Session 7

Assignment 3

|  |  |
| --- | --- |
| **Prepared For:** | AcadGild |
|  |  |
| **Document Approval:** | **AcadGild** |
|  |  |
|  |  |
|  |  |
|  |  |
| **Project Title:** | Session 7 – Assignment 3 |
|  |  |
| **Prepared By:** | Duncan Burgess |
|  |  |
|  | dburgess@duncb.com |
|  |  |
| **Primary Engineer:** | Duncan Burgess |
|  |  |
| **Document Reference:** | **Session 7 – Assignment 3** |
|  |  |
| **Start Date:** | 21/09/2017 |
|  |  |
|  |  |

# 

# Contents

[Contents 2](#_Toc493771067)

[Change History 3](#_Toc493771068)

[1. Problem Statement 4](#_Toc493771069)

[2. DDL Commands on Databases in Hive 5](#_Toc493771070)

[2.1. Create Database in Hive 5](#_Toc493771071)

[2.2. Drop Database in Hive 5](#_Toc493771072)

[2.3. Describe Database Command in Hive 5](#_Toc493771073)

[2.4. Alter Database Command in Hive 6](#_Toc493771074)

[2.5. Show Database Command in Hive 6](#_Toc493771075)

[2.6. Use Database Command in Hive 6](#_Toc493771076)

[2.7. Create Table Command in Hive 7](#_Toc493771077)

[2.8. DROP Table Command in Hive 7](#_Toc493771078)

[2.9. TRUNCATE Table Command in Hive 8](#_Toc493771079)

[2.10. ALTER Table Command in Hive 8](#_Toc493771080)

[2.11. DESCRIBE Table Command in Hive 8](#_Toc493771081)

[2.12. Show Table Command in Hive 9](#_Toc493771082)

[3. DML Commands in Hive 9](#_Toc493771083)

[3.1. Usage of LOAD Command for Inserting Data Into Hive Tables 10](#_Toc493771084)

[3.2. Exporting Data 11](#_Toc493771085)

# Change History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Document Revision** | **Date** | **Authored By** | **Authorised By** | **Sections Affected** | **Reason for Change** |
| Rev 01 | 21/09/2017 | Duncan Burgess |  | All | Initial release. |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Problem Statement

Explain the below concepts with an example in brief.

● Hive Data Definitions

● Hive Data Manipulations

● HiveQL Manipulations

Examples will be given using the following dataset

**emp\_details.txt'**

Amit,Big Data,1,BBSR

Venkat,Web Technology,2,BBSR

Aditya,DBA,1,BNG

Ravinder,Java,2,BBSR

Sunil,C#,1,BBSR

Anil,ASP,2,BNG

Mihir,Big Data,3,BBSR

Mohit,Java,1,BBSR

# DDL Commands on Databases in Hive

## Create Database in Hive

As the name implies, this DDL command in Hive is used for creating databases.

CREATE (DATABASE) [IF NOT EXISTS] database\_name

[COMMENT database\_comment]

[LOCATION hdfs\_path]

[WITH DBPROPERTIES (property\_name=property\_value, ...)];

In the above syntax for create database command, the values mentioned in square brackets [] are optional.

Usage of Create Database Command in Hive

*create database if not exists HiveDB comment "This is a Hive database" location '/user/hive/warehouse/HiveDB' with DBPROPERTIES ('createdby'='Duncan','createdfor'='acadgild');*

*OK*

*Time taken: 0.042 seconds*

*hive (Demo)> describe database HiveDB;*

*OK*

*hivedb This is a Hive database hdfs://localhost:9000/user/hive/warehouse/HiveDB acadgild USER*

*Time taken: 0.026 seconds, Fetched: 1 row(s)*

## Drop Database in Hive

This command is used for deleting an already created database in Hive and the syntax is as follows -

DROP (DATABASE) [IF EXISTS] database\_name [RESTRICT|CASCADE];

Usage of Drop Database Command in Hive

*hive> drop database if exists HiveDB CASCADE;*

*OK*

*Time taken: 0.099 seconds*

In Hadoop Hive, the mode is set as RESTRICT by default and users cannot delete it unless it is non-empty. For deleting a database in Hive along with the existing tables, users must change the mode from RESTRICT to CASCADE.

In the syntax for drop database Hive command, “if exists” clause is used to avoid any errors that might occur if the programmer tries to delete a database which does not exist.

## Describe Database Command in Hive

This command is used to check any associated metadata for the databases.

*hive (Demo)> describe database extended HiveDB;*

*OK*

*hivedb This is a Hive database hdfs://localhost:9000/user/hive/warehouse/HiveDB acadgild USER {createdfor=acadgild, createdby=Duncan}*

*Time taken: 0.018 seconds, Fetched: 1 row(s)*

## Alter Database Command in Hive

Whenever the developers need to change the metadata of any of the databases, alter hive DDL command can be used as follows –

ALTER (DATABASE) database\_name SET DBPROPERTIES (property\_name=property\_value, ...);

Usage of ALTER database command in Hive –

Let’s use the Alter command to modify the OWNER property and specify the role for the owner –

ALTER (DATABASE) database\_name SET OWNER [USER|ROLE] user\_or\_role;

hive (Demo)> alter database HiveDB SET OWNER ROLE admin;

OK

Time taken: 0.046 seconds

hive (Demo)> describe database extended HiveDB;

OK

hivedb This is a Hive database hdfs://localhost:9000/user/hive/warehouse/HiveDB admin ROLE {createdfor=acadgild, createdby=Duncan}

Time taken: 0.03 seconds, Fetched: 1 row(s)

## Show Database Command in Hive

Programmers can view the list of existing databases in the current schema.

Usage of Show Database Command

*hive (Demo)> show databases;*

*OK*

*custom*

*default*

*demo*

*dunctest*

*hivedb*

*Time taken: 0.023 seconds, Fetched: 5 row(s)*

## Use Database Command in Hive

This hive command is used to select a specific database for the session on which hive queries would be executed.

Usage of Use Database Command in Hive

*hive (Demo)> use HiveDB;*

*OK*

*Time taken: 0.013 seconds*

*hive (HiveDB)>*

**DDL Commands on Tables in Hive**

## Create Table Command in Hive

Hive create table command is used to create a table in the existing database that is in use for a particular session.

CREATE TABLE [IF NOT EXISTS] [db\_name.]table\_name --

[(col\_name data\_type [COMMENT col\_comment], ...)]

[COMMENT table\_comment]

[LOCATION hdfs\_path]

**Hive Create Table Usage**

*create table employee*

*(*

*emp\_name string,*

*skill string,*

*exp int,*

*location string*

*)*

*row format delimited*

*fields terminated by ',';*

In the above step, we have created a hive table named employee in the database college with various fields like id, name, experience and location. Comments can mentioned for each column so that anybody referring to the table gets an overview about what the columns mean.

The LOCATION keyword can be used for specifying where the table should be stored on HDFS.

How to create a table in hive by copying an existing table schema?

Hive lets programmers create a new table by replicating the schema of an existing table but remember only the schema of the new table is replicated but not the data. When creating the new table, the location parameter can be specified.

CREATE TABLE [IF NOT EXISTS] [db\_name.]table\_name Like [db\_name].existing\_table [LOCATION hdfs\_path]

*hive (HiveDB)> create table if not exists HiveDB.employee2table\_name Like HiveDB.employee;*

*OK*

*Time taken: 0.05 seconds*

*hive (HiveDB)> show tables;*

*OK*

*employee*

*employee2table\_name*

*Time taken: 0.035 seconds, Fetched: 2 row(s)*

## DROP Table Command in Hive

Drops the table and all the data associated with it in the Hive metastore.

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

**Usage of DROP Table command in Hive**

*hive (HiveDB)> drop table if exists employee2 purge;*

*OK*

*Time taken: 0.008 seconds*

DROP table command removes the metadata and data for a particular table. Data is usually moved to .Trash/Current directory if Trash is configured. If PURGE option is specified then the table data will not go to the trash directory and there will be no scope to retrieve the data in case of erroneous DROP command execution.

## TRUNCATE Table Command in Hive

This hive command is used to truncate all the rows present in a table i.e. it deletes all the data from the Hive meta store and the data cannot be restored.

TRUNCATE TABLE [db\_name].table\_name

**Usage of TRUNCATE Table in Hive**

*hive (HiveDB)> truncate table employee2;*

*OK*

*Time taken: 0.04 seconds*

*hive (HiveDB)>*

## ALTER Table Command in Hive

Using ALTER Table command, the structure and metadata of the table can be modified even after the table has been created. Let’s try to change the name of an existing table using the ALTER command –

ALTER TABLE [db\_name].old\_table\_name RENAME TO [db\_name].new\_table\_name;

OK

Time taken: 0.07 seconds

hive (HiveDB)>

Syntax to ALTER Table Properties

ALTER TABLE [db\_name].tablename SET TBLPROPERTIES (‘property\_key’=’property\_new\_value’)

In the above step, we have set the creator attribute for the table and similarly we can later or modify other table properties also.

## DESCRIBE Table Command in Hive

Gives the information of a particular table and the syntax is as follows –

DESCRIBE [EXTENDED|FORMATTED] [db\_name.] table\_name[.col\_name ( [.field\_name]

Usage of Describe Table Command

*hive (HiveDB)> describe employee;*

*OK*

*emp\_name string*

*skill string*

*exp int*

*location string*

*Time taken: 0.055 seconds, Fetched: 4 row(s)*

*hive (HiveDB)>*

*Also describe extended to give full information.*

*hive (HiveDB)> describe extended employee;*

*OK*

*emp\_name string*

*skill string*

*exp int*

*location string*

*Detailed Table Information Table(tableName:employee, dbName:hivedb, owner:acadgild, createTime:1505997324, lastAccessTime:0, retention:0, sd:StorageDescriptor(cols:[FieldSchema(name:emp\_name, type:string, comment:null), FieldSchema(name:skill, type:string, comment:null), FieldSchema(name:exp, type:int, comment:null), FieldSchema(name:location, type:string, comment:null)], location:hdfs://localhost:9000/user/hive/warehouse/HiveDB/employee, inputFormat:org.apache.hadoop.mapred.TextInputFormat, outputFormat:org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat, compressed:false, numBuckets:-1, serdeInfo:SerDeInfo(name:null, serializationLib:org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe, parameters:{field.delim=,, serialization.format=,}), bucketCols:[], sortCols:[], parameters:{}, skewedInfo:SkewedInfo(skewedColNames:[], skewedColValues:[], skewedColValueLocationMaps:{}), storedAsSubDirectories:false), partitionKeys:[], parameters:{transient\_lastDdlTime=1505997324}, viewOriginalText:null, viewExpandedText:null, tableType:MANAGED\_TABLE)*

*Time taken: 0.052 seconds, Fetched: 6 row(s)*

*hive (HiveDB)>*

We can also check the description for a specific column from the table as follows –

*hive (HiveDB)> hive (HiveDB)> describe extended employee skill;*

*OK*

*skill string from deserializer*

*Time taken: 0.093 seconds, Fetched: 1 row(s)*

*hive (HiveDB)> ;*

## Show Table Command in Hive

Gives the list of existing tables in the current database schema.

*hive (HiveDB)> show tables;*

*OK*

*employee*

*employee2*

*employee2table\_name*

*Time taken: 0.021 seconds, Fetched: 3 row(s)*

# DML Commands in Hive

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

**Syntax for Load Command in Hive**

LOAD DATA [LOCAL] INPATH 'hdfsfilepath/localfilepath' [OVERWRITE] INTO TABLE existing\_table\_name

Example

Let’s load a structured file that contains information about different employee's.

The file – /home/acadgild/Datasets/emp\_details.txt

*Amit,Big Data,1,BBSR*

*Venkat,Web Technology,2,BBSR*

*Aditya,DBA,1,BNG*

*Ravinder,Java,2,BBSR*

*Sunil,C#,1,BBSR*

*Anil,ASP,2,BNG*

*Mihir,Big Data,3,BBSR*

*Mohit,Java,1,BBSR*

The file is a ‘,’ delimited file where each row can be inserted as a table record.

A table can be created for employee based on the contents in the file –

* The ROW FORMAT DELIMITED must appear before any of the other clauses, with the exception of the STORED AS … clause.
* The clause ROW FORMAT DELIMITED FIELDS TERMINATED BY '| means I character will be used as field separator by hive.
* The clause LINES TERMINATED BY ‘\n' means that the line delimiter will be new line.
* The clause LINES TERMINATED BY ‘\n' and STORED AS … do not require the ROW FORMAT DELIMITED keywords.

*create table employee*

*(*

*emp\_name string,*

*skill string,*

*exp int,*

*location string*

*)*

*row format delimited*

*fields terminated by ',';*

**Let’s load the file into the employee table –**

*load data local inpath '/home/acadgild/Datasets/emp\_details.txt'*

*into table employee;*

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.

Let’s check if the data has been inserted into the table –

**To test**

*hive (HiveDB)> select \* from employee;*

*OK*

*Amit Big Data 1 BBSR*

*Venkat Web Technology 2 BBSR*

*Aditya DBA 1 BNG*

*Ravinder Java 2 BBSR*

*Sunil C# 1 BBSR*

*Anil ASP 2 BNG*

*Mihir Big Data 3 BBSR*

*Mohit Java 1 BBSR*

*Time taken: 0.025 seconds, Fetched: 8 row(s)*

**Now, let's try to retrieve only 2 records using the limit option -**

*hive (HiveDB)> select \* from employee limit 2;*

*OK*

*Amit Big Data 1 BBSR*

*Venkat Web Technology 2 BBSR*

*Time taken: 0.035 seconds, Fetched: 2 row(s)*

**Let's count the total number of records in the table -**

**Note MapReduce will be initiated**

*hive (HiveDB)> select count(\*) from employee;*

*Total MapReduce CPU Time Spent: 3 seconds 130 msec*

*OK*

*8*

*Time taken: 20.959 seconds, Fetched: 1 row(s)*

**Let’s check the number of employees from each location.**

**Note MapReduce will be initiated**

*hive (HiveDB)> select location ,COUNT(\*) from employee group by location;*

*Total MapReduce CPU Time Spent: 3 seconds 50 msec*

*OK*

*BBSR 6*

*BNG 2*

*Time taken: 21.493 seconds, Fetched: 2 row(s)*

## Exporting Data

INSERT OVERWRITE LOCAL DIRECTORY '/home/acadgild/hiveintro'

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '|'

File will be stored locally.