5 via PyEnv
4. Pandas
5. Dask
6. PyArrow
7. SQLite
8. IDE for Python (Eclipse or any other preferable IDE)

Homework Tasks

Create local CSV file “mydata.csv” with 1000000 rows with columns (id, fruit, price, color). Use random value for rows, where fruit has one of the values ['Orange', 'Grape', 'Apple', 'Banana', 'Pineapple', 'Avocado'] and colors are ['Red', 'Green', 'Yellow', 'Blue']. Price should be random integer between 10 and 100. Filed id should be an index number starting from 1.

Task 1: CSV and SQL

1. Write Python code to create SQLite database “mydb.db” and create a table “mydata” with the schema of the “mydata.csv”
2. Write Python code to load “mydata.csv” into “mydata” table.
3. Write 2 different SQL statements with different conditions to retrieve different rows. Explain which parts of the statement are predicate and which parts are projection.

Task 2: CSV and Parquet

1. Write Python program that reads “mydata.csv” file and count number of lines
2. By using PyArrow, create Parquet file from the “mydata.csv”. Name Parquet file as “mydatapyarrow.parquet”
3. By using Dask, create Parquet file from the “mydata.csv”. Name Parquet file as “mydatadask.parquet”
4. By using Pandas, create Parquet file from the “mydata.csv”. Name Parquet file as “mydatapandas.parquet”

5. Examine generated Parquet files. Why do you think Dask generated Parquet file differently than PyArrow and Pandas? What might be explanation for this?

Task 3: Split CSV files

1. Write Python code that calculates size of “mydata.csv” in bytes. Define an integer variable “middle” which is the size of “mydata.csv” divided by 2.
2. Write a Python function first_chunk that count number of rows by reading **the byte range of the CSV** file, from 0 till the “middle”. Write a function last_chunk that count number of rows **by reading byte range of CSV** file from the “middle”+1 till the end of the file.
   Notice: Use seek to position into middle + 1 location. To correctly seek and read, you need to open file into binary mode and use decoder, for example to read the second chunk

```
f2= open("mydata.csv", "rb")
f2.seek(middle + 1, 0)
d2 = f2.read(p2).decode(encoding='utf-8')
```

   Notice: implement this function using pure python (no pandas)

3. Explain why total number of lines from the first chunk and second chunk is larger than the number of lines calculated in the step (1) of Task 2.
4. Suggest an algorithm to resolve the issue from the step (3) and implement it.
5. Check the algorithm of step (4) with multiple chunks. Define a chunk size to be 16MB. Write a function that process “mydata.csv “ in chunks and count number of lines for each chunk. For example, first chunk will be 0-16MB, second chunk 16MB-32BM, and so on, until the last chunk, which might be smaller.

Good luck!