Project description:

This project will create tables in HDFS using Hive. Then, analyze the airline data using sparkSQL.

Open Azure, Open Putty, upload data using FileZilla

Step 1 Download the data file “airline\_data.zip”

step 2 upload the data via filezilla

unzip the file: unzip airline\_data.zip

step3. create a folder in HDFS called airline\_data:

hadoop fs -mkdir /airline\_data

put the file train\_df.csv into HDFS on your sandbox under the folder airline\_data

hadoop fs -copyFromLocal airline\_data/train\_df.csv /airline\_data

step4:

Open hive: directly typeh “hive”

Next run the following with **Hive** to create **airline\_delay data**:

CREATE EXTERNAL TABLE airline\_delay(YEAR INT,MONTH

INT,DAY\_OF\_MONTH INT,DAY\_OF\_WEEK INT,CARRIER

STRING,FL\_NUM INT,ORIGIN STRING, DEST STRING,DEP\_TIME

INT,DEP\_DELAY INT,ARR\_TIME INT,ARR\_DELAY INT,CANCELLED

STRING,CANCELLATION\_CODE STRING,AIR\_TIME INT,DISTANCE

INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE

LOCATION '/airline\_data'

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

CREATE EXTERNAL TABLE airline\_delay(YEAR INT,MONTH

INT,DAY\_OF\_MONTH INT,DAY\_OF\_WEEK INT,CARRIER

STRING,FL\_NUM INT,ORIGIN STRING, DEST STRING,DEP\_TIME

INT,DEP\_DELAY INT,ARR\_TIME INT,ARR\_DELAY INT,CANCELLED

STRING,CANCELLATION\_CODE STRING,AIR\_TIME INT,DISTANCE

INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

STORED AS TEXTFILE

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

LOAD DATA INPATH '/airline\_data/train\_df.csv' OVERWRITE INTO TABLE airline\_delay;

exit hive mode🡪 type: exit;

step5: Create a folder

mkdir exercise11

step6:

nano ex11.py

step7

**change spark1.6 to spark2.0**

[**https://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.6.4/bk\_spark-component-guide/content/spark-choose-version.html**](https://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.6.4/bk_spark-component-guide/content/spark-choose-version.html)

from pyspark.sql import SparkSession

spark = SparkSession \

.builder \

.appName("qy\_ex11") \

.enableHiveSupport().getOrCreate()

step8: selects all the data from the table we just created

input = spark.sql("SELECT \* FROM airline\_delay")

**step9:** prints out the count of records

print input.count()

**step10: shows the schema**

input.printSchema()

**step11:** compute statistics for the columns **dep\_delay** and **arr\_delay**

input.describe(['dep\_delay','arr\_delay']).show()

**step12:**

creates a new rdd called **df1** by filtering the rdd **input** by the **origin** being equal

to **DFW**

**pay attention to the way call the column**

df1 = input.filter("origin == 'DFW'")

**step13: count of df1**

print df1.count()

**step14:** compute the average arrival delay arr\_delay and the average departure delay dep\_delay per month grouping

df2 = df1.groupBy("year","month").avg("arr\_delay","dep\_delay").collect()

for i in df2:

print i;

***Step 15*** Run your script using **spark-submit** and pipe the output to a file

**run: spark-submit ex11.py > ex11\_answer.txt**

**error1:**

**ImportError: cannot import name SparkSession**

**change spark1.6 to spark2.0**

[**https://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.6.4/bk\_spark-component-guide/content/spark-choose-version.html**](https://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.6.4/bk_spark-component-guide/content/spark-choose-version.html)

**error2:**

**'Table or view not found: airline\_day;**

**view the tables in hive:**

**hive**

**show tables;**

**exits;**

**error3:**

**Traceback (most recent call last):**

**File "/root/exercise11/ex11.py", line 15, in <module>**

**df1 = input.filter("origin" == "DFW")**

**File "/usr/hdp/2.5.0.0-1245/spark2/python/lib/pyspark.zip/pyspark/sql/datafra$**

**TypeError: condition should be string or Column**