coreservlets.com – Hadoop Course

Hive

In this exercise, you will have a chance to create Hive tables, load data in various ways, execute HiveQL queries, join tables and store them in a result table. You will also get a chance to create a Hive table with partitions.

Approx. Time: 60 minutes

Perform

1. Perform the following in Hive:
    a. Start hive in interactive mode
    b. Create a table called books with 3 columns that will be capable of storing data from $PLAY_AREA/data/books.txt; Load data from books.txt (stored locally) into books table; books table should have the following schema:

        (id INT, title STRING, publishDate STRING)

    c. select record from books table with id of 2
    d. create new table named purchases by re-use existing HDFS location '/training/exercises/hive/ex1'; purchases table will have 3 columns:

        (id INT, buyer STRING, purchaseDate STRING)

    e. select 5 rows from purchases table
    f. create books_purchases table with the following schema

        (id INT, title STRING, buyer STRING, purchaseDate STRING)

    g. populate books_purchases table by joining books and purchases tables via id column
    h. select 10 records from books_purchases table
    i. drop books, purchases and books_purchases tables
2. In this exercise we'll create a partitioned table called media. Media table will be made of the following schema:

```
id          int
title       string
releaseDate string
type            string
```

    type column will be the partition column. Load data from 3 local files into three partitions:

```
partition book => $PLAY_AREA/data/media/books.txt
partition cd     => $PLAY_AREA/data/media/cds.txt
partition dvd    => $PLAY_AREA/data/media/dvds.txt
```

    a. select records of type cd
    b. select records of type dvd
    c. select records of type book
    d. display all the partitions for media table
    e. open new terminal and display partition directory structure in HDFS

f. drop media table

Solution

1. Here are the steps:

   a. $ cd $PLAY_AREA
      $ hive

   b. CREATE TABLE books (id INT, title STRING, publishDate STRING)
      ROW FORMAT DELIMITED
      FIELDS TERMINATED BY ','
      STORED AS TEXTFILE;

      LOAD DATA LOCAL INPATH 'data/books.txt'
      OVERWRITE INTO TABLE books;

   c. select * from books where id=2;

   d. CREATE EXTERNAL TABLE purchases
      (id INT, buyer STRING, purchaseDate STRING)
      ROW FORMAT DELIMITED
      FIELDS TERMINATED BY '\t'
      STORED AS TEXTFILE
      LOCATION '/training/exercises/hive/ex1';

   e. select * from purchases limit 5;

   f. CREATE TABLE books_purchases (id INT, title STRING, buyer STRING, purchaseDate STRING)
      ROW FORMAT DELIMITED
      FIELDS TERMINATED BY ','
      STORED AS TEXTFILE;

   g. INSERT OVERWRITE TABLE books_purchases
      SELECT b.id, b.title, p.buyer, p.purchaseDate
      FROM books b JOIN purchases p ON (b.id=p.id);

   h. select * from books_purchases limit 10;

   i. drop table books_purchases;
      drop table purchases;
      drop table books;

2. Here are the steps:
   $ cd $PLAY_AREA
   $ hive
   CREATE TABLE media (id INT, title STRING, releaseDate STRING)
   PARTITIONED BY(type STRING)
   ROW FORMAT DELIMITED

```
FIELDS TERMINATED BY ','
STORED AS TEXTFILE;

LOAD DATA LOCAL INPATH 'data/media/books.txt'
OVERWRITE INTO TABLE media PARTITION(type='book');

LOAD DATA LOCAL INPATH 'data/media/cds.txt'
OVERWRITE INTO TABLE media PARTITION(type='cd');

LOAD DATA LOCAL INPATH 'data/media/dvds.txt'
OVERWRITE INTO TABLE media PARTITION(type='dvd');
```

a. select * from media where type = "cd";
b. select * from media where type = "dvd";
c. select * from media where type = "book";
d. show partitions media;
e. $hdfs dfs -ls /user/hive/warehouse/media/
f. drop table media;