Hadoop: The cases of Pig and Hive in Hadoop

1. Data Type Change using Pig

Set environment:

pig -x local

records = LOAD '/home/student4/sample.txt' AS (year:int, temperature:int, quality:int);

DESCRIBE records;

Screenshot:

A picture containing table, bird, holding

Description automatically generated

records = LOAD '/home/student4/sample.txt' AS (year, temperature, quality);

DESCRIBE records;

Screenshot:

A picture containing table, bird

Description automatically generated

records = LOAD '/home/student4/sample.txt' AS (year, temperature:int, quality:int);

DESCRIBE records;

Screenshot:

A picture containing table, bird, holding

Description automatically generated

records = LOAD '/home/student4/sample.txt';

DESCRIBE records;

Screenshot:

A picture containing bird

Description automatically generated

projected\_records = FOREACH records GENERATE $0, $1, $2;

DUMP projected\_records;

DESCRIBE projected\_records;

Screenshot:

A screenshot of a cell phone

Description automatically generated

1. Loading UDF

Command for compiling Java file:

javac -classpath /home/student4/hadoop-common-2.6.1.jar:/home/student4/hadoop-mapreduce-client-core-2.6.1.jar:/home/student4/commons-cli-2.0.jar:/home/student4/pig-0.11.0.jar:/home/student4/commons-logging-1.2.jar -d . CutLoadFunc.java Range.java

Command for creating Jar file:

jar -cvf CutLoadFunc.jar com/hadoopbook/pig/CutLoadFunc.class

Copy file from Local Disk to HDFS:

hdfs dfs -copyFromLocal sample.txt /home/student4/

run pig:

pig -x local

Register jar in pig:

REGISTER CutLoadFunc.jar;

Command for Loading records:

records = LOAD 'sample.txt' USING com.hadoopbook.pig.CutLoadFunc('16-19,88-92,93-93') AS (year:int, temperature:int, quality:int);

DUMP records;

Screenshot:

A screenshot of text

Description automatically generated

1. Running Hive

Command for compiling Java file:

javac -classpath /home/student4/hadoop-common-2.6.1.jar:/home/student4/hadoop-mapreduce-client-core-2.6.1.jar:/home/student4/commons-cli-2.0.jar:/home/student4/hive-exec-0.13.0.jar -d . Strip.java

Command for creating Jar file:

jar -cvf hive-strip.jar com/hadoopbook/hive/Strip.class

Command for copying a file from local disk to HDFS:

hdfs dfs -copyFromLocal sample.txt /home/student4/

Commends for activating Hive:

mv metastore\_db metastore\_db.old

schematool -dbType derby -initSchema

hive

Add jar file:

ADD JAR hive-strip.jar;

Create a table to hold the data:

CREATE TEMPORARY FUNCTION strip AS 'com.hadoopbook.hive.Strip';

DROP TABLE IF EXISTS records;

CREATE TABLE records (year STRING, temperature INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t';

Populate Hive with the data:

LOAD DATA LOCAL INPATH 'sample.txt'

OVERWRITE INTO TABLE records;

Run the query:

SELECT year, MAX(temperature)

FROM records

WHERE temperature != 9999

AND (quality = 0 OR quality = 1 OR quality = 4 OR quality = 5 OR quality = 9)

GROUP BY year;

Screenshot:

A screenshot of a cell phone

Description automatically generated

1. The codes to create table records2 are as follows:

DROP TABLE IF exists records2;

CREATE TABLE records2 (station STRING, year STRING, temperature INT, quality INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t';

LOAD DATA LOCAL INPATH '/your/path/to /sample2.txt'

OVERWRITE INTO TABLE records2;

Command for compiling Java file:

javac -classpath /home/student4/hadoop-common-2.6.1.jar:/home/student4/hadoop-mapreduce-client-core-2.6.1.jar:/home/student4/commons-cli-2.0.jar:/home/student4/hive-exec-0.13.0.jar -d . Strip.java

Command for creating Jar file:

jar -cvf hive-strip.jar com/hadoopbook/hive/Strip.class

Command for copying a file from local disk to HDFS:

hdfs dfs -copyFromLocal sample2.txt /home/student4/

Commends for activating Hive:

mv metastore\_db metastore\_db.old

schematool -dbType derby -initSchema

hive

Add jar file:

ADD JAR hive-strip.jar;

Create a table to hold the data:

CREATE TEMPORARY FUNCTION strip AS 'com.hadoopbook.hive.Strip';

DROP TABLE IF exists records2;

CREATE TABLE records2 (station STRING, year STRING, temperature INT, quality INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t';

Populate Hive with the data:

LOAD DATA LOCAL INPATH 'sample2.txt'

OVERWRITE INTO TABLE records2;

Run the query:

FROM records2

SELECT year, temperature

DISTRIBUTE BY year

SORT BY year ASC, temperature DESC;

Screenshot:

A screenshot of a cell phone

Description automatically generated