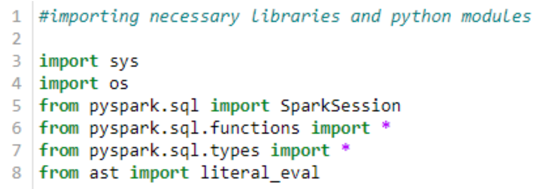
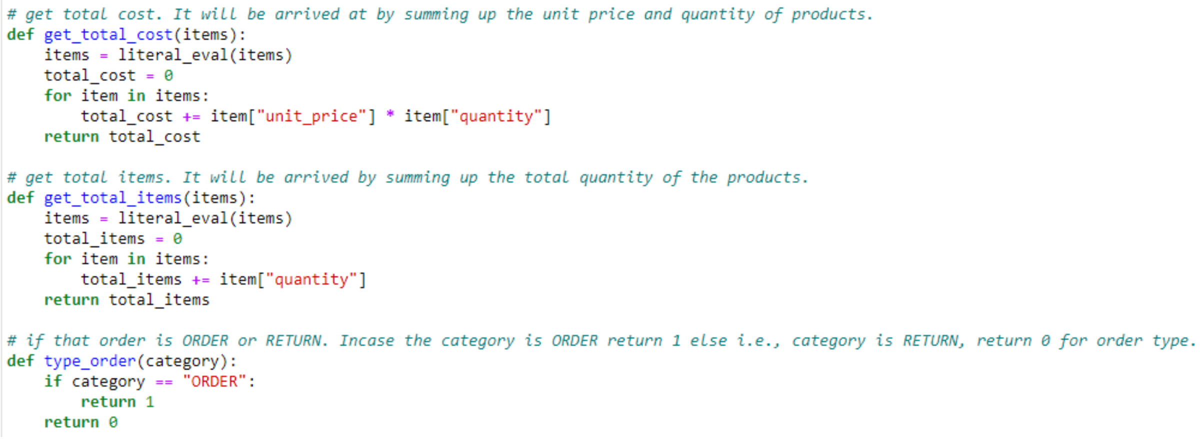
Code Logic - Retail Data Analysis



```
#importing necessary libraries and python modules

import sys
import os
from pyspark.sql import SparkSession
from pyspark.sql.functions import *
from pyspark.sql.types import *
from ast import literal_eval
```

Required python modules and libraries are imported here. Literal_eval will be used to convert string from items column into a proper python list of dictionary.



```
# get total cost. It will be arrived at by summing up the unit price and quantity of products.
def get_total_cost(items):
    items = literal_eval(items)
    total_cost = 0
    for item in items:
        total_cost += item["unit_price"] * item["quantity"]
    return total_cost

# get total items. It will be arrived by summing up the total quantity of the products.
def get_total_items(items):
    items = literal_eval(items)
    total_items = 0
    for item in items:
        total_items += item["quantity"]
    return total_items

# if that order is ORDER or RETURN. Incase the category is ORDER return 1 else i.e., category is RETURN, return 0 for order type.
def type_order(category):
    if category == "ORDER":
        return 1
    return 0
```

Below are the details of the custom functions:

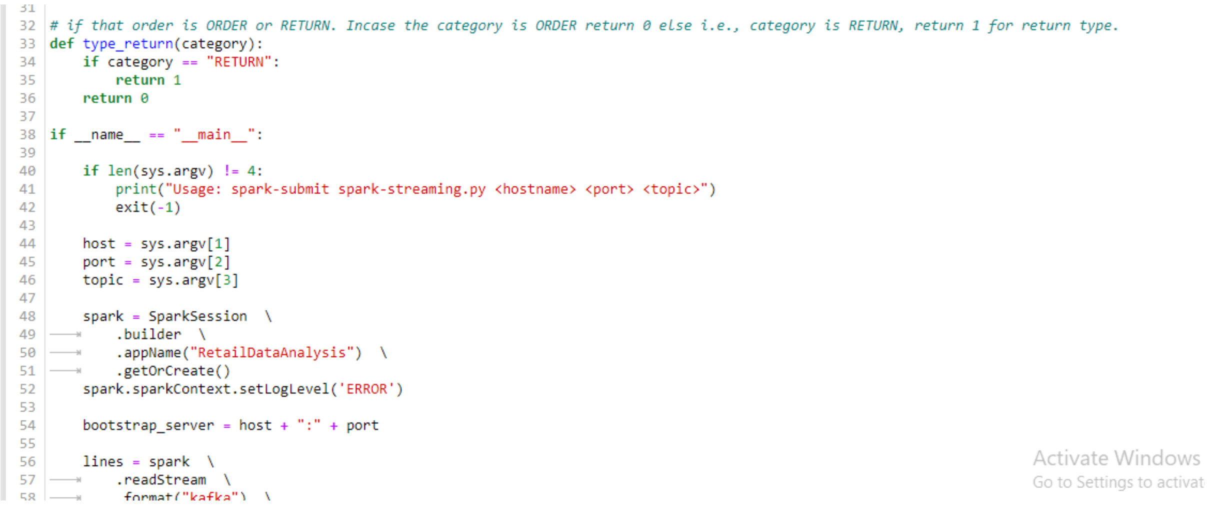
1. get_total_cost(items): This function takes item as an argument and calculate the total cost by summing up the unit price and quantity of products. The formulae to calculate get total cost is :

$$\sum(\text{quantity} * \text{unitprice})$$

2. get_total_items(items): This function takes item as an argument and used to retrieve the total items by summing up total quantity of the products. The formulae to calculate the total item is:

$$\sum(\text{quantity})$$

3. type_order(category): This function takes category as an argument, and it is used to map type of order. If that order is ORDER or RETURN. In case the category is ORDER return 1 else i.e., category is RETURN, return 0 for order type.
4. type_return(category): This function takes category as an argument, and it is used to map type of order. If that order is ORDER or RETURN. In case the category is ORDER return 0 else i.e., category is RETURN, return 1 for return type.



```python
# if that order is ORDER or RETURN. Incase the category is ORDER return 0 else i.e., category is RETURN, return 1 for return type.
def type_return(category):
    if category == "RETURN":
        return 1
    return 0

if __name__ == "__main__":

    if len(sys.argv) != 4:
        print("Usage: spark-submit spark-streaming.py <hostname> <port> <topic>")
        exit(-1)

    host = sys.argv[1]
    port = sys.argv[2]
    topic = sys.argv[3]

    spark = SparkSession  \
        .builder  \
        .appName("RetailDataAnalysis")  \
        .getOrCreate()
    spark.sparkContext.setLogLevel('ERROR')

    bootstrap_server = host + ":" + port

    lines = spark  \
        .readStream  \
        .format("kafka")  \
```

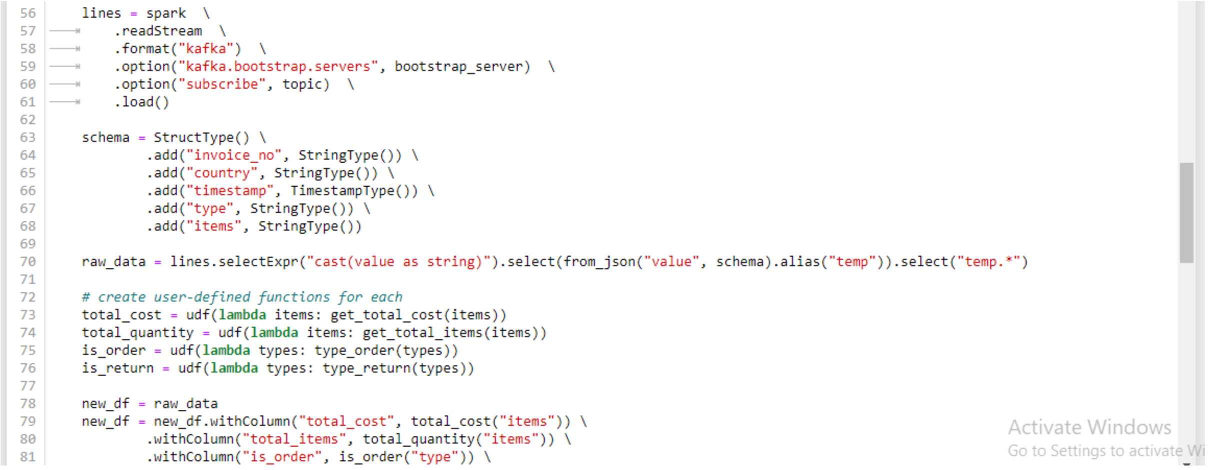
In the beginning, the host, port number and topic is received from the command line argument.

host: 18.211.252.152

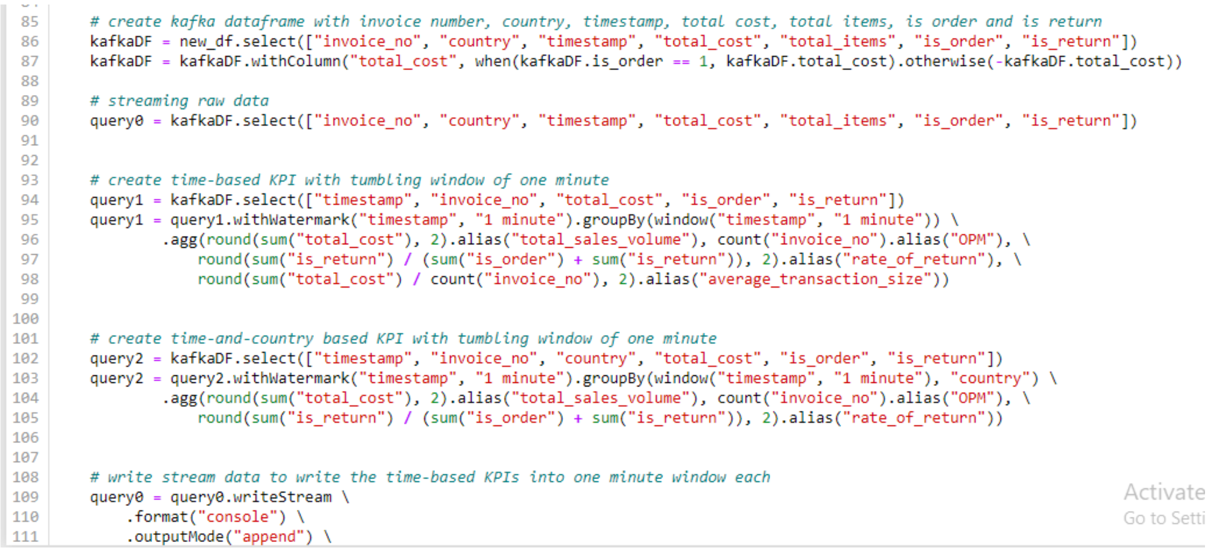
port: 9092

topic: real-time-project

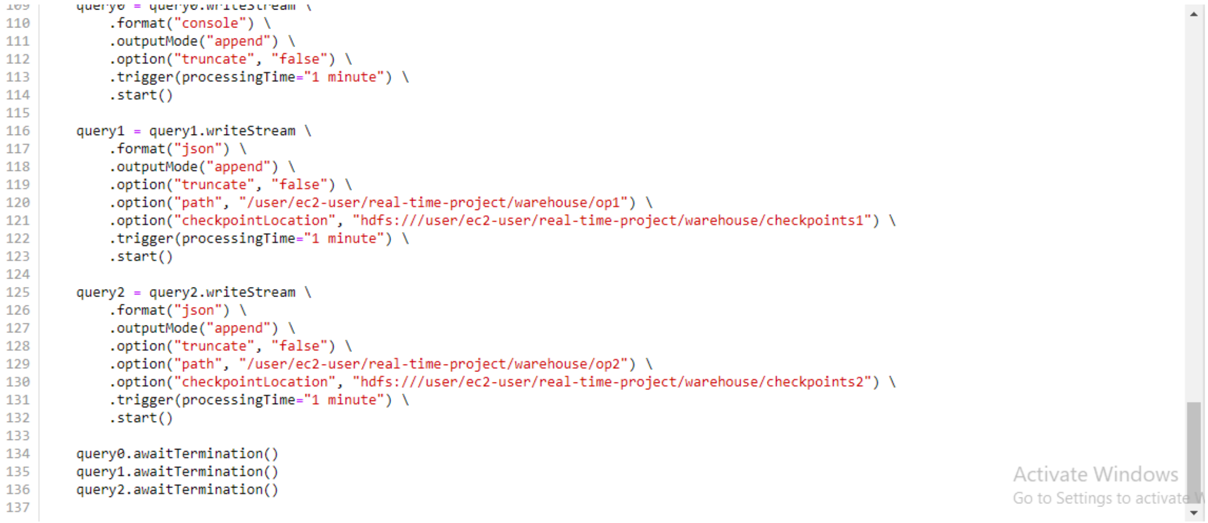
This is used to read the spark stream from kafka bootstrap server.



```python
    lines = spark  \
        .readStream  \
        .format("kafka")  \
        .option("kafka.bootstrap.servers", bootstrap_server)  \
        .option("subscribe", topic)  \
        .load()

    schema = StructType() \
            .add("invoice_no", StringType()) \
            .add("country", StringType()) \
            .add("timestamp", TimestampType()) \
            .add("type", StringType()) \
            .add("items", StringType())

    raw_data = lines.selectExpr("cast(value as string)").select(from_json("value", schema).alias("temp")).select("temp.*")

    # create user-defined functions for each
    total_cost = udf(lambda items: get_total_cost(items))
    total_quantity = udf(lambda items: get_total_items(items))
    is_order = udf(lambda types: type_order(types))
    is_return = udf(lambda types: type_return(types))

    new_df = raw_data
    new_df = new_df.withColumn("total_cost", total_cost("items")) \
            .withColumn("total_items", total_quantity("items")) \
            .withColumn("is_order", is_order("type")) \
```



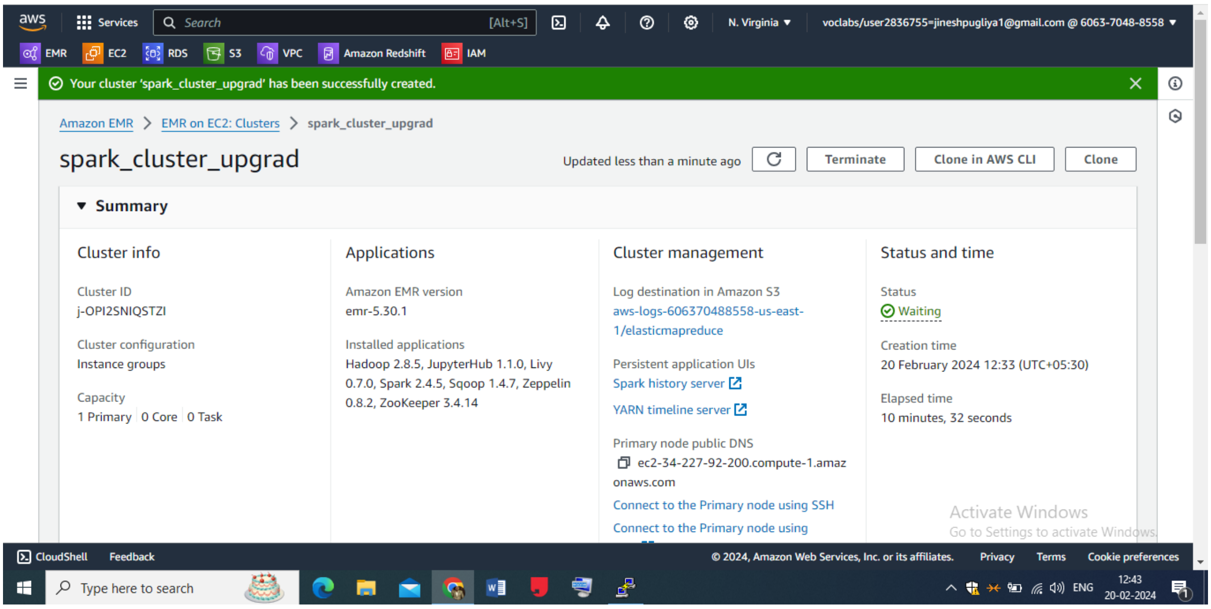
```python
    # create kafka dataframe with invoice number, country, timestamp, total cost, total items, is order and is return
    kafkaDF = new_df.select(["invoice_no", "country", "timestamp", "total_cost", "total_items", "is_order", "is_return"])
    kafkaDF = kafkaDF.withColumn("total_cost", when(kafkaDF.is_order == 1, kafkaDF.total_cost).otherwise(-kafkaDF.total_cost))

    # streaming raw data
    query0 = kafkaDF.select(["invoice_no", "country", "timestamp", "total_cost", "total_items", "is_order", "is_return"])


    # create time-based KPI with tumbling window of one minute
    query1 = kafkaDF.select(["timestamp", "invoice_no", "total_cost", "is_order", "is_return"])
    query1 = query1.withWatermark("timestamp", "1 minute").groupBy(window("timestamp", "1 minute")) \
            .agg(round(sum("total_cost"), 2).alias("total_sales_volume"), count("invoice_no").alias("OPM"), \
                round(sum("is_return") / (sum("is_order") + sum("is_return")), 2).alias("rate_of_return"), \
                round(sum("total_cost") / count("invoice_no"), 2).alias("average_transaction_size"))


    # create time-and-country based KPI with tumbling window of one minute
    query2 = kafkaDF.select(["timestamp", "invoice_no", "country", "total_cost", "is_order", "is_return"])
    query2 = query2.withWatermark("timestamp", "1 minute").groupBy(window("timestamp", "1 minute"), "country") \
            .agg(round(sum("total_cost"), 2).alias("total_sales_volume"), count("invoice_no").alias("OPM"), \
                round(sum("is_return") / (sum("is_order") + sum("is_return")), 2).alias("rate_of_return"))


    # write stream data to write the time-based KPIs into one minute window each
    query0 = query0.writeStream \
        .format("console") \
        .outputMode("append") \
```



```
    query0 = query0.writeStream \
        .format("console") \
        .outputMode("append") \
        .option("truncate", "false") \
        .trigger(processingTime="1 minute") \
        .start()

    query1 = query1.writeStream \
        .format("json") \
        .outputMode("append") \
        .option("truncate", "false") \
        .option("path", "/user/ec2-user/real-time-project/warehouse/op1") \
        .option("checkpointLocation", "hdfs:///user/ec2-user/real-time-project/warehouse/checkpoints1") \
        .trigger(processingTime="1 minute") \
        .start()

    query2 = query2.writeStream \
        .format("json") \
        .outputMode("append") \
        .option("truncate", "false") \
        .option("path", "/user/ec2-user/real-time-project/warehouse/op2") \
        .option("checkpointLocation", "hdfs:///user/ec2-user/real-time-project/warehouse/checkpoints2") \
        .trigger(processingTime="1 minute") \
        .start()

    query0.awaitTermination()
    query1.awaitTermination()
    query2.awaitTermination()
```

Code Deployment and Execution Steps:

1. Create an EMR Instance with **Spark and ZooKeeper** applications installed in it



Your cluster 'spark_cluster_upgrad' has been successfully created.

spark_cluster_upgrad

Updated less than a minute ago

▼ Summary

Cluster info

Cluster ID
j-OPI2SNIQSTZI

Cluster configuration
Instance groups

Capacity
1 Primary | 0 Core | 0 Task

Applications

Amazon EMR version
emr-5.30.1

Installed applications
Hadoop 2.8.5, JupyterHub 1.1.0, Livy 0.7.0, Spark 2.4.5, Sqoop 1.4.7, Zeppelin 0.8.2, ZooKeeper 3.4.14

Cluster management

Log destination in Amazon S3
aws-logs-606370488558-us-east-1/elasticmapreduce

Persistent application UIs
Spark history server
YARN timeline server

Primary node public DNS
ec2-34-227-92-200.compute-1.amazonaws.com

Connect to the Primary node using SSH
Connect to the Primary node using

Status and time

Status
Waiting

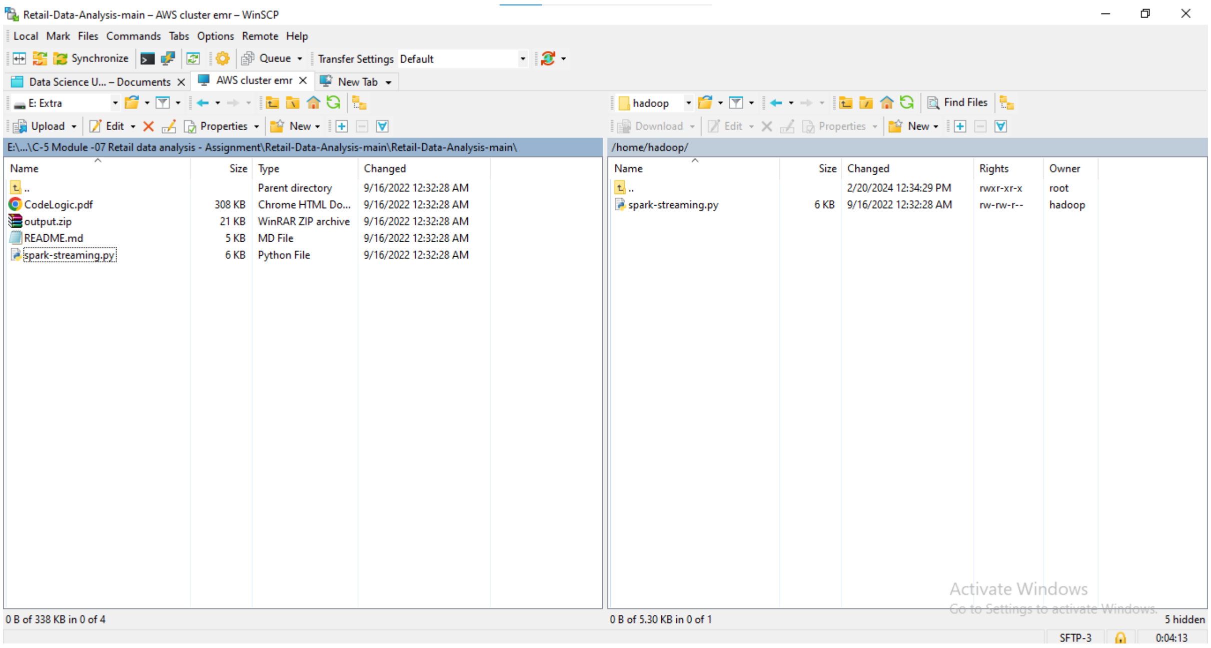
Creation time
20 February 2024 12:33 (UTC+05:30)

Elapsed time
10 minutes, 32 seconds

© 2024, Amazon Web Services, Inc. or its affiliates.

12:43
20-02-2024

2. Copy the python file (spark-streaming.py) under the path: /home/hadoop using WinSCP files transfer application (Windows OS).


Retail-Data-Analysis-main – AWS cluster emr – WinSCP
Local Mark Files Commands Tabs Options Remote Help
Synchronize
Queue
Transfer Settings Default
Data Science U... – Documents
AWS cluster emr
New Tab
E: Extra
Upload
Edit
Properties
New
hadoop
Find Files
Download
Edit
Properties
New
Activate Windows
Go to Settings to activate Windows.

E:\...\C-5 Module -07 Retail data analysis - Assignment\Retail-Data-Analysis-main\Retail-Data-Analysis-main\

Name	Size	Type	Changed
..		Parent directory	9/16/2022 12:32:28 AM
CodeLogic.pdf	308 KB	Chrome HTML Do...	9/16/2022 12:32:28 AM
output.zip	21 KB	WinRAR ZIP archive	9/16/2022 12:32:28 AM
README.md	5 KB	MD File	9/16/2022 12:32:28 AM
spark-streaming.py	6 KB	Python File	9/16/2022 12:32:28 AM

0 B of 338 KB in 0 of 4

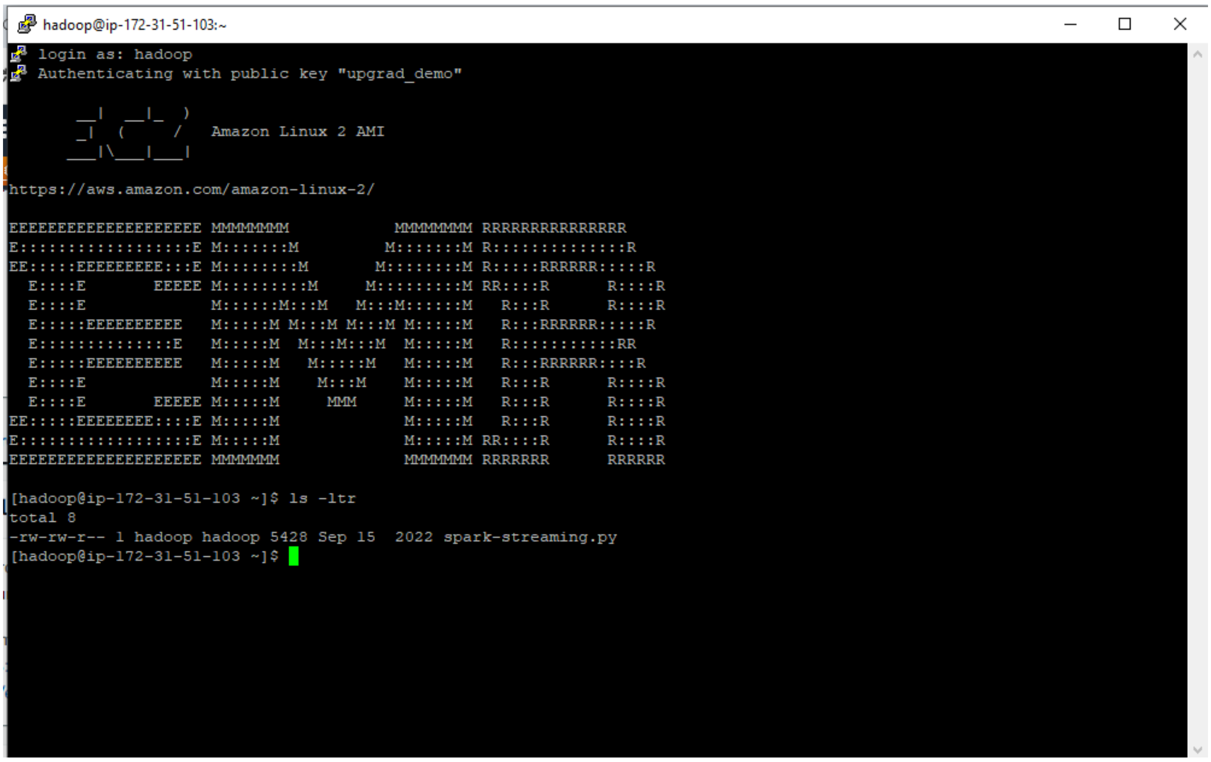
/home/hadoop/

Name	Size	Changed	Rights	Owner
..		2/20/2024 12:34:29 PM	rwxr-xr-x	root
spark-streaming.py	6 KB	9/16/2022 12:32:28 AM	rw-rw-r--	hadoop

0 B of 5.30 KB in 0 of 1

5 hidden

SFTP-3 0:04:13



```
login as: hadoop
Authenticating with public key "upgrad_demo"

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-51-103 ~]$ ls -ltr
total 8
-rw-rw-r-- 1 hadoop hadoop 5428 Sep 15  2022 spark-streaming.py
[hadoop@ip-172-31-51-103 ~]$
```

3. Run the following command to enable Kafka Integration with Apache Spark.
   export SPARK_KAFKA_VERSION=0.10
4. Execute the python file using spark-submit command providing Kafka jar package as an argument. Save the console output in a text file (Console-output.txt).

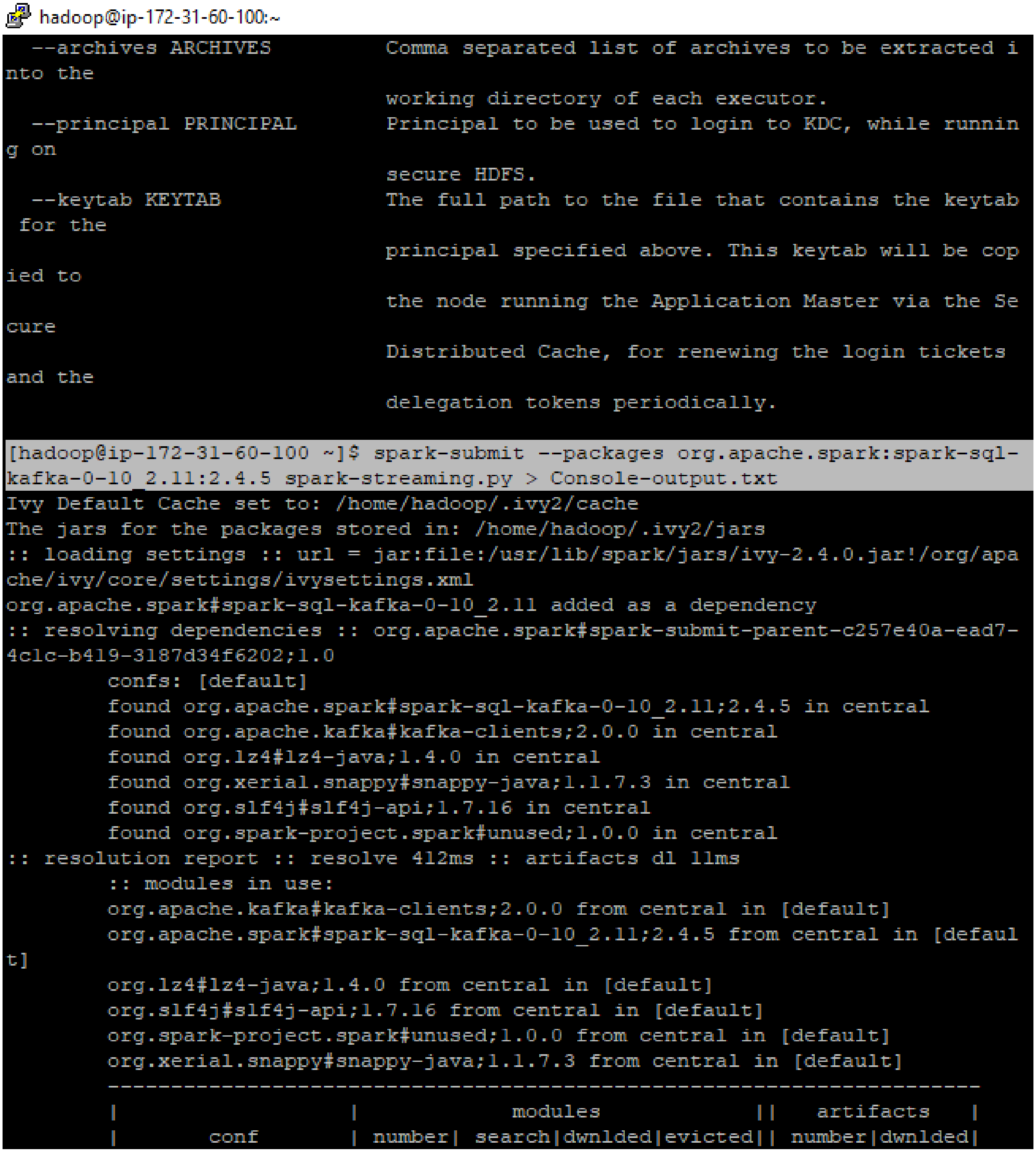
   Bootstrap Server - 18.211.252.152

   Port - 9092

   Topic - real-time-project

spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.5 spark-streaming.py > Console-output.txt

spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.5 spark-streaming.py > 18.211.252.152 9092 real-time-project



hadoop@ip-172-31-60-100:~

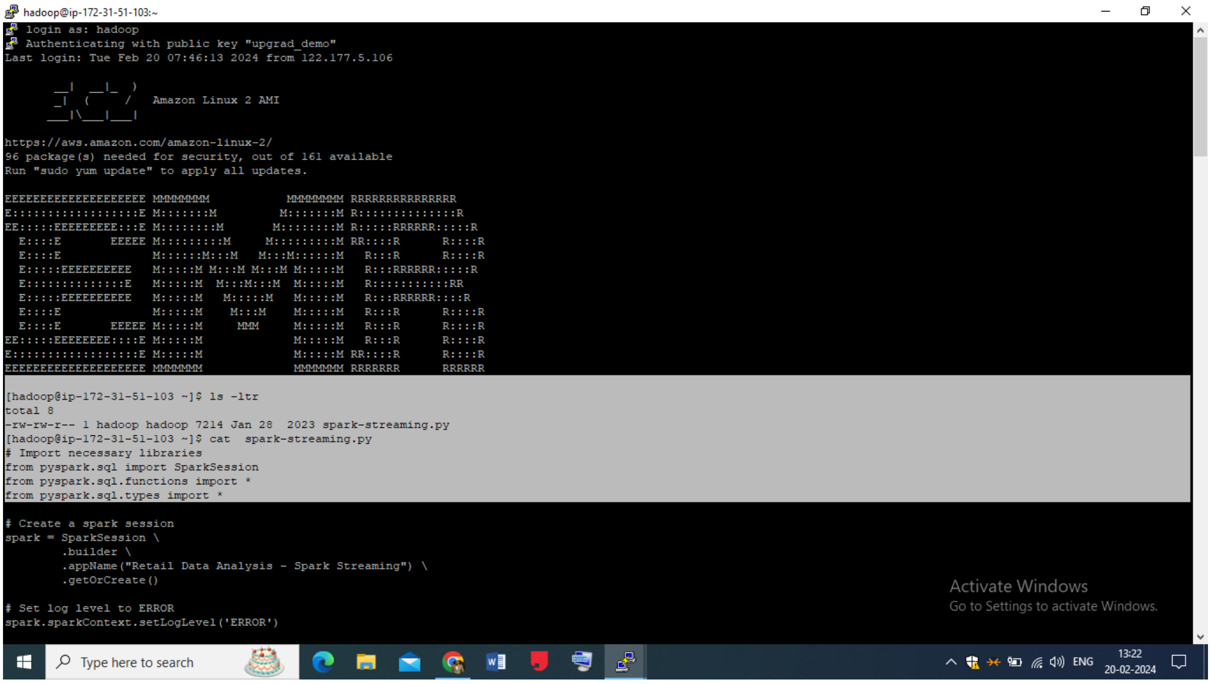
```
  --archives ARCHIVES         Comma separated list of archives to be extracted i
nto the
                              working directory of each executor.
  --principal PRINCIPAL       Principal to be used to login to KDC, while runnin
g on
                              secure HDFS.
  --keytab KEYTAB             The full path to the file that contains the keytab
 for the
                              principal specified above. This keytab will be cop
ied to
                              the node running the Application Master via the Se
cure
                              Distributed Cache, for renewing the login tickets
and the
                              delegation tokens periodically.

[hadoop@ip-172-31-60-100 ~]$ spark-submit --packages org.apache.spark:spark-sql-
kafka-0-10_2.11:2.4.5 spark-streaming.py > Console-output.txt
Ivy Default Cache set to: /home/hadoop/.ivy2/cache
The jars for the packages stored in: /home/hadoop/.ivy2/jars
:: loading settings :: url = jar:file:/usr/lib/spark/jars/ivy-2.4.0.jar!/org/apa
che/ivy/core/settings/ivysettings.xml
org.apache.spark#spark-sql-kafka-0-10_2.11 added as a dependency
:: resolving dependencies :: org.apache.spark#spark-submit-parent-c257e40a-ead7-
4c1c-b419-3187d34f6202;1.0
        confs: [default]
        found org.apache.spark#spark-sql-kafka-0-10_2.11;2.4.5 in central
        found org.apache.kafka#kafka-clients;2.0.0 in central
        found org.lz4#lz4-java;1.4.0 in central
        found org.xerial.snappy#snappy-java;1.1.7.3 in central
        found org.slf4j#slf4j-api;1.7.16 in central
        found org.spark-project.spark#unused;1.0.0 in central
:: resolution report :: resolve 412ms :: artifacts dl 11ms
        :: modules in use:
        org.apache.kafka#kafka-clients;2.0.0 from central in [default]
        org.apache.spark#spark-sql-kafka-0-10_2.11;2.4.5 from central in [defau
t]
        org.lz4#lz4-java;1.4.0 from central in [default]
        org.slf4j#slf4j-api;1.7.16 from central in [default]
        org.spark-project.spark#unused;1.0.0 from central in [default]
        org.xerial.snappy#snappy-java;1.1.7.3 from central in [default]
        ---------------------------------------------------------------------
        |                  |            modules            ||   artifacts   |
        |       conf       | number| search|dwnlded|evicted|| number|dwnlded|
```

5. Read the console output file using the command:

    ls -ltr

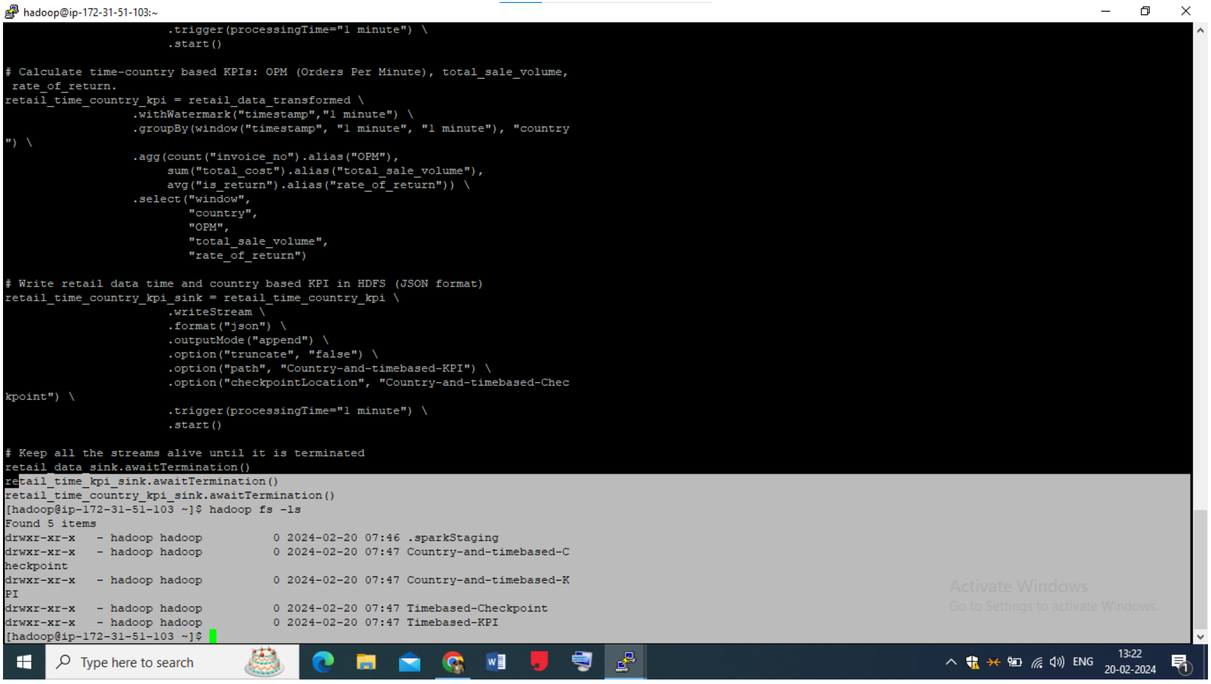
    cat Console-output.txt

    check whether we can see the transformed data as per our requirement.


hadoop@ip-172-31-51-103:~
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
13:22
20-02-2024

```
login as: hadoop
Authenticating with public key "upgrad_demo"
Last login: Tue Feb 20 07:46:13 2024 from 122.177.5.106

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
96 package(s) needed for security, out of 161 available
Run "sudo yum update" to apply all updates.

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-51-103 ~]$ ls -ltr
total 8
-rw-rw-r-- 1 hadoop hadoop 7214 Jan 28  2023 spark-streaming.py
[hadoop@ip-172-31-51-103 ~]$ cat  spark-streaming.py
# Import necessary libraries
from pyspark.sql import SparkSession
from pyspark.sql.functions import *
from pyspark.sql.types import *

# Create a spark session
spark = SparkSession \
        .builder \
        .appName("Retail Data Analysis - Spark Streaming") \
        .getOrCreate()

# Set log level to ERROR
spark.sparkContext.setLogLevel('ERROR')
```

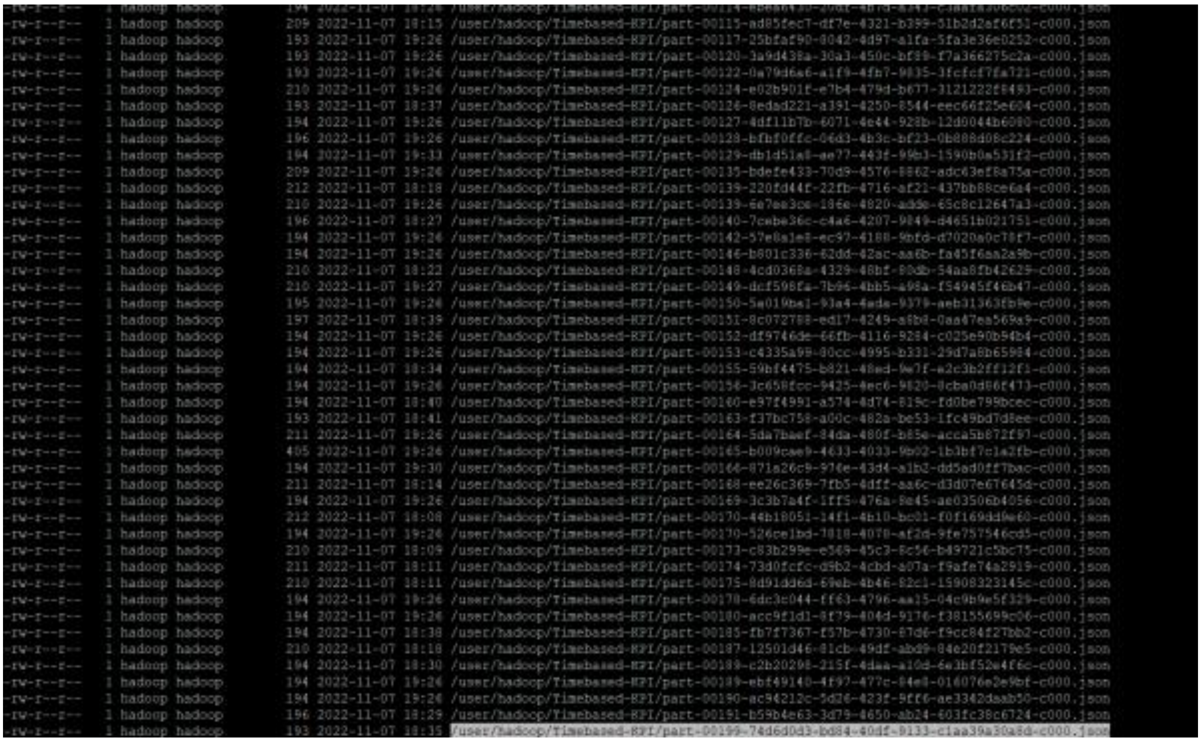
6. Now check whether all the JSON files are created for Time based and Time-andCountry based KPIs in HDFS **(Path: /user/hadoop/).**
   hadoop fs -ls


hadoop@ip-172-31-51-103:~
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
13:22
20-02-2024

```
                        .trigger(processingTime="1 minute") \
                        .start()

# Calculate time-country based KPIs: OPM (Orders Per Minute), total_sale_volume,
 rate_of_return.
retail_time_country_kpi = retail_data_transformed \
                   .withWatermark("timestamp","1 minute") \
                   .groupBy(window("timestamp", "1 minute", "1 minute"), "country
") \
                   .agg(count("invoice_no").alias("OPM"),
                        sum("total_cost").alias("total_sale_volume"),
                        avg("is_return").alias("rate_of_return")) \
                   .select("window",
                           "country",
                           "OPM",
                           "total_sale_volume",
                           "rate_of_return")

# Write retail data time and country based KPI in HDFS (JSON format)
retail_time_country_kpi_sink = retail_time_country_kpi \
                        .writeStream \
                        .format("json") \
                        .outputMode("append") \
                        .option("truncate", "false") \
                        .option("path", "Country-and-timebased-KPI") \
                        .option("checkpointLocation", "Country-and-timebased-Chec
kpoint") \
                        .trigger(processingTime="1 minute") \
                        .start()

# Keep all the streams alive until it is terminated
retail_data_sink.awaitTermination()
retail_time_kpi_sink.awaitTermination()
retail_time_country_kpi_sink.awaitTermination()
[hadoop@ip-172-31-51-103 ~]$ hadoop fs -ls
Found 5 items
drwxr-xr-x   - hadoop hadoop          0 2024-02-20 07:46 .sparkStaging
drwxr-xr-x   - hadoop hadoop          0 2024-02-20 07:47 Country-and-timebased-C
heckpoint
drwxr-xr-x   - hadoop hadoop          0 2024-02-20 07:47 Country-and-timebased-K
PI
drwxr-xr-x   - hadoop hadoop          0 2024-02-20 07:47 Timebased-Checkpoint
drwxr-xr-x   - hadoop hadoop          0 2024-02-20 07:47 Timebased-KPI
[hadoop@ip-172-31-51-103 ~]$
```

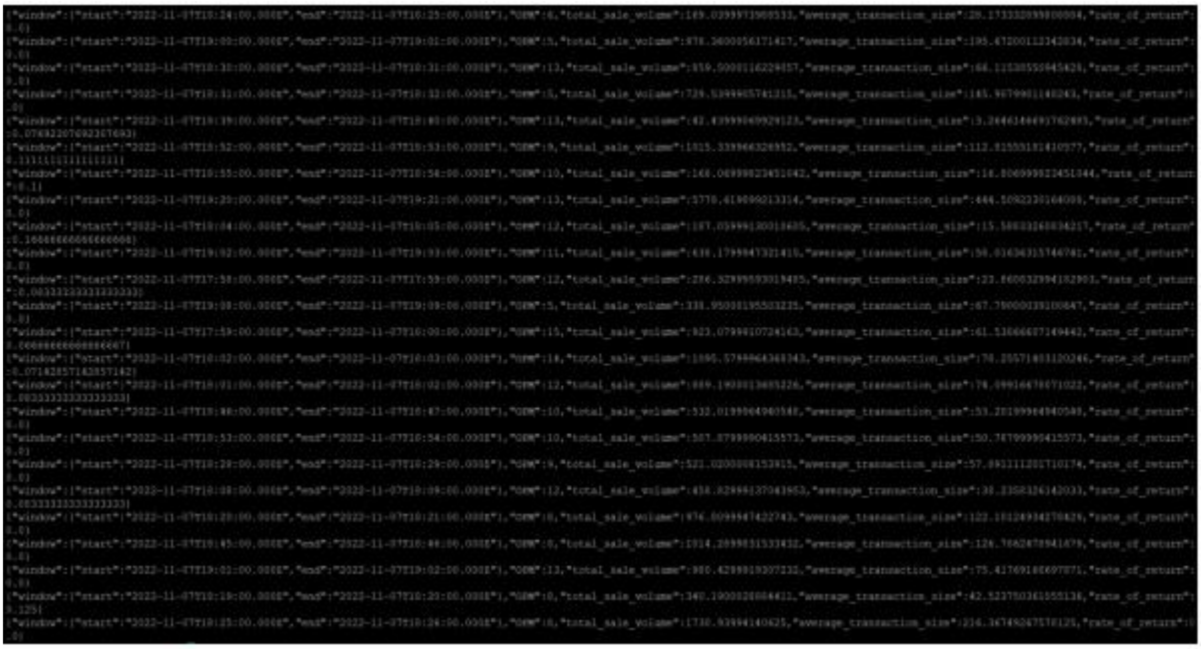
Read Timebased-KPI JSON files:

hadoop fs -ls /user/hadoop/Timebased-KPI



```
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00114-4be6a430-20df-4b7d-a343-c3aafa306b02-c000.json
-rw-r--r--   1 hadoop hadoop        209 2022-11-07 18:15 /user/hadoop/Timebased-KPI/part-00115-ad85fec7-df7e-4321-b399-51b2d2af6f51-c000.json
-rw-r--r--   1 hadoop hadoop        193 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00117-25bfaf90-0042-4d97-a1fa-5fa3e36e0252-c000.json
-rw-r--r--   1 hadoop hadoop        193 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00120-3a9d438a-30a3-450c-bf59-f7a366275c2a-c000.json
-rw-r--r--   1 hadoop hadoop        193 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00122-0a79d6a6-a1f9-4fb7-9835-3fcfcf7fa721-c000.json
-rw-r--r--   1 hadoop hadoop        210 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00124-e02b901f-e7b4-479d-b677-3121222f8493-c000.json
-rw-r--r--   1 hadoop hadoop        193 2022-11-07 18:37 /user/hadoop/Timebased-KPI/part-00126-8edad221-a391-4250-8544-eec66f25e604-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00127-4df11b7b-6071-4e44-928b-12d6044b6080-c000.json
-rw-r--r--   1 hadoop hadoop        196 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00128-bfbf0ffc-06d3-4b3c-bf23-0b888d08c224-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:33 /user/hadoop/Timebased-KPI/part-00129-db1d51a8-ae77-443f-99b3-1590b0a531f2-c000.json
-rw-r--r--   1 hadoop hadoop        209 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00135-bdefe433-70d9-4576-8862-adc63ef8a75a-c000.json
-rw-r--r--   1 hadoop hadoop        212 2022-11-07 18:18 /user/hadoop/Timebased-KPI/part-00139-220fd44f-22fb-4716-af21-437bb88ce6a4-c000.json
-rw-r--r--   1 hadoop hadoop        210 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00139-6e7ea3ce-186e-4820-adde-65c8c12647a3-c000.json
-rw-r--r--   1 hadoop hadoop        196 2022-11-07 18:27 /user/hadoop/Timebased-KPI/part-00140-7cebe36c-c4a6-4207-9849-d4651b021751-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00142-57e8a1e8-ec97-4188-9bfd-d7020a0c78f7-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00146-b801c336-62dd-42ac-aa6b-fa45f6aa2a9b-c000.json
-rw-r--r--   1 hadoop hadoop        210 2022-11-07 18:22 /user/hadoop/Timebased-KPI/part-00148-4cd0368a-4329-48bf-80db-54aa8fb42629-c000.json
-rw-r--r--   1 hadoop hadoop        210 2022-11-07 19:27 /user/hadoop/Timebased-KPI/part-00149-dcf598fa-7b96-4bb5-a98a-f54945f46b47-c000.json
-rw-r--r--   1 hadoop hadoop        195 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00150-5a019ba1-93a4-4ada-9379-aeb31363fb9e-c000.json
-rw-r--r--   1 hadoop hadoop        197 2022-11-07 18:39 /user/hadoop/Timebased-KPI/part-00151-8c072788-ed17-4249-a8b8-0aa47ea569a9-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00152-df9746de-66fb-4116-9284-c025e90b94b4-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00153-c4335a99-80cc-4995-b331-29d7a8b65984-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 18:34 /user/hadoop/Timebased-KPI/part-00155-59bf4475-b821-48ed-9e7f-e2c3b2ff12f1-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00156-3c658fcc-9425-4ec6-9820-8cba0d86f473-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 18:40 /user/hadoop/Timebased-KPI/part-00160-e97f4991-a574-4d74-819c-fd0be799bcec-c000.json
-rw-r--r--   1 hadoop hadoop        193 2022-11-07 18:41 /user/hadoop/Timebased-KPI/part-00163-f37bc758-a00c-482a-be53-1fc49bd7d8ee-c000.json
-rw-r--r--   1 hadoop hadoop        211 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00164-5da7baef-84da-480f-b85e-acca5b872f97-c000.json
-rw-r--r--   1 hadoop hadoop        405 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00165-b009cae9-4633-4033-9b02-1b3bf7c1a2fb-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:30 /user/hadoop/Timebased-KPI/part-00166-871a26c9-976e-43d4-a1b2-dd5ad0ff7bac-c000.json
-rw-r--r--   1 hadoop hadoop        211 2022-11-07 18:14 /user/hadoop/Timebased-KPI/part-00168-ee26c369-7fb5-4dff-aa6c-d3d07e67645d-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00169-3c3b7a4f-1ff5-476a-8e45-ae03506b4056-c000.json
-rw-r--r--   1 hadoop hadoop        212 2022-11-07 18:08 /user/hadoop/Timebased-KPI/part-00170-44b18051-14f1-4b10-bc01-f0f169dd9e60-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00170-526ce1bd-7818-4078-af2d-9fe757546cd5-c000.json
-rw-r--r--   1 hadoop hadoop        210 2022-11-07 18:09 /user/hadoop/Timebased-KPI/part-00173-c83b299e-e569-45c3-8c56-b49721c5bc75-c000.json
-rw-r--r--   1 hadoop hadoop        211 2022-11-07 18:11 /user/hadoop/Timebased-KPI/part-00174-73d0fcfc-d9b2-4cbd-a07a-f9afe74a2919-c000.json
-rw-r--r--   1 hadoop hadoop        210 2022-11-07 18:11 /user/hadoop/Timebased-KPI/part-00175-8d91dd6d-69eb-4b46-82c1-15908323145c-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00178-6dc3c044-ff63-4796-aa15-04c9b9e5f329-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00180-acc9f1d1-8f79-404d-9176-f38155699c06-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 18:38 /user/hadoop/Timebased-KPI/part-00185-fb7f7367-f57b-4730-87d6-f9cc84f27bb2-c000.json
-rw-r--r--   1 hadoop hadoop        210 2022-11-07 18:18 /user/hadoop/Timebased-KPI/part-00187-12501d46-81cb-49df-abd9-84e20f2179e5-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 18:30 /user/hadoop/Timebased-KPI/part-00189-c2b20298-215f-4daa-a10d-6e3bf52e4f6c-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00189-ebf49140-4f97-477c-84e8-016076e2e9bf-c000.json
-rw-r--r--   1 hadoop hadoop        194 2022-11-07 19:26 /user/hadoop/Timebased-KPI/part-00190-ac94212c-5d26-423f-9ff6-ae3342daab50-c000.json
-rw-r--r--   1 hadoop hadoop        196 2022-11-07 18:29 /user/hadoop/Timebased-KPI/part-00191-b59b4e63-3d79-4650-ab24-603fc38c6724-c000.json
-rw-r--r--   1 hadoop hadoop        193 2022-11-07 18:35 /user/hadoop/Timebased-KPI/part-00199-74d6d0d3-bd84-40df-9133-c1aa39a30a8d-c000.json
```

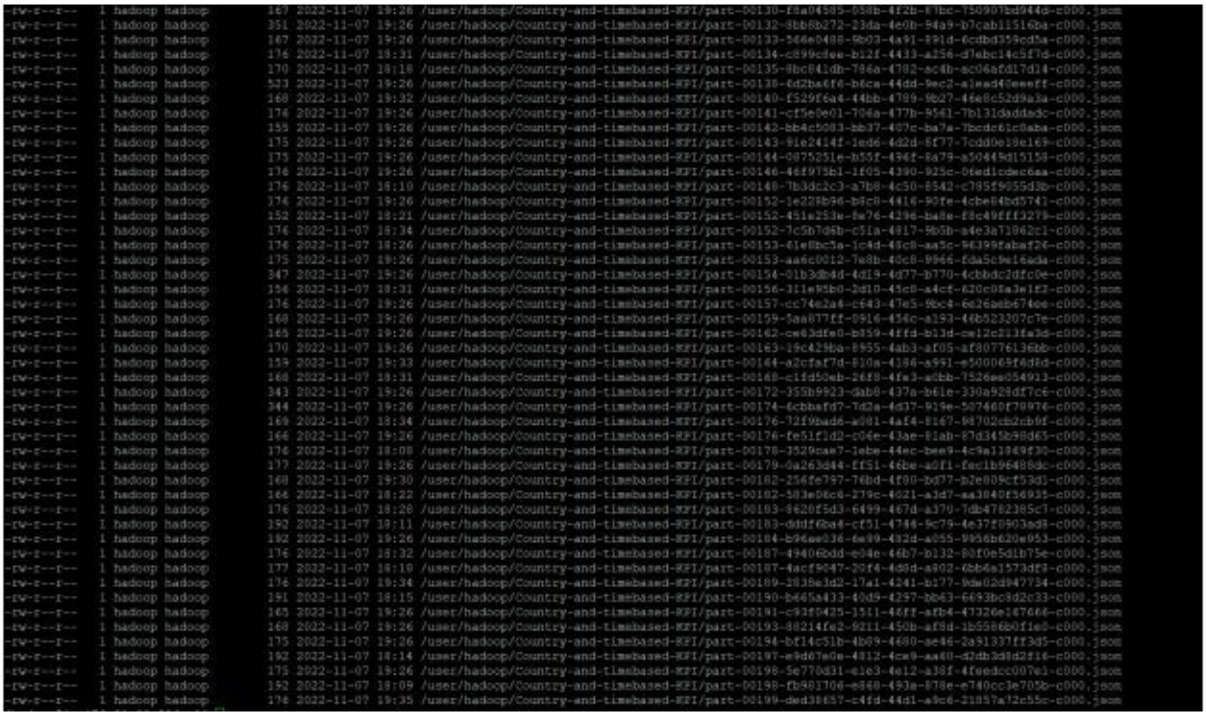
hadoop fs -cat /user/hadoop/Timebased-KPI/part*



```
{"window":{"start":"2022-11-07T18:24:00.000Z","end":"2022-11-07T18:25:00.000Z"},"OPM":6,"total_sale_volume":169.0399973988533,"average_transaction_size":28.173332899800004,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T19:00:00.000Z","end":"2022-11-07T19:01:00.000Z"},"OPM":5,"total_sale_volume":976.3600056171417,"average_transaction_size":195.47200112342834,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T18:30:00.000Z","end":"2022-11-07T18:31:00.000Z"},"OPM":13,"total_sale_volume":859.5000116229457,"average_transaction_size":66.11538555945426,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T18:31:00.000Z","end":"2022-11-07T18:32:00.000Z"},"OPM":5,"total_sale_volume":729.5399965741215,"average_transaction_size":145.9079931148243,"rate_of_return":0
.0}
{"window":{"start":"2022-11-07T18:39:00.000Z","end":"2022-11-07T18:40:00.000Z"},"OPM":13,"total_sale_volume":42.43999069929123,"average_transaction_size":3.2646144691762093,"rate_of_return"
:0.07692307692307693}
{"window":{"start":"2022-11-07T18:52:00.000Z","end":"2022-11-07T18:53:00.000Z"},"OPM":9,"total_sale_volume":1015.339966328552,"average_transaction_size":112.81555181410577,"rate_of_return":
0.1111111111111111}
{"window":{"start":"2022-11-07T18:55:00.000Z","end":"2022-11-07T18:56:00.000Z"},"OPM":10,"total_sale_volume":168.06999823451042,"average_transaction_size":16.806999823451044,"rate_of_return
":0.1}
{"window":{"start":"2022-11-07T19:20:00.000Z","end":"2022-11-07T19:21:00.000Z"},"OPM":13,"total_sale_volume":5778.419999213314,"average_transaction_size":444.5092230164095,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T18:04:00.000Z","end":"2022-11-07T18:05:00.000Z"},"OPM":12,"total_sale_volume":187.05999130018605,"average_transaction_size":15.588332608348217,"rate_of_return"
:0.16666666666666666}
{"window":{"start":"2022-11-07T19:02:00.000Z","end":"2022-11-07T19:03:00.000Z"},"OPM":11,"total_sale_volume":638.1799947321415,"average_transaction_size":58.01636315746741,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T17:58:00.000Z","end":"2022-11-07T17:59:00.000Z"},"OPM":12,"total_sale_volume":286.3299959301948,"average_transaction_size":23.86083299418290,"rate_of_return
":0.08333333333333333}
{"window":{"start":"2022-11-07T19:08:00.000Z","end":"2022-11-07T19:09:00.000Z"},"OPM":5,"total_sale_volume":338.95000195503235,"average_transaction_size":67.79000039100647,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T17:59:00.000Z","end":"2022-11-07T18:00:00.000Z"},"OPM":15,"total_sale_volume":923.0799810724163,"average_transaction_size":61.53866607149442,"rate_of_return":
0.06666666666666667}
{"window":{"start":"2022-11-07T18:02:00.000Z","end":"2022-11-07T18:03:00.000Z"},"OPM":14,"total_sale_volume":1095.5799964368343,"average_transaction_size":78.25571403120245,"rate_of_return"
:0.07142857142857142}
{"window":{"start":"2022-11-07T18:01:00.000Z","end":"2022-11-07T18:02:00.000Z"},"OPM":12,"total_sale_volume":889.1900013685226,"average_transaction_size":74.09916678071022,"rate_of_return":
0.08333333333333333}
{"window":{"start":"2022-11-07T18:46:00.000Z","end":"2022-11-07T18:47:00.000Z"},"OPM":10,"total_sale_volume":532.0199984940580,"average_transaction_size":53.20199984940580,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T18:53:00.000Z","end":"2022-11-07T18:54:00.000Z"},"OPM":10,"total_sale_volume":507.8799990415573,"average_transaction_size":50.78799990415573,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T18:28:00.000Z","end":"2022-11-07T18:29:00.000Z"},"OPM":9,"total_sale_volume":521.0200008153915,"average_transaction_size":57.89111120171017,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T18:08:00.000Z","end":"2022-11-07T18:09:00.000Z"},"OPM":12,"total_sale_volume":458.8299913704395,"average_transaction_size":38.23583261420329,"rate_of_return":
0.08333333333333333}
{"window":{"start":"2022-11-07T18:20:00.000Z","end":"2022-11-07T18:21:00.000Z"},"OPM":8,"total_sale_volume":976.8099947422743,"average_transaction_size":122.10124934278429,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T18:45:00.000Z","end":"2022-11-07T18:46:00.000Z"},"OPM":8,"total_sale_volume":1014.2099931533432,"average_transaction_size":126.77624914416790,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T19:01:00.000Z","end":"2022-11-07T19:02:00.000Z"},"OPM":13,"total_sale_volume":980.4299919307232,"average_transaction_size":75.41769168697871,"rate_of_return":
0.0}
{"window":{"start":"2022-11-07T18:19:00.000Z","end":"2022-11-07T18:20:00.000Z"},"OPM":8,"total_sale_volume":340.1900020886441,"average_transaction_size":42.52375026108051,"rate_of_return":
0.125}
{"window":{"start":"2022-11-07T18:25:00.000Z","end":"2022-11-07T18:26:00.000Z"},"OPM":8,"total_sale_volume":1730.9399414062,"average_transaction_size":216.36749267578125,"rate_of_return":0
.0}
```

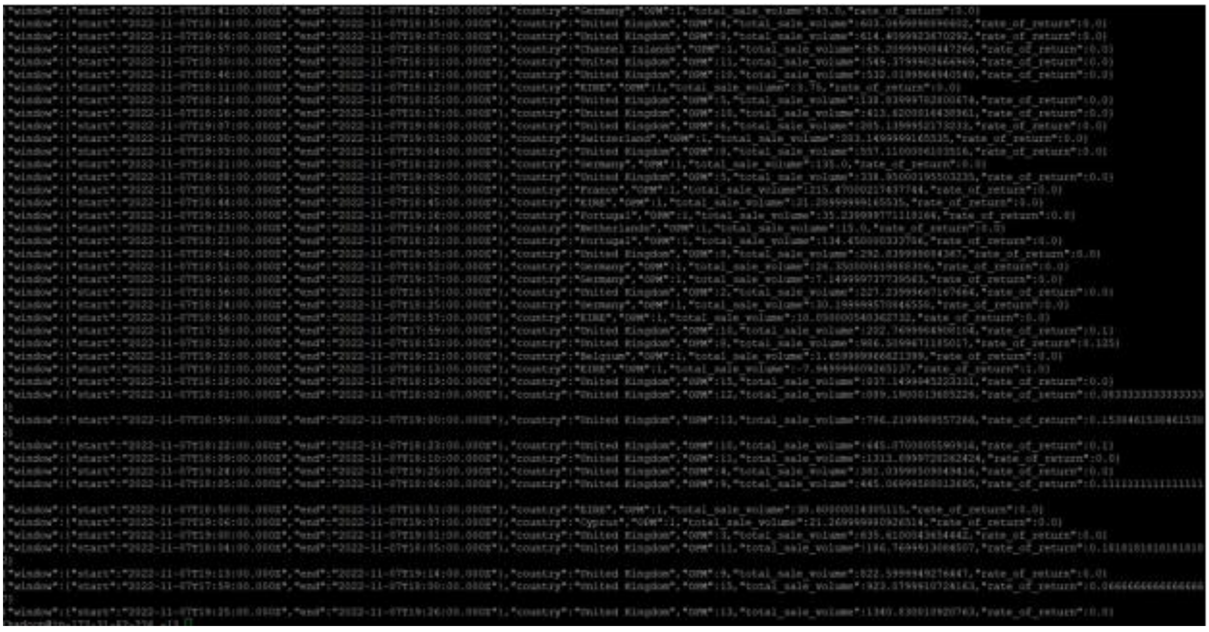
Read Country-and-timebased-KPI JSON files:

hadoop fs -ls /user/hadoop/Country-and-timebased-KPI



```
-rw-r--r--   1 hadoop hadoop        167 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00130-f8a04585-058b-4f2b-87bc-750907bd944d-c000.json
-rw-r--r--   1 hadoop hadoop        351 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00132-8bb8b272-23da-4e0b-94a9-b7cab11516ba-c000.json
-rw-r--r--   1 hadoop hadoop        167 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00133-566e0488-9b03-4a91-891d-6cdbd359cd5a-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 18:31 /user/hadoop/Country-and-timebased-KPI/part-00134-c899c6ee-b12f-4433-a256-d7ebc14c5f7d-c000.json
-rw-r--r--   1 hadoop hadoop        170 2022-11-07 18:18 /user/hadoop/Country-and-timebased-KPI/part-00135-8bc841db-786a-4782-ac4b-ac06afd17d14-c000.json
-rw-r--r--   1 hadoop hadoop        523 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00138-6d2ba6f6-b6ca-44dd-9ec2-a1ead40eeeff-c000.json
-rw-r--r--   1 hadoop hadoop        168 2022-11-07 19:32 /user/hadoop/Country-and-timebased-KPI/part-00140-f529f6a4-44bb-4789-9b27-46e8c52d9a3a-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00141-cf5e0e01-706a-477b-9561-7b131daddadc-c000.json
-rw-r--r--   1 hadoop hadoop        155 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00142-bb4c5083-bb37-407c-ba7a-7bcdc61c0aba-c000.json
-rw-r--r--   1 hadoop hadoop        175 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00143-91e2414f-1ed6-4d2d-8f77-7cdd0e18e169-c000.json
-rw-r--r--   1 hadoop hadoop        175 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00144-0875251e-b55f-496f-8a79-a50449d15158-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00146-46f975b1-1f05-4390-925c-06ed1cdec6aa-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 18:10 /user/hadoop/Country-and-timebased-KPI/part-00148-7b3dc2c3-a7b8-4c50-8542-c785f9055d3b-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00152-1e228b96-b8c8-4416-90fe-4cbe84bd5741-c000.json
-rw-r--r--   1 hadoop hadoop        152 2022-11-07 18:21 /user/hadoop/Country-and-timebased-KPI/part-00152-451e253e-8e76-4296-ba8e-f8c49fff3279-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 18:34 /user/hadoop/Country-and-timebased-KPI/part-00152-7c5b7d6b-c51a-4817-9b5b-a4e3a71862c1-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 18:26 /user/hadoop/Country-and-timebased-KPI/part-00153-61e8bc5a-1c4d-48c8-aa5c-96399fabaf26-c000.json
-rw-r--r--   1 hadoop hadoop        175 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00153-aa6c0012-7e8b-40c8-9966-fda5c9e16ada-c000.json
-rw-r--r--   1 hadoop hadoop        347 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00154-01b3db4d-4d19-4d77-b770-4cbbdc2dfc0e-c000.json
-rw-r--r--   1 hadoop hadoop        156 2022-11-07 18:31 /user/hadoop/Country-and-timebased-KPI/part-00156-311e95b0-2d10-45c8-a4cf-620c08a3e1f2-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00157-cc74e2a4-c643-47e5-9bc4-6e26aeb674ee-c000.json
-rw-r--r--   1 hadoop hadoop        168 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00159-5aa877ff-0916-456c-a193-46b523207c7e-c000.json
-rw-r--r--   1 hadoop hadoop        165 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00162-ce63dfe0-b859-4ffd-b13d-ce12c213fa3d-c000.json
-rw-r--r--   1 hadoop hadoop        170 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00163-19c429ba-8955-4ab3-af05-af80776136bb-c000.json
-rw-r--r--   1 hadoop hadoop        159 2022-11-07 19:33 /user/hadoop/Country-and-timebased-KPI/part-00164-a2cfaf7d-810a-4186-a991-e500069f6d8d-c000.json
-rw-r--r--   1 hadoop hadoop        168 2022-11-07 18:31 /user/hadoop/Country-and-timebased-KPI/part-00168-c1fd50eb-26f8-4fe3-a0bb-7526ee054913-c000.json
-rw-r--r--   1 hadoop hadoop        343 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00172-355b9923-dab8-437a-b61e-330a928df7c6-c000.json
-rw-r--r--   1 hadoop hadoop        344 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00174-6cbbafd7-7d2a-4d37-919e-507460f78976-c000.json
-rw-r--r--   1 hadoop hadoop        169 2022-11-07 18:34 /user/hadoop/Country-and-timebased-KPI/part-00176-72f9bad6-a081-4af4-8167-98702cb2cb9f-c000.json
-rw-r--r--   1 hadoop hadoop        166 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00176-fe51f1d2-c06e-43ae-81ab-87d345b98d65-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 18:08 /user/hadoop/Country-and-timebased-KPI/part-00178-3529cae7-1ebe-44ec-bee9-4c9a11849f30-c000.json
-rw-r--r--   1 hadoop hadoop        177 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00179-0a263d44-ff51-46be-a0f1-fec1b96488dc-c000.json
-rw-r--r--   1 hadoop hadoop        168 2022-11-07 19:30 /user/hadoop/Country-and-timebased-KPI/part-00182-256fe797-76bd-4f80-bd77-b2e809cf53d1-c000.json
-rw-r--r--   1 hadoop hadoop        166 2022-11-07 18:22 /user/hadoop/Country-and-timebased-KPI/part-00182-583e06c6-279c-4021-a3d7-aa3840f56935-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 18:28 /user/hadoop/Country-and-timebased-KPI/part-00183-8628f5d3-6499-467d-a370-7db4782385c7-c000.json
-rw-r--r--   1 hadoop hadoop        192 2022-11-07 18:11 /user/hadoop/Country-and-timebased-KPI/part-00183-dddf6ba4-cf51-4744-9c79-4e37f0903ad8-c000.json
-rw-r--r--   1 hadoop hadoop        192 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00184-b96ae036-6e99-482d-a055-9956b620e953-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 18:32 /user/hadoop/Country-and-timebased-KPI/part-00187-49406bdd-e04e-46b7-b132-80f0e5d1b75e-c000.json
-rw-r--r--   1 hadoop hadoop        177 2022-11-07 18:10 /user/hadoop/Country-and-timebased-KPI/part-00187-4acf9047-20f4-4d8d-a802-6bb4a1573df9-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 18:34 /user/hadoop/Country-and-timebased-KPI/part-00189-2838e3d2-17a1-4241-b177-9de02d947734-c000.json
-rw-r--r--   1 hadoop hadoop        191 2022-11-07 18:15 /user/hadoop/Country-and-timebased-KPI/part-00190-b665a433-40d9-4297-bb63-6693bc8d2c33-c000.json
-rw-r--r--   1 hadoop hadoop        165 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00191-c93f0425-1511-46ff-afb4-47326e147666-c000.json
-rw-r--r--   1 hadoop hadoop        168 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00193-88214fe2-9211-450b-af8d-1b5986b0f1e0-c000.json
-rw-r--r--   1 hadoop hadoop        175 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00194-bf14c51b-4b89-4680-ae46-2a91337ff3d5-c000.json
-rw-r--r--   1 hadoop hadoop        192 2022-11-07 18:14 /user/hadoop/Country-and-timebased-KPI/part-00197-e9d67e0e-4812-4ce9-aa40-d2db3d8d2f16-c000.json
-rw-r--r--   1 hadoop hadoop        175 2022-11-07 19:26 /user/hadoop/Country-and-timebased-KPI/part-00198-5e770d31-e1e3-4e12-a38f-4feedcc007e1-c000.json
-rw-r--r--   1 hadoop hadoop        192 2022-11-07 18:09 /user/hadoop/Country-and-timebased-KPI/part-00198-fb981706-e868-493a-878e-e740cc3e705b-c000.json
-rw-r--r--   1 hadoop hadoop        176 2022-11-07 19:35 /user/hadoop/Country-and-timebased-KPI/part-00199-ded38657-c4fd-44d1-a9c6-21857a72c55c-c000.json
```

hadoop fs -cat /user/hadoop/Country-and-timebased-KPI/part*



```
{"window":{"start":"2022-11-07T18:41:00.000Z","end":"2022-11-07T18:42:00.000Z"},"country":"Germany","OPM":1,"total_sale_volume":45.0,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:34:00.000Z","end":"2022-11-07T18:35:00.000Z"},"country":"United Kingdom","OPM":4,"total_sale_volume":403.06999999996002,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:06:00.000Z","end":"2022-11-07T19:07:00.000Z"},"country":"United Kingdom","OPM":2,"total_sale_volume":614.4099923670292,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:57:00.000Z","end":"2022-11-07T18:58:00.000Z"},"country":"Channel Islands","OPM":1,"total_sale_volume":43.28999908447266,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:50:00.000Z","end":"2022-11-07T18:51:00.000Z"},"country":"United Kingdom","OPM":11,"total_sale_volume":545.3799982666969,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:46:00.000Z","end":"2022-11-07T18:47:00.000Z"},"country":"United Kingdom","OPM":10,"total_sale_volume":532.0189968940540,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:11:00.000Z","end":"2022-11-07T18:12:00.000Z"},"country":"EIRE","OPM":1,"total_sale_volume":3.75,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:24:00.000Z","end":"2022-11-07T18:25:00.000Z"},"country":"United Kingdom","OPM":5,"total_sale_volume":138.03999782800674,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:16:00.000Z","end":"2022-11-07T18:17:00.000Z"},"country":"United Kingdom","OPM":10,"total_sale_volume":413.8200814438961,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:07:00.000Z","end":"2022-11-07T19:08:00.000Z"},"country":"United Kingdom","OPM":6,"total_sale_volume":205.1399952173233,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:00:00.000Z","end":"2022-11-07T19:01:00.000Z"},"country":"Switzerland","OPM":1,"total_sale_volume":233.1499999165535,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:03:00.000Z","end":"2022-11-07T19:04:00.000Z"},"country":"United Kingdom","OPM":8,"total_sale_volume":557.1100096103516,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:21:00.000Z","end":"2022-11-07T18:22:00.000Z"},"country":"Germany","OPM":1,"total_sale_volume":135.0,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:08:00.000Z","end":"2022-11-07T19:09:00.000Z"},"country":"United Kingdom","OPM":5,"total_sale_volume":338.95000195503235,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:51:00.000Z","end":"2022-11-07T18:52:00.000Z"},"country":"France","OPM":1,"total_sale_volume":215.47000217437744,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:44:00.000Z","end":"2022-11-07T18:45:00.000Z"},"country":"EIRE","OPM":1,"total_sale_volume":21.28999999165535,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:15:00.000Z","end":"2022-11-07T19:16:00.000Z"},"country":"Portugal","OPM":1,"total_sale_volume":35.23999977111816,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:23:00.000Z","end":"2022-11-07T19:24:00.000Z"},"country":"Netherlands","OPM":1,"total_sale_volume":15.0,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:21:00.000Z","end":"2022-11-07T18:22:00.000Z"},"country":"Portugal","OPM":1,"total_sale_volume":134.4500003337786,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:04:00.000Z","end":"2022-11-07T19:05:00.000Z"},"country":"United Kingdom","OPM":8,"total_sale_volume":292.0399990084367,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:51:00.000Z","end":"2022-11-07T18:52:00.000Z"},"country":"Germany","OPM":1,"total_sale_volume":26.350000619888306,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:16:00.000Z","end":"2022-11-07T19:17:00.000Z"},"country":"Germany","OPM":1,"total_sale_volume":7.149999737739563,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:56:00.000Z","end":"2022-11-07T18:57:00.000Z"},"country":"United Kingdom","OPM":2,"total_sale_volume":227.2399966716766,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:24:00.000Z","end":"2022-11-07T18:25:00.000Z"},"country":"Germany","OPM":1,"total_sale_volume":30.199999570846558,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:56:00.000Z","end":"2022-11-07T18:57:00.000Z"},"country":"EIRE","OPM":1,"total_sale_volume":10.080000540362732,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T17:58:00.000Z","end":"2022-11-07T17:59:00.000Z"},"country":"United Kingdom","OPM":10,"total_sale_volume":222.76999968900104,"rate_of_return":0.1}
{"window":{"start":"2022-11-07T18:52:00.000Z","end":"2022-11-07T18:53:00.000Z"},"country":"United Kingdom","OPM":8,"total_sale_volume":986.5899967118501,"rate_of_return":0.125}
{"window":{"start":"2022-11-07T19:20:00.000Z","end":"2022-11-07T19:21:00.000Z"},"country":"Belgium","OPM":1,"total_sale_volume":1.6599999666213989,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:12:00.000Z","end":"2022-11-07T18:13:00.000Z"},"country":"EIRE","OPM":1,"total_sale_volume":-7.949999809265137,"rate_of_return":1.0}
{"window":{"start":"2022-11-07T18:18:00.000Z","end":"2022-11-07T18:19:00.000Z"},"country":"United Kingdom","OPM":15,"total_sale_volume":837.1499943222331,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:01:00.000Z","end":"2022-11-07T18:02:00.000Z"},"country":"United Kingdom","OPM":12,"total_sale_volume":899.1900013685226,"rate_of_return":0.08333333333333333
3}
{"window":{"start":"2022-11-07T18:59:00.000Z","end":"2022-11-07T19:00:00.000Z"},"country":"United Kingdom","OPM":13,"total_sale_volume":786.2199998557286,"rate_of_return":0.15384615384615385
5}
{"window":{"start":"2022-11-07T18:22:00.000Z","end":"2022-11-07T18:23:00.000Z"},"country":"United Kingdom","OPM":10,"total_sale_volume":645.0700805590916,"rate_of_return":0.1}
{"window":{"start":"2022-11-07T18:09:00.000Z","end":"2022-11-07T18:10:00.000Z"},"country":"United Kingdom","OPM":11,"total_sale_volume":1313.0999728282424,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:24:00.000Z","end":"2022-11-07T19:25:00.000Z"},"country":"United Kingdom","OPM":4,"total_sale_volume":381.0399930984341,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:05:00.000Z","end":"2022-11-07T18:06:00.000Z"},"country":"United Kingdom","OPM":9,"total_sale_volume":445.0699958801369,"rate_of_return":0.1111111111111111
1}
{"window":{"start":"2022-11-07T18:50:00.000Z","end":"2022-11-07T18:51:00.000Z"},"country":"EIRE","OPM":1,"total_sale_volume":30.60000014305115,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:06:00.000Z","end":"2022-11-07T19:07:00.000Z"},"country":"Cyprus","OPM":1,"total_sale_volume":21.269999980926514,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T19:00:00.000Z","end":"2022-11-07T19:01:00.000Z"},"country":"United Kingdom","OPM":3,"total_sale_volume":635.4100043034442,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T18:04:00.000Z","end":"2022-11-07T18:05:00.000Z"},"country":"United Kingdom","OPM":11,"total_sale_volume":186.76999130008507,"rate_of_return":0.18181818181818182
2}
{"window":{"start":"2022-11-07T19:13:00.000Z","end":"2022-11-07T19:14:00.000Z"},"country":"United Kingdom","OPM":9,"total_sale_volume":822.5999949276447,"rate_of_return":0.0}
{"window":{"start":"2022-11-07T17:59:00.000Z","end":"2022-11-07T18:00:00.000Z"},"country":"United Kingdom","OPM":15,"total_sale_volume":923.3799910724163,"rate_of_return":0.06666666666666667
7}
{"window":{"start":"2022-11-07T19:25:00.000Z","end":"2022-11-07T19:26:00.000Z"},"country":"United Kingdom","OPM":13,"total_sale_volume":1340.8300108920763,"rate_of_return":0.0}
[hadoop@ip-172-31-42-236 ~]$
```

Note : Post completion of all the above steps, terminate the EMR instance from AWS console.