Logic For First Submission

<Properly explain the code, list the steps to run the code provided by you and attach screenshots of code execution>

->
The main project is regarding capturing the cab rides data from different sources(Kafka and AWS RDS) and making them by consumable by creating aggregated data and storing the aggregated and non-aggregated data into hive tables.

Breaking the project down to multiple steps :

Step 1 : Ingesting Click stream data from a Kafka topic into Hadoop

- We take the help of spark to read the data from the Kafka Topic and then store the data into the hadoop
- The data read from the Kafka topic is in the Json format so we will pull the Value of the Key Value in Json format and store them in hadoop
- For the above point we will use readstream and dataframes to store the data which is then written in the form of Json file again into Hadoop
- The Spark code is as follows:

```
from pyspark.sql import SparkSession
from pyspark.sql.functions import *


# Initializing Spark session
spark = SparkSession \
        .builder \
        .appName("Kafka-to-local") \
        .getOrCreate()

#Reading Streaming data from de-capstone3 kafka topic[Shared by Upgrad]
# we are using starting offsets and setting it as earliest to pull all the data
df = spark.readStream \
        .format("kafka") \
        .option("kafka.bootstrap.servers", "18.211.252.152:9092") \
        .option("startingOffsets", "earliest") \
        .option("subscribe", "de-capstone3") \
        .load()
```

```
# We only copy the value part from the json format data
# we transform the same to string and then drop all the remaining value pulled
df= df \
      .withColumn('value_str',df['value'].cast('string').drop('value') \
      .drop('key','topic','partition','offset','timestamp','timestampType')

#Writing data from kakfa to Hadoop
df.writeStream \
  .format("json") \
  .outputMode("append") \
  .option("path", "/user/root/clickstream_data_dump") \
  .option("checkpointLocation", "/user/root/clickstream_data_dump_cp") \
  .start() \
  .awaitTermination()
```

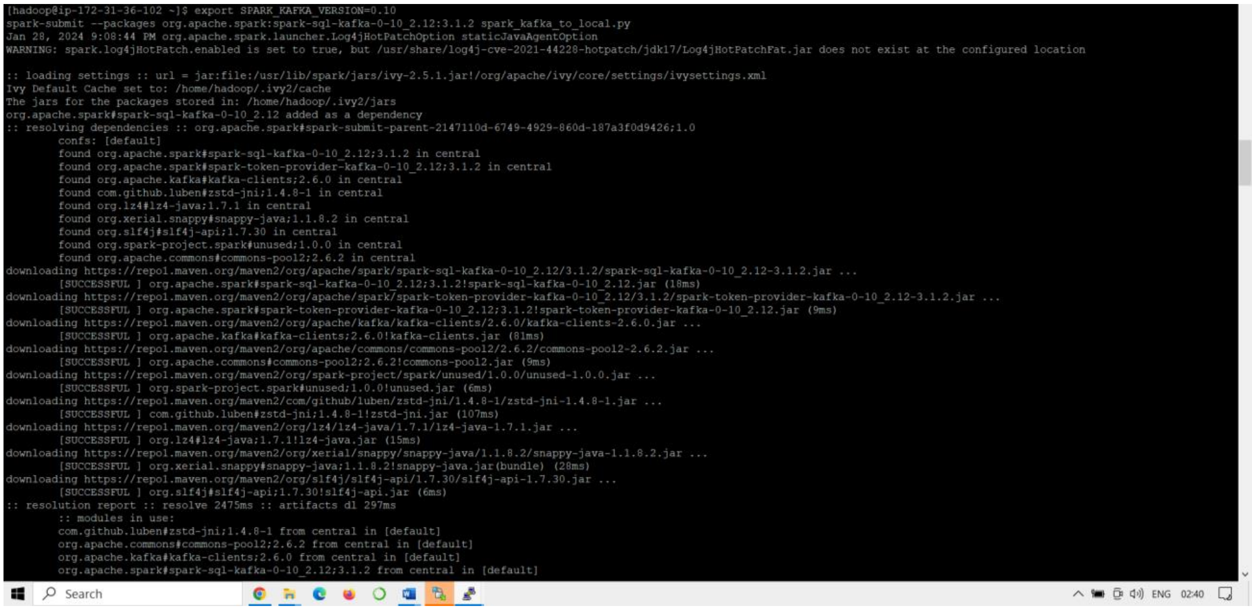
- Copy this spark file to hadoop using winscp
- Then Copy this file to /user/livy for the code to be accesible to be run for spark submit job
- The File attached by the name “spark_kafka_to_local.py” contains the spark code which would need to be run as spark job using spark submit using the following code:

  “

  export SPARK_KAFKA_VERSION=0.10
  spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.1.2 spark_kafka_to_local.py

  ”

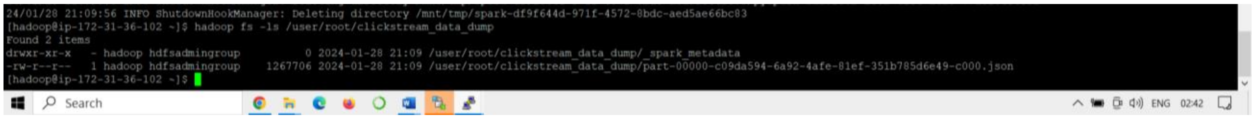
  The above code transfers the spark kafka dependecies for the spark job to interact with the kafka topic
- Screenshot for running the code



```
[hadoop@ip-172-31-36-102 ~]$ export SPARK_KAFKA_VERSION=0.10
spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.1.2 spark_kafka_to_local.py
Jan 28, 2024 9:08:44 PM org.apache.spark.launcher.Log4jHotPatchOption staticJavaAgentOption
WARNING: spark.log4jHotPatch.enabled is set to true, but /usr/share/log4j-cve-2021-44228-hotpatch/jdk17/Log4jHotPatchFat.jar does not exist at the configured location

:: loading settings :: url = jar:file:/usr/lib/spark/jars/ivy-2.5.1.jar!/org/apache/ivy/core/settings/ivysettings.xml
Ivy Default Cache set to: /home/hadoop/.ivy2/cache
The jars for the packages stored in: /home/hadoop/.ivy2/jars
org.apache.spark#spark-sql-kafka-0-10_2.12 added as a dependency
:: resolving dependencies :: org.apache.spark#spark-submit-parent-2147110d-6749-4929-860d-187a3f0d9426;1.0
        confs: [default]
        found org.apache.spark#spark-sql-kafka-0-10_2.12;3.1.2 in central
        found org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.1.2 in central
        found org.apache.kafka#kafka-clients;2.6.0 in central
        found com.github.luben#zstd-jni;1.4.8-1 in central
        found org.lz4#lz4-java;1.7.1 in central
        found org.xerial.snappy#snappy-java;1.1.8.2 in central
        found org.slf4j#slf4j-api;1.7.30 in central
        found org.spark-project.spark#unused;1.0.0 in central
        found org.apache.commons#commons-pool2;2.6.2 in central
downloading https://repo1.maven.org/maven2/org/apache/spark/spark-sql-kafka-0-10_2.12/3.1.2/spark-sql-kafka-0-10_2.12-3.1.2.jar ...
        [SUCCESSFUL ] org.apache.spark#spark-sql-kafka-0-10_2.12;3.1.2!spark-sql-kafka-0-10_2.12.jar (18ms)
downloading https://repo1.maven.org/maven2/org/apache/spark/spark-token-provider-kafka-0-10_2.12/3.1.2/spark-token-provider-kafka-0-10_2.12-3.1.2.jar ...
        [SUCCESSFUL ] org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.1.2!spark-token-provider-kafka-0-10_2.12.jar (9ms)
downloading https://repo1.maven.org/maven2/org/apache/kafka/kafka-clients/2.6.0/kafka-clients-2.6.0.jar ...
        [SUCCESSFUL ] org.apache.kafka#kafka-clients;2.6.0!kafka-clients.jar (81ms)
downloading https://repo1.maven.org/maven2/org/apache/commons/commons-pool2/2.6.2/commons-pool2-2.6.2.jar ...
        [SUCCESSFUL ] org.apache.commons#commons-pool2;2.6.2!commons-pool2.jar (9ms)
downloading https://repo1.maven.org/maven2/org/spark-project/spark/unused/1.0.0/unused-1.0.0.jar ...
        [SUCCESSFUL ] org.spark-project.spark#unused;1.0.0!unused.jar (6ms)
downloading https://repo1.maven.org/maven2/com/github/luben/zstd-jni/1.4.8-1/zstd-jni-1.4.8-1.jar ...
        [SUCCESSFUL ] com.github.luben#zstd-jni;1.4.8-1!zstd-jni.jar (107ms)
downloading https://repo1.maven.org/maven2/org/lz4/lz4-java/1.7.1/lz4-java-1.7.1.jar ...
        [SUCCESSFUL ] org.lz4#lz4-java;1.7.1!lz4-java.jar (15ms)
downloading https://repo1.maven.org/maven2/org/xerial/snappy/snappy-java/1.1.8.2/snappy-java-1.1.8.2.jar ...
        [SUCCESSFUL ] org.xerial.snappy#snappy-java;1.1.8.2!snappy-java.jar(bundle) (28ms)
downloading https://repo1.maven.org/maven2/org/slf4j/slf4j-api/1.7.30/slf4j-api-1.7.30.jar ...
        [SUCCESSFUL ] org.slf4j#slf4j-api;1.7.30!slf4j-api.jar (6ms)
:: resolution report :: resolve 2475ms :: artifacts dl 297ms
        :: modules in use:
        com.github.luben#zstd-jni;1.4.8-1 from central in [default]
        org.apache.commons#commons-pool2;2.6.2 from central in [default]
        org.apache.kafka#kafka-clients;2.6.0 from central in [default]
        org.apache.spark#spark-sql-kafka-0-10_2.12;3.1.2 from central in [default]
```

- ➔ Verifying if the json file was generated at the target location "/user/root/clickstream_data_dump"
    - ○ Code: hadoop fs -ls /user/root/clickstream_data_dump

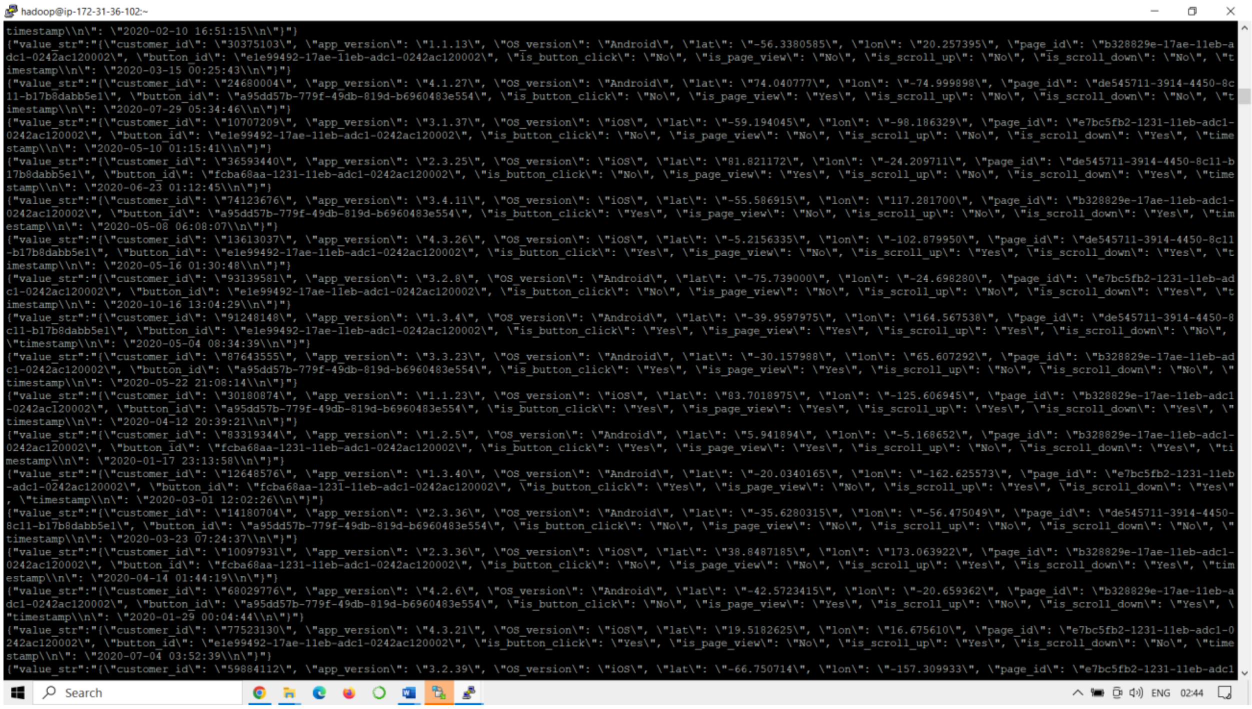
Screen shot for the same :



```
24/01/28 21:09:56 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-df9f644d-971f-4572-8bdc-aed5ae66bc83
[hadoop@ip-172-31-36-102 ~]$ hadoop fs -ls /user/root/clickstream_data_dump
Found 2 items
drwxr-xr-x   - hadoop hdfsadmingroup          0 2024-01-28 21:09 /user/root/clickstream_data_dump/_spark_metadata
-rw-r--r--   1 hadoop hdfsadmingroup    1267706 2024-01-28 21:09 /user/root/clickstream_data_dump/part-00000-c09da594-6a92-4afe-81ef-351b785d6e49-c000.json
[hadoop@ip-172-31-36-102 ~]$
```

The screen shot shows that json file is generated with data copied from the kafka Topic

Screenshot of the data stored in the part file :



```
timestamp\\n\": \"2020-02-10 16:51:15\\n\"}"}
{"value_str":"{\"customer_id\": \"30375103\", \"app_version\": \"1.1.13\", \"OS_version\": \"Android\", \"lat\": \"-56.3380585\", \"lon\": \"20.257395\", \"page_id\": \"b328829e-17ae-11eb-a
dc1-0242ac120002\", \"button_id\": \"e1e99492-17ae-11eb-adc1-0242ac120002\", \"is_button_click\": \"No\", \"is_page_view\": \"No\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"No\", \"t
imestamp\\n\": \"2020-03-15 00:25:43\\n\"}"}
{"value_str":"{\"customer_id\": \"24680004\", \"app_version\": \"4.1.27\", \"OS_version\": \"Android\", \"lat\": \"74.040777\", \"lon\": \"-74.999898\", \"page_id\": \"de545711-3914-4450-8c
11-b17b8dabb5e1\", \"button_id\": \"a95dd57b-779f-49db-819d-b6960483e554\", \"is_button_click\": \"No\", \"is_page_view\": \"Yes\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"No\", \"t
imestamp\\n\": \"2020-07-29 05:34:46\\n\"}"}
{"value_str":"{\"customer_id\": \"10707209\", \"app_version\": \"3.1.37\", \"OS_version\": \"iOS\", \"lat\": \"-59.194045\", \"lon\": \"-98.186329\", \"page_id\": \"e7bc5fb2-1231-11eb-adc1-
0242ac120002\", \"button_id\": \"e1e99492-17ae-11eb-adc1-0242ac120002\", \"is_button_click\": \"No\", \"is_page_view\": \"No\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"Yes\", \"time
stamp\\n\": \"2020-05-10 01:15:41\\n\"}"}
{"value_str":"{\"customer_id\": \"36593440\", \"app_version\": \"2.3.25\", \"OS_version\": \"iOS\", \"lat\": \"81.821172\", \"lon\": \"-24.209711\", \"page_id\": \"de545711-3914-4450-8c11-b
17b8dabb5e1\", \"button_id\": \"fcba68aa-1231-11eb-adc1-0242ac120002\", \"is_button_click\": \"No\", \"is_page_view\": \"Yes\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"Yes\", \"time
stamp\\n\": \"2020-06-23 01:12:45\\n\"}"}
{"value_str":"{\"customer_id\": \"74123676\", \"app_version\": \"3.4.11\", \"OS_version\": \"iOS\", \"lat\": \"-55.586915\", \"lon\": \"117.281700\", \"page_id\": \"b328829e-17ae-11eb-adc1-
0242ac120002\", \"button_id\": \"a95dd57b-779f-49db-819d-b6960483e554\", \"is_button_click\": \"Yes\", \"is_page_view\": \"No\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"Yes\", \"tim
estamp\\n\": \"2020-05-08 06:08:07\\n\"}"}
{"value_str":"{\"customer_id\": \"13613037\", \"app_version\": \"4.3.26\", \"OS_version\": \"iOS\", \"lat\": \"-5.2156335\", \"lon\": \"-102.879950\", \"page_id\": \"de545711-3914-4450-8c11
-b17b8dabb5e1\", \"button_id\": \"e1e99492-17ae-11eb-adc1-0242ac120002\", \"is_button_click\": \"Yes\", \"is_page_view\": \"No\", \"is_scroll_up\": \"Yes\", \"is_scroll_down\": \"Yes\", \"t
imestamp\\n\": \"2020-05-16 01:30:48\\n\"}"}
{"value_str":"{\"customer_id\": \"93139581\", \"app_version\": \"3.2.8\", \"OS_version\": \"Android\", \"lat\": \"-75.739000\", \"lon\": \"-24.698280\", \"page_id\": \"e7bc5fb2-1231-11eb-ad
c1-0242ac120002\", \"button_id\": \"e1e99492-17ae-11eb-adc1-0242ac120002\", \"is_button_click\": \"No\", \"is_page_view\": \"No\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"Yes\", \"t
imestamp\\n\": \"2020-10-16 13:04:29\\n\"}"}
{"value_str":"{\"customer_id\": \"91248148\", \"app_version\": \"1.3.4\", \"OS_version\": \"Android\", \"lat\": \"-39.9597975\", \"lon\": \"164.567538\", \"page_id\": \"de545711-3914-4450-8
c11-b17b8dabb5e1\", \"button_id\": \"e1e99492-17ae-11eb-adc1-0242ac120002\", \"is_button_click\": \"Yes\", \"is_page_view\": \"Yes\", \"is_scroll_up\": \"Yes\", \"is_scroll_down\": \"No\",
\"timestamp\\n\": \"2020-05-04 08:34:39\\n\"}"}
{"value_str":"{\"customer_id\": \"87643555\", \"app_version\": \"3.3.23\", \"OS_version\": \"Android\", \"lat\": \"-30.157988\", \"lon\": \"65.607292\", \"page_id\": \"b328829e-17ae-11eb-ad
c1-0242ac120002\", \"button_id\": \"a95dd57b-779f-49db-819d-b6960483e554\", \"is_button_click\": \"Yes\", \"is_page_view\": \"Yes\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"No\", \"
timestamp\\n\": \"2020-05-22 21:08:14\\n\"}"}
{"value_str":"{\"customer_id\": \"30180874\", \"app_version\": \"1.1.23\", \"OS_version\": \"iOS\", \"lat\": \"83.7018975\", \"lon\": \"-125.606945\", \"page_id\": \"b328829e-17ae-11eb-adc1
-0242ac120002\", \"button_id\": \"a95dd57b-779f-49db-819d-b6960483e554\", \"is_button_click\": \"Yes\", \"is_page_view\": \"Yes\", \"is_scroll_up\": \"Yes\", \"is_scroll_down\": \"Yes\", \"
timestamp\\n\": \"2020-04-12 20:39:21\\n\"}"}
{"value_str":"{\"customer_id\": \"83319344\", \"app_version\": \"1.2.5\", \"OS_version\": \"Android\", \"lat\": \"5.941894\", \"lon\": \"-5.168652\", \"page_id\": \"b328829e-17ae-11eb-adc1-
0242ac120002\", \"button_id\": \"fcba68aa-1231-11eb-adc1-0242ac120002\", \"is_button_click\": \"Yes\", \"is_page_view\": \"Yes\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"Yes\", \"ti
mestamp\\n\": \"2020-01-17 23:13:58\\n\"}"}
{"value_str":"{\"customer_id\": \"12648576\", \"app_version\": \"1.3.40\", \"OS_version\": \"Android\", \"lat\": \"-20.0340165\", \"lon\": \"-162.625573\", \"page_id\": \"e7bc5fb2-1231-11eb
-adc1-0242ac120002\", \"button_id\": \"fcba68aa-1231-11eb-adc1-0242ac120002\", \"is_button_click\": \"Yes\", \"is_page_view\": \"No\", \"is_scroll_up\": \"Yes\", \"is_scroll_down\": \"Yes\"
, \"timestamp\\n\": \"2020-03-01 12:02:26\\n\"}"}
{"value_str":"{\"customer_id\": \"14180704\", \"app_version\": \"2.3.36\", \"OS_version\": \"Android\", \"lat\": \"-35.6280315\", \"lon\": \"-56.475049\", \"page_id\": \"de545711-3914-4450-
8c11-b17b8dabb5e1\", \"button_id\": \"a95dd57b-779f-49db-819d-b6960483e554\", \"is_button_click\": \"No\", \"is_page_view\": \"No\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"No\", \"
timestamp\\n\": \"2020-03-23 07:24:37\\n\"}"}
{"value_str":"{\"customer_id\": \"10097931\", \"app_version\": \"2.3.36\", \"OS_version\": \"iOS\", \"lat\": \"38.8487185\", \"lon\": \"173.063922\", \"page_id\": \"b328829e-17ae-11eb-adc1-
0242ac120002\", \"button_id\": \"fcba68aa-1231-11eb-adc1-0242ac120002\", \"is_button_click\": \"No\", \"is_page_view\": \"No\", \"is_scroll_up\": \"Yes\", \"is_scroll_down\": \"Yes\", \"tim
estamp\\n\": \"2020-04-14 01:44:19\\n\"}"}
{"value_str":"{\"customer_id\": \"68029776\", \"app_version\": \"4.2.6\", \"OS_version\": \"Android\", \"lat\": \"-42.5723415\", \"lon\": \"-20.659362\", \"page_id\": \"b328829e-17ae-11eb-a
dc1-0242ac120002\", \"button_id\": \"a95dd57b-779f-49db-819d-b6960483e554\", \"is_button_click\": \"No\", \"is_page_view\": \"Yes\", \"is_scroll_up\": \"No\", \"is_scroll_down\": \"Yes\", \
"timestamp\\n\": \"2020-01-29 00:04:44\\n\"}"}
{"value_str":"{\"customer_id\": \"77523130\", \"app_version\": \"4.3.21\", \"OS_version\": \"iOS\", \"lat\": \"19.5182625\", \"lon\": \"16.675610\", \"page_id\": \"e7bc5fb2-1231-11eb-adc1-0
242ac120002\", \"button_id\": \"e1e99492-17ae-11eb-adc1-0242ac120002\", \"is_button_click\": \"Yes\", \"is_page_view\": \"No\", \"is_scroll_up\": \"Yes\", \"is_scroll_down\": \"No\", \"time
stamp\\n\": \"2020-07-04 03:52:39\\n\"}"}
{"value_str":"{\"customer_id\": \"59884112\", \"app_version\": \"3.2.39\", \"OS_version\": \"iOS\", \"lat\": \"-66.750714\", \"lon\": \"-157.309933\", \"page_id\": \"e7bc5fb2-1231-11eb-adc1
```

<u>Step 2:</u> The second step is to actually flatten the data

- ➔ The current data is string of data of the form of nested Json
- ➔ We write a Spark query to read this json file and then flatten the data -> make it in the form of a csv file [tabular format having all the data]
- ➔ Spark query:

```
# importing the important classes/libraries
from pyspark.sql import SparkSession
from pyspark.sql.functions import *
from pyspark.sql.types import *
```

```
# Initializing Spark session
spark = SparkSession.builder \
    .appName("Flatten") \
    .master("local")\
    .getOrCreate()

# Reading the json file using spark.read.json function
df=spark.read.json("/user/root/clickstream_data_dump/part-00000-c09da594-6a92-4afe-81ef-351b785d6e49-c000.json")

#Do not forget to change the name for the part file


#We take help of “get_json_object” to deal with nested json format to refer to the nested
#columns and providing alias at the same time
df_final=df.select(get_json_object(df['value_str'],"$.customer_id").alias("customer_id"),
            get_json_object(df['value_str'],"$.app_version").alias("app_version"),
            get_json_object(df['value_str'],"$.OS_version").alias("OS_version"),
            get_json_object(df['value_str'],"$.lat").alias("lat"),
            get_json_object(df['value_str'],"$.lon").alias("lon"),
            get_json_object(df['value_str'],"$.page_id").alias("page_id"),
            get_json_object(df['value_str'],"$.button_id").alias("button_id"),
            get_json_object(df['value_str'],"$.is_button_click").alias("is_button_click"),
            get_json_object(df['value_str'],"$.is_page_view").alias("is_page_view"),
            get_json_object(df['value_str'],"$.is_scroll_up").alias("is_scroll_up"),
            get_json_object(df['value_str'],"$.is_scroll_down").alias("is_scroll_down"),
            get_json_object(df['value_str'],"$.timestamp").alias("timestamp")
             )

# Finally writing the data into csv file and storing at the below given location
df_final.write.format('csv').mode('overwrite').save('/user/root/clickstream_flattened',header='true')
```

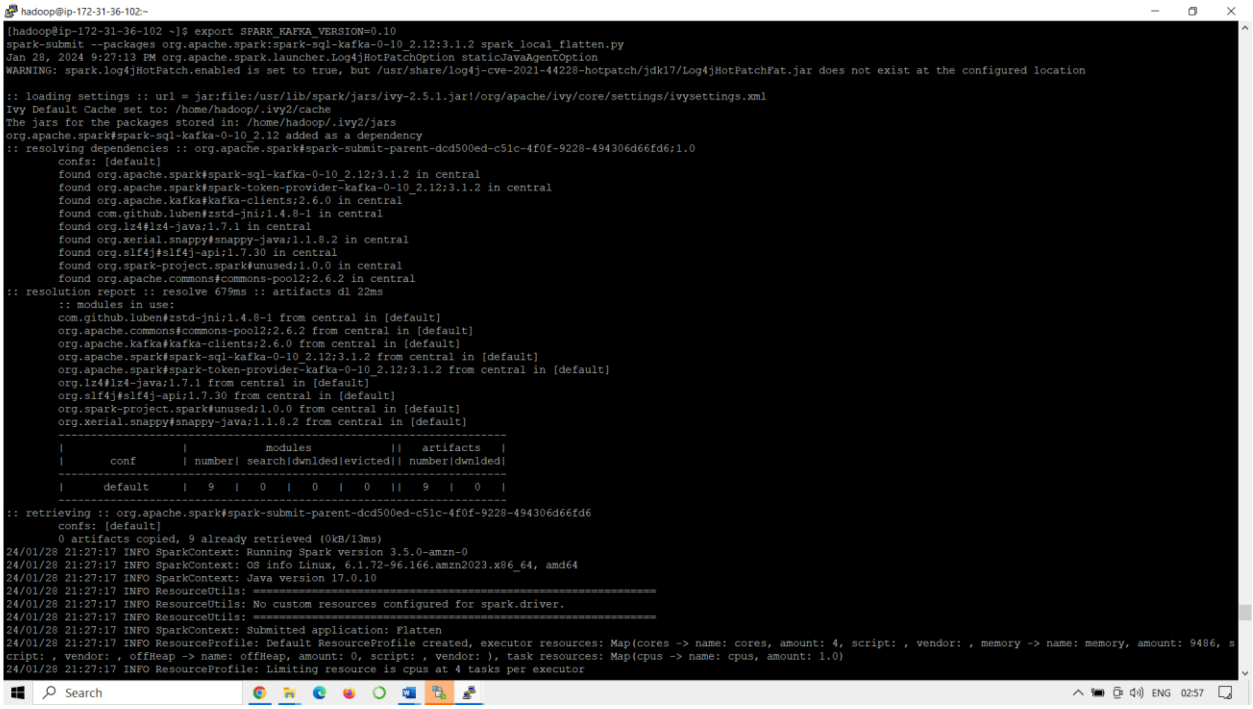
➔ In order to run the spark submit program we need to run the following code(ensure that the spark file name “spark_local_flatten.py” has been copied to /user/livy:

“

export SPARK_KAFKA_VERSION=0.10
spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.1.2 spark_local_flatten.py

“

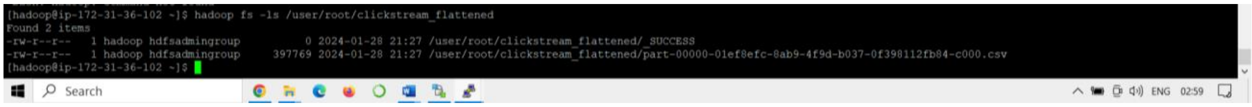
➔ Screenshot of running the code :



```
[hadoop@ip-172-31-36-102 ~]$ export SPARK_KAFKA_VERSION=0.10
spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.1.2 spark_local_flatten.py
Jan 28, 2024 9:27:13 PM org.apache.spark.launcher.Log4jHotPatchOption staticJavaAgentOption
WARNING: spark.log4jHotPatch.enabled is set to true, but /usr/share/log4j-cve-2021-44228-hotpatch/jdk17/Log4jHotPatchFat.jar does not exist at the configured location

:: loading settings :: url = jar:file:/usr/lib/spark/jars/ivy-2.5.1.jar!/org/apache/ivy/core/settings/ivysettings.xml
Ivy Default Cache set to: /home/hadoop/.ivy2/cache
The jars for the packages stored in: /home/hadoop/.ivy2/jars
org.apache.spark#spark-sql-kafka-0-10_2.12 added as a dependency
:: resolving dependencies :: org.apache.spark#spark-submit-parent-dcd500ed-c51c-4f0f-9228-494306d66fd6;1.0
        confs: [default]
        found org.apache.spark#spark-sql-kafka-0-10_2.12;3.1.2 in central
        found org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.1.2 in central
        found org.apache.kafka#kafka-clients;2.6.0 in central
        found com.github.luben#zstd-jni;1.4.8-1 in central
        found org.lz4#lz4-java;1.7.1 in central
        found org.xerial.snappy#snappy-java;1.1.8.2 in central
        found org.slf4j#slf4j-api;1.7.30 in central
        found org.spark-project.spark#unused;1.0.0 in central
        found org.apache.commons#commons-pool2;2.6.2 in central
:: resolution report :: resolve 679ms :: artifacts dl 22ms
        :: modules in use:
        com.github.luben#zstd-jni;1.4.8-1 from central in [default]
        org.apache.commons#commons-pool2;2.6.2 from central in [default]
        org.apache.kafka#kafka-clients;2.6.0 from central in [default]
        org.apache.spark#spark-sql-kafka-0-10_2.12;3.1.2 from central in [default]
        org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.1.2 from central in [default]
        org.lz4#lz4-java;1.7.1 from central in [default]
        org.slf4j#slf4j-api;1.7.30 from central in [default]
        org.spark-project.spark#unused;1.0.0 from central in [default]
        org.xerial.snappy#snappy-java;1.1.8.2 from central in [default]
        ---------------------------------------------------------------------
        |                  |            modules            ||   artifacts   |
        |       conf       | number| search|dwnlded|evicted|| number|dwnlded|
        ---------------------------------------------------------------------
        |      default     |   9   |   0   |   0   |   0   ||   9   |   0   |
        ---------------------------------------------------------------------
:: retrieving :: org.apache.spark#spark-submit-parent-dcd500ed-c51c-4f0f-9228-494306d66fd6
        confs: [default]
        0 artifacts copied, 9 already retrieved (0kB/13ms)
24/01/28 21:27:17 INFO SparkContext: Running Spark version 3.5.0-amzn-0
24/01/28 21:27:17 INFO SparkContext: OS info Linux, 6.1.72-96.166.amzn2023.x86_64, amd64
24/01/28 21:27:17 INFO SparkContext: Java version 17.0.10
24/01/28 21:27:17 INFO ResourceUtils: ==============================================================
24/01/28 21:27:17 INFO ResourceUtils: No custom resources configured for spark.driver.
24/01/28 21:27:17 INFO ResourceUtils: ==============================================================
24/01/28 21:27:17 INFO SparkContext: Submitted application: Flatten
24/01/28 21:27:17 INFO ResourceProfile: Default ResourceProfile created, executor resources: Map(cores -> name: cores, amount: 4, script: , vendor: , memory -> name: memory, amount: 9486, s
cript: , vendor: , offHeap -> name: offHeap, amount: 0, script: , vendor: ), task resources: Map(cpus -> name: cpus, amount: 1.0)
24/01/28 21:27:17 INFO ResourceProfile: Limiting resource is cpus at 4 tasks per executor
```

Verifying that the csv file is generated at the target location
Code: hadoop fs -ls /user/root/clickstream_flattened

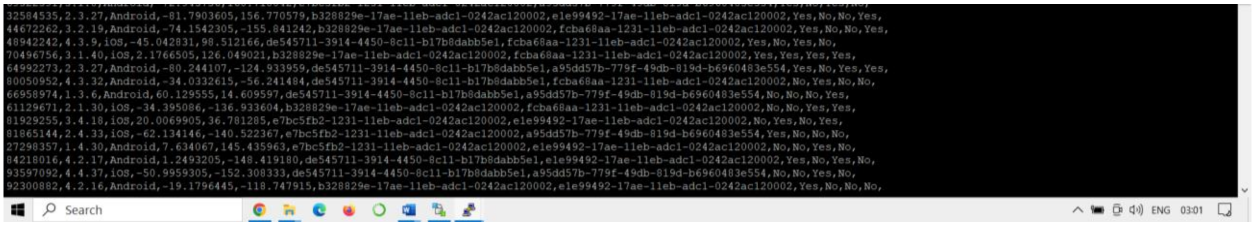
Screenshot :



```
[hadoop@ip-172-31-36-102 ~]$ hadoop fs -ls /user/root/clickstream_flattened
Found 2 items
-rw-r--r--   1 hadoop hdfsadmingroup          0 2024-01-28 21:27 /user/root/clickstream_flattened/_SUCCESS
-rw-r--r--   1 hadoop hdfsadmingroup     397769 2024-01-28 21:27 /user/root/clickstream_flattened/part-00000-01ef8efc-8ab9-4f9d-b037-0f398112fb84-c000.csv
[hadoop@ip-172-31-36-102 ~]$
```

Screenshot of the flattened data in the above csv file:
Code: hadoop fs -cat /user/root/clickstream_flattened/part-00000-01ef8efc-8ab9-4f9d-b037-0f398112fb84-c000.csv

##change the name of the csv file as per the new name generated when run



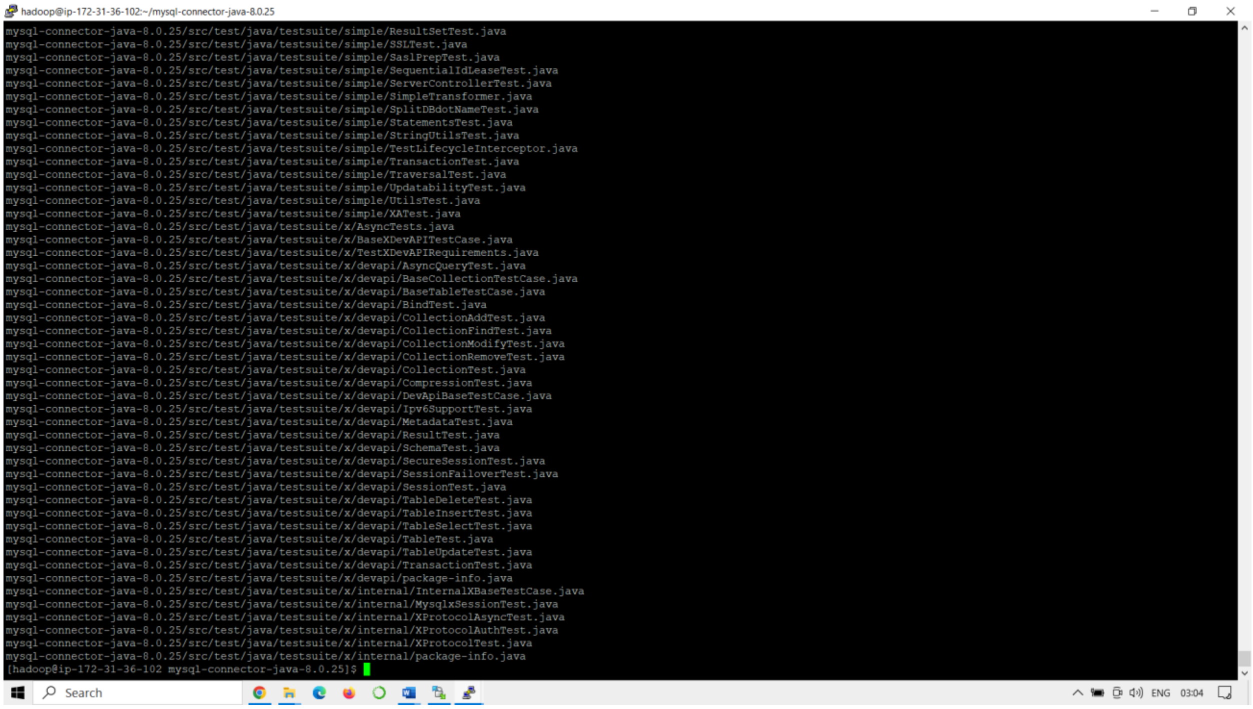
```
32584535,2.3.27,Android,-81.7903605,156.770579,b328829e-17ae-11eb-adc1-0242ac120002,e1e99492-17ae-11eb-adc1-0242ac120002,Yes,No,No,Yes,
44672262,3.2.19,Android,-74.1542305,-155.841242,b328829e-17ae-11eb-adc1-0242ac120002,fcba68aa-1231-11eb-adc1-0242ac120002,Yes,No,No,Yes,
48942242,4.3.9,iOS,-45.042831,98.512166,de545711-3914-4450-8c11-b17b8dabb5e1,fcba68aa-1231-11eb-adc1-0242ac120002,Yes,No,Yes,No,
70496756,3.1.40,iOS,2.1766505,126.049021,b328829e-17ae-11eb-adc1-0242ac120002,fcba68aa-1231-11eb-adc1-0242ac120002,Yes,Yes,Yes,Yes,
64992273,2.3.27,Android,-80.244107,-124.933959,de545711-3914-4450-8c11-b17b8dabb5e1,a95dd57b-779f-49db-819d-b6960483e554,Yes,No,Yes,Yes,
80050952,4.3.32,Android,-34.0332615,-56.241484,de545711-3914-4450-8c11-b17b8dabb5e1,fcba68aa-1231-11eb-adc1-0242ac120002,No,Yes,No,No,
66958974,1.3.6,Android,60.129555,14.609597,de545711-3914-4450-8c11-b17b8dabb5e1,a95dd57b-779f-49db-819d-b6960483e554,No,No,No,Yes,
61129671,2.1.30,iOS,-34.395086,-136.933604,b328829e-17ae-11eb-adc1-0242ac120002,fcba68aa-1231-11eb-adc1-0242ac120002,No,No,Yes,Yes,
81929255,3.4.18,iOS,20.0069905,36.781285,e7bc5fb2-1231-11eb-adc1-0242ac120002,e1e99492-17ae-11eb-adc1-0242ac120002,No,Yes,No,Yes,
81865144,2.4.33,iOS,-62.134146,-140.522367,e7bc5fb2-1231-11eb-adc1-0242ac120002,a95dd57b-779f-49db-819d-b6960483e554,Yes,No,No,No,
27298357,1.4.30,Android,7.634067,145.435963,e7bc5fb2-1231-11eb-adc1-0242ac120002,e1e99492-17ae-11eb-adc1-0242ac120002,No,No,Yes,No,
84218016,4.2.17,Android,1.2493205,-148.419180,de545711-3914-4450-8c11-b17b8dabb5e1,e1e99492-17ae-11eb-adc1-0242ac120002,Yes,No,Yes,No,
93597092,4.4.37,iOS,-50.9959305,-152.308333,de545711-3914-4450-8c11-b17b8dabb5e1,a95dd57b-779f-49db-819d-b6960483e554,No,No,Yes,No,
92300882,4.2.16,Android,-19.1796445,-118.747915,b328829e-17ae-11eb-adc1-0242ac120002,e1e99492-17ae-11eb-adc1-0242ac120002,Yes,No,No,No,
```

Step 3: The next steps involves ingesting data from AWS RDS table to the hadoop

- ➔ We take help of sqoop
- ➔ First we need to install Mysql connectors in our cluster so that we can use the jdbc connectors to connect to the database and pull the data for the same
- ➔ Code:

```
wget https://de-mysql-connector.s3.amazonaws.com/mysql-connector-java-8.0.25.tar.gz
tar -xvf mysql-connector-java-8.0.25.tar.gz
cd mysql-connector-java-8.0.25/
sudo cp mysql-connector-java-8.0.25.jar /usr/lib/sqoop/lib/
```

Screenshot

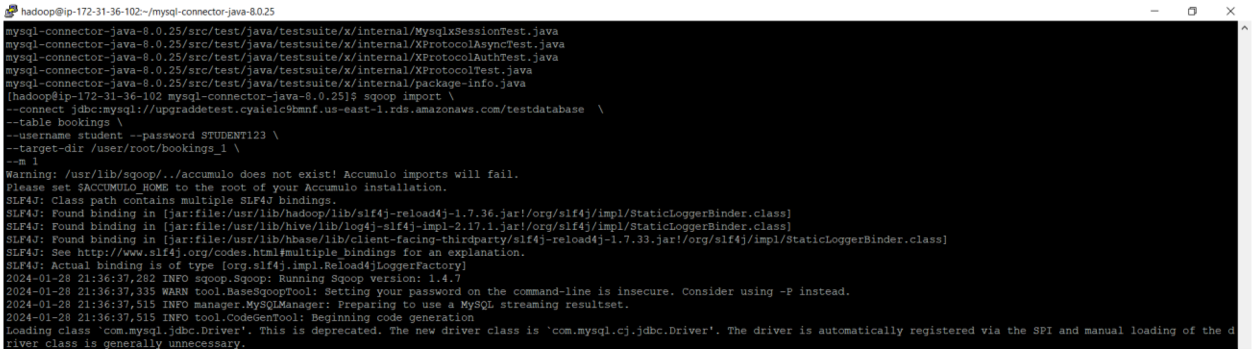


```
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/ResultSetTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/SSLTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/SaslPrepTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/SequentialIdLeaseTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/ServerControllerTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/SimpleTransformer.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/SplitDBdotNameTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/StatementsTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/StringUtilsTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/TestLifecycleInterceptor.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/TransactionTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/TraversalTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/UpdatabilityTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/UtilsTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/simple/XATest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/AsyncTests.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/BaseXDevAPITestCase.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/TestXDevAPIRequirements.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/AsyncQueryTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/BaseCollectionTestCase.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/BaseTableTestCase.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/BindTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/CollectionAddTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/CollectionFindTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/CollectionModifyTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/CollectionRemoveTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/CollectionTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/CompressionTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/DevApiBaseTestCase.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/Ipv6SupportTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/MetadataTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/ResultTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/SchemaTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/SecureSessionTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/SessionFailoverTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/SessionTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/TableDeleteTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/TableInsertTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/TableSelectTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/TableTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/TableUpdateTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/TransactionTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/devapi/package-info.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/InternalXBaseTestCase.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/MysqlxSessionTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/XProtocolAsyncTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/XProtocolAuthTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/XProtocolTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/package-info.java
[hadoop@ip-172-31-36-102 mysql-connector-java-8.0.25]$ 
```

- ➔ Now we can connect to the RDS using the jdbc connector and the user id and password shared
- ➔ Code:

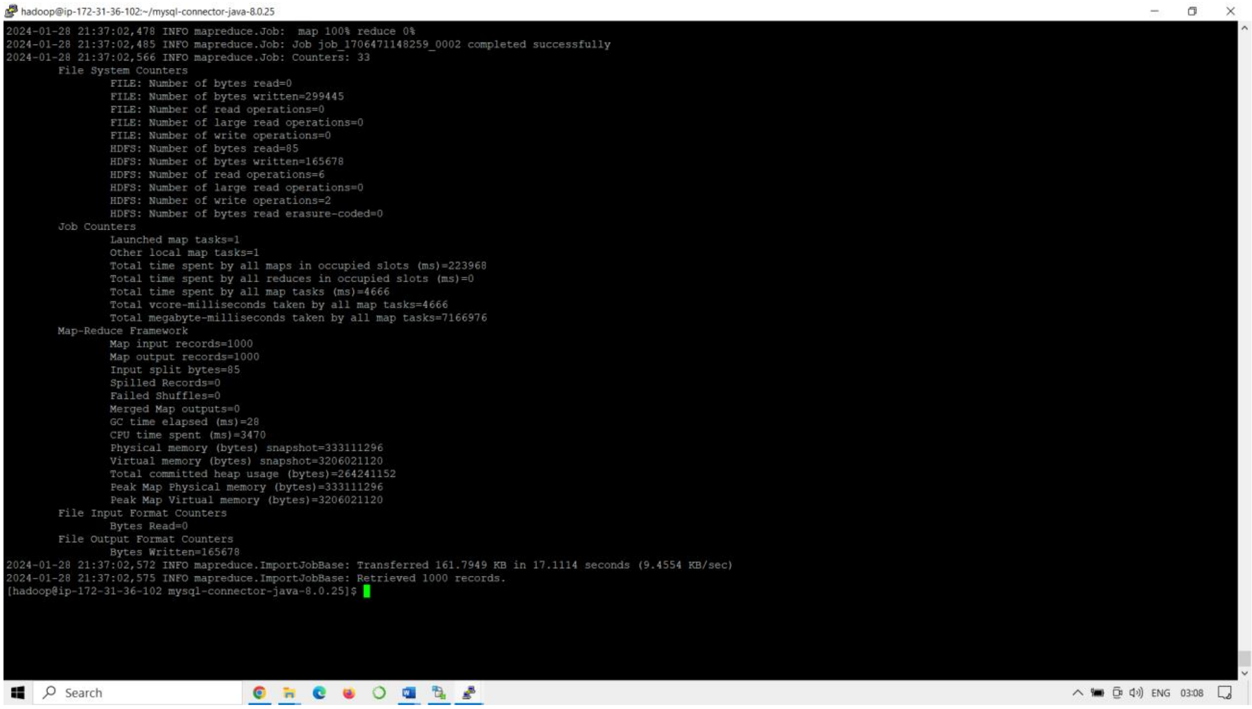
```
sqoop import \
--connect jdbc:mysql://upgraddetest.cyaielc9bmnf.us-east-
1.rds.amazonaws.com/testdatabase  \
--table bookings \
--username student --password STUDENT123 \
--target-dir /user/root/bookings_1 \
--m 1
```

-> Screen shot



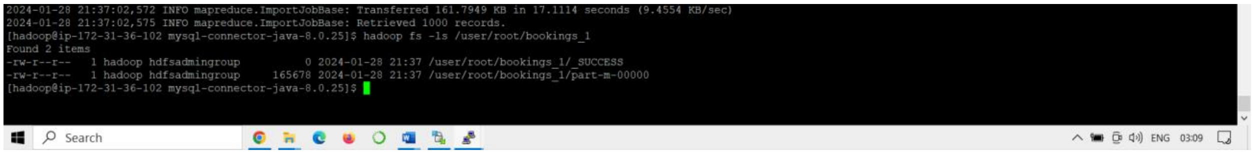
```
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/MysqlxSessionTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/XProtocolAsyncTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/XProtocolAuthTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/XProtocolTest.java
mysql-connector-java-8.0.25/src/test/java/testsuite/x/internal/package-info.java
[hadoop@ip-172-31-36-102 mysql-connector-java-8.0.25]$ sqoop import \
--connect jdbc:mysql://upgraddetest.cyaielc9bmnf.us-east-1.rds.amazonaws.com/testdatabase  \
--table bookings \
--username student --password STUDENT123 \
--target-dir /user/root/bookings_1 \
--m 1
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-reload4j-1.7.36.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/lib/hive/lib/log4j-slf4j-impl-2.17.1.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/lib/hbase/lib/client-facing-thirdparty/slf4j-reload4j-1.7.33.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Reload4jLoggerFactory]
2024-01-28 21:36:37,282 INFO sqoop.Sqoop: Running Sqoop version: 1.4.7
2024-01-28 21:36:37,335 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
2024-01-28 21:36:37,515 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
2024-01-28 21:36:37,515 INFO tool.CodeGenTool: Beginning code generation
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'. The driver is automatically registered via the SPI and manual loading of the d
river class is generally unnecessary.
```

Screen shot showing 1000 records pulled:



```
2024-01-28 21:37:02,478 INFO mapreduce.Job:  map 100% reduce 0%
2024-01-28 21:37:02,485 INFO mapreduce.Job: Job job_1706471148259_0002 completed successfully
2024-01-28 21:37:02,566 INFO mapreduce.Job: Counters: 33
        File System Counters
                FILE: Number of bytes read=0
                FILE: Number of bytes written=299445
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=85
                HDFS: Number of bytes written=165678
                HDFS: Number of read operations=6
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=1
                Other local map tasks=1
                Total time spent by all maps in occupied slots (ms)=223968
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=4666
                Total vcore-milliseconds taken by all map tasks=4666
                Total megabyte-milliseconds taken by all map tasks=7166976
        Map-Reduce Framework
                Map input records=1000
                Map output records=1000
                Input split bytes=85
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=28
                CPU time spent (ms)=3470
                Physical memory (bytes) snapshot=333111296
                Virtual memory (bytes) snapshot=3206021120
                Total committed heap usage (bytes)=264241152
                Peak Map Physical memory (bytes)=333111296
                Peak Map Virtual memory (bytes)=3206021120
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=165678
2024-01-28 21:37:02,572 INFO mapreduce.ImportJobBase: Transferred 161.7949 KB in 17.1114 seconds (9.4554 KB/sec)
2024-01-28 21:37:02,575 INFO mapreduce.ImportJobBase: Retrieved 1000 records.
[hadoop@ip-172-31-36-102 mysql-connector-java-8.0.25]$
```

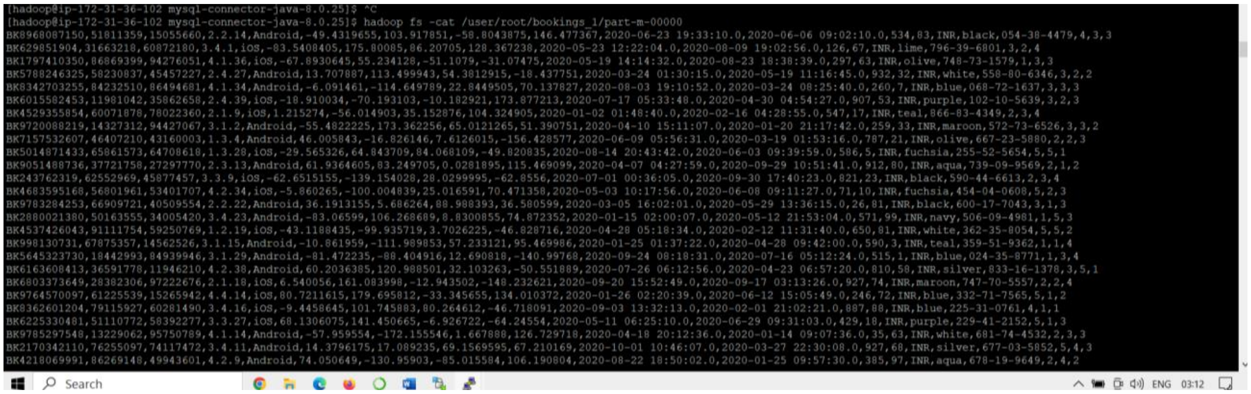
Verifying the file generated at the target location:
Code: hadoop fs -ls /user/root/bookings_1



```
2024-01-28 21:37:02,572 INFO mapreduce.ImportJobBase: Transferred 161.7949 KB in 17.1114 seconds (9.4554 KB/sec)
2024-01-28 21:37:02,575 INFO mapreduce.ImportJobBase: Retrieved 1000 records.
[hadoop@ip-172-31-36-102 mysql-connector-java-8.0.25]$ hadoop fs -ls /user/root/bookings_1
Found 2 items
-rw-r--r--   1 hadoop hdfsadmingroup          0 2024-01-28 21:37 /user/root/bookings_1/_SUCCESS
-rw-r--r--   1 hadoop hdfsadmingroup     165678 2024-01-28 21:37 /user/root/bookings_1/part-m-00000
[hadoop@ip-172-31-36-102 mysql-connector-java-8.0.25]$
```

The image shows a part file was generated with the data pulled form the RDS

Screenshot of the data stored in the part file :



```
[hadoop@ip-172-31-36-102 mysql-connector-java-8.0.25]$ ^C
[hadoop@ip-172-31-36-102 mysql-connector-java-8.0.25]$ hadoop fs -cat /user/root/bookings_1/part-m-00000
BK8968087150,51811359,15055660,2.2.14,Android,-49.4319655,103.917851,-58.8043875,146.477367,2020-06-23 19:33:10.0,2020-06-06 09:02:10.0,534,83,INR,black,054-38-4479,4,3,3
BK629851904,31663218,60872180,3.4.1,iOS,-83.5408405,175.80085,86.20705,128.367238,2020-05-23 12:22:04.0,2020-08-09 19:02:56.0,126,67,INR,lime,796-39-6801,3,2,4
BK1797410350,86869399,94276051,4.1.36,iOS,-67.8930645,55.234128,-51.1079,-31.07475,2020-05-19 14:14:32.0,2020-08-23 18:38:39.0,297,63,INR,olive,748-73-1579,1,3,3
BK5788246325,58230837,45457227,2.4.27,Android,13.707887,113.499943,54.3812915,-18.437751,2020-03-24 01:30:15.0,2020-05-19 11:16:45.0,932,32,INR,white,558-80-6346,3,2,2
BK8342703255,84232510,86494681,4.1.34,Android,-6.091461,-114.649789,22.8449505,70.137827,2020-08-03 19:10:52.0,2020-03-24 08:25:40.0,260,7,INR,blue,068-72-1637,3,3,3
BK6015582453,11981042,35862658,2.4.39,iOS,-18.910034,-70.193103,-10.182921,173.877213,2020-07-17 05:33:48.0,2020-04-30 04:54:27.0,907,53,INR,purple,102-10-5639,3,2,3
BK4529355854,60071878,78022360,2.1.9,iOS,1.215274,-56.014903,35.152876,104.324905,2020-01-02 01:48:40.0,2020-02-16 04:28:55.0,547,17,INR,teal,866-83-4349,2,3,4
BK9720088219,14327312,94427067,3.1.2,Android,-55.4822225,173.362256,65.0121265,51.390751,2020-04-10 15:11:07.0,2020-01-20 21:17:42.0,259,33,INR,maroon,572-73-6526,3,3,2
BK7157532607,46407210,43160003,1.3.4,Android,46.005843,-16.826146,7.6126015,-156.428577,2020-06-09 05:56:31.0,2020-03-19 01:53:16.0,787,21,INR,olive,667-23-5880,2,2,3
BK5014871433,65861573,64708618,1.3.28,iOS,-29.565326,64.843709,84.068109,-49.820835,2020-08-14 20:43:42.0,2020-06-03 09:39:59.0,586,5,INR,fuchsia,255-52-5654,5,5,1
BK9051488736,37721758,27297770,2.3.13,Android,61.9364605,83.249705,0.0281895,115.469099,2020-04-07 04:27:59.0,2020-09-29 10:51:41.0,912,80,INR,aqua,739-09-9569,2,1,2
BK243762319,62552969,45877457,3.3.9,iOS,-62.6515155,-139.154028,28.0299995,-62.8556,2020-07-01 00:36:05.0,2020-09-30 17:40:23.0,821,23,INR,black,590-44-6613,2,3,4
BK4683595168,56801961,53401707,4.2.34,iOS,-5.860265,-100.004839,25.016591,70.471358,2020-05-03 10:17:56.0,2020-06-08 09:11:27.0,71,10,INR,fuchsia,454-04-0608,5,2,3
BK9783284253,66909721,40509554,2.2.22,Android,36.1913155,5.686264,88.988393,36.580599,2020-03-05 16:02:01.0,2020-05-29 13:36:15.0,26,81,INR,black,600-17-7043,3,1,3
BK2880021380,50163555,34005420,3.4.23,Android,-83.06599,106.268689,8.8300855,74.872352,2020-01-15 02:00:07.0,2020-05-12 21:53:04.0,571,99,INR,navy,506-09-4981,1,5,3
BK4537426043,91111754,59250769,1.2.19,iOS,-43.1188435,-99.935719,3.7026225,-46.828716,2020-04-28 05:18:34.0,2020-02-12 11:31:40.0,650,81,INR,white,362-35-8054,5,5,2
BK998130731,67875357,14562526,3.1.15,Android,-10.861959,-111.989853,57.233121,95.469986,2020-01-25 01:37:22.0,2020-04-28 09:42:00.0,590,3,INR,teal,359-51-9362,1,1,4
BK5645323730,18442993,84939946,3.1.29,Android,-81.472235,-88.404916,12.690818,-140.99768,2020-09-24 08:18:31.0,2020-07-16 05:12:24.0,515,1,INR,blue,024-35-8771,1,3,4
BK6163608413,36591778,11946210,4.2.38,Android,60.2036385,120.988501,32.103263,-50.551889,2020-07-26 06:12:56.0,2020-04-23 06:57:20.0,810,58,INR,silver,833-16-1378,3,5,1
BK6803373649,28382306,97222676,2.1.18,iOS,6.540056,161.083998,-12.943502,-148.232621,2020-09-20 15:52:49.0,2020-09-17 03:13:26.0,927,74,INR,maroon,747-70-5557,2,2,4
BK9764570097,61225539,15265942,4.4.14,iOS,80.7211615,179.695812,-33.345655,134.010372,2020-01-26 02:20:39.0,2020-06-12 15:05:49.0,246,72,INR,blue,332-71-7565,5,1,2
BK8362601204,79115927,60281490,3.4.16,iOS,-9.4458645,101.745883,80.264612,-46.718091,2020-09-03 13:32:13.0,2020-02-01 21:02:21.0,887,88,INR,blue,225-31-0761,4,1,1
BK6225330481,51110772,58392277,3.3.27,iOS,68.1306075,141.450665,-6.926722,-64.24554,2020-05-11 06:25:10.0,2020-06-29 09:31:03.0,429,18,INR,purple,229-41-2152,5,1,3
BK9785297548,13229062,95750789,4.1.14,Android,-57.959554,-172.155546,1.667888,126.729718,2020-04-18 20:12:36.0,2020-01-14 09:07:36.0,35,63,INR,white,681-74-4532,2,3,3
BK2170342110,76255097,74117472,3.4.11,Android,14.3796175,17.089235,69.1569595,67.210169,2020-10-01 10:46:07.0,2020-03-27 22:30:08.0,927,68,INR,silver,677-03-5852,5,4,3
BK4218069991,86269148,49943601,4.2.9,Android,74.050649,-130.95903,-85.015584,106.190804,2020-08-22 18:50:02.0,2020-01-25 09:57:30.0,385,97,INR,aqua,678-19-9649,2,4,2
```

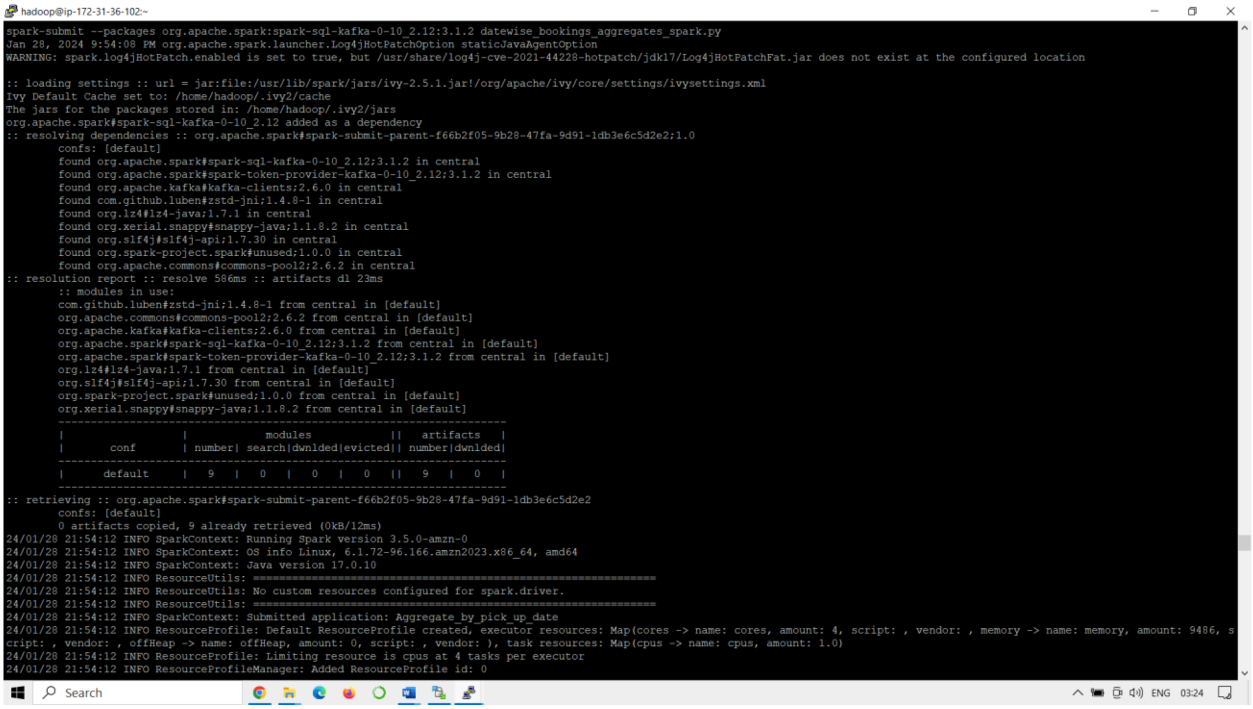
Step 4: The next step involves creating a aggregated file-> the number of bookings needs to be aggregated at daywise level.

The code is as follows:

```
from pyspark.sql import SparkSession
from pyspark.sql.functions import *

Spark=
SparkSession.builder.appName("Aggregate_by_pick_up_date").master("local").getOrCreate()

df=spark.read.csv("/user/root/bookings_1/part-m-00000")
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"]

#Creating a new Dataframe form the old data frame and assigning a new schema
new_df = df.toDF(*new_col)

new_df.show(5)

# Crearing a new column which has only the date part from the pickup_timestamp column
df_new= new_df.withColumn("Pickupdate",col("pickup_timestamp").cast("date"))
df_new.show(5)
```

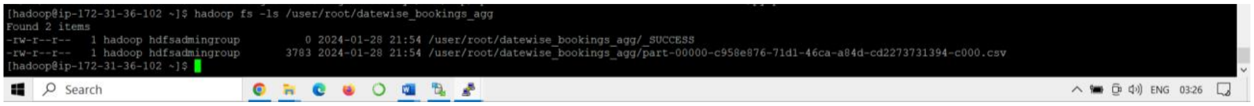
```
# grouping the data by the new column Pickupdate
agg_df =
df_new.groupBy("Pickupdate").agg(count("booking_id").alias("booking_count")).orderBy("Pickupdate")


# Copying the data to a csv file
agg_df.coalesce(1).write.format('csv').mode('overwrite').save('/user/root/datewise_bookings_agg',header='true')
```

- ➔ Ensure the spark file is copied to /user/livy folder
- ➔ The attached file name is “datewise_bookings_aggregates_spark.py”
- ➔ Code to run the spark job -> “export SPARK_KAFKA_VERSION=0.10
  spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.1.2
  datewise_bookings_aggregates_spark.py”
- ➔ Screenshot of running the file :



```
spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.1.2 datewise_bookings_aggregates_spark.py
Jan 28, 2024 9:54:08 PM org.apache.spark.launcher.Log4jHotPatchOption staticJavaAgentOption
WARNING: spark.log4jHotPatch.enabled is set to true, but /usr/share/log4j-cve-2021-44228-hotpatch/jdk17/Log4jHotPatchFat.jar does not exist at the configured location

:: loading settings :: url = jar:file:/usr/lib/spark/jars/ivy-2.5.1.jar!/org/apache/ivy/core/settings/ivysettings.xml
Ivy Default Cache set to: /home/hadoop/.ivy2/cache
The jars for the packages stored in: /home/hadoop/.ivy2/jars
org.apache.spark#spark-sql-kafka-0-10_2.12 added as a dependency
:: resolving dependencies :: org.apache.spark#spark-submit-parent-f66b2f05-9b28-47fa-9d91-1db3e6c5d2e2;1.0
        confs: [default]
        found org.apache.spark#spark-sql-kafka-0-10_2.12;3.1.2 in central
        found org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.1.2 in central
        found org.apache.kafka#kafka-clients;2.6.0 in central
        found com.github.luben#zstd-jni;1.4.8-1 in central
        found org.lz4#lz4-java;1.7.1 in central
        found org.xerial.snappy#snappy-java;1.1.8.2 in central
        found org.slf4j#slf4j-api;1.7.30 in central
        found org.spark-project.spark#unused;1.0.0 in central
        found org.apache.commons#commons-pool2;2.6.2 in central
:: resolution report :: resolve 586ms :: artifacts dl 23ms
        :: modules in use:
        com.github.luben#zstd-jni;1.4.8-1 from central in [default]
        org.apache.commons#commons-pool2;2.6.2 from central in [default]
        org.apache.kafka#kafka-clients;2.6.0 from central in [default]
        org.apache.spark#spark-sql-kafka-0-10_2.12;3.1.2 from central in [default]
        org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.1.2 from central in [default]
        org.lz4#lz4-java;1.7.1 from central in [default]
        org.slf4j#slf4j-api;1.7.30 from central in [default]
        org.spark-project.spark#unused;1.0.0 from central in [default]
        org.xerial.snappy#snappy-java;1.1.8.2 from central in [default]
        ---------------------------------------------------------------------
        |                  |            modules            ||   artifacts   |
        |       conf       | number| search|dwnlded|evicted|| number|dwnlded|
        ---------------------------------------------------------------------
        |      default     |   9   |   0   |   0   |   0   ||   9   |   0   |
        ---------------------------------------------------------------------
:: retrieving :: org.apache.spark#spark-submit-parent-f66b2f05-9b28-47fa-9d91-1db3e6c5d2e2
        confs: [default]
        0 artifacts copied, 9 already retrieved (0kB/12ms)
24/01/28 21:54:12 INFO SparkContext: Running Spark version 3.5.0-amzn-0
24/01/28 21:54:12 INFO SparkContext: OS info Linux, 6.1.72-96.166.amzn2023.x86_64, amd64
24/01/28 21:54:12 INFO SparkContext: Java version 17.0.10
24/01/28 21:54:12 INFO ResourceUtils: ==============================================================
24/01/28 21:54:12 INFO ResourceUtils: No custom resources configured for spark.driver.
24/01/28 21:54:12 INFO ResourceUtils: ==============================================================
24/01/28 21:54:12 INFO SparkContext: Submitted application: Aggregate_by_pick_up_date
24/01/28 21:54:12 INFO ResourceProfile: Default ResourceProfile created, executor resources: Map(cores -> name: cores, amount: 4, script: , vendor: , memory -> name: memory, amount: 9486, s
cript: , vendor: , offHeap -> name: offHeap, amount: 0, script: , vendor: ), task resources: Map(cpus -> name: cpus, amount: 1.0)
24/01/28 21:54:12 INFO ResourceProfile: Limiting resource is cpus at 4 tasks per executor
24/01/28 21:54:12 INFO ResourceProfileManager: Added ResourceProfile id: 0
```

- ➔ Verifying the csv file got generated at the target location:
  - Code: hadoop fs -ls /user/root/datewise_bookings_agg



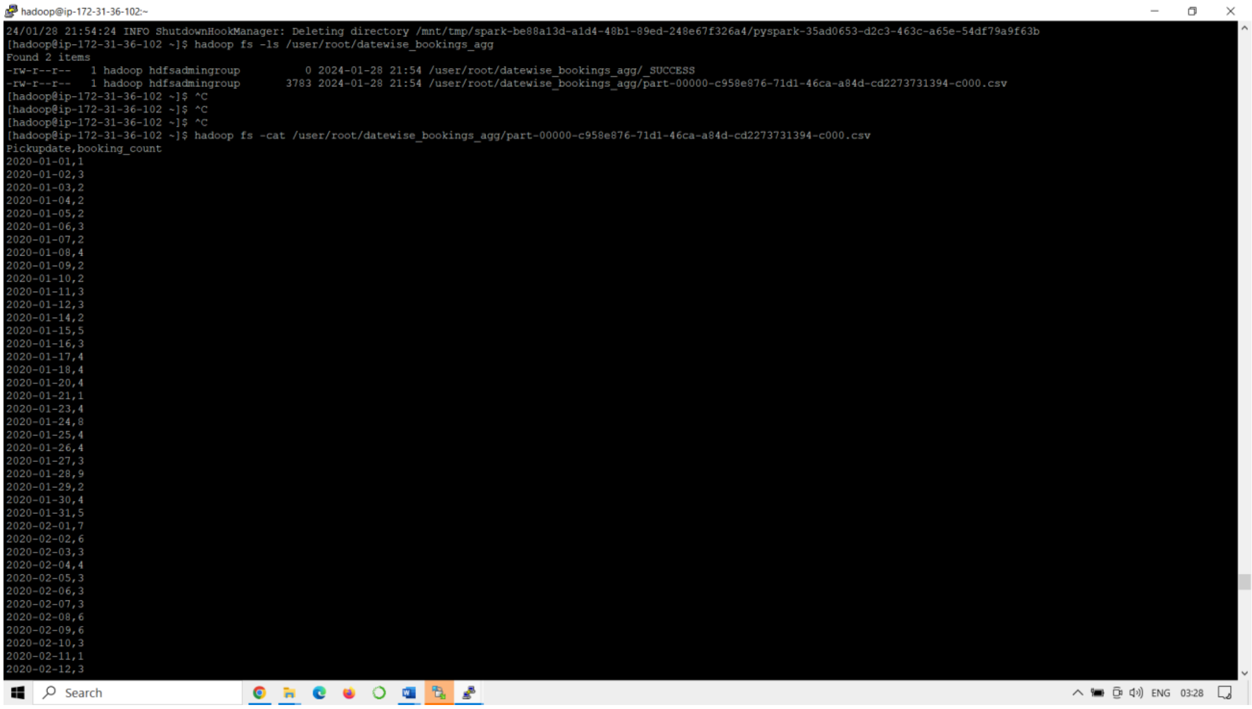
```
[hadoop@ip-172-31-36-102 ~]$ hadoop fs -ls /user/root/datewise_bookings_agg
Found 2 items
-rw-r--r--   1 hadoop hdfsadmingroup          0 2024-01-28 21:54 /user/root/datewise_bookings_agg/_SUCCESS
-rw-r--r--   1 hadoop hdfsadmingroup       3783 2024-01-28 21:54 /user/root/datewise_bookings_agg/part-00000-c958e876-71d1-46ca-a84d-cd2273731394-c000.csv
[hadoop@ip-172-31-36-102 ~]$ 
```

- ➔ Screenshot of the data stored in the csv file

- Code: hadoop fs -cat /user/root/datewise_bookings_agg/part-00000-c958e876-71d1-46ca-a84d-cd2273731394-c000.csv

## change the csv file name

➔ Screenshot of the above point :



```
24/01/28 21:54:24 INFO ShutdownHookManager: Deleting directory /mnt/tmp/spark-be88a13d-a1d4-48b1-89ed-248e67f326a4/pyspark-35ad0653-d2c3-463c-a65e-54df79a9f63b
[hadoop@ip-172-31-36-102 ~]$ hadoop fs -ls /user/root/datewise_bookings_agg
Found 2 items
-rw-r--r--   1 hadoop hdfsadmingroup          0 2024-01-28 21:54 /user/root/datewise_bookings_agg/_SUCCESS
-rw-r--r--   1 hadoop hdfsadmingroup       3783 2024-01-28 21:54 /user/root/datewise_bookings_agg/part-00000-c958e876-71d1-46ca-a84d-cd2273731394-c000.csv
[hadoop@ip-172-31-36-102 ~]$ ^C
[hadoop@ip-172-31-36-102 ~]$ ^C
[hadoop@ip-172-31-36-102 ~]$ ^C
[hadoop@ip-172-31-36-102 ~]$ hadoop fs -cat /user/root/datewise_bookings_agg/part-00000-c958e876-71d1-46ca-a84d-cd2273731394-c000.csv
Pickupdate,booking_count
2020-01-01,1
2020-01-02,3
2020-01-03,2
2020-01-04,2
2020-01-05,2
2020-01-06,3
2020-01-07,2
2020-01-08,4
2020-01-09,2
2020-01-10,2
2020-01-11,3
2020-01-12,3
2020-01-14,2
2020-01-15,5
2020-01-16,3
2020-01-17,4
2020-01-18,4
2020-01-20,4
2020-01-21,1
2020-01-23,4
2020-01-24,8
2020-01-25,4
2020-01-26,4
2020-01-27,3
2020-01-28,9
2020-01-29,2
2020-01-30,4
2020-01-31,5
2020-02-01,7
2020-02-02,6
2020-02-03,3
2020-02-04,4
2020-02-05,3
2020-02-06,3
2020-02-07,3
2020-02-08,6
2020-02-09,6
2020-02-10,3
2020-02-11,1
2020-02-12,3
```

<u>Step 5:</u> The next step involves generating the hive tables on the ingested data

Before creating the hive tables, Follow the following commands:

Type Hive to get into Hive CLI
Then Create a Database:

➔ Create Database CabRideProject;

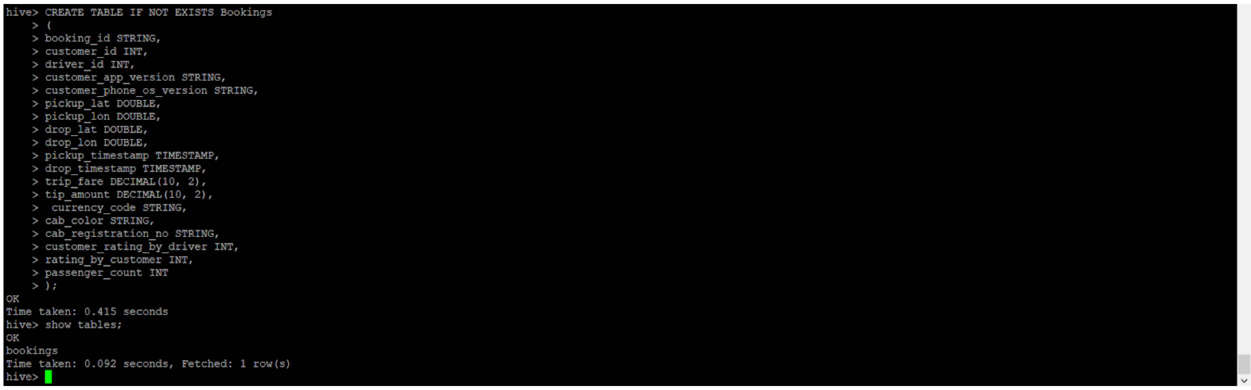
➔ Use CabRideProject;

1. Creating the Hive table to store the Booking data

➔

```
CREATE TABLE IF NOT EXISTS Bookings
(
booking_id STRING,
customer_id INT,
driver_id INT,
customer_app_version STRING,
customer_phone_os_version STRING,
pickup_lat DOUBLE,
pickup_lon DOUBLE,
drop_lat DOUBLE,
drop_lon DOUBLE,
pickup_timestamp TIMESTAMP,
drop_timestamp TIMESTAMP,
trip_fare DECIMAL(10, 2),
tip_amount DECIMAL(10, 2),
 currency_code STRING,
cab_color STRING,
cab_registration_no STRING,
customer_rating_by_driver INT,
rating_by_customer INT,
passenger_count INT
);
```

Screen shot of the table creation



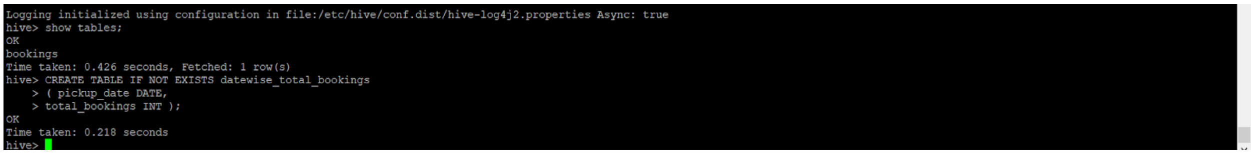
```
hive> CREATE TABLE IF NOT EXISTS Bookings
    > (
    > booking_id STRING,
    > customer_id INT,
    > driver_id INT,
    > customer_app_version STRING,
    > customer_phone_os_version STRING,
    > pickup_lat DOUBLE,
    > pickup_lon DOUBLE,
    > drop_lat DOUBLE,
    > drop_lon DOUBLE,
    > pickup_timestamp TIMESTAMP,
    > drop_timestamp TIMESTAMP,
    > trip_fare DECIMAL(10, 2),
    > tip_amount DECIMAL(10, 2),
    >  currency_code STRING,
    > cab_color STRING,
    > cab_registration_no STRING,
    > customer_rating_by_driver INT,
    > rating_by_customer INT,
    > passenger_count INT
    > );
OK
Time taken: 0.415 seconds
hive> show tables;
OK
bookings
Time taken: 0.092 seconds, Fetched: 1 row(s)
hive>
```

2. **Creating table for storing then aggregated booking values [daywise]**

   ➔

```
CREATE TABLE IF NOT EXISTS datewise_total_bookings
( pickup_date DATE,
total_bookings INT );
```

Screenshot for the same :



```
Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: true
hive> show tables;
OK
bookings
Time taken: 0.426 seconds, Fetched: 1 row(s)
hive> CREATE TABLE IF NOT EXISTS datewise_total_bookings
    > ( pickup_date DATE,
    > total_bookings INT );
OK
Time taken: 0.218 seconds
hive>
```

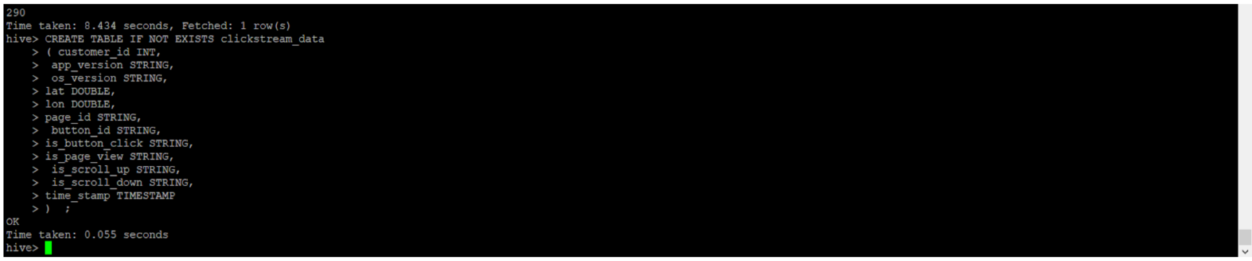
3. **Creating table for storing data for the click stream**

   ➔

```
CREATE TABLE IF NOT EXISTS clickstream_data
( customer_id INT,
 app_version STRING,
```

 os_version STRING,
lat DOUBLE,
lon DOUBLE,
page_id STRING,
 button_id STRING,
is_button_click STRING,
is_page_view STRING,
 is_scroll_up STRING,
 is_scroll_down STRING,
time_stamp TIMESTAMP
) ;

Screenshot for the table creation command execution:



```
290
Time taken: 8.434 seconds, Fetched: 1 row(s)
hive> CREATE TABLE IF NOT EXISTS clickstream_data
    > ( customer_id INT,
    >  app_version STRING,
    >  os_version STRING,
    > lat DOUBLE,
    > lon DOUBLE,
    > page_id STRING,
    >  button_id STRING,
    > is_button_click STRING,
    > is_page_view STRING,
    >  is_scroll_up STRING,
    >  is_scroll_down STRING,
    > time_stamp TIMESTAMP
    > )  ;
OK
Time taken: 0.055 seconds
hive>
```

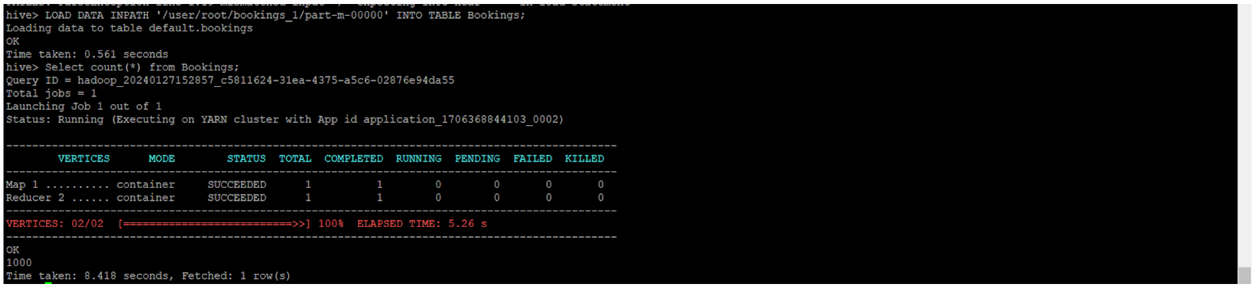
<Command to load the data into Hive tables>

Loading the data from hadoop to the hive tables;

1. Uploading the data from hadoop to Hive table for storing the bookings data ingested from AWS RDS to hadoop using sqoop
   ➔

   LOAD DATA INPATH '/user/root/bookings_1/part-m-00000' OVERWRITE INTO TABLE Bookings

   Screenshot for the running the same command and verifying the number of rows in the hive table



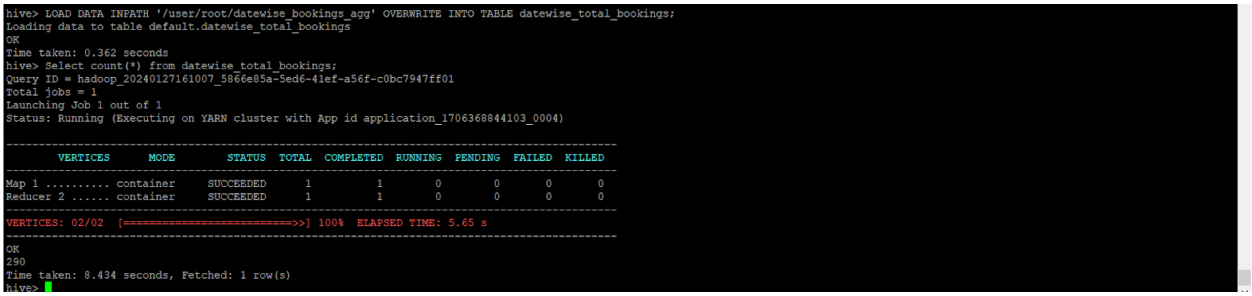
```
hive> LOAD DATA INPATH '/user/root/bookings_1/part-m-00000' INTO TABLE Bookings;
Loading data to table default.bookings
OK
Time taken: 0.561 seconds
hive> Select count(*) from Bookings;
Query ID = hadoop_20240127152857_c5811624-31ea-4375-a5c6-02876e94da55
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1706368844103_0002)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 5.26 s
----------------------------------------------------------------------------------------------
OK
1000
Time taken: 8.418 seconds, Fetched: 1 row(s)
```

2. Uploading the data from hadoop to Hive table for the aggregated data created as a datewise aggregation of bookings

   ➔

   LOAD DATA INPATH '/user/root/datewise_bookings_agg' OVERWRITE INTO TABLE datewise_total_bookings;

The image also contains verification of rows present in the hive table

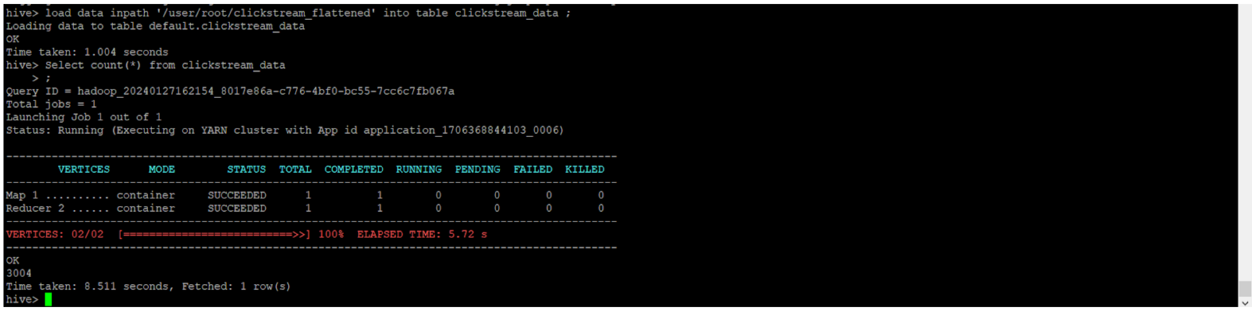


```
hive> LOAD DATA INPATH '/user/root/datewise_bookings_agg' OVERWRITE INTO TABLE datewise_total_bookings;
Loading data to table default.datewise_total_bookings
OK
Time taken: 0.362 seconds
hive> Select count(*) from datewise_total_bookings;
Query ID = hadoop_20240127161007_5866e85a-5ed6-41ef-a56f-c0bc7947ff01
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1706368844103_0004)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 5.65 s
----------------------------------------------------------------------------------------------
OK
290
Time taken: 8.434 seconds, Fetched: 1 row(s)
hive>
```

3. Uploading the data form hadoop to hive -> This is for the data pulled form kafka and stored in the hadoop

   ➔

   LOAD DATA INPATH ‘/user/root/clickstream_flattened’ into table clickstream_data;



```
hive> load data inpath '/user/root/clickstream_flattened' into table clickstream_data ;
Loading data to table default.clickstream_data
OK
Time taken: 1.004 seconds
hive> Select count(*) from clickstream_data
    > ;
Query ID = hadoop_20240127162154_8017e86a-c776-4bf0-bc55-7cc6c7fb067a
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1706368844103_0006)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 5.72 s
----------------------------------------------------------------------------------------------
OK
3004
Time taken: 8.511 seconds, Fetched: 1 row(s)
hive>
```

Below image also holds the verification query to run the numbers of rows