PROJECT: Mini-HIVE
BIG DATA UE17CS313

SOURCE CODE DOCUMENTATION

AUTHORS

ANANTHARAM RU	PES1201700088
SHARANYA VENKAT	PES1201700218
KIRTHIKA GURUMURTHY	PES1201700230
RICHA	PES1201700688

Our Implementation Consists of the following files

1) Driver
2) Mapper
3) Identity mapper
4) Select_Reducer
5) Aggregate_Reducer
6) Project_Reducer

>> Table/Database is stored in the Hadoop HDFS directory. The schema for the table/database is stored on hdfs and the local system during the execution of the load command.

Driver.py

- Driver.py file is the main file aggregates all the mappers and reducers and decides which mapper and reducer to call based on the query passed by the user
- Schema is loaded in the driver
- Input database/Table is taken here
- Provides Error checking Mechanisms which include syntactical errors as well as ensures schema and table is loaded first before querying
- Maintains 2 log files
    - Hadoop_Logs - the logfile logs all the output of Hadoop jar commands run while calling respective mapper and reducer
    - Error_logs - captures any error that gets printed onto stderr
- Schema is deleted in the driver

Mapper.py

- The mapper takes input from the database and does the required action based on the query and outputs key-value pairs.
- The mapper implements the SELECT and PROJECT queries and handles the different errors that can occur.

    PROJECT :

        sys.argv[1] = 0

        sys.argv[2] = Column to project

    SELECT with WHERE:

        sys.argv[1] = 1

        sys.argv[2] = Column to be printed after processing the query

        sys.argv[3] = Column used in the WHERE condition

        sys.argv[4] = condition used

        sys.argv[5] = Condition value used

    SIMPLE SELECT:

        sys.argv[1] = 2

- For project query or queries where we have to select a column the mapper outputs key-value pairs where key is the column to be projected or selected and value is the value of each row for that column , that is, (column_name,column values).
- For 'SELECT *' queries the mapper outputs key-value pairs where each row is both the key and value, that is, (row,row).

Select_reducer.py

Takes in the key value pairs from the mapper and prints the final output in the desired format without making any changes.

Project_reducer.py

Takes the input from the mapper and removes all the duplicates and prints all the distinct values in that column.

Aggregate_reducer.py

Takes input from the mapper and performs a suitable aggregate function. It takes system arguments as 0,1,2 and performs count, min and max accordingly

Running Our scripts

Cmd: `python3 driver.py`

```
>> outputs the miniHIVE terminal where you can load database and query
```