**Assignment 7.3**

**Hive Data Definitions:**

Hive Data Definition Language (DDL) is a subset of Hive SQL statements that describe the data structure in Hive by creating, deleting, or altering schema objects such as databases, tables, views, partitions, and buckets.

HiveQL DDL statements are:

* CREATE DATABASE/SCHEMA, TABLE, VIEW, FUNCTION, INDEX
* DROP DATABASE/SCHEMA, TABLE, VIEW, INDEX
* TRUNCATE TABLE
* ALTER DATABASE/SCHEMA, TABLE, VIEW
* MSCK REPAIR TABLE (or ALTER TABLE RECOVER PARTITIONS)
* SHOW DATABASES/SCHEMAS, TABLES, TBLPROPERTIES, VIEWS, PARTITIONS, FUNCTIONS, INDEX[ES], COLUMNS, CREATE TABLE
* DESCRIBE DATABASE/SCHEMA, table\_name, view\_name

Create Database Statement

1. hive**>** CREATE SCHEMA userdb;
2. hive**>** SHOW DATABASES;

Drop database

hive**>** DROP DATABASE IF EXISTS userdb;

**Hive Data Manipulations**:

These commands are used to put data into tables and to extract data from tables to the filesystem.

DML (Data Manipulation Language) commands in Hive are used for inserting and querying the data from hive tables once the structure and architecture of the database has been defined using the DDL commands listed above.

Data can be loaded into Hive tables using –

* LOAD command
* Insert command

Usage of LOAD Command for Inserting Data Into Hive Tables



If the keyword LOCAL is not specified, then Hive will need absolute URI of the file. However, if local is specified then it assumes the following rules -

It will assume it’s an HDFS path and will try to search for the file in HDFS.

If the path is not absolute, then hive will try to locate the file in the /user/ in HDFS.

Using the OVERWRITE keyword while importing means the data will be ingested i.e. it will delete old data and put new data otherwise it would just append the new data. The contents of the target table will be deleted and replaced by the files referred to by file path; otherwise the files referred by file path will be added to the table.

**Hive QL Manipulations:**

The Hive Query Language (HiveQL) is a query language for Hive to process and analyze structured data in a Metastore.

Eg. Select statement with where clause

SELECT [ALL | DISTINCT] select\_expr, select\_expr, ...

FROM table\_reference

[WHERE where\_condition]

[GROUP BY col\_list]

[HAVING having\_condition]

[CLUSTER BY col\_list | [DISTRIBUTE BY col\_list] [SORT BY col\_list]]

[LIMIT number];

There are four types of operators in Hive:

* Relational Operators
* Arithmetic Operators
* Logical Operators
* Complex Operators

Hive also has many built in functions for data operations.Some of the examples are

Round(),Upper(),from\_unixtime(),to\_date() etc.

### Example

The following queries demonstrate some built-in functions:

### round() function

hive> SELECT round(2.6) from temp;

On successful execution of query, you get to see the following response:

3.0

Joins are also supported by HQL.

JOIN is a clause that is used for combining specific fields from two tables by using values common to each one. It is used to combine records from two or more tables in the database. It is more or less similar to SQL JOIN.

## Syntax

join\_table:

table\_reference JOIN table\_factor [join\_condition]

| table\_reference {LEFT|RIGHT|FULL} [OUTER] JOIN table\_reference

join\_condition

| table\_reference LEFT SEMI JOIN table\_reference join\_condition

| table\_reference CROSS JOIN table\_reference [join\_condition]

There are different types of joins given as follows:

* JOIN

JOIN clause is used to combine and retrieve the records from multiple tables. JOIN is same as OUTER JOIN in SQL. A JOIN condition is to be raised using the primary keys and foreign keys of the tables.

* LEFT OUTER JOIN

A LEFT JOIN returns all the values from the left table, plus the matched values from the right table, or NULL in case of no matching JOIN predicate.

* RIGHT OUTER JOIN

A RIGHT JOIN returns all the values from the right table, plus the matched values from the left table, or NULL in case of no matching join predicate.

* FULL OUTER JOIN

The HiveQL FULL OUTER JOIN combines the records of both the left and the right outer tables that fulfil the JOIN condition. The joined table contains either all the records from both the tables, or fills in NULL values for missing matches on either side.