## Load data from AWS RDS to Hadoop

**<Command to run the python file>**

I have created a ipynb file and used pyspark to load datewise pickup data from AWS RDS to hadoop

1. First import all required libraries

import os

import sys

os.environ["PYSPARK\_PYTHON"] = "/opt/cloudera/parcels/Anaconda/bin/python"

os.environ["JAVA\_HOME"] = "/usr/java/jdk1.8.0\_232-cloudera/jre"

os.environ["SPARK\_HOME"]="/opt/cloudera/parcels/SPARK2-2.3.0.cloudera2-1.cdh5.13.3.p0.316101/lib/spark2/"

os.environ["PYLIB"] = os.environ["SPARK\_HOME"] + "/python/lib"

sys.path.insert(0, os.environ["PYLIB"] +"/py4j-0.10.6-src.zip")

sys.path.insert(0, os.environ["PYLIB"] +"/pyspark.zip")

from pyspark.sql import SparkSession

from pyspark.sql.functions import \*

1. Initialize the Spark session.

spark=SparkSession.builder.appName("datewise\_bookings\_aggregates\_spark").master("local").getOrCreate()

1. Read the data from HDFS which was loaded using Sqoop.

df=spark.read.csv("/user/root/cab\_rides/part-m-00000")

1. Rename the columns

* new\_col = ["booking\_id","customer\_id","driver\_id","customer\_app\_version","customer\_phone\_os\_version","pickup\_lat","pickup\_lon","drop\_lat","drop\_lon","pickup\_timestamp","drop\_timestamp","trip\_fare","tip\_amount","currency\_code","cab\_color","cab\_registration\_no","customer\_rating\_by\_driver","rating\_by\_customer","passenger\_count"]
* new\_df = df.toDF(\*new\_col)

1. Converting pickup\_timestamp to date by extracting date from pickup\_timestamp for aggregation

new\_df=new\_df.select("booking\_id","customer\_id","driver\_id","customer\_app\_version","customer\_phone\_os\_version","pickup\_lat","pickup\_lon","drop\_lat", "drop\_lon",to\_date(col('pickup\_timestamp')).alias('pickup\_date').cast("date"),"drop\_timestamp","trip\_fare","tip\_amount","currency\_code","cab\_color","cab\_registration\_no","customer\_rating\_by\_driver", "rating\_by\_customer","passenger\_count")

1. Aggregation on pickup date.

agg\_df=new\_df.groupBy("pickup\_date").count().orderBy("pickup\_date")

1. Command to move above grouped data into HDFS.

agg\_df.coalesce(1).write.format('csv').mode('overwrite').save('/user/root/datewise\_bookings\_agg',header='true')

1. **Note**: I have removed pickup\_date and count column in csv while loading the data into hive table

**<Command to move the csv file to HDFS>**

agg\_df.coalesce(1).write.format('csv').mode('overwrite').save('/user/root/datewise\_bookings\_agg',header='true')

**<Screenshot of the file in HDFS>**

