Project description:

This project will create tables in HDFS using Hive. Then, create view, merge table and analyze the customer demographic information using the web log data.

Open Azure, Open Putty, upload data using FileZilla

step1:

make a new directory: mkdir ex9

move the file into the directory: cp ex9data.zip ex9

step2:

unzip the file: unzip ex9data.zip

step3

create a directory in HDFS called **/hive\_exercise**:

Hadoop fs -mkdir /hive\_exercise

step4: move the data from local to hdfs

[root@sandbox ~]# hadoop fs -copyFromLocal ex9/products.tsv /hive\_exercise

[root@sandbox ~]# hadoop fs -copyFromLocal ex9/omniture-logs.tsv /hive\_exercise

[root@sandbox ~]# hadoop fs -ls /hive\_exercise ex9/users.tsv /hive\_exercise

***Step 5*** Create a table for the **users**:

CREATE TABLE **users** (swid STRING, birth\_dt STRING,

gender\_cd CHAR(1))

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

STORED AS TEXTFILE

TBLPROPERTIES ("skip.header.line.count"="1");

***Step 6*** Create a table for the **products**:

CREATE TABLE **products** (url STRING, category STRING)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

STORED AS TEXTFILE

TBLPROPERTIES ("skip.header.line.count"="1");

***Step 7*** Create a table for the **omniturelogs**:

CREATE TABLE **omniturelogs** (col\_1 STRING,col\_2 STRING,col\_3

STRING,col\_4 STRING,col\_5 STRING,col\_6 STRING,col\_7

STRING,col\_8 STRING,col\_9 STRING,col\_10 STRING,col\_11

STRING,col\_12 STRING,col\_13 STRING,col\_14 STRING,col\_15

STRING,col\_16 STRING,col\_17 STRING,col\_18 STRING,col\_19

STRING,col\_20 STRING,col\_21 STRING,col\_22 STRING,col\_23

STRING,col\_24 STRING,col\_25 STRING,col\_26 STRING,col\_27

STRING,col\_28 STRING,col\_29 STRING,col\_30 STRING,col\_31

STRING,col\_32 STRING,col\_33 STRING,col\_34 STRING,col\_35

STRING,col\_36 STRING,col\_37 STRING,col\_38 STRING,col\_39

STRING,col\_40 STRING,col\_41 STRING,col\_42 STRING,col\_43

STRING,col\_44 STRING,col\_45 STRING,col\_46 STRING,col\_47

STRING,col\_48 STRING,col\_49 STRING,col\_50 STRING,col\_51

STRING,col\_52 STRING,col\_53 STRING)

ROW FORMAT DELIMITED

FIELDS TERMINATED by '\t'

STORED AS TEXTFILE

TBLPROPERTIES ("skip.header.line.count"="1");

step8: check the tables

show tables;

Step 9 Next we’ll load the data for the users table:

LOAD DATA INPATH '/hive\_exercise/users.tsv' OVERWRITE

INTO TABLE users;

step10: Next load the data for the table **products** and **omniturelogs** using the **LOAD**

statement

LOAD DATA INPATH '/hive\_exercise/**products**.tsv' OVERWRITE

INTO TABLE **products**;

LOAD DATA INPATH '/hive\_exercise/omniture-logs.tsv' OVERWRITE

INTO TABLE omniturelogs;

omniturelogs

***Step 11*** Check that the tables were loaded. Do a count on the number of lines in the tables

drivers and timesheet.

***Step 12*** Let’s create a view. The table omniturelog doesn’t make much sense.

Run the following:

CREATE VIEW omniture AS

SELECT col\_2 ts, col\_8 ip, col\_13 url, col\_14 swid,

col\_50 city, col\_51 country, col\_53 state

FROM omniturelogs;

step 13. Simple Join:

select a.swid,a.city,a.country,a.state,b.category

from omniture a INNER JOIN products b ON (a.url=b.url limit 10);

step14 (pay attention to how to join three tables)

select a.swid,a.city,a.country,a.state,b.category, u.gender\_cd, u.birth\_dt from omniture a INNER JOIN products b ON a.url=b.url

LEFT OUTER JOIN users u

ON a.swid=concat('{', u.swid, '}') LIMIT 10;

select regexp\_replace(swid, "}", "") from omniture limit 10; 🡪 replace certain string in hive, this can run

but this fail 🡪select regexp\_replace(swid, "{", "") from omniture limit 10;

step 15: how to calculate age, how to get date, how to use timestamp, pay attention to typo(especially, the bracket)

**Logdate DATE, URL STRING, IP STRING, CITY STRING, STATE STRING (uppercase), COUNTRY STRING, CATEGORY,** **CURRENT\_AGE INT, GENDER\_CODE STRING**

select TO\_DATE(o.ts) logdate, o.url, o.ip, o.city, upper(o.state), o.country, p.category,

FLOOR(DATEDIFF(from\_unixtime(unix\_timestamp()), from\_unixtime(unix\_timestamp(u.birth\_dt, 'DD-MMM-YY')))/365.25) CURRENT\_AGE, u.gender\_cd gender\_code

from users u left join omniture o ON o.swid=concat('{', u.swid, '}')

left join products p on o.url = p.url limit 10;

CREATE TABLE WEBLOGANALYTICS AS

select TO\_DATE(o.ts) logdate, o.url, o.ip, o.city, upper(o.state), o.country,

p.category,

FLOOR(DATEDIFF(from\_unixtime(unix\_timestamp()), from\_unixtime(unix\_timestamp(u.birth\_dt, 'DD-MMM-YY')))/365.25) CURRENT\_AGE, u.gender\_cd gender\_code

from omniture o left join products p on o.url = p.url

left join users u on o.swid=concat('{', u.swid, '}') limit 10;

from users u left join omniture o ON o.swid=concat('{', u.swid, '}')

left join products p on o.url = p.url limit 10;

step16: Using the new table **WEBLOGANALYTICS,** write a query to generate an output that

contains the count of people between 19 and 45 for the state of Texas grouped by

**GENDER\_CD,CITY,** and **CATEGORY.**

pay attention to the question: get the number of users, not get the number of url, so in this question, the webloganalytics table should join with user table.

~~select gender\_code, city, category, count(url)~~

~~from webloganalytics~~

~~where state = "TX" and current\_age between 19 and 45~~

~~group by gender\_code, city, category limit 10~~

SELECT w.GENDER\_CODE,w.CITY,w.CATEGORY,COUNT(DISTINCT o.swid)

FROM WEBLOGANALYTICS w LEFT JOIN omniture o

ON w.Logdate = o.ts AND w.IP = o.ip

where w.CURRENT\_AGE BETWEEN 19 AND 45 AND w.STATE = 'TX'

GROUP BY w.GENDER\_CODE, w.CITY, w.CATEGORY

limit 10;

--add distinct user,