**Data Analytics Laboratory**

**Task 11**

**Database Manipulation using HiveQL**

## Aim

To perform HiveQL commands to create, alter, drop databases and views

## Introduction to Hive

* Apache Hive is a data warehouse system for Apache Hadoop. Hive enables data summarization, querying, and analysis of data.
* Hive queries are written in HiveQL, which is a query language similar to SQL.
* Hive allows you to project structure on largely unstructured data. After you define the structure, you can use HiveQL to query the data without knowledge of Java or MapReduce.
* SELECT statement is used to retrieve the data from a table.
* WHERE clause works similar to a condition. It filters the data using the condition and gives you a finite result.
* The built-in operators and functions generate an expression, which fulfils the condition.

## Prerequisites

**1. What is Apache Hive?**

Apache Hive is a data warehouse system built on top of Hadoop and is used for analysing structured and semi-structured data. Hive gives SQL queries to perform an analysis. Hive it is not a database it gives you logical abstraction over the databases and the tables.

**2. Is Hive suitable to be used for OLTP systems? Why?**

No, it is not suitable for OLTP system since it does not offer insert and update at the row level.

**3. Where does the data of a Hive table gets stored?**

By default the hive tables are stored in HDFS directory – */user/hive/warehouse*. The user can configure the storage location making necessary changes in hive.metastore.warehouse.dir configuration parameter present in the hive-site.xml file.

**4. When should we use SORT BY instead of ORDER BY?**

SORT BY clause sorts the data using multiple reducers. ORDER BY sorts all of the data together using a single reducer. Hence, using ORDER BY will take a lot of time to execute a large number of inputs.

**5. What kind of applications is supported by Apache Hive?**

Hive supports all those client applications that are written in:

Java, PHP, Python, C++, Ruby by exposing its Thrift server.

## In-Lab Tasks

Start the hive environment

**hadoop@hadoop-laptop:~$ hive**

Creating and deletion of databases and tables

**hive> show databases;**

OK

default

Time taken: 3.602 seconds

**hive> create schema vrsec;**

OK

Time taken: 0.082 seconds

**hive> drop database if exists vrsec;**

OK

Time taken: 0.838 seconds

**hive> create database student;**

OK

Time taken: 0.024 seconds

Selecting a database and inserting rows into table

**hive> use student;**

OK

Time taken: 0.0070 seconds

**hive> show tables;**

OK

Time taken: 0.074 seconds

**hive> create table marks(regno string, sub1 int, sub2 int, sub3 int);**

OK

Time taken: 0.191 seconds

**hive> describe marks;**

OK

regno string

sub1 int

sub2 int

sub3 int

Time taken: 0.139 seconds

Altering the tables in database

**hive> alter table marks rename to subjectmarks;**

OK

Time taken: 0.349 seconds

**hive> describe marks;**

OK

Table marks does not exist

Time taken: 0.064 seconds

**hive> describe subjectmarks;**

OK

regno string

sub1 int

sub2 int

sub3 int

Time taken: 0.079 seconds

**hive> show tables;**

OK

subjectmarks

Time taken: 0.061 seconds

**hive> alter table subjectmarks add columns(sub4 int);**

OK

Time taken: 0.087 seconds

**hive> describe subjectmarks;**

OK

regno string

sub1 int

sub2 int

sub3 int

sub4 int

Time taken: 0.069 seconds

**hive> alter table subjectmarks replace columns(name string, total int);**

OK

Time taken: 0.063 seconds

**hive> describe subjectmarks;**

OK

name string

total int

Time taken: 0.051 seconds

Importing data from csv file to a table

**hive> create table studentdetails(id string, name string, age int) row format delimited fields terminated by ',' stored as textfile;**

OK

Time taken: 0.051 seconds

**hive> show tables;**

OK

studentdetails

subjectmarks

Time taken: 0.047 seconds

**hadoop@hadoop-laptop:~/Desktop$ touch hivefile.txt**

**hadoop@hadoop-laptop:~/Desktop$ nano hivefile.txt**

**hadoop@hadoop-laptop:~/Desktop$ cat hivefile.txt**

001,arun,30

002,siddhu,27

003,magilan,35

004,mark,39

**hadoop@hadoop-laptop:~/Desktop$ hdfs dfs -put hivefile.txt /hivefile.txt**

**hive> load data inpath '/hivefile.txt' overwrite into table studentdetails;**

Loading data to table student.studentdetails

rmr: DEPRECATED: Please use 'rm -r' instead.

Deleted /user/hive/warehouse/student.db/studentdetails

OK

Time taken: 0.235 seconds

Selecting, displaying the content stored in tables.

**hive> select \* from studentdetails;**

OK

001 arun 30

002 siddhu 27

003 magilan 35

004 mark 39

Time taken: 0.162 seconds

**hive> select \* from studentdetails where id='001';**

Total MapReduce jobs = 1

Launching Job 1 out of 1

Number of reduce tasks is set to 0 since there's no reduce operator

Starting Job = job\_1613923031783\_0001, Tracking URL = http://localhost:8088/proxy/application\_1613923031783\_0001/

Kill Command = /home/hadoop/Training/CDH4/hadoop-2.0.0-cdh4.0.0/bin/hadoop job -Dmapred.job.tracker=localhost:10040 -kill job\_1613923031783\_0001

Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0

2021-02-21 11:08:28,236 Stage-1 map = 0%, reduce = 0%

2021-02-21 11:08:32,474 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 0.53 sec

MapReduce Total cumulative CPU time: 530 msec

Ended Job = job\_1613923031783\_0001

MapReduce Jobs Launched:

Job 0: Map: 1 Accumulative CPU: 0.53 sec HDFS Read: 0 HDFS Write: 0 SUCESS

Total MapReduce CPU Time Spent: 530 msec

OK

001 arun 30

Time taken: 9.764 seconds

**hive> select \* from studentdetails sort by id;**

Total MapReduce jobs = 1

Launching Job 1 out of 1

Number of reduce tasks not specified. Estimated from input data size: 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 mapred.reduce.tasks=<number>

Starting Job = job\_1613923031783\_0002, Tracking URL = http://localhost:8088/proxy/application\_1613923031783\_0002/

Kill Command = /home/hadoop/Training/CDH4/hadoop-2.0.0-cdh4.0.0/bin/hadoop job -Dmapred.job.tracker=localhost:10040 -kill job\_1613923031783\_0002

Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1

2021-02-21 11:08:46,430 Stage-1 map = 0%, reduce = 0%

2021-02-21 11:08:50,622 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 0.64 sec

2021-02-21 11:08:51,659 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 0.64 sec

2021-02-21 11:08:52,713 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 1.62 sec

MapReduce Total cumulative CPU time: 1 seconds 620 msec

Ended Job = job\_1613923031783\_0002

MapReduce Jobs Launched:

Job 0: Map: 1 Reduce: 1 Accumulative CPU: 1.62 sec HDFS Read: 0 HDFS Write: 0 SUCESS

Total MapReduce CPU Time Spent: 1 seconds 620 msec

OK

001 arun 30

002 siddhu 27

003 magilan 35

004 mark 39

Time taken: 11.034 seconds

**hive> select \* from studentdetails order by name;**

Total MapReduce jobs = 1

Launching Job 1 out of 1

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 mapred.reduce.tasks=<number>

Starting Job = job\_1613923031783\_0003, Tracking URL = http://localhost:8088/proxy/application\_1613923031783\_0003/

Kill Command = /home/hadoop/Training/CDH4/hadoop-2.0.0-cdh4.0.0/bin/hadoop job -Dmapred.job.tracker=localhost:10040 -kill job\_1613923031783\_0003

Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1

2021-02-21 11:09:09,729 Stage-1 map = 0%, reduce = 0%

2021-02-21 11:09:12,863 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 0.51 sec

2021-02-21 11:09:13,932 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 0.51 sec

2021-02-21 11:09:14,966 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 0.51 sec

2021-02-21 11:09:16,016 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 1.85 sec

MapReduce Total cumulative CPU time: 1 seconds 850 msec

Ended Job = job\_1613923031783\_0003

MapReduce Jobs Launched:

Job 0: Map: 1 Reduce: 1 Accumulative CPU: 1.85 sec HDFS Read: 0 HDFS Write: 0 SUCESS

Total MapReduce CPU Time Spent: 1 seconds 850 msec

OK

001 arun 30

003 magilan 35

004 mark 39

002 siddhu 27

Time taken: 11.05 seconds

**Results**

Create, alter, drop databases and views was performed using HiveQL Commands

**Faculty Signature**