Hive notes

Arturo Alatriste Trujillo.

Contents

[Hive Samples 1](#_Toc491684501)

[Start Beeline by specifying the URL for a Hive2 server 1](#_Toc491684502)

[Query ends with semicolon 1](#_Toc491684503)

[Use ! to execute beeline commands 1](#_Toc491684504)

[create database 1](#_Toc491684505)

[create database if not exist 1](#_Toc491684506)

[delete database 1](#_Toc491684507)

[delete database if not exist 1](#_Toc491684508)

[Create table sintax 1](#_Toc491684509)

[Create a table similar to an existing one 1](#_Toc491684510)

[Create a table based on SELECT 1](#_Toc491684511)

[Create a table stored as avro – schema is another file 1](#_Toc491684512)

[Create a table stored as avro – schema iline 1](#_Toc491684513)

[Create a table stored as parquet 1](#_Toc491684514)

[Create table partitioned by state column. 1](#_Toc491684515)

[Create table nested partitioned by state and zipcode columns. 1](#_Toc491684516)

[Use LOCATION to specify the directory where table data resides 1](#_Toc491684517)

[Using EXTERNAL when creating the table avoids this behavior 1](#_Toc491684518)

[List - Show all the tables 1](#_Toc491684519)

[The DESCRIBE Command lists the fields in the specified table 1](#_Toc491684520)

[DESCRIBE FORMATTED also shows table properties 1](#_Toc491684521)

[SHOW CREATE TABLE displays The SQL Command to create the table 1](#_Toc491684522)

[Upload a file to the data directory 1](#_Toc491684523)

[Or use LOAD DATA 1](#_Toc491684524)

[Use OVERRITE to delete previous data 1](#_Toc491684525)

[Populate a table through a query 1](#_Toc491684526)

[Using sqoop 1](#_Toc491684527)

# Hive Samples

## Start Beeline by specifying the URL for a Hive2 server

beeline -u jdbc:hive2://*host*:10000 \

-n *username* -p *password*

## Query ends with semicolon

SELECT lname,fname FROM customers

. . . > WHERE state = 'CA' LIMIT 50;

## Use ! to execute beeline commands

!connect url – connect to a different Hive2 server

!exit – exit the shell

!help – show the full list of commands

!verbose – show added details of queries

Execute a script file

beeline -u -f myscript.hql

use beeline and query directly

beeline -u -e ‘SELECT \* FROM users’

use silent option to suppress informational messages

beeline -u ... --silent

## create database

CREATE DATABASE customers;

## create database if not exist

CREATE DATABASE IF NOT EXISTS customers;

## delete database

DROP DATABASE customers;

## delete database if not exist

DROP DATABASE IF NOT EXISTS customers;

## Create table sintax

CREATE TABLE tablename (colname DATATYPE, ...)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY char

STORED AS {TEXTFILE|SEQUENCEFILE|…}

## Create a table similar to an existing one

CREATE TABLE good\_customers LIKE customers;

## Create a table based on SELECT

CREATE TABLE ny\_customers AS

SELECT cust\_id, fname, lname

FROM customers

WHERE state = 'NY';

## Create a table stored as avro – schema is another file

CREATE TABLE order\_details\_avro

STORED AS AVRO

TBLPROPERTIES ( 'avro.schema.url' = 'hdfs://localhost/loudacre/accounts\_schema.json' );

## Create a table stored as avro – schema iline

CREATE TABLE order\_details\_avro

STORED AS AVRO

TBLPROPERTIES ('avro.schema.literal'=

'{

"name": "order",

"type": "record",

"fields": [

{ "name":"order\_id", "type":"int" },

{ "name":"cust\_id", "type":"int" },

{ "name":"order\_date", "type":"string" }

]}');

## Create a table stored as parquet

CREATE TABLE order\_details\_parquet (

order\_id INT,

prod\_id INT)

STORED AS PARQUET;

## Create table partitioned by state column.

CREATE EXTERNAL TABLE accounts\_by\_state(

cust\_id INT,

fname STRING,

lname STRING,

address STRING,

city STRING,

state STRING,

zipcode STRING)

PARTITIONED BY (state STRING)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

LOCATION '/loudacre/accounts\_by\_state';

## Create table nested partitioned by state and zipcode columns.

CREATE EXTERNAL TABLE accounts\_by\_state(

cust\_id INT,

fname STRING,

lname STRING,

address STRING,

city STRING,

state STRING,

zipcode STRING)

PARTITIONED BY (state STRING, zipcode STRING)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

LOCATION '/loudacre/accounts\_by\_state';

## Use LOCATION to specify the directory where table data resides

CREATE TABLE jobs (

id INT,

title STRING,

salary INT,

posted TIMESTAMP

)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

LOCATION '/loudacre/jobs';

**CAUTION: Dropping a table removes its data in HDFS**

– Tables are “managed” or “internal” by default

## Using EXTERNAL when creating the table avoids this behavior

– Dropping An *external* table removes only its *metadata*

CREATE EXTERNAL TABLE adclicks

( campaign\_id STRING,

click\_time TIMESTAMP,

keyword STRING,

site STRING,

placement STRING,

was\_clicked BOOLEAN,

cost SMALLINT)

LOCATION '/loudacre/ad\_data';

## List - Show all the tables

SHOW TABLES

## The DESCRIBE Command lists the fields in the specified table

DESCRIBE customers;

## DESCRIBE FORMATTED also shows table properties

DESCRIBE FORMATTED customers;

## SHOW CREATE TABLE displays The SQL Command to create the table

SHOW CREATE TABLE customers;

HUE Metastore can create, load, preview, databases and tables.

To import data to tale

## Upload a file to the data directory

$ hdfs dfs -mv /tmp/sales.txt /user/hive/warehouse/sales/

## Or use LOAD DATA

LOAD DATA INPATH '/tmp/sales.txt'

INTO TABLE sales;

## Use OVERRITE to delete previous data

LOAD DATA INPATH '/tmp/sales.txt'

OVERWRITE INTO TABLE sales;

## Populate a table through a query

INSERT INTO TABLE loyal\_customers

SELECT \* FROM accounts

WHERE YEAR(acct\_create\_dt) = 2008

AND acct\_close\_dt IS NULL;

## Using sqoop

sqoop import \

--connect jdbc:mysql://localhost/loudacre \

--username training \

--password training \

--fields-terminated-by '\t' \

--table employees \

--hive-import