Database Operations in HIVE

**1. Create a database**

Syntax:

create database database\_name;

Example:

create database myhiveDB;

Output:

**2. Creating a table**

**Syntax:**

create table database.tablename(columns);

**Example:**

create table myhiveDB.myhivetbl01(id int,name string);

Here id and string are the two columns.

**Output :**

**3. Display Database**

**Syntax:**

show databases;

**Output:**Display the databases created.

**4. Describe Database**

**Syntax:**

describe database database\_name;

**Example:**

describe database myhiveDB;

**Output:**Display the HDFS path of a particular database.

5. Use database;

Syntax

use database\_name;

**6. DROP TABLE**

**It** used to drop a table inside the hive. Hive will remove all of its data and metadata from the hive meta-store. The hive DROP TABLE statement comes with a **PURGE**option.  In case if the PURGE option is mentioned the data will be completely lost and cannot be recovered later but if not mentioned then data will move to .Trash/current directory.

**Syntax:**

DROP TABLE [IF EXISTS] table\_name [PURGE];

7. ALTER TABLE

Used to perform alterations on the tables. We can modify multiple numbers of properties associated with the table schema in the Hive.

**i. Renaming Table Name**

ALTER TABLE with RENAME is used to change the name of an already existing table in the hive.

**Syntax:**

ALTER TABLE <current\_table\_name> RENAME TO <new\_table\_name>;

**ii. ADD Columns**

**Syntax:**

ALTER TABLE <table\_name> ADD COLUMNS (<col-name>  <data-type>  COMMENT ”, <col-name>  <data-type>  COMMENT ”, ….. )

**Command:**

Let’s add a column contact to the customer table that we have obtained after renaming the demo.

ALTER TABLE customer ADD COLUMNS ( contact BIGINT COMMENT ‘Store the customer contact number’);

**iii. CHANGE Column**

CHANGE in ALTER TABLE is used to change the name or data type of an existing column or attribute.

**Syntax:**

ALTER TABLE <table\_name> CHANGE <column\_name> <new\_column\_name> <new\_data\_type>;

**Command:**

Let’s change the demo\_name attribute to customer\_name.

ALTER TABLE customer CHANGE demo\_name customer\_name STRING;

**iv. REPLACE Column**

The REPLACE with ALTER TABLE is used to remove all the existing columns from the table in Hive. The attributes or columns which are added in the ALTER TABLE REPLACE statement will be replaced with the older columns.

**Syntax:**

ALTER TABLE <table\_name> REPLACE COLUMNS (

<attribute\_name> <data\_type>,

<attribute\_name> <data\_type>,

.

.

.

);

For example in our customer table, we have 2 attributes customer\_name and contact. If we want to remove the contact attribute the query should be like as shown below.

**Command:**

ALTER TABLE customer REPLACE COLUMNS (

customer\_name STRING

);

### 8.ALTER Database Command -------

With the helpof the below command, we can add database properties or modify the properties we have added. DBPROPERTIES takes multiple arguments in the form of a key-value pair.

**Syntax:**

DATABASE or SCHEMA is the same thing we can use any name. SCHEMA in ALTER is added in hive 0.14.0 and later.

ALTER (DATABASE|SCHEMA) <database\_name> SET DBPROPERTIES ('<property\_name>'='<property\_value>',..);

Use ALTER to add properties to the database

ALTER DATABASE student SET DBPROPERTIES ( ' owner ' = ' GFG' , ' Date ' = ' 2020-5-6 ');

### 9. Truncate table

### All the data in the table is deleted, but the table metadata and structure remain intact.

### Truncating a table is a faster operation than dropping a table because it doesn’t involve deleting and recreating the table.

### Truncating a table cannot be undone, and the deleted data cannot be recovered.

Syntax:

TRUNCATE [TABLE] table\_name [PARTITION partition\_spec];

partition\_spec: (partition\_column = partition\_col\_value, partition\_column = partition\_col\_value, ...)

Example:

TRUNCATE TABLE hql.customer;

DML Commands

1. LOAD command

Load data from any file to the database

In Hive, if unmatched data is loaded then it stores the Null value at the position of unmatched tuple. It will not throw any exception.

Syntax :

LOAD data [LOCAL] inpath <file path> into table [tablename]

Example

LOAD DATA LOCAL INPATH '/home/training/Desktop/clients.csv' INTO TABLE clientstbl;

2. SELECT COMMAND

**The SELECT statement in Hive is similar to the SELECT statement in SQL used for retrieving data from the database.**

**Syntax:**

**SELECT col1,col2 FROM tablename;**

**Example**

**Select \* from emp;**

**Select salesid,salesamount from sales;**

### 3. INSERT Command

The **INSERT** command in Hive loads the data into a Hive table and partition.

#### a. INSERT INTO

INSERT INTO statement appends the data into existing data in the table or partition. INSERT INTO statement works from Hive version 0.8.

Syntax:

INSERT INTO TABLE tablename1 [PARTITION (partcol1=val1, partcol2=val2 ...)] select\_statement1 FROM from\_statement;

Example:

#### b. INSERT OVERWRITE

The **INSERT OVERWRITE** table overwrites the existing data in the table or partition.

**Syntax:**

INSERT OVERWRITE TABLE tablename1 [PARTITION (partcol1=val1, ..) [IF NOT EXISTS]] select\_statement FROM from\_statement;

**Example:**

INSERT OVERWRITE DIRECTORY '/user/data/output/export' ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' SELECT \* FROM emp.employee;

### 4. DELETE command

The DELETE statement in Hive deletes the table data. If the WHERE clause is specified, then it deletes the rows that satisfy the condition in where clause.

The DELETE statement can only be used on the hive tables that support ACID.

Syntax:

DELETE FROM tablename [WHERE expression];

### 5. UPDATE Command

The update can be performed on the hive tables that support ACID.

The UPDATE statement in Hive deletes the table data. If the WHERE clause is specified, then it updates the column of the rows that satisfy the condition in WHERE clause.

Partitioning and Bucketing columns cannot be updated.

Syntax:

UPDATE tablename SET column = value [, column = value ...] [WHERE expression];

6. EXPORT Command

The Hive EXPORT statement exports the table or partition data along with the metadata to the specified output location in the HDFS.

Metadata is exported in a \_metadata file, and data is exported in a subdirectory ‘data.’

Syntax:

EXPORT TABLE tablename [PARTITION (part\_column="value"[, ...])] TO

'export\_target\_path' [ FOR replication('eventid') ];

Example:

export table user\_info to 'hdfs\_exports\_location/user\_info';

7. IMPORT Command

The Hive IMPORT command imports the data from a specified location to a new table or already existing table.

Syntax:

IMPORT [[EXTERNAL] TABLE new\_or\_original\_tablename [PARTITION (part\_column="value"[, ...])]]

FROM 'source\_path' [LOCATION 'import\_target\_path']

Example

import table emp from 'hdfs\_imports\_location/emp1;