Exp.No.: 5a

Design and test various schema models to optimize data storage and retrieval Using Hive

Aim:

To Design and test various schema models to optimize data storage and retrieval Using Hbase.

Procedure:

Step 1: Start Hive

Open a terminal and start Hive by running:

$hive

Step 2: Create a Database

Create a new database in Hive: hive>CREATE DATABASE financials;

```
hive> CREATE DATABASE financials;
OK
Time taken: 0.063 seconds
```

Step 3: Use the Database:

Switch to the newly created database: hive>use financials;

```
hive> use financials;
OK
Time taken: 0.57 seconds
```

Step 4: Create a Table:

Create a simple table in your database:

hive>CREATE TABLE finance_table(id INT, name STRING);

```
hive> CREATE TABLE finance_table( id INT, name STRING );
OK
Time taken: 2.013 seconds
```

Step 5: Load Sample Data:

You can insert sample data into the table:

hive>INSERT INTO finance_tableVALUES (1, 'Alice'), (2, 'Bob'), (3, 'Charlie');

```
hive> CREATE TABLE financial_table(id INT,name STRING);
OK
Time taken: 1.92 seconds
hive> INSERT INTO financial_table VALUES
    > (1,'Alice')
    > ,
    > (2,'Bob'),
    > (3,'Charlie');
Query ID = vinisha_20240922202602_b5ac396a-f9bc-4b55-89a5-a0e9f82f0544
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1727014182163_0001, Tracking URL = http://ubuntu.myguest.virtualbox.org:8088/proxy/application_1727014182163_0001
/
Kill Command = /home/vinisha/hadoop-3.4.0//bin/mapred job  -kill job_1727014182163_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-09-22 20:29:49,316 Stage-1 map = 0%,  reduce = 0%
2024-09-22 20:30:09,626 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 11.31 sec
2024-09-22 20:30:48,411 Stage-1 map = 100%,  reduce = 67%, Cumulative CPU 18.91 sec
2024-09-22 20:30:50,866 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 19.89 sec
MapReduce Total cumulative CPU time: 19 seconds 890 msec
Ended Job = job_1727014182163_0001
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to directory hdfs://localhost:9000/user/hive/warehouse/financials.db/financial_table/.hive-staging_hive_2024-09-22_20-26
-02_582_6525596755121584090-1/-ext-10000
Loading data to table financials.financial_table
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 19.89 sec   HDFS Read: 15725 HDFS Write: 293 SUCCESS
Total MapReduce CPU Time Spent: 19 seconds 890 msec
OK
Time taken: 324.436 seconds
hive>
```

Step 6: Query Your Data

Use SQL-like queries to retrieve data from your table:
hive>CREATE VIEW myview AS SELECT name, id FROM finance_table;

```
hive> CREATE VIEW myview AS SELECT name, id FROM finance_table;
OK
Time taken: 0.244 seconds
```

Step 7: View the data:

*To see the data in the view, you would need to query the view hive>SELECT*FROM myview;*

```
hive> SELECT*FROM myview;
OK
Alice   1
Bob     2
Charlie 3
Time taken: 0.22 seconds, Fetched: 3 row(s)
```

Step 8: Describe a Table:

You can describe the structure of a table using the DESCRIBE command:
hive>DESCRIBE finance_table;

```
hive> DESCRIBE finance_table;
OK
id                      int
name                    string
age                     int
Time taken: 0.729 seconds, Fetched: 3 row(s)
```

Step 9: Alter a Table:

You can alter the table structure by adding a new column: hive>ALTER TABLE finance_table ADD COLUMNS (age INT);

```
hive> ALTER TABLE finance_table ADD COLUMNS (age INT);
OK
Time taken: 0.188 seconds
```

Step 10: Quit Hive:

To exit the Hive CLI, simply type: hive>quit;

```
Time taken: 1.244 seconds, Fetched: 3 row(s)
hive> DESCRIBE financial_table;
OK
id                      int
name                    string
Time taken: 0.363 seconds, Fetched: 2 row(s)
hive> ALTER TABLE financial_table ADD COLUMNS(age INT);
OK
Time taken: 0.599 seconds
hive> quit;
```

Result:

Thus, the usage of various commands in Hive has been successfully completed.