Muralikrishnan Subbaian

ASSIGNMENT

BIS Academy

AGENDA

- Prerequisites
- Assignment

Prerequisites

Input Data

- aeroplanes.csv
  - Contains all domestic flight data of US for 2007
- aircarriers.csv
  - Contains Airlines information
- airports.csv
  - Contains Airport information

Data Description

- Data description of input files aeroplanes.csv
  - Refer the aeroplanes_data_description
- Data description of input files aircarriers.csv
  - Refer the aircarriers_data_description
- Data description of input files airports.csv
  - Refer the airports_data_description

Preliminary Preparation

- **Note: Execute in Sequence.**
- Edit all the 3 files to delete the first line i.e. header containing the field names (first check if the files do have a header).
- Create Unix/Linux directory – "inputfiles" in user root (/home/cloudera) directory.
- FTP the 3 files from windows to unix/linux directory.

Preliminary Preparation …

- Create directory “assignment/inputfiles” in HDFS user root (/user/cloudera) directory.
- Create directory “/root/inputfiles” in HDFS system root (/) directory.

And get ready for the action…

Assignment

Mysql

- Create “hadoopdb” DataBase in “mysql”.
- Create Table “AEROPLANES” inside “hadoopdb”.
- Describe “AEROPLANES” table and ensure all the fields, field names and datatypes are defined.
- Load data from “aeroplanes.csv” to “AEROPLANES” table.
- Count the entire “AEROPLANES” records and ensure “360” records are available.

Note: Use Data input file “aeroplanes_data_description”

Mysql …

- Create Table “AIRCARRIERS” inside “hadoopdb”.
- Describe “AIRCARRIERS” table and ensure all the fields, field names and datatypes are defined.
- Load data from “aircarriers.csv” to “AIRCARRIERS” table.
- Count the entire “AIRCARRIERS” records and ensure “1491” records are available.

Note: Use Data input file “aircarriers_data_description”

Mysql …

- Create Table “AIRPORTS” inside “hadoopdb”.
- Describe “AIRPORTS” table and ensure all the fields, field names and datatypes are defined.
- Load data from “airports.csv” to “AIRPORTS” table.
- Count the entire “AIRPORTS” records and ensure “3376” records are available.

Note: Use Data input file “airports_data_description”

Sqoop & Hive

- Copy the mysql AEROPLANES table structure to hive using sqoop.
- Describe the hive “aeroplanes” table and ensure all the fields, field names and datatypes are defined.
- Load data from mysql AEROPLANES table to hive using sqoop.
- Count the entire hive “aeroplanes” records and ensure “360” records are copied.

Sqoop & HDFS

- Copy the mysql “AIRCARRIERS” table into HDFS using sqoop.
- Ensure the “AIRCARRIERS” directory created in HDFS and inside file “part-*-00000” available. Note * can be ‘r’ or ‘m’.
- Copy the file “part-*-00000” to the directory assignment/inputfiles in new file name “aircarriers”. Using hadoop fs –cp AIRCARRIERS/part-*-00000 assignment/inputfiles.

Sqoop & HDFS …

- Rename using Hadoop fs –mv assignment/inputfiles/part-*-00000 assignment/inputfiles/aircarrier
- Copy the file “aircarriers” to directory /root/inputfiles Using hadoop fs –cp assignment/inputfiles/aircarriers /root/inputfiles.

Hive

- Create **table external** "aircarriers" in hive pointing to the Location /root/inputfiles.
- Describe the hive "aircarriers" table and ensure all the fields, field names and datatypes are defined.
- Count the entire hive "aircarriers" records and ensure "1491" records are available.

Hive …

- Create **table** “airports” in hive.
- Describe the hive “airports” table and ensure all the fields, field names and datatypes are defined.
- **Load data as local** from file “airports.csv” available in unix/linux directory “inputfiles”.
- Count the entire hive “airports” records and ensure “3376” records are loaded.

Sqoop & HDFS …

- Copy the mysql “AEROPLANES” table into HDFS using sqoop.
- Ensure the “AEROPLANES” directory created in HDFS and inside file “part-*-00000” available. Note * can be ‘r’ or ‘m’.
- Copy the file “part-*-00000” to the directory assignment/inputfiles in new file name “aeroplanes”. Using hadoop fs –cp AEROPLANES/part-*-00000 assignment/inputfiles.

Sqoop & HDFS …

- Rename using Hadoop fs –mv assignment/inputfiles/part-*-00000 assignment/inputfiles/aeroplanes
- Copy the file “aeroplanes” to directory /root/inputfiles Using hadoop fs –cp assignment/inputfiles/aeroplanes /root/inputfiles.

HDFS

- Copy the “airports.csv” file available in unix/linux directory to HDFS using hadoop commands.
- hadoop fs –copyFromLocal inputfiles/airports.csv assignment/inputfiles.
- Copy the file “airports” to directory /root/inputfiles Using hadoop fs –cp assignment/inputfiles/airports.csv /root/inputfiles.

Hive Exercise

- Display all the records of aeroplanes table where carrier is equal to 'OH'.
- Display the records of aeroplane table after sorting the carrier in ascending order.
- In aeroplane table, sort the "origin" in descending order and display the records.
- In aeroplane table, sort the "destination" in ascending order and display the records.
- Group the aircarriers in ascending order by name.

Hive Exercise …

- Group the aircarriers in ascending order by code.
- Group the aircarriers in descending order by name.
- Group the aircarriers in descending order by code.
- Sort the airports table in ascending order by airname.
- Sort the airports table in ascending order by aircode.

Hive Exercise …

- Sort the airports table in descending order by airname.
- Sort the airports table in descending order by aircode.
- Join the aeroplanes and aircarriers table with carrier and carrcode columns and display only the matching records. (Inner join)
- Join the aeroplanes and aircarriers table with carrier and carrcode columns and display all the records of aeroplanes table and matching records of aircarriers table. (Left outer join)

Hive Exercise …

- Join the aeroplanes and aircarriers table with carrier and carrcode columns and display matching records of aeroplanes table and all the records of aircarriers table. (Right outer join)

- Join the aeroplanes and aircarriers table with carrier and carrcode columns and display all the records of aeroplanes table and all the records of aircarriers table. (Full outer join)

Pig Exercise

- Load aeroplanes file into pig. Describe & Dump aeroplanes bag.
- Display the number of record count of aeroplanes bag, ensure 360 records are present.
- Load aircarriers file into pig. Describe & Dump aircarriers bag.
- Display the number of record count of aircarriers bag, ensure 1491 are present.
- Load airports.csvs file into pig. Describe & Dump airports bag.

Pig Exercise …

- Display the number of record count of airports bag, ensure 3376 records are present.
- Display the carrier field alone in aeroplanes bag (use foreach statement, move to another bag called aero2).
- Group the carrier field in aero2 bag and dump.
- Count the no. of carriers repeated in aero2 bag and dump.
- Display the carrier name in first field and then carrier code in second field from aircarriers bag (use foreach statement and display field2 first and then field1 next).

Pig Exercise …

- Display the carrier name alone from aircarriers bag in ascending order (use foreach statement, move to another bag called aircarriers2 bag, then order the aircarriers2 in ascending).

- Display the carrier code alone from aircarriers bag in descending order (use foreach statement, move to another bag called aircarriers3 bag, then order the aircarriers3 in descending).

- Display the airport code and airport name of first 100 entries in the airports bag.

Pig Exercise …

- Display the airports name alone from airports bag in ascending order (use foreach statement, move to another bag called airport2 bag, then order the airport2 in ascending).

- Display the airports code alone from airports bag in descending order (use foreach statement, move to another bag called airport3 bag, then order the airport3 in descending).

- Join the aeroplanes and airports table with origin and aircode columns and display only the matching records. (Inner join)

Pig Exercise …

- Join the aeroplanes and airports table with origin and aircode columns and display all the records of aeroplanes table and matching records of airports table. (Left outer join).

- Join the aeroplanes and airports table with origin and aircode columns and display matching records of aeroplanes table and all the records of airports table. (Right outer join).

- Join the aeroplanes and airports table with origin and aircode columns and display all the records of aeroplanes table and all the records of airports table. (Full outer join).

Oozie Exercise

- Create 1st work flow to execute the following pig job.
  - Load aeroplanes file into pig. Store in HDFS.
- Create 2nd work flow to execute the following hive job.
  - Create table emp with empid and empname fields (data types of empid is INT and empname is STRING).
- Create 3rd work flow to execute 1st the above pig job and on its success execute 2nd the above hive job.

THANK YOU

Muralikrishnan Subbaian