Code Logic - Retail Data Analysis

In this case study, we gone through with a real-world use case from the retail sector. Data from a centralized Kafka server in real-time are streamed and processed to calculate various KPIs or key performance indicators.

1. Various SQL functions were imported from pyspark.SQL.functions module. The functions include window, udf etc.

2. Various SQL types were imported from pyspark.SQL.types module. The types include StringType, ArrayType, TimestampType, IntegerType, DoubleType etc.

3. SparkSession was imported from pyspark.SQL module

4. Initialized the spark session using

```
spark = SparkSession |
.builder |
.appName("KafkaRead") |
.getOrCreate()
```

5. Streamed the data from kafka producer using

```
orderRaw = spark |
.readStream |
.format("kafka") |
.option("kafka.bootstrap.servers","ec2-18-211-252-152.compute-1.amazonaws.com:9092") |
.option("subscribe","real-time-project") |
.load()
```

   From Bootstrap Server - 18.211.252.152, Port – 9092, Topic - real-time-project

6. Schema is defined using

```
jsonSchema = StructType() |
.add("invoice_no", StringType()) |
.add("country", StringType()) |
.add("timestamp", TimestampType()) |
.add("type", StringType()) |
.add("items", ArrayType(StructType([
StructField("SKU", StringType()),
StructField("title", StringType()),
StructField("unit_price", DoubleType()),
StructField("quantity", IntegerType())
])))
```

7. Python function is written to compute total cost of an order using
   $Total\ cost = \Sigma(quantity * unitprice)$

8. The above function is transformed into udf (user defined function) using
   add_total_cost = udf(get_total_cost, DoubleType())

9. UDF is written to find total items in an order.
   add_total_count = udf(get_total_item, IntegerType())

10. UDF is written to find if the type of transaction is order
    add_is_order_flag = udf(get_is_order, IntegerType())

11. UDF is written to find if the type of transaction is return.
    add_is_return_flag = udf(get_is_return, IntegerType())

12. Selected data ("invoice_no", "country", "timestamp", "Total_Items", "Total_Cost", "is_order", "is_return") is written to the console.

13. Time based KPI (“Window”, ”OPM”, ”Total Sales Volume”, ”Average rate of return”, “Average Transaction Size”) is calculated using tumbling window function for every 1 minute

14. Time and country based KPI ((“Window”, ”Country” ,”OPM”, ”Total Sales Volume”, ”Average rate of return”) is calculated using tumbling window function for every 1 minute

15. The Computed KPI were written to files and stored on HDFS in json form

16. The streaming process is manually killed after 10 mins.

List of Commands used and Output

1. ********* Setting up kafka version ********

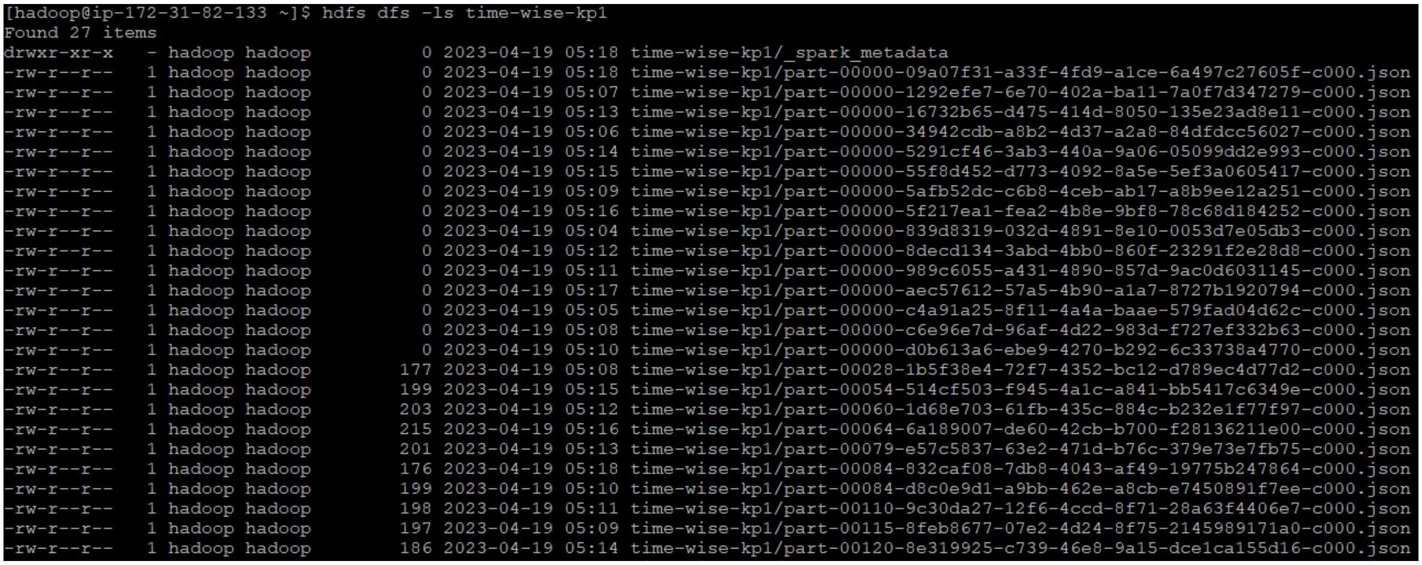
```
[hadoop@ip-172-31-85-155 ~]$ export SPARK_KAFKA_VERSION=0.10
```

2. ********* Spark submit execution to read data from Kafka Server *********

```
[hadoop@ip-172-31-85-155 ~]$ spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.5 spark-streaming.py > consoleoutput.txt
```

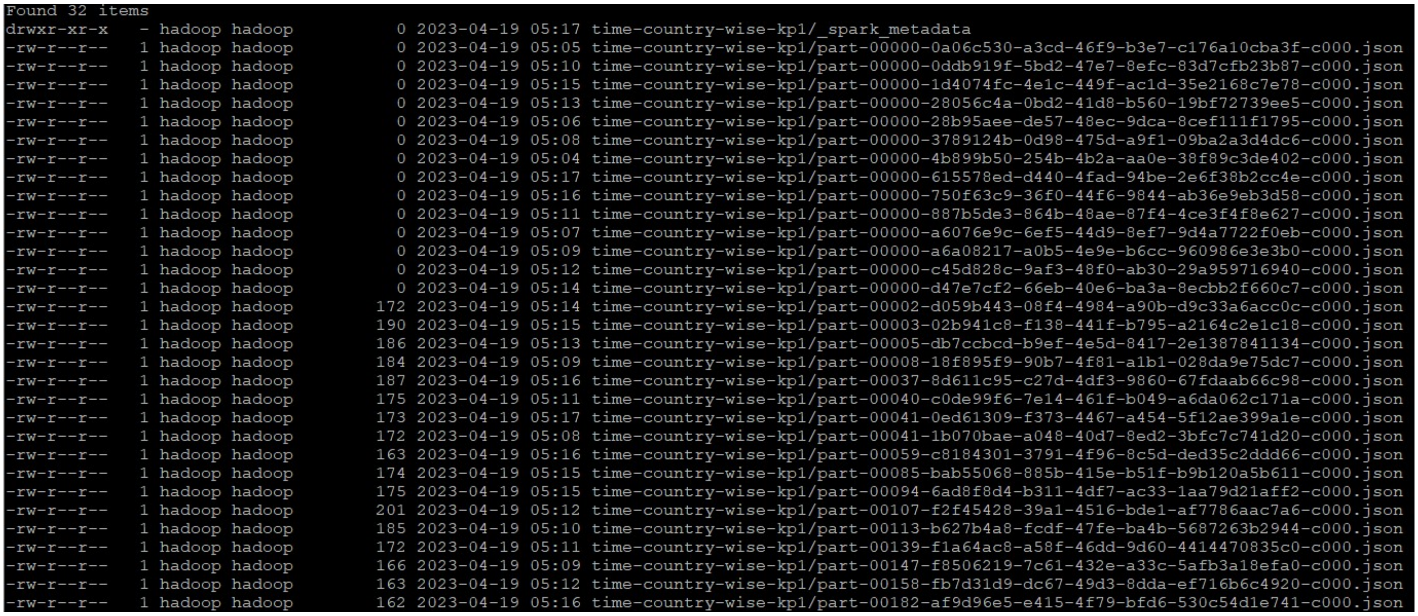
3. ********** List output json files generated by Spark-submit ***********

```
4. [hadoop@ip-172-31-85-155 ~]$hdfsdfs -ls time-wise-kp1
```



```
[hadoop@ip-172-31-82-133 ~]$ hdfs dfs -ls time-wise-kp1
Found 27 items
drwxr-xr-x   - hadoop hadoop          0 2023-04-19 05:18 time-wise-kp1/_spark_metadata
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:18 time-wise-kp1/part-00000-09a07f31-a33f-4fd9-a1ce-6a497c27605f-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:07 time-wise-kp1/part-00000-1292efe7-6e70-402a-ba11-7a0f7d347279-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:13 time-wise-kp1/part-00000-16732b65-d475-414d-8050-135e23ad8e11-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:06 time-wise-kp1/part-00000-34942cdb-a8b2-4d37-a2a8-84dfdcc56027-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:14 time-wise-kp1/part-00000-5291cf46-3ab3-440a-9a06-05099dd2e993-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:15 time-wise-kp1/part-00000-55f8d452-d773-4092-8a5e-5ef3a0605417-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:09 time-wise-kp1/part-00000-5afb52dc-c6b8-4ceb-ab17-a8b9ee12a251-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:16 time-wise-kp1/part-00000-5f217ea1-fea2-4b8e-9bf8-78c68d184252-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:04 time-wise-kp1/part-00000-839d8319-032d-4891-8e10-0053d7e05db3-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:12 time-wise-kp1/part-00000-8decd134-3abd-4bb0-860f-23291f2e28d8-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:11 time-wise-kp1/part-00000-989c6055-a431-4890-857d-9ac0d6031145-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:17 time-wise-kp1/part-00000-aec57612-57a5-4b90-a1a7-8727b1920794-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:05 time-wise-kp1/part-00000-c4a91a25-8f11-4a4a-baae-579fad04d62c-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:08 time-wise-kp1/part-00000-c6e96e7d-96af-4d22-983d-f727ef332b63-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:10 time-wise-kp1/part-00000-d0b613a6-ebe9-4270-b292-6c33738a4770-c000.json
-rw-r--r--   1 hadoop hadoop        177 2023-04-19 05:08 time-wise-kp1/part-00028-1b5f38e4-72f7-4352-bc12-d789ec4d77d2-c000.json
-rw-r--r--   1 hadoop hadoop        199 2023-04-19 05:15 time-wise-kp1/part-00054-514cf503-f945-4a1c-a841-bb5417c6349e-c000.json
-rw-r--r--   1 hadoop hadoop        203 2023-04-19 05:12 time-wise-kp1/part-00060-1d68e703-61fb-435c-884c-b232e1f77f97-c000.json
-rw-r--r--   1 hadoop hadoop        215 2023-04-19 05:16 time-wise-kp1/part-00064-6a189007-de60-42cb-b700-f28136211e00-c000.json
-rw-r--r--   1 hadoop hadoop        201 2023-04-19 05:13 time-wise-kp1/part-00079-e57c5837-63e2-471d-b76c-379e73e7fb75-c000.json
-rw-r--r--   1 hadoop hadoop        176 2023-04-19 05:18 time-wise-kp1/part-00084-832caf08-7db8-4043-af49-19775b247864-c000.json
-rw-r--r--   1 hadoop hadoop        199 2023-04-19 05:10 time-wise-kp1/part-00084-d8c0e9d1-a9bb-462e-a8cb-e7450891f7ee-c000.json
-rw-r--r--   1 hadoop hadoop        198 2023-04-19 05:11 time-wise-kp1/part-00110-9c30da27-12f6-4ccd-8f71-28a63f4406e7-c000.json
-rw-r--r--   1 hadoop hadoop        197 2023-04-19 05:09 time-wise-kp1/part-00115-8feb8677-07e2-4d24-8f75-2145989171a0-c000.json
-rw-r--r--   1 hadoop hadoop        186 2023-04-19 05:14 time-wise-kp1/part-00120-8e319925-c739-46e8-9a15-dce1ca155d16-c000.json
```

5. [hadoop@ip-172-31-85-155 ~]$hdfsdfs -ls time-country-wise-kp1



```
Found 32 items
drwxr-xr-x   - hadoop hadoop          0 2023-04-19 05:17 time-country-wise-kp1/_spark_metadata
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:05 time-country-wise-kp1/part-00000-0a06c530-a3cd-46f9-b3e7-c176a10cba3f-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:10 time-country-wise-kp1/part-00000-0ddb919f-5bd2-47e7-8efc-83d7cfb23b87-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:15 time-country-wise-kp1/part-00000-1d4074fc-4e1c-449f-ac1d-35e2168c7e78-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:13 time-country-wise-kp1/part-00000-28056c4a-0bd2-41d8-b560-19bf72739ee5-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:06 time-country-wise-kp1/part-00000-28b95aee-de57-48ec-9dca-8cef111f1795-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:08 time-country-wise-kp1/part-00000-3789124b-0d98-475d-a9f1-09ba2a3d4dc6-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:04 time-country-wise-kp1/part-00000-4b899b50-254b-4b2a-aa0e-38f89c3de402-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:17 time-country-wise-kp1/part-00000-615578ed-d440-4fad-94be-2e6f38b2cc4e-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:16 time-country-wise-kp1/part-00000-750f63c9-36f0-44f6-9844-ab36e9eb3d58-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:11 time-country-wise-kp1/part-00000-887b5de3-864b-48ae-87f4-4ce3f4f8e627-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:07 time-country-wise-kp1/part-00000-a6076e9c-6ef5-44d9-8ef7-9d4a7722f0eb-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:09 time-country-wise-kp1/part-00000-a6a08217-a0b5-4e9e-b6cc-960986e3e3b0-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:12 time-country-wise-kp1/part-00000-c45d828c-9af3-48f0-ab30-29a959716940-c000.json
-rw-r--r--   1 hadoop hadoop          0 2023-04-19 05:14 time-country-wise-kp1/part-00000-d47e7cf2-66eb-40e6-ba3a-8ecbb2f660c7-c000.json
-rw-r--r--   1 hadoop hadoop        172 2023-04-19 05:14 time-country-wise-kp1/part-00002-d059b443-08f4-4984-a90b-d9c33a6acc0c-c000.json
-rw-r--r--   1 hadoop hadoop        190 2023-04-19 05:15 time-country-wise-kp1/part-00003-02b941c8-f138-441f-b795-a2164c2e1c18-c000.json
-rw-r--r--   1 hadoop hadoop        186 2023-04-19 05:13 time-country-wise-kp1/part-00005-db7ccbcd-b9ef-4e5d-8417-2e1387841134-c000.json
-rw-r--r--   1 hadoop hadoop        184 2023-04-19 05:09 time-country-wise-kp1/part-00008-18f895f9-90b7-4f81-a1b1-028da9e75dc7-c000.json
-rw-r--r--   1 hadoop hadoop        187 2023-04-19 05:16 time-country-wise-kp1/part-00037-8d611c95-c27d-4df3-9860-67fdaab66c98-c000.json
-rw-r--r--   1 hadoop hadoop        175 2023-04-19 05:11 time-country-wise-kp1/part-00040-c0de99f6-7e14-461f-b049-a6da062c171a-c000.json
-rw-r--r--   1 hadoop hadoop        173 2023-04-19 05:17 time-country-wise-kp1/part-00041-0ed61309-f373-4467-a454-5f12ae399a1e-c000.json
-rw-r--r--   1 hadoop hadoop        172 2023-04-19 05:08 time-country-wise-kp1/part-00041-1b070bae-a048-40d7-8ed2-3bfc7c741d20-c000.json
-rw-r--r--   1 hadoop hadoop        163 2023-04-19 05:16 time-country-wise-kp1/part-00059-c8184301-3791-4f96-8c5d-ded35c2ddd66-c000.json
-rw-r--r--   1 hadoop hadoop        174 2023-04-19 05:15 time-country-wise-kp1/part-00085-bab55068-885b-415e-b51f-b9b120a5b611-c000.json
-rw-r--r--   1 hadoop hadoop        175 2023-04-19 05:15 time-country-wise-kp1/part-00094-6ad8f8d4-b311-4df7-ac33-1aa79d21aff2-c000.json
-rw-r--r--   1 hadoop hadoop        201 2023-04-19 05:12 time-country-wise-kp1/part-00107-f2f45428-39a1-4516-bde1-af7786aac7a6-c000.json
-rw-r--r--   1 hadoop hadoop        185 2023-04-19 05:10 time-country-wise-kp1/part-00113-b627b4a8-fcdf-47fe-ba4b-5687263b2944-c000.json
-rw-r--r--   1 hadoop hadoop        172 2023-04-19 05:11 time-country-wise-kp1/part-00139-f1a64ac8-a58f-46dd-9d60-4414470835c0-c000.json
-rw-r--r--   1 hadoop hadoop        166 2023-04-19 05:09 time-country-wise-kp1/part-00147-f8506219-7c61-432e-a33c-5afb3a18efa0-c000.json
-rw-r--r--   1 hadoop hadoop        163 2023-04-19 05:12 time-country-wise-kp1/part-00158-fb7d31d9-dc67-49d3-8dda-ef716b6c4920-c000.json
-rw-r--r--   1 hadoop hadoop        162 2023-04-19 05:16 time-country-wise-kp1/part-00182-af9d96e5-e415-4f79-bfd6-530c54d1e741-c000.json
```

6. ********* Copying kafka files from hdfsfilesystem into emr file system ********

[hadoop@ip-172-31-85-155 ~]$hdfsdfs -copyToLocaltime-country-wise-kp1.
[hadoop@ip-172-31-85-155 ~]$hdfsdfs -copyToLocaltime-wise-kp1.