Distributed Database Design For Supermarkets

check-out System

Introduction And Motivation:

In today's Growing World the demand for grocery shopping increasing very quickly so to facilitate the increase in number of people and to process their check out process smoothly we need to have high scalable system with high availability by which the end user would not feel they are waiting too long in the check-out lines which in intern helps in customer retention and customer salinification with store

To implement the high scalable system, we have used a data set form git which contains supermarket check-out system data, which we have used as inputs/transaction to our databases.

System Design and Architecture

System Requirements:

1) Once customer/user added the product to cart in the store/online we need to calculate each product prize and quantity based on inventory and process the payment and update the database.

Scale of The System:

To estimate the scale of the system we need to perform some assumptions of user traffic and what are their day-to-day interaction with the system, and is the ops on our system

Assumptions:

1) Assuming a xyzzy supermarket franchise run 10000 retail stores and each store as 20 check-out machine operating, so the total systems operating daily would be around 10000*20=200000 and we way except around 100000 active user online who is using the same system so the total amount of system interacting with our database should come approx. 200000+100000=300000 lakh users or systems
2) Assuming each user perform one write operation in databases when they check out
3) To estimate the ops we can assume at a particular time period let's say 10 min we may have 30000 transactions in peak time so it comes out be 3000 per min so if done it to second it comes out to be 3000/60 =50 so in peak time our system may expect 50 ops/sec . so we need to build a reliable system to operate 50 ops/sec traffic.

Functional Requirement's.

1)user need to add the products to cart and remove the product

2)They should be able to check-out the products that are added to cart

3)performing analytics on each user transactions (Here is is already present in data set)

Non-Functional Requirements:

1) System Should support 50 ops/sec
2) Implementing CAP Theron Availability and partition tolerance should be high but consistency can be ignored

Data set Discussion:

To mimic the above system we have found a data set form GitHub, where once the user add the thing to cart and check and enter the payment we have all those details in the dataset

supermarket_Sales (1)

Invoice ID	Branch	City	Customer type	Gender	Product line	Unit price	Quantity	Tax 5%	Total	Date	Time	Payment	Cost of goods sold	Gross margin percentage	Gross income	Customer stratification rating
750-67-8428	A	Yangon	Member	Female	Health and beauty	74.69	7	26.1415	548.9715	1/5/2019	13:08	Ewallet	522.83	4.761904762	26.1415	9.1
226-31-3081	C	Naypyitaw	Normal	Female	Electronic accessories	15.28	5	3.82	80.22	3/8/2019	10:29	Cash	76.4	4.761904762	3.82	9.6
631-41-3108	A	Yangon	Normal	Male	Home and lifestyle	46.33	7	16.2155	340.5255	3/3/2019	13:23	Credit card	324.31	4.761904762	16.2155	7.4
123-19-1176	A	Yangon	Member	Male	Health and beauty	58.22	8	23.288	489.048	1/27/2019	20:33	Ewallet	465.76	4.761904762	23.288	8.4
373-73-7910	A	Yangon	Normal	Male	Sports and travel	86.31	7	30.2085	634.3785	2/8/2019	10:37	Ewallet	604.17	4.761904762	30.2085	5.3
699-14-3026	C	Naypyitaw	Normal	Male	Electronic accessories	85.39	7	29.8865	627.6165	3/25/2019	18:30	Ewallet	597.73	4.761904762	29.8865	4.1
355-53-5943	A	Yangon	Member	Female	Electronic accessories	68.84	6	20.652	433.692	2/25/2019	14:36	Ewallet	413.04	4.761904762	20.652	5.8
315-22-5665	C	Naypyitaw	Normal	Female	Home and lifestyle	73.56	10	36.78	772.38	2/24/2019	11:38	Ewallet	735.6	4.761904762	36.78	8
665-32-9167	A	Yangon	Member	Female	Health and beauty	36.26	2	3.626	76.146	1/10/2019	17:15	Credit card	72.52	4.761904762	3.626	7.2
692-92-5582	B	Mandalay	Member	Female	Food and beverages	54.84	3	8.226	172.746	2/20/2019	13:27	Credit card	164.52	4.761904762	8.226	5.9
351-62-0822	B	Mandalay	Member	Female	Fashion accessories	14.48	4	2.896	60.816	2/6/2019	18:07	Ewallet	57.92	4.761904762	2.896	4.5
529-56-3974	B	Mandalay	Member	Male	Electronic accessories	25.51	4	5.102	107.142	3/9/2019	17:03	Cash	102.04	4.761904762	5.102	6.8
365-64-0515	A	Yangon	Normal	Female	Electronic accessories	46.95	5	11.7375	246.4875	2/12/2019	10:25	Ewallet	234.75	4.761904762	11.7375	7.1
252-56-2699	A	Yangon	Normal	Male	Food and beverages	43.19	10	21.595	453.495	2/7/2019	16:48	Ewallet	431.9	4.761904762	21.595	8.2
829-34-3910	A	Yangon	Normal	Female	Health and beauty	71.38	10	35.69	749.49	3/29/2019	19:21	Cash	713.8	4.761904762	35.69	5.7
299-46-1805	B	Mandalay	Member	Female	Sports and travel	93.72	6	28.116	590.436	1/15/2019	16:19	Cash	562.32	4.761904762	28.116	4.5
656-95-9349	A	Yangon	Member	Female	Health and beauty	68.93	7	24.1255	506.6355	3/11/2019	11:03	Credit card	482.51	4.761904762	24.1255	4.6
765-26-6951	A	Yangon	Normal	Male	Sports and travel	72.61	6	21.783	457.443	1/1/2019	10:39	Credit card	435.66	4.761904762	21.783	6.9
329-62-1586	A	Yangon	Normal	Male	Food and beverages	54.67	3	8.2005	172.2105	1/21/2019	18:00	Credit card	164.01	4.761904762	8.2005	8.6
319-50-3348	B	Mandalay	Normal	Female	Home and lifestyle	40.3	2	4.03	84.63	3/11/2019	15:30	Ewallet	80.6	4.761904762	4.03	4.4
300-71-4605	C	Naypyitaw	Member	Male	Electronic accessories	86.04	5	21.51	451.71	2/25/2019	11:24	Ewallet	430.2	4.761904762	21.51	4.8
371-85-5789	B	Mandalay	Normal	Male	Health and beauty	87.98	3	13.197	277.137	3/5/2019	10:40	Ewallet	263.94	4.761904762	13.197	5.1
273-16-6619	B	Mandalay	Normal	Male	Home and lifestyle	33.2	2	3.32	69.72	3/15/2019	12:20	Credit card	66.4	4.761904762	3.32	4.4

Dataset consists of 17 columns of the user interaction, and we will be treated each row as one data base transaction and design and architect the system according to that

Architecture Design:

To design the above system and support the scale of the customers we need to handle few different things,

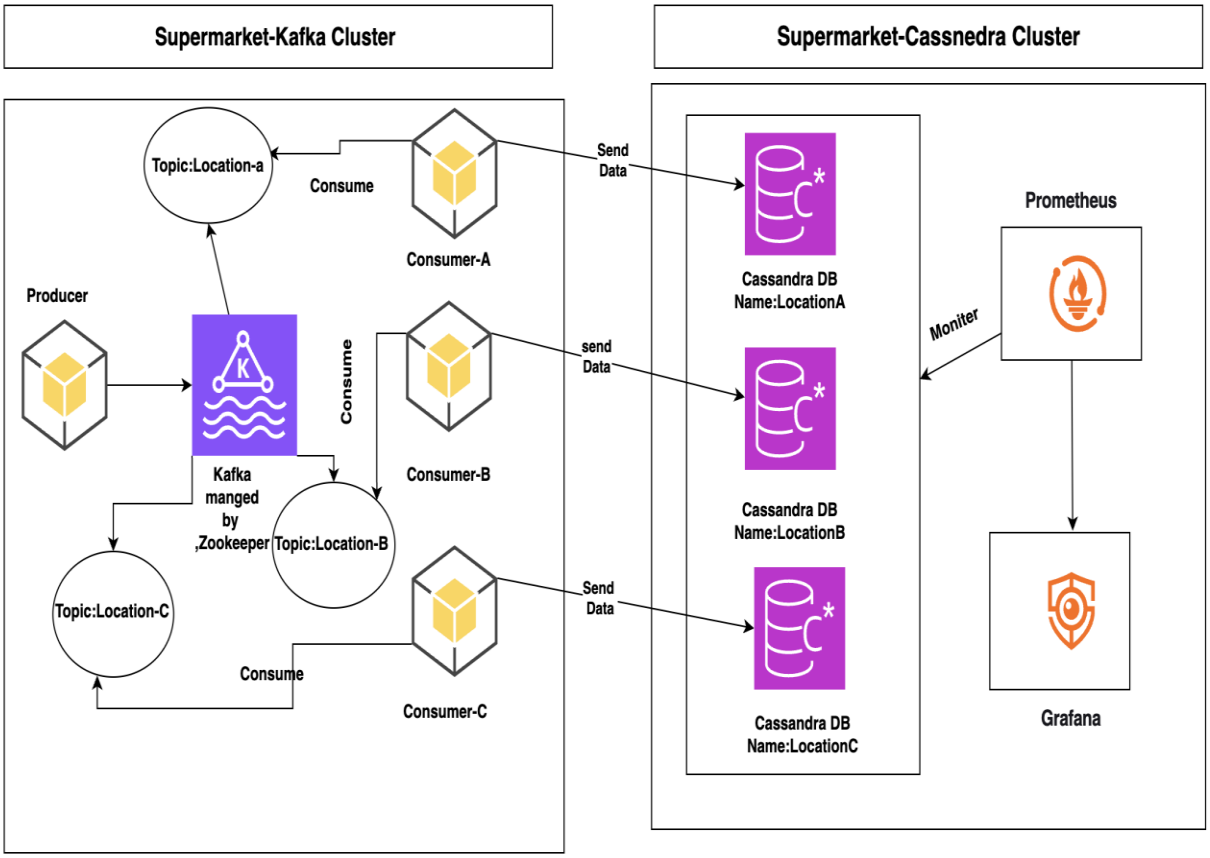
1)as we are experiencing 50 ops from our assumptions we need a system that accept these many request per sec and be fault tolerant so to achieve this we will

be using Kafka Streaming's which is known for handling high traffic and then distribute to different nodes and be fault tolerant in case of any failure

2)As the supermarket retail store in located in multiple regions requesting or sending data from single data center would take time which increase in latency ,if suppose we place our dc in south-east , but we have supermarket retail store in north so to get or post the data to this data center come with cost called latency to avoid this me need to maintain the data center near to location so in our data set we have city and few city grouped to one Brach where we can place the data center so every operation take place with less latency .and most the time our system is performing write operation in data base according to our requirement and as our system is having 50 ops we need to fault tolerant so we can ensure high availability and scale up easily to support the distributed data center and connect them together and which has high write throughput and which scale horizontally according to our requirement the best database the satisfy these requirement is Cassandra where it as a distributed nature and communicate with each other data center over a Cassandra network ,and the replication across the nodes help Cassandra to be fault tolerant and if the ops is gradually increasing we can spin up as many as Cassandra and join to the network which helps in achieve Scalability

[**Note:** The below Architecture is limited to local env as we are doing it in single, the same architecture diagram can support multiple system or nodes interlinking and the scale of the system will look completely different then in one system.]

High Level Design of super market retail store check out system

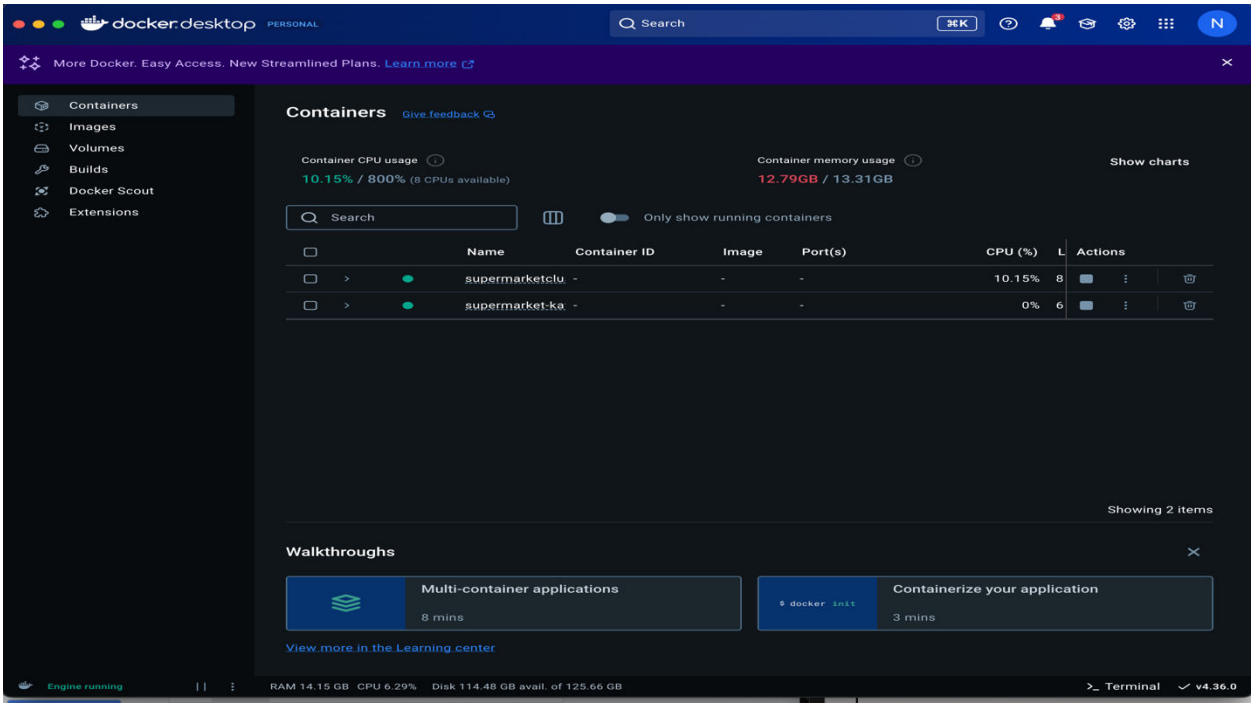


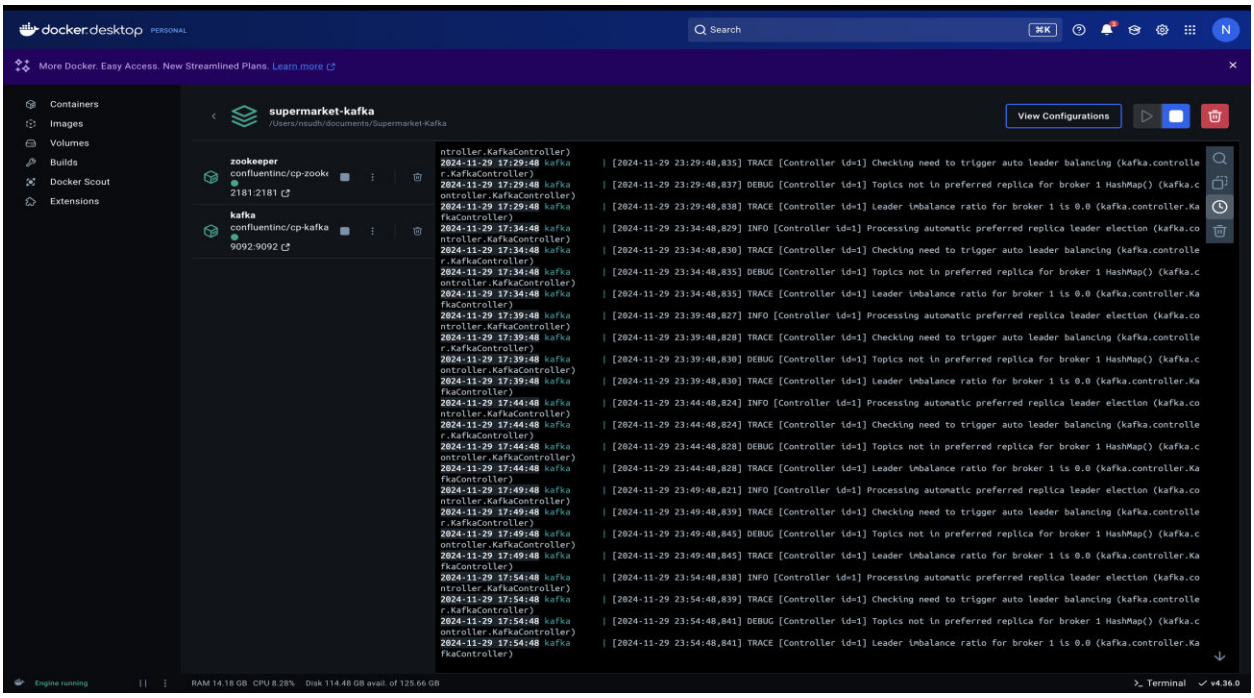
so here have setup two cluster because of system infrastructure constraint in Supermarket-Kafka cluster we have kaka and its zookeeper and produce and consumer as python scripts in supermarket-Cassandra cluster we have three nodes with location-a and b c and Prometheus and graphene to monitor the Cassandra nodes

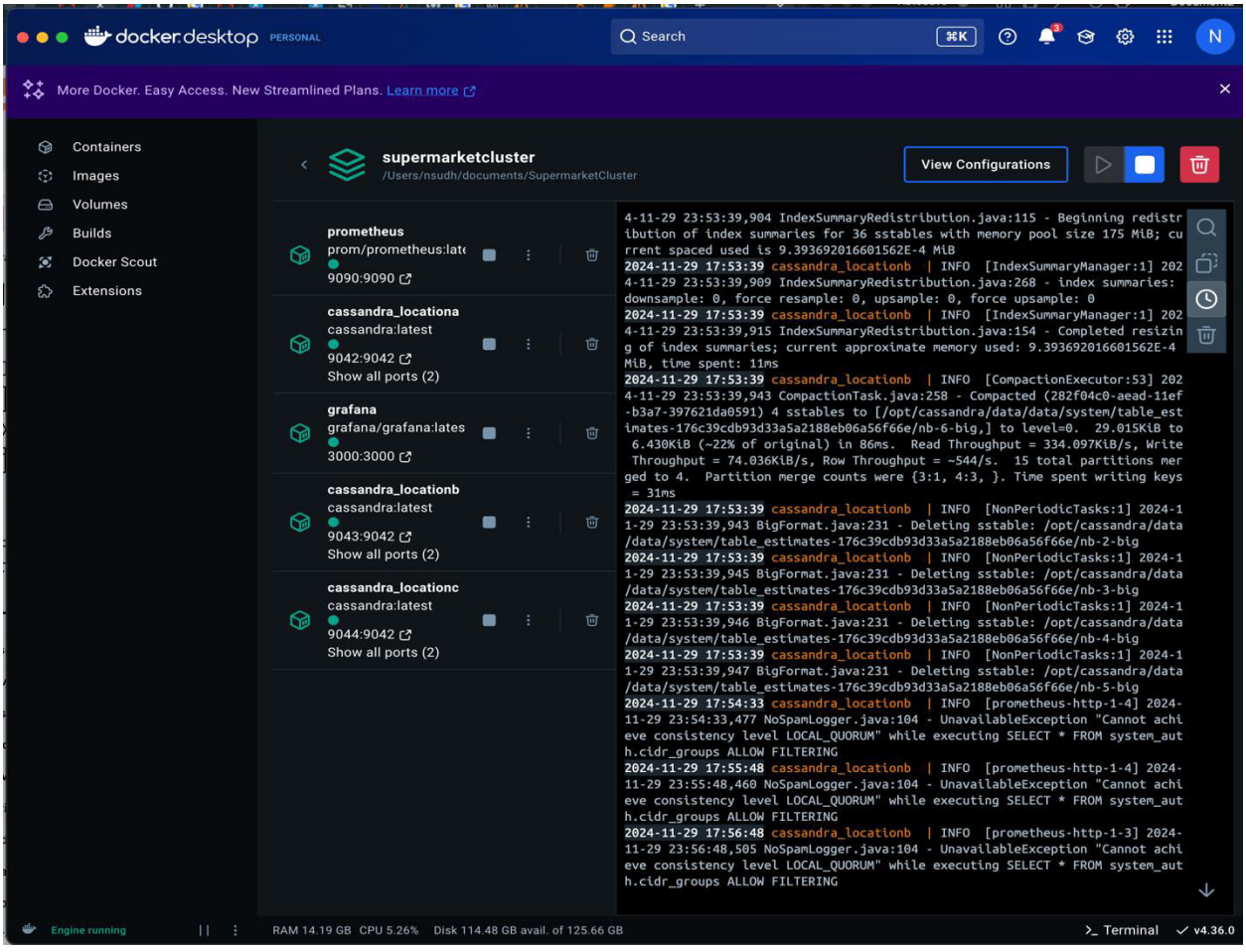
Flow of the System:

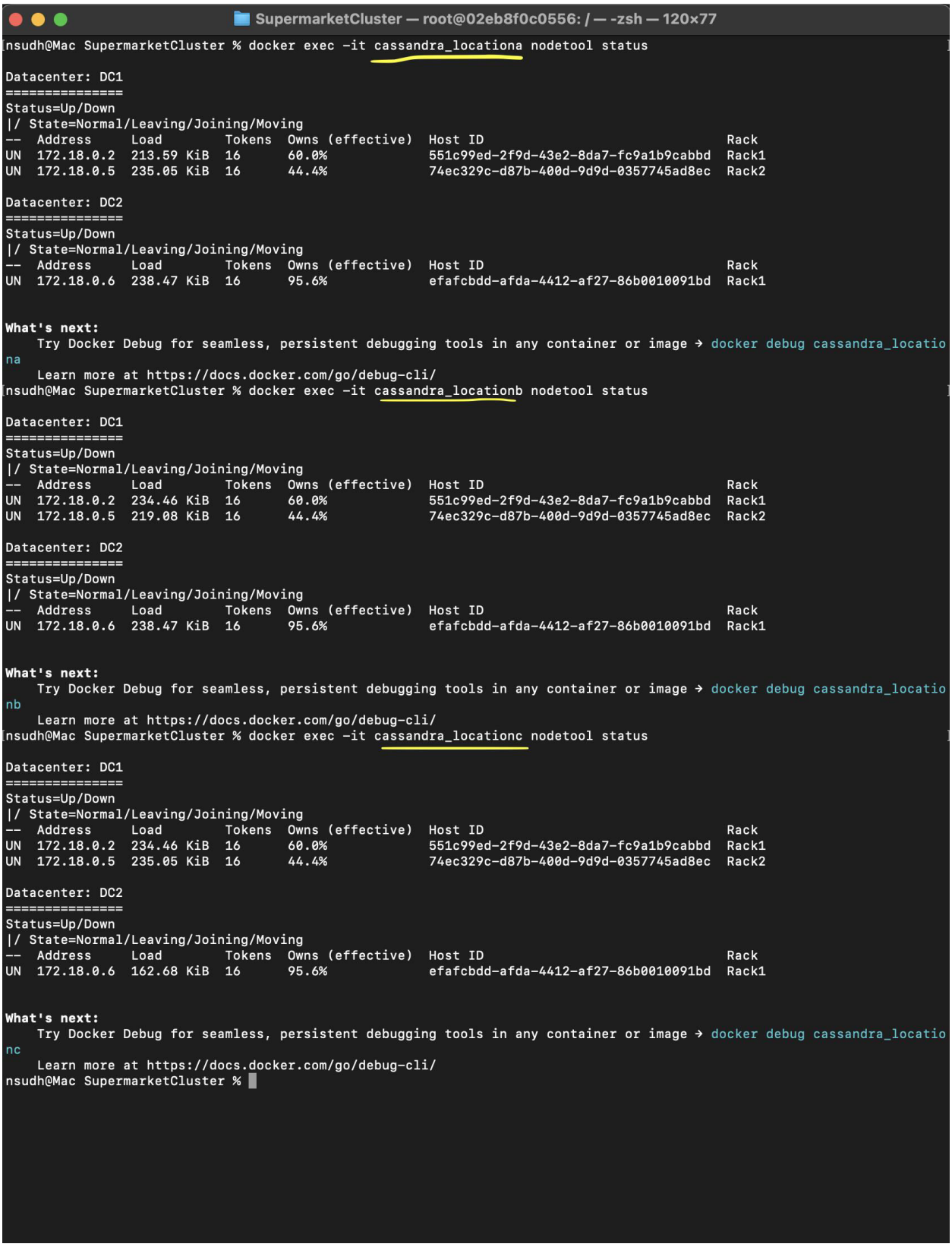
So will be running produce.py and read the data set and send to the topics in Kafka and then consumer.py(a,bc) picks those things and send it approximate Cassandra data base location

Setup of the System:




docker desktop PERSONAL
Search
More Docker. Easy Access. New Streamlined Plans. Learn more
Containers
Images
Volumes
Builds
Docker Scout
Extensions
supermarket-kafka
/Users/nsudh/documents/Supermarket-Kafka
View Configurations
zookeeper
confluentinc/cp-zooke
2181:2181
kafka
confluentinc/cp-kafka
9092:9092
Engine running
RAM 14.18 GB CPU 8.28% Disk 114.48 GB avail. of 125.66 GB
Terminal
v4.36.0

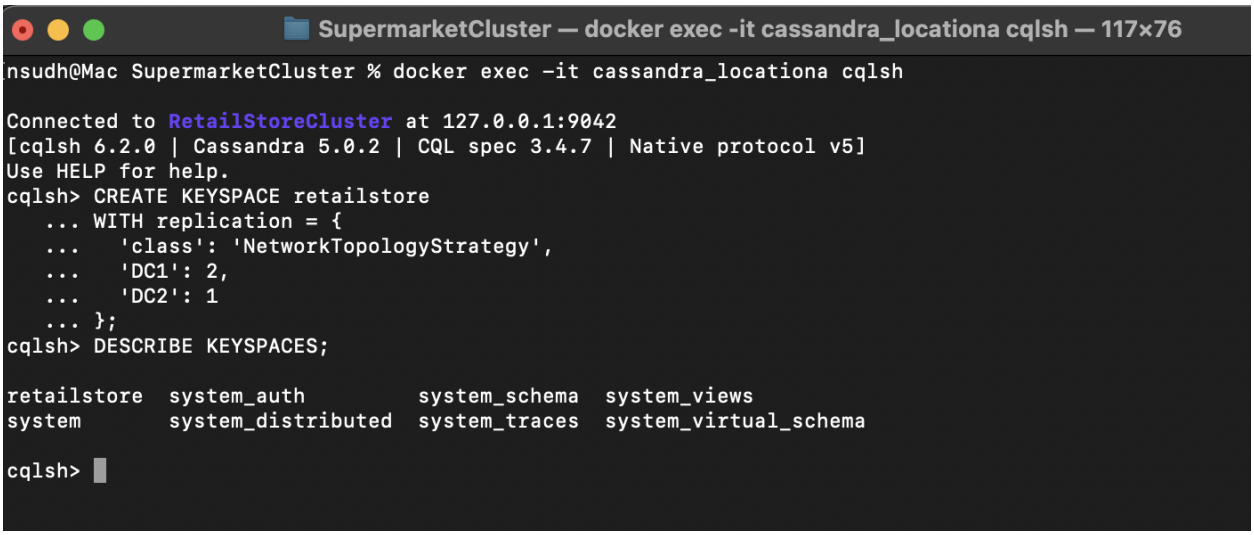

docker desktop PERSONAL
Search
More Docker. Easy Access. New Streamlined Plans. Learn more
Containers
Images
Volumes
Builds
Docker Scout
Extensions
supermarketcluster
/Users/nsudh/documents/SupermarketCluster
View Configurations
prometheus
prom/prometheus:late
9090:9090
cassandra_locationa
cassandra:latest
9042:9042
Show all ports (2)
grafana
grafana/grafana:lates
3000:3000
cassandra_locationb
cassandra:latest
9043:9042
Show all ports (2)
cassandra_locationc
cassandra:latest
9044:9042
Show all ports (2)
Engine running
RAM 14.19 GB CPU 5.26% Disk 114.48 GB avail. of 125.66 GB
Terminal
v4.36.0


SupermarketCluster — root@02eb8f0c0556: / — -zsh — 120×77
nsudh@Mac SupermarketCluster % docker exec -it cassandra_locationa nodetool status
Datacenter: DC1
==========
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address Load Tokens Owns (effective) Host ID Rack
UN 172.18.0.2 213.59 KiB 16 60.0% 551c99ed-2f9d-43e2-8da7-fc9a1b9cabbd Rack1
UN 172.18.0.5 235.05 KiB 16 44.4% 74ec329c-d87b-400d-9d9d-0357745ad8ec Rack2
Datacenter: DC2
==========
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address Load Tokens Owns (effective) Host ID Rack
UN 172.18.0.6 238.47 KiB 16 95.6% efafcbdd-afda-4412-af27-86b0010091bd Rack1
What's next:
Try Docker Debug for seamless, persistent debugging tools in any container or image → docker debug cassandra_locatio
na
Learn more at https://docs.docker.com/go/debug-cli/
nsudh@Mac SupermarketCluster % docker exec -it cassandra_locationb nodetool status
Datacenter: DC1
==========
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address Load Tokens Owns (effective) Host ID Rack
UN 172.18.0.2 234.46 KiB 16 60.0% 551c99ed-2f9d-43e2-8da7-fc9a1b9cabbd Rack1
UN 172.18.0.5 219.08 KiB 16 44.4% 74ec329c-d87b-400d-9d9d-0357745ad8ec Rack2
Datacenter: DC2
==========
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address Load Tokens Owns (effective) Host ID Rack
UN 172.18.0.6 238.47 KiB 16 95.6% efafcbdd-afda-4412-af27-86b0010091bd Rack1
What's next:
Try Docker Debug for seamless, persistent debugging tools in any container or image → docker debug cassandra_locatio
nb
Learn more at https://docs.docker.com/go/debug-cli/
nsudh@Mac SupermarketCluster % docker exec -it cassandra_locationc nodetool status
Datacenter: DC1
==========
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address Load Tokens Owns (effective) Host ID Rack
UN 172.18.0.2 234.46 KiB 16 60.0% 551c99ed-2f9d-43e2-8da7-fc9a1b9cabbd Rack1
UN 172.18.0.5 235.05 KiB 16 44.4% 74ec329c-d87b-400d-9d9d-0357745ad8ec Rack2
Datacenter: DC2
==========
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address Load Tokens Owns (effective) Host ID Rack
UN 172.18.0.6 162.68 KiB 16 95.6% efafcbdd-afda-4412-af27-86b0010091bd Rack1
What's next:
Try Docker Debug for seamless, persistent debugging tools in any container or image → docker debug cassandra_locatio
nc
Learn more at https://docs.docker.com/go/debug-cli/
nsudh@Mac SupermarketCluster %

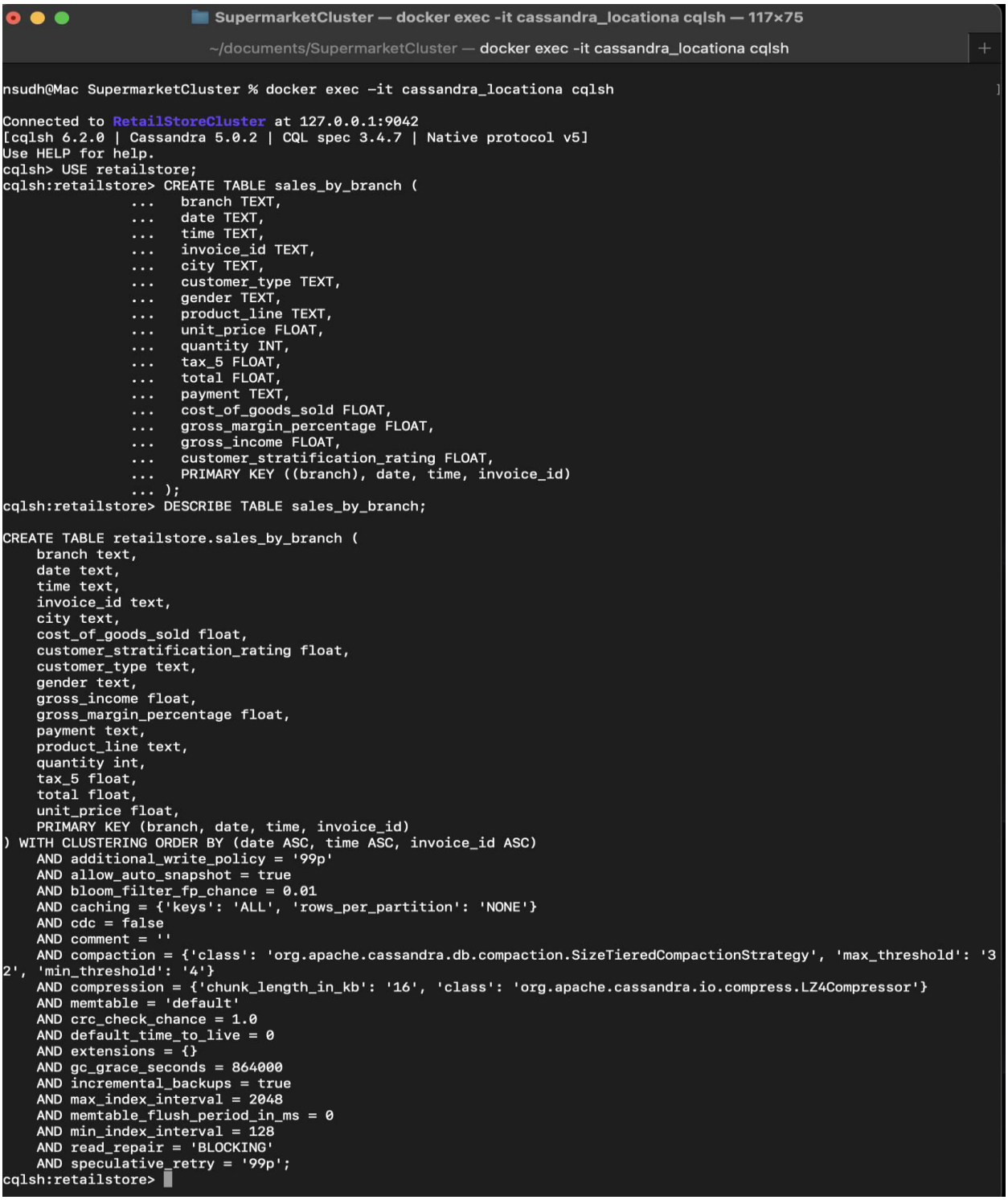
Data Modeling:

1) we will be created key space with dc1 with replication of 2 and dc2 s one as there is only one node
1) we will be using only one as he details server only that is requirement check out processing based on our data set
2) We will be using Patrion key as branch which by ensuring branch specific data is stored together
3) we will be using data and time as clustering keys which allow us for sorting and querying sales chronologically and we also invoice Ida s clustering key which ensure uniqueness within the partition

Creating the Key space

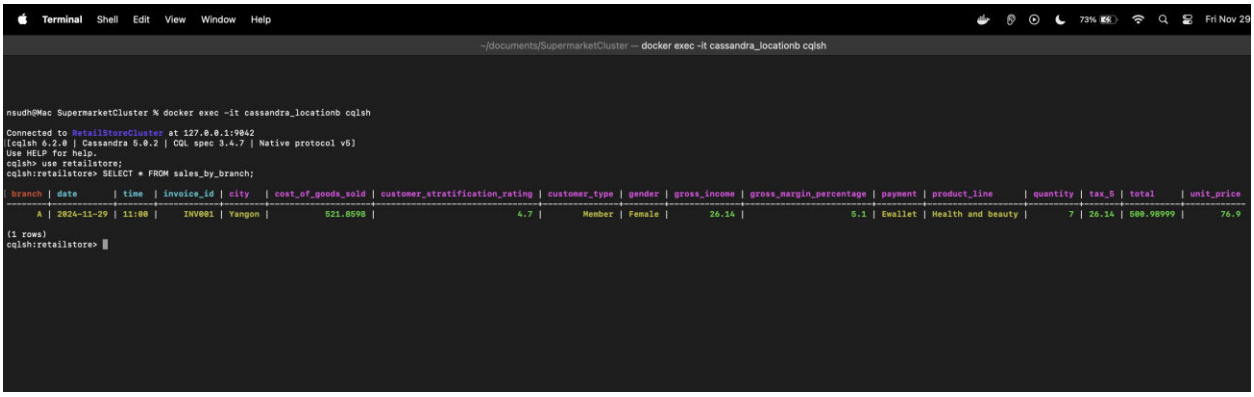


Creation of Table



Inserting a record and verifying the table in all location :

```
nsudh@Mac SupermarketCluster % docker exec -it cassandra_locationa cqlsh

Connected to RetailStoreCluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> USE retailstore;
cqlsh:retailstore> INSERT INTO sales_by_branch (
              ...   branch, date, time, invoice_id, city, customer_type, gender, product_line,
              ...   unit_price, quantity, tax_5, total, payment, cost_of_goods_sold,
              ...   gross_margin_percentage, gross_income, customer_stratification_rating
              ... ) VALUES (
              ...   'A', '2024-11-29', '11:00', 'INV001', 'Yangon', 'Member', 'Female', 'Health and beauty',
              ...   76.90, 7, 26.14, 500.99, 'Ewallet', 521.85982, 5.1, 26.14, 4.70
              ... );
cqlsh:retailstore> SELECT * FROM sales_by_branch;

 branch | date       | time  | invoice_id | city   | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment | product_line      | quantity | tax_5 | total     | unit_price
--------+------------+-------+------------+--------+--------------------+--------------------------------+---------------+--------+--------------+-------------------------+---------+-------------------+----------+-------+-----------+------------
      A | 2024-11-29 | 11:00 |     INV001 | Yangon |          521.8598 |                            4.7 |        Member | Female |        26.14 |                     5.1 | Ewallet | Health and beauty |        7 | 26.14 | 500.98999 |       76.9

(1 rows)
cqlsh:retailstore>
```



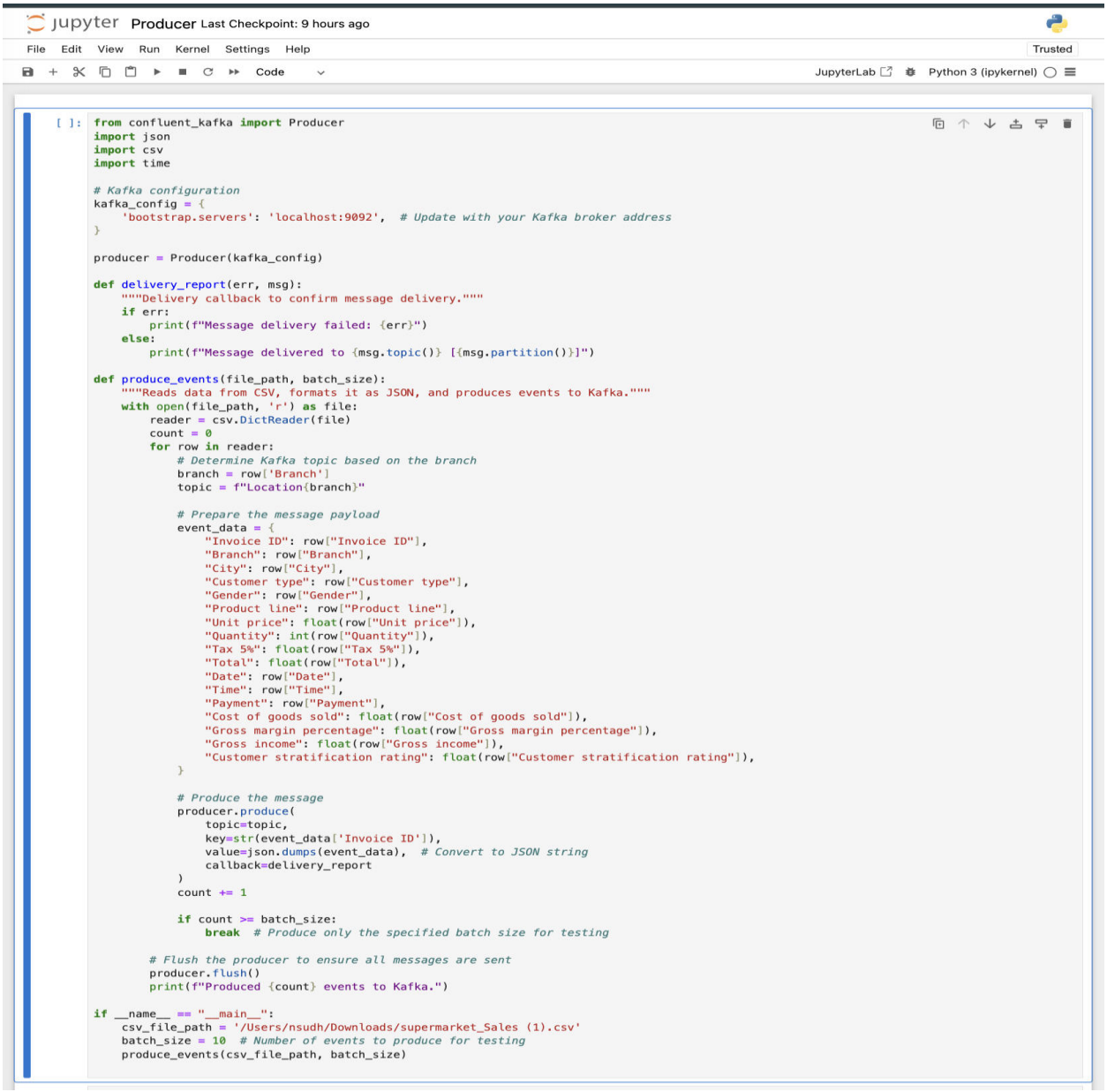
```
Terminal  Shell  Edit  View  Window  Help                                                    73%  Fri Nov 29

~/documents/SupermarketCluster — docker exec -it cassandra_locationb cqlsh

nsudh@Mac SupermarketCluster % docker exec -it cassandra_locationb cqlsh

Connected to RetailStoreCluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> use retailstore;
cqlsh:retailstore> SELECT * FROM sales_by_branch;

 branch | date       | time  | invoice_id | city   | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment | product_line      | quantity | tax_5 | total     | unit_price
--------+------------+-------+------------+--------+--------------------+--------------------------------+---------------+--------+--------------+-------------------------+---------+-------------------+----------+-------+-----------+------------
      A | 2024-11-29 | 11:00 |     INV001 | Yangon |          521.8598 |                            4.7 |        Member | Female |        26.14 |                     5.1 | Ewallet | Health and beauty |        7 | 26.14 | 500.98999 |       76.9

(1 rows)
cqlsh:retailstore>
```

```
nsudh@Mac SupermarketCluster % docker exec -it cassandra_locationc cqlsh

Connected to RetailStoreCluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> use retailstore;
cqlsh:retailstore> SELECT * FROM sales_by_branch;

 branch | date       | time  | invoice_id | city   | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment | product_line      | quantity | tax_5 | total     | unit_price
--------+------------+-------+------------+--------+--------------------+--------------------------------+---------------+--------+--------------+-------------------------+---------+-------------------+----------+-------+-----------+------------
      A | 2024-11-29 | 11:00 |     INV001 | Yangon |          521.8598 |                            4.7 |        Member | Female |        26.14 |                     5.1 | Ewallet | Health and beauty |        7 | 26.14 | 500.98999 |       76.9

(1 rows)
cqlsh:retailstore>
```

Implementation:

To implement the flow we will be pushing 10 rows from the dataset for easy recording (Note: Data base in Databases are truncated before performing these)

We be Setting the batch size as 10 in producser.py:



```
                key=str(event_data['Invoice ID']),
                value=json.dumps(event_data),  # Convert to JSON string
                callback=delivery_report
            )
            count += 1

            if count >= batch_size:
                break  # Produce only the specified batch size for testing

        # Flush the producer to ensure all messages are sent
        producer.flush()
        print(f"Produced {count} events to Kafka.")

if __name__ == "__main__":
    csv_file_path = '/Users/nsudh/Downloads/supermarket_Sales (1).csv'
    batch_size = 10  # Number of events to produce for testing
    produce_events(csv_file_path, batch_size)
```

```
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationC [0]
Message delivered to LocationC [0]
Message delivered to LocationC [0]
Message delivered to LocationB [0]
Produced 10 events to Kafka.
```

Consuming From Topic: Locationa

jupyter ConsumerA Last Checkpoint: 10 hours ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python 3 (ipykernel)

[ ]:
```python
from confluent_kafka import Consumer
import json
from cassandra.cluster import Cluster

# Kafka configuration
kafka_config = {
    'bootstrap.servers': 'localhost:9092',
    'group.id': 'consumer_a_group',
    'auto.offset.reset': 'earliest',
}

# Cassandra configuration
cassandra_cluster = Cluster(['127.0.0.1'])
session = cassandra_cluster.connect()
session.set_keyspace('retailstore')

def consume_events():
    consumer = Consumer(kafka_config)
    consumer.subscribe(['LocationA'])

    while True:
        msg = consumer.poll(1.0)  # Timeout in seconds
        if msg is None:
            continue
        if msg.error():
            print(f"Consumer error: {msg.error()}")
            continue

        try:
            record_value = json.loads(msg.value().decode('utf-8'))  # Decode and parse JSON
            print(f"Consumed event: {record_value}")

            # Insert into Cassandra
            session.execute("""
                INSERT INTO sales_by_branch(
                    invoice_id, branch, city, customer_type, gender, product_line,
                    unit_price, quantity, tax_5, total, date, time, payment,
                    cost_of_goods_sold, gross_margin_percentage, gross_income, customer_stratification_rating
                ) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
            """, (
                record_value['Invoice ID'], record_value['Branch'], record_value['City'],
                record_value['Customer type'], record_value['Gender'], record_value['Product line'],
                record_value['Unit price'], record_value['Quantity'], record_value['Tax 5%'],
                record_value['Total'], record_value['Date'], record_value['Time'],
                record_value['Payment'], record_value['Cost of goods sold'],
                record_value['Gross margin percentage'], record_value['Gross income'],
                record_value['Customer stratification rating']
            ))
        except json.JSONDecodeError as e:
            print(f"JSON decoding error: {e}. Message: {msg.value()}")
        except Exception as e:
            print(f"Error processing message: {e}")

if __name__ == "__main__":
    consume_events()
```

[ ]:

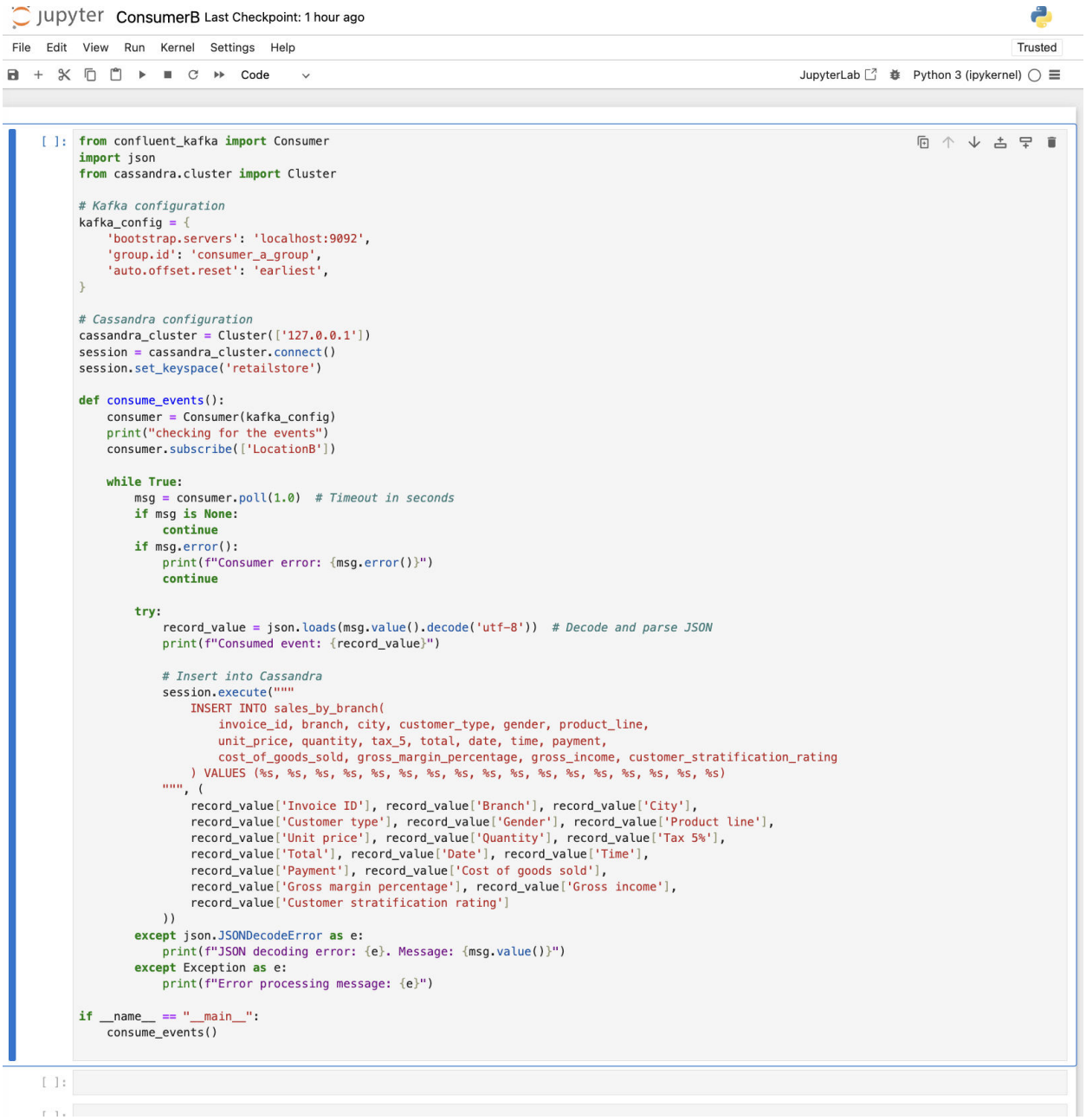
[ ]:

```
            ))
        except json.JSONDecodeError as e:
            print(f"JSON decoding error: {e}. Message: {msg.value()}")
        except Exception as e:
            print(f"Error processing message: {e}")

if __name__ == "__main__":
    consume_events()
```

```
Consumed event: {'Invoice ID': '750-67-8428', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Female', 'Product line':
'Health and beauty', 'Unit price': 74.69, 'Quantity': 7, 'Tax 5%': 26.1415, 'Total': 548.9715, 'Date': '1/5/2019', 'Time': '13:08', 'Paymen
t': 'Ewallet', 'Cost of goods sold': 522.83, 'Gross margin percentage': 4.761904762, 'Gross income': 26.1415, 'Customer stratification ratin
g': 9.1}
Consumed event: {'Invoice ID': '631-41-3108', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Normal', 'Gender': 'Male', 'Product line':
'Home and lifestyle', 'Unit price': 46.33, 'Quantity': 7, 'Tax 5%': 16.2155, 'Total': 340.5255, 'Date': '3/3/2019', 'Time': '13:23', 'Paymen
t': 'Credit card', 'Cost of goods sold': 324.31, 'Gross margin percentage': 4.761904762, 'Gross income': 16.2155, 'Customer stratification ra
ting': 7.4}
Consumed event: {'Invoice ID': '123-19-1176', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Male', 'Product line':
'Health and beauty', 'Unit price': 58.22, 'Quantity': 8, 'Tax 5%': 23.288, 'Total': 489.048, 'Date': '1/27/2019', 'Time': '20:33', 'Payment':
'Ewallet', 'Cost of goods sold': 465.76, 'Gross margin percentage': 4.761904762, 'Gross income': 23.288, 'Customer stratification rating': 8.
4}
Consumed event: {'Invoice ID': '373-73-7910', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Normal', 'Gender': 'Male', 'Product line':
'Sports and travel', 'Unit price': 86.31, 'Quantity': 7, 'Tax 5%': 30.2085, 'Total': 634.3785, 'Date': '2/8/2019', 'Time': '10:37', 'Paymen
t': 'Ewallet', 'Cost of goods sold': 604.17, 'Gross margin percentage': 4.761904762, 'Gross income': 30.2085, 'Customer stratification ratin
g': 5.3}
Consumed event: {'Invoice ID': '355-53-5943', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Female', 'Product line':
'Electronic accessories', 'Unit price': 68.84, 'Quantity': 6, 'Tax 5%': 20.652, 'Total': 433.692, 'Date': '2/25/2019', 'Time': '14:36', 'Paym
ent': 'Ewallet', 'Cost of goods sold': 413.04, 'Gross margin percentage': 4.761904762, 'Gross income': 20.652, 'Customer stratification ratin
g': 5.8}
Consumed event: {'Invoice ID': '665-32-9167', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Female', 'Product line':
'Health and beauty', 'Unit price': 36.26, 'Quantity': 2, 'Tax 5%': 3.626, 'Total': 76.146, 'Date': '1/10/2019', 'Time': '17:15', 'Payment':
'Credit card', 'Cost of goods sold': 72.52, 'Gross margin percentage': 4.761904762, 'Gross income': 3.626, 'Customer stratification rating':
7.2}
```

```
SupermarketCluster — root@348fa6287f94: / — docker exec -it cassandra_locationa cqlsh — 118×76

nsudh@SUDHEERs-MacBook-Pro-3 SupermarketCluster % docker exec -it cassandra_locationa cqlsh
Connected to RetailStoreCluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> use retailstore;
cqlsh:retailstore> select * from sales_by_branch;

 branch | date      | time  | invoice_id  | city   | cost_of_goods_sold | customer_stratification_rating | customer_ty
pe | gender | gross_income | gross_margin_percentage | payment     | product_line           | quantity | tax_5    | tot
al     | unit_price
--------+-----------+-------+-------------+--------+--------------------+--------------------------------+------------
---+--------+--------------+-------------------------+-------------+------------------------+----------+----------+----
-------+------------
      A | 1/10/2019 | 17:15 | 665-32-9167 | Yangon |              72.52 |                            7.2 |        Memb
er | Female |        3.626 |                  4.7619 | Credit card |      Health and beauty |        2 |    3.626 |
76.146 |      36.26
      A | 1/27/2019 | 20:33 | 123-19-1176 | Yangon |         465.76001 |                            8.4 |        Memb
er |   Male |       23.288 |                  4.7619 |     Ewallet |      Health and beauty |        8 |   23.288 |    4
89.048 |      58.22
      A |  1/5/2019 | 13:08 | 750-67-8428 | Yangon |         522.83002 |                            9.1 |        Memb
er | Female |      26.1415 |                  4.7619 |     Ewallet |      Health and beauty |        7 | 26.1415 |   54
8.9715 |      74.69
      A | 2/25/2019 | 14:36 | 355-53-5943 | Yangon |         413.04001 |                            5.8 |        Memb
er | Female |       20.652 |                  4.7619 |     Ewallet | Electronic accessories |        6 |   20.652 | 433
.69199 |      68.84
      A |  2/8/2019 | 10:37 | 373-73-7910 | Yangon |         604.16998 |                            5.3 |        Norm
al |   Male |      30.2085 |                  4.7619 |     Ewallet |      Sports and travel |        7 | 30.2085 | 634
.37848 |      86.31
      A |  3/3/2019 | 13:23 | 631-41-3108 | Yangon |            324.31 |                            7.4 |        Norm
al |   Male |      16.2155 |                  4.7619 | Credit card |     Home and lifestyle |        7 | 16.2155 | 340
.52551 |      46.33

(6 rows)
cqlsh:retailstore>
```

Consuming From TopicB:



```
                record_value['Customer stratification rating']
            ))
        except json.JSONDecodeError as e:
            print(f"JSON decoding error: {e}. Message: {msg.value()}")
        except Exception as e:
            print(f"Error processing message: {e}")

if __name__ == "__main__":
    consume_events()
```

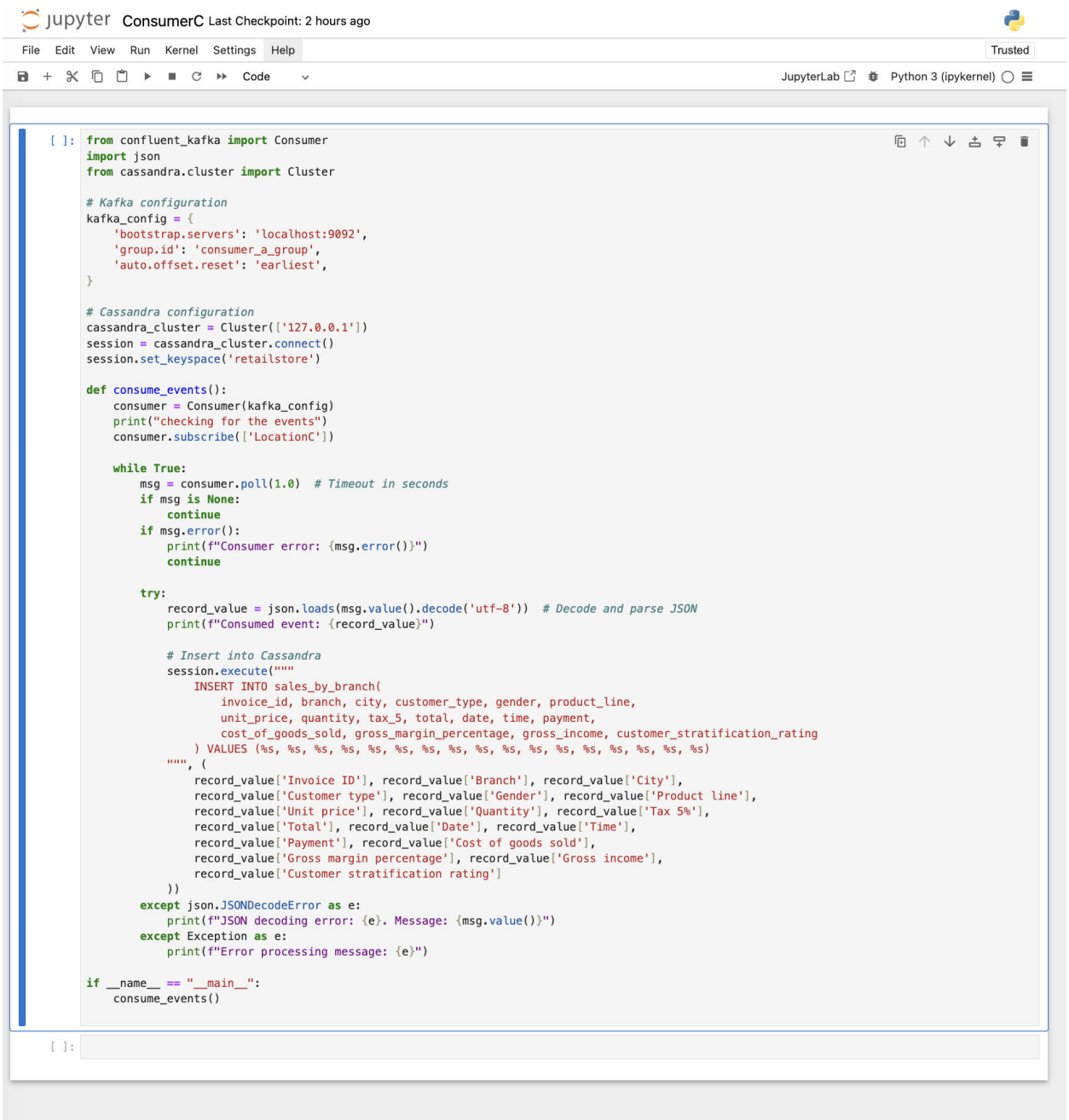
```
checking for the events
Consumed event: {'Invoice ID': '692-92-5582', 'Branch': 'B', 'City': 'Mandalay', 'Customer type': 'Member', 'Gender': 'Female', 'Product lin
e': 'Food and beverages', 'Unit price': 54.84, 'Quantity': 3, 'Tax 5%': 8.226, 'Total': 172.746, 'Date': '2/20/2019', 'Time': '13:27', 'Payme
nt': 'Credit card', 'Cost of goods sold': 164.52, 'Gross margin percentage': 4.761904762, 'Gross income': 8.226, 'Customer stratification rat
ing': 5.9}
```

[ ]:

SupermarketCluster — root@348fa6287f94: / — docker exec -it cassandra_locationb cqlsh — 118×76

```
nsudh@SUDHEERs-MacBook-Pro-3 SupermarketCluster % docker exec -it cassandra_locationb cqlsh
Connected to RetailStoreCluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> use retailstore;
cqlsh:retailstore> select * from retailstore.sales_by_branch;

 branch | date      | time  | invoice_id  | city     | cost_of_goods_sold | customer_stratification_rating | customer_
type | gender | gross_income | gross_margin_percentage | payment     | product_line           | quantity | tax_5    | t
otal     | unit_price
--------+-----------+-------+-------------+----------+--------------------+--------------------------------+----------
-----+--------+--------------+-------------------------+-------------+------------------------+----------+----------+--
---------+------------

      B | 2/20/2019 | 13:27 | 692-92-5582 | Mandalay |             164.52 |                            5.9 |        Me
mber | Female |        8.226 |                  4.7619 | Credit card |     Food and beverages |        3 |    8.226 |
 172.746 |      54.84
      A | 1/10/2019 | 17:15 | 665-32-9167 |   Yangon |              72.52 |                            7.2 |        Me
mber | Female |        3.626 |                  4.7619 | Credit card |      Health and beauty |        2 |    3.626 |
  76.146 |      36.26
      A | 1/27/2019 | 20:33 | 123-19-1176 |   Yangon |          465.76001 |                            8.4 |        Me
mber |   Male |       23.288 |                  4.7619 |     Ewallet |      Health and beauty |        8 |   23.288 |
 489.048 |      58.22
      A |  1/5/2019 | 13:08 | 750-67-8428 |   Yangon |          522.83002 |                            9.1 |        Me
mber | Female |      26.1415 |                  4.7619 |     Ewallet |      Health and beauty |        7 |  26.1415 |
548.9715 |      74.69
      A | 2/25/2019 | 14:36 | 355-53-5943 |   Yangon |          413.04001 |                            5.8 |        Me
mber | Female |       20.652 |                  4.7619 |     Ewallet | Electronic accessories |        6 |   20.652 | 4
33.69199 |      68.84
      A |  2/8/2019 | 10:37 | 373-73-7910 |   Yangon |          604.16998 |                            5.3 |        No
rmal |   Male |      30.2085 |                  4.7619 |     Ewallet |      Sports and travel |        7 |  30.2085 | 6
34.37848 |      86.31
      A |  3/3/2019 | 13:23 | 631-41-3108 |   Yangon |             324.31 |                            7.4 |        No
rmal |   Male |      16.2155 |                  4.7619 | Credit card |     Home and lifestyle |        7 |  16.2155 | 3
40.52551 |      46.33

(7 rows)
cqlsh:retailstore>
```

Consume from c:



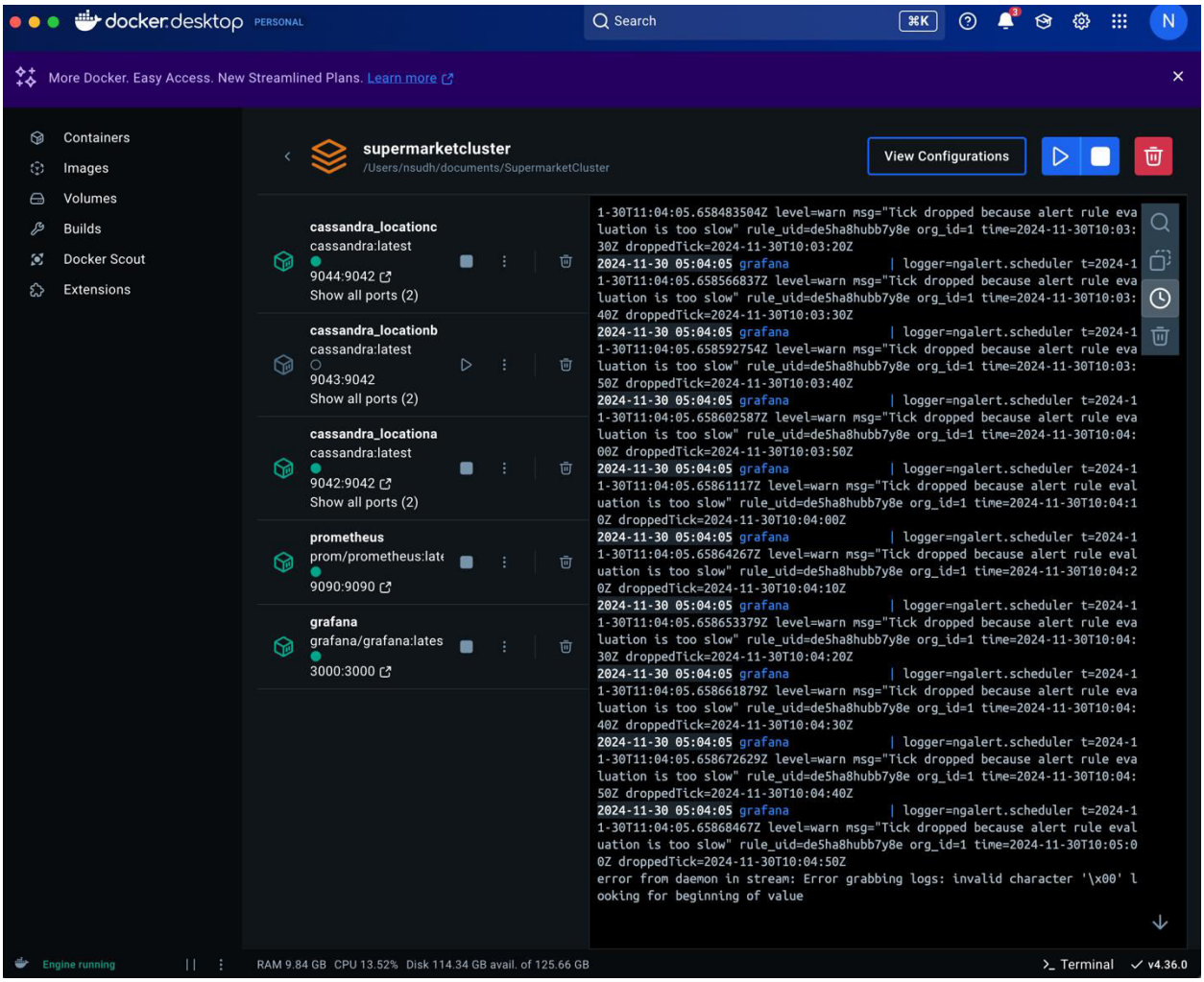
```
            ))
        except json.JSONDecodeError as e:
            print(f"JSON decoding error: {e}. Message: {msg.value()}")
        except Exception as e:
            print(f"Error processing message: {e}")

if __name__ == "__main__":
    consume_events()
```

```
checking for the events
Consumed event: {'Invoice ID': '226-31-3081', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Normal', 'Gender': 'Female', 'Product lin
e': 'Electronic accessories', 'Unit price': 15.28, 'Quantity': 5, 'Tax 5%': 3.82, 'Total': 80.22, 'Date': '3/8/2019', 'Time': '10:29', 'Payme
nt': 'Cash', 'Cost of goods sold': 76.4, 'Gross margin percentage': 4.761904762, 'Gross income': 3.82, 'Customer stratification rating': 9.6}
Consumed event: {'Invoice ID': '699-14-3026', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Normal', 'Gender': 'Male', 'Product lin
e': 'Electronic accessories', 'Unit price': 85.39, 'Quantity': 7, 'Tax 5%': 29.8865, 'Total': 627.6165, 'Date': '3/25/2019', 'Time': '18:30',
'Payment': 'Ewallet', 'Cost of goods sold': 597.73, 'Gross margin percentage': 4.761904762, 'Gross income': 29.8865, 'Customer stratification
rating': 4.1}
Consumed event: {'Invoice ID': '315-22-5665', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Normal', 'Gender': 'Female', 'Product lin
e': 'Home and lifestyle', 'Unit price': 73.56, 'Quantity': 10, 'Tax 5%': 36.78, 'Total': 772.38, 'Date': '2/24/2019', 'Time': '11:38', 'Payme
nt': 'Ewallet', 'Cost of goods sold': 735.6, 'Gross margin percentage': 4.761904762, 'Gross income': 36.78, 'Customer stratification rating':
8.0}
```

SupermarketCluster — root@348fa6287f94: / — docker exec -it cassandra_locationc cqlsh — 118×76

```
insudh@SUDHEERs-MacBook-Pro-3 SupermarketCluster % docker exec -it cassandra_locationc cqlsh
Connected to RetailStoreCluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> use retailstore;
cqlsh:retailstore> SELECT * FROM sales_by_branch;

 branch | date      | time  | invoice_id  | city      | cost_of_goods_sold | customer_stratification_rating | customer
_type | gender | gross_income | gross_margin_percentage | payment     | product_line           | quantity | tax_5    |
total     | unit_price
--------+-----------+-------+-------------+-----------+--------------------+--------------------------------+---------
---------+--------+--------------+-------------------------+-------------+------------------------+----------+----------+-
----------+------------
      C | 2/24/2019 | 11:38 | 315-22-5665 | Naypyitaw |          735.59998 |                              8 |        N
ormal | Female |        36.78 |                  4.7619 |     Ewallet |     Home and lifestyle |       10 |   36.78 |
   772.38 |      73.56
      C | 3/25/2019 | 18:30 | 699-14-3026 | Naypyitaw |          597.72998 |                            4.1 |        N
ormal |   Male |      29.8865 |                  4.7619 |     Ewallet | Electronic accessories |        7 | 29.8865 |
627.61652 |      85.39
      C |  3/8/2019 | 10:29 | 226-31-3081 | Naypyitaw |               76.4 |                            9.6 |        N
ormal | Female |         3.82 |                  4.7619 |        Cash | Electronic accessories |        5 |    3.82 |
   80.22 |      15.28
      B | 2/20/2019 | 13:27 | 692-92-5582 |  Mandalay |             164.52 |                            5.9 |        M
ember | Female |        8.226 |                  4.7619 | Credit card |     Food and beverages |        3 |   8.226 |
  172.746 |      54.84
      A | 1/10/2019 | 17:15 | 665-32-9167 |    Yangon |              72.52 |                            7.2 |        M
ember | Female |        3.626 |                  4.7619 | Credit card |      Health and beauty |        2 |   3.626 |
   76.146 |      36.26
      A | 1/27/2019 | 20:33 | 123-19-1176 |    Yangon |          465.76001 |                            8.4 |        M
ember |   Male |       23.288 |                  4.7619 |     Ewallet |      Health and beauty |        8 |  23.288 |
  489.048 |      58.22
      A |  1/5/2019 | 13:08 | 750-67-8428 |    Yangon |          522.83002 |                            9.1 |        M
ember | Female |      26.1415 |                  4.7619 |     Ewallet |      Health and beauty |        7 | 26.1415 |
 548.9715 |      74.69
      A | 2/25/2019 | 14:36 | 355-53-5943 |    Yangon |          413.04001 |                            5.8 |        M
ember | Female |       20.652 |                  4.7619 |     Ewallet | Electronic accessories |        6 |  20.652 |
433.69199 |      68.84
      A |  2/8/2019 | 10:37 | 373-73-7910 |    Yangon |          604.16998 |                            5.3 |        N
ormal |   Male |      30.2085 |                  4.7619 |     Ewallet |      Sports and travel |        7 | 30.2085 |
634.37848 |      86.31
      A |  3/3/2019 | 13:23 | 631-41-3108 |    Yangon |             324.31 |                            7.4 |        N
ormal |   Male |      16.2155 |                  4.7619 | Credit card |     Home and lifestyle |        7 | 16.2155 |
340.52551 |      46.33

(10 rows)
cqlsh:retailstore>
```

Implementation for fault tolerance and consistency level when one node down:



Cassandra db. is emptied:

```
nsudh@SUDHEERs-MacBook-Pro-3 SupermarketCluster % docker exec -it cassandra_locationa cqlsh
Connected to RetailStoreCluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> use retailstore;
cqlsh:retailstore> select * from sales_by_branch;

 branch | date | time | invoice_id | city | cost_of_goods_sold | customer_stratification_rating | customer_type | gend
er | gross_income | gross_margin_percentage | payment | product_line | quantity | tax_5 | total | unit_price
--------+------+------+------------+------+--------------------+--------------------------------+---------------+-----
---+--------------+-------------------------+---------+--------------+----------+-------+-------+------------

(0 rows)
cqlsh:retailstore>
```

jupyter **Producer** Last Checkpoint: 23 hours ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python 3 (ipykernel)

[1]:
```
from confluent_kafka import Producer
import json
import csv
import time

# Kafka configuration
kafka_config = {
    'bootstrap.servers': 'localhost:9092',  # Update with your Kafka broker address
}

producer = Producer(kafka_config)

def delivery_report(err, msg):
    """Delivery callback to confirm message delivery."""
    if err:
        print(f"Message delivery failed: {err}")
    else:
        print(f"Message delivered to {msg.topic()} [{msg.partition()}]")

def produce_events(file_path, batch_size):
    """Reads data from CSV, formats it as JSON, and produces events to Kafka."""
    with open(file_path, 'r') as file:
        reader = csv.DictReader(file)
        count = 0
        for row in reader:
            # Determine Kafka topic based on the branch
            branch = row['Branch']
            topic = f"Location{branch}"

            # Prepare the message payload
            event_data = {
                "Invoice ID": row["Invoice ID"],
                "Branch": row["Branch"],
                "City": row["City"],
                "Customer type": row["Customer type"],
                "Gender": row["Gender"],
                "Product line": row["Product line"],
                "Unit price": float(row["Unit price"]),
                "Quantity": int(row["Quantity"]),
                "Tax 5%": float(row["Tax 5%"]),
                "Total": float(row["Total"]),
                "Date": row["Date"],
                "Time": row["Time"],
                "Payment": row["Payment"],
                "Cost of goods sold": float(row["Cost of goods sold"]),
                "Gross margin percentage": float(row["Gross margin percentage"]),
                "Gross income": float(row["Gross income"]),
                "Customer stratification rating": float(row["Customer stratification rating"]),
            }

            # Produce the message
            producer.produce(
                topic=topic,
                key=str(event_data['Invoice ID']),
                value=json.dumps(event_data),  # Convert to JSON string
                callback=delivery_report
            )
            count += 1

            if count >= batch_size:
                break  # Produce only the specified batch size for testing

        # Flush the producer to ensure all messages are sent
        producer.flush()
        print(f"Produced {count} events to Kafka.")

if __name__ == "__main__":
    csv_file_path = '/Users/nsudh/Downloads/supermarket_Sales (1).csv'
    batch_size = 10  # Number of events to produce for testing
    produce_events(csv_file_path, batch_size)
```

```
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationA [0]
Message delivered to LocationC [0]
Message delivered to LocationC [0]
Message delivered to LocationC [0]
Message delivered to LocationB [0]
Produced 10 events to Kafka.
```

jupyter ConsumerA Last Checkpoint: 23 hours ago

File Edit View Run Kernel Settings Help Trusted

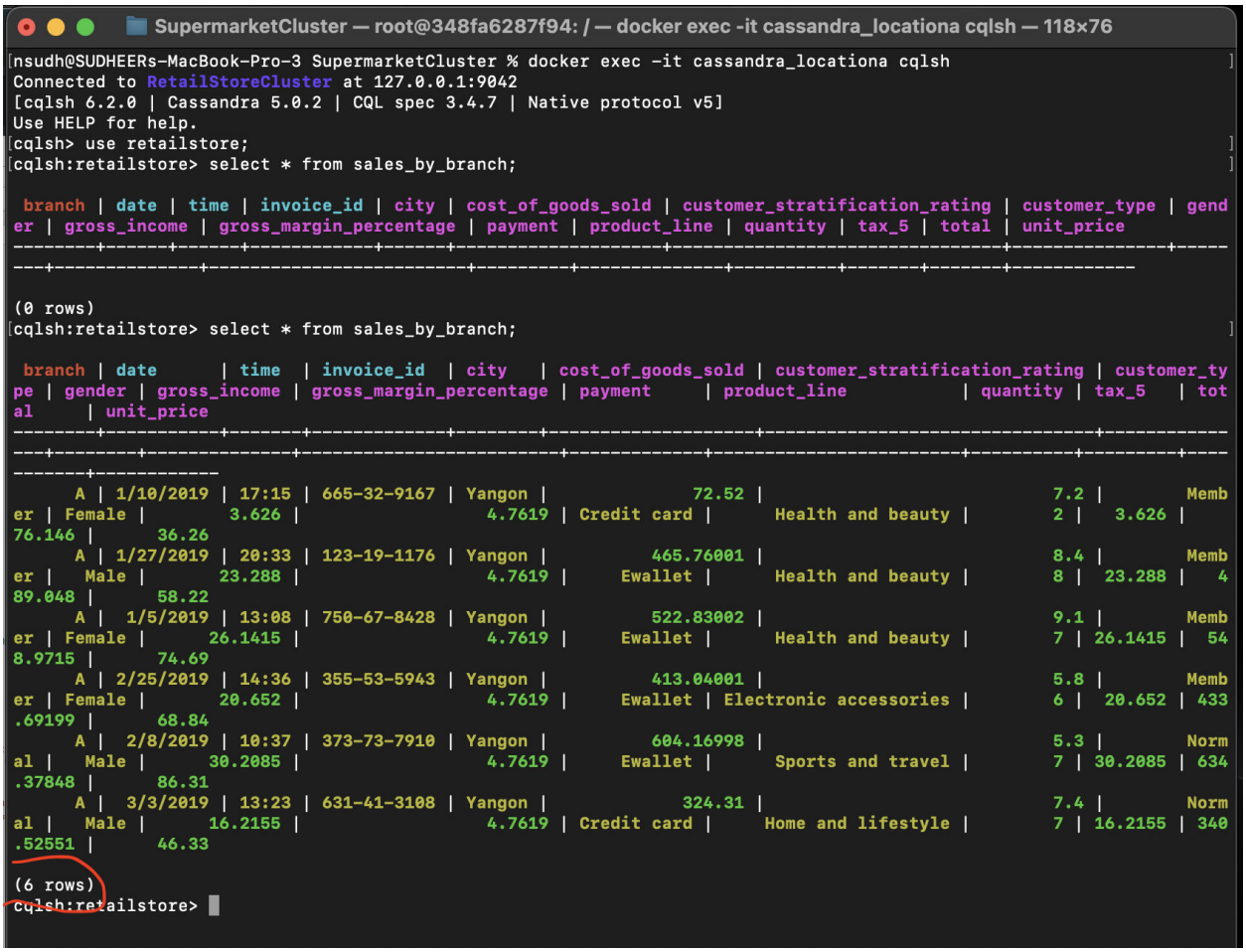
Code JupyterLab Python 3 (ipykernel)

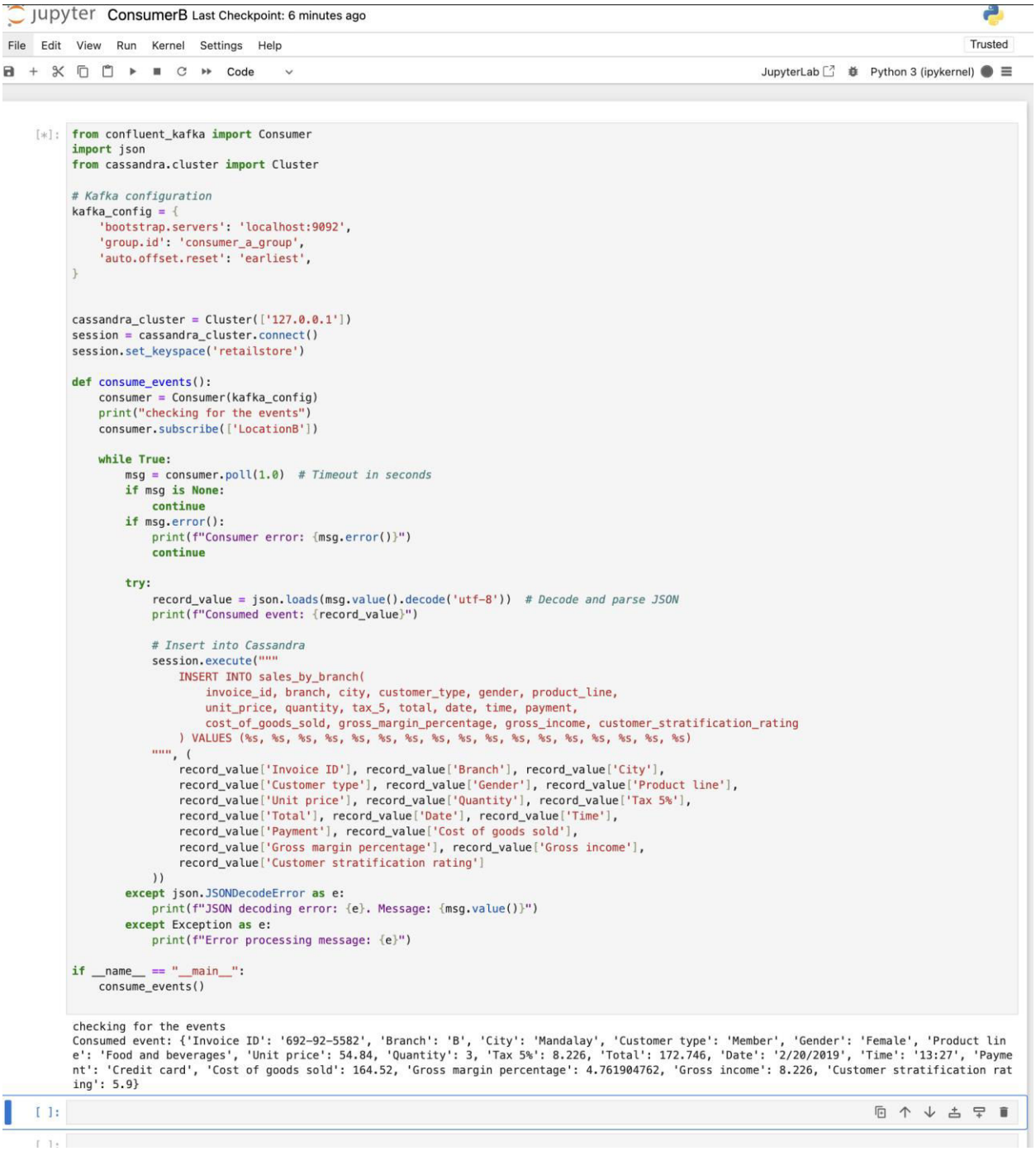
[*]:
```python
from confluent_kafka import Consumer
import json
from cassandra.cluster import Cluster

# Kafka configuration
kafka_config = {
    'bootstrap.servers': 'localhost:9092',
    'group.id': 'consumer_a_group',
    'auto.offset.reset': 'earliest',
}

# Cassandra configuration
cassandra_cluster = Cluster(['127.0.0.1'])
session = cassandra_cluster.connect()
session.set_keyspace('retailstore')

def consume_events():
    consumer = Consumer(kafka_config)
    consumer.subscribe(['LocationA'])

    while True:
        msg = consumer.poll(1.0)  # Timeout in seconds
        if msg is None:
            continue
        if msg.error():
            print(f"Consumer error: {msg.error()}")
            continue

        try:
            record_value = json.loads(msg.value().decode('utf-8'))  # Decode and parse JSON
            print(f"Consumed event: {record_value}")

            # Insert into Cassandra
            session.execute("""
                INSERT INTO sales_by_branch(
                    invoice_id, branch, city, customer_type, gender, product_line,
                    unit_price, quantity, tax_5, total, date, time, payment,
                    cost_of_goods_sold, gross_margin_percentage, gross_income, customer_stratification_rating
                ) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
            """, (
                record_value['Invoice ID'], record_value['Branch'], record_value['City'],
                record_value['Customer type'], record_value['Gender'], record_value['Product line'],
                record_value['Unit price'], record_value['Quantity'], record_value['Tax 5%'],
                record_value['Total'], record_value['Date'], record_value['Time'],
                record_value['Payment'], record_value['Cost of goods sold'],
                record_value['Gross margin percentage'], record_value['Gross income'],
                record_value['Customer stratification rating']
            ))
        except json.JSONDecodeError as e:
            print(f"JSON decoding error: {e}. Message: {msg.value()}")
        except Exception as e:
            print(f"Error processing message: {e}")

if __name__ == "__main__":
    consume_events()
```

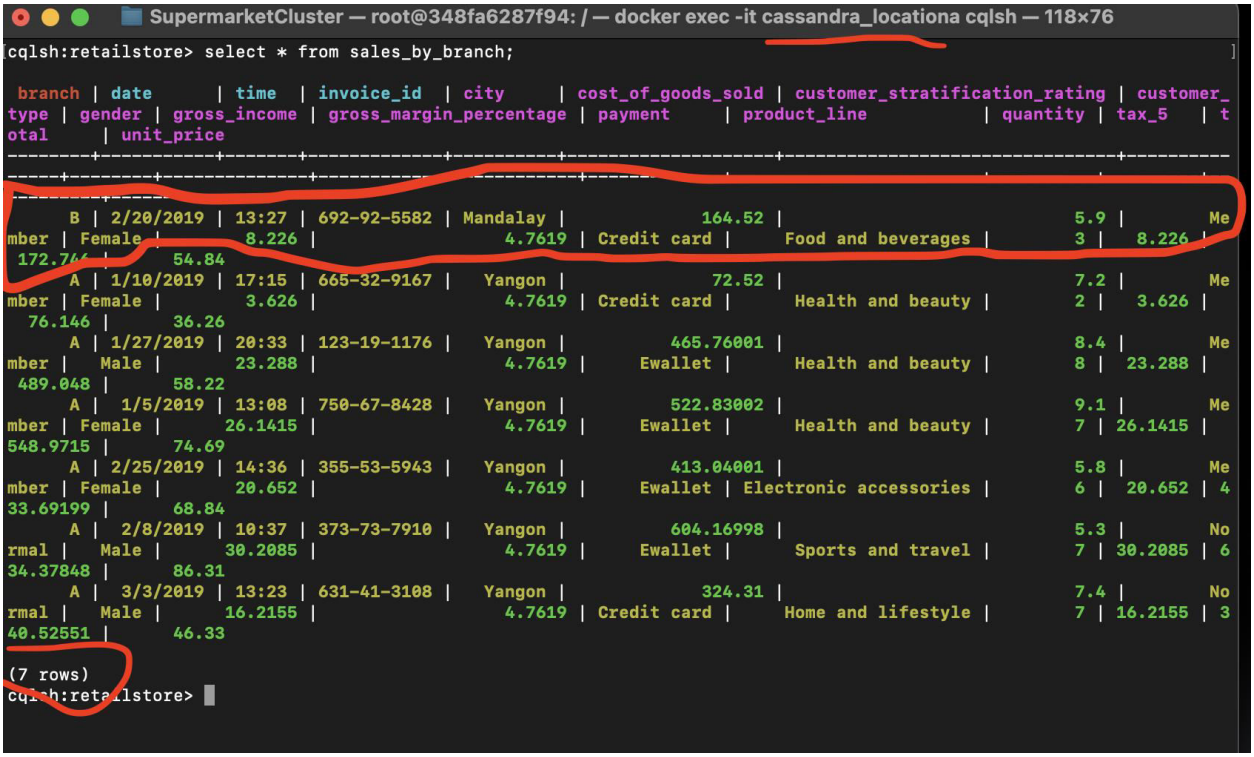
```
Consumed event: {'Invoice ID': '750-67-8428', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Female', 'Product line':
'Health and beauty', 'Unit price': 74.69, 'Quantity': 7, 'Tax 5%': 26.1415, 'Total': 548.9715, 'Date': '1/5/2019', 'Time': '13:08', 'Paymen
t': 'Ewallet', 'Cost of goods sold': 522.83, 'Gross margin percentage': 4.761904762, 'Gross income': 26.1415, 'Customer stratification ratin
g': 9.1}
Consumed event: {'Invoice ID': '631-41-3108', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Normal', 'Gender': 'Male', 'Product line':
'Home and lifestyle', 'Unit price': 46.33, 'Quantity': 7, 'Tax 5%': 16.2155, 'Total': 340.5255, 'Date': '3/3/2019', 'Time': '13:23', 'Paymen
t': 'Credit card', 'Cost of goods sold': 324.31, 'Gross margin percentage': 4.761904762, 'Gross income': 16.2155, 'Customer stratification ra
ting': 7.4}
Consumed event: {'Invoice ID': '123-19-1176', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Male', 'Product line':
'Health and beauty', 'Unit price': 58.22, 'Quantity': 8, 'Tax 5%': 23.288, 'Total': 489.048, 'Date': '1/27/2019', 'Time': '20:33', 'Payment':
'Ewallet', 'Cost of goods sold': 465.76, 'Gross margin percentage': 4.761904762, 'Gross income': 23.288, 'Customer stratification rating': 8.
4}
Consumed event: {'Invoice ID': '373-73-7910', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Normal', 'Gender': 'Male', 'Product line':
'Sports and travel', 'Unit price': 86.31, 'Quantity': 7, 'Tax 5%': 30.2085, 'Total': 634.3785, 'Date': '2/8/2019', 'Time': '10:37', 'Paymen
t': 'Ewallet', 'Cost of goods sold': 604.17, 'Gross margin percentage': 4.761904762, 'Gross income': 30.2085, 'Customer stratification ratin
g': 5.3}
Consumed event: {'Invoice ID': '355-53-5943', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Female', 'Product line':
'Electronic accessories', 'Unit price': 68.84, 'Quantity': 6, 'Tax 5%': 20.652, 'Total': 433.692, 'Date': '2/25/2019', 'Time': '14:36', 'Paym
ent': 'Ewallet', 'Cost of goods sold': 413.04, 'Gross margin percentage': 4.761904762, 'Gross income': 20.652, 'Customer stratification ratin
g': 5.8}
Consumed event: {'Invoice ID': '665-32-9167', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Female', 'Product line':
'Health and beauty', 'Unit price': 36.26, 'Quantity': 2, 'Tax 5%': 3.626, 'Total': 76.146, 'Date': '1/10/2019', 'Time': '17:15', 'Payment':
'Credit card', 'Cost of goods sold': 72.52, 'Gross margin percentage': 4.761904762, 'Gross income': 3.626, 'Customer stratification rating':
7.2}
```

[ ]:

[ ]:


SupermarketCluster — root@348fa6287f94: / — docker exec -it cassandra_locationa cqlsh — 118×76
nsudh@SUDHEERs-MacBook-Pro-3 SupermarketCluster % docker exec -it cassandra_locationa cqlsh
Connected to RetailStoreCluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> use retailstore;
cqlsh:retailstore> select * from sales_by_branch;
branch | date | time | invoice_id | city | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment | product_line | quantity | tax_5 | total | unit_price
(0 rows)
cqlsh:retailstore> select * from sales_by_branch;
branch | date | time | invoice_id | city | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment | product_line | quantity | tax_5 | total | unit_price
A | 1/10/2019 | 17:15 | 665-32-9167 | Yangon | 72.52 | 7.2 | Member | Female | 3.626 | 4.7619 | Credit card | Health and beauty | 2 | 3.626 | 76.146 | 36.26
A | 1/27/2019 | 20:33 | 123-19-1176 | Yangon | 465.76001 | 8.4 | Member | Male | 23.288 | 4.7619 | Ewallet | Health and beauty | 8 | 23.288 | 489.048 | 58.22
A | 1/5/2019 | 13:08 | 750-67-8428 | Yangon | 522.83002 | 9.1 | Member | Female | 26.1415 | 4.7619 | Ewallet | Health and beauty | 7 | 26.1415 | 548.9715 | 74.69
A | 2/25/2019 | 14:36 | 355-53-5943 | Yangon | 413.04001 | 5.8 | Member | Female | 20.652 | 4.7619 | Ewallet | Electronic accessories | 6 | 20.652 | 433.69199 | 68.84
A | 2/8/2019 | 10:37 | 373-73-7910 | Yangon | 604.16998 | 5.3 | Normal | Male | 30.2085 | 4.7619 | Ewallet | Sports and travel | 7 | 30.2085 | 634.37848 | 86.31
A | 3/3/2019 | 13:23 | 631-41-3108 | Yangon | 324.31 | 7.4 | Normal | Male | 16.2155 | 4.7619 | Credit card | Home and lifestyle | 7 | 16.2155 | 340.52551 | 46.33
(6 rows)
cqlsh:retailstore>


jupyter ConsumerB Last Checkpoint: 6 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[*]:
from confluent_kafka import Consumer
import json
from cassandra.cluster import Cluster
# Kafka configuration
kafka_config = {
'bootstrap.servers': 'localhost:9092',
'group.id': 'consumer_a_group',
'auto.offset.reset': 'earliest',
}
cassandra_cluster = Cluster(['127.0.0.1'])
session = cassandra_cluster.connect()
session.set_keyspace('retailstore')
def consume_events():
consumer = Consumer(kafka_config)
print("checking for the events")
consumer.subscribe(['LocationB'])
while True:
msg = consumer.poll(1.0) # Timeout in seconds
if msg is None:
continue
if msg.error():
print(f"Consumer error: {msg.error()}")
continue
try:
record_value = json.loads(msg.value().decode('utf-8')) # Decode and parse JSON
print(f"Consumed event: {record_value}")
# Insert into Cassandra
session.execute("""
INSERT INTO sales_by_branch(
invoice_id, branch, city, customer_type, gender, product_line,
unit_price, quantity, tax_5, total, date, time, payment,
cost_of_goods_sold, gross_margin_percentage, gross_income, customer_stratification_rating
) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
""", (
record_value['Invoice ID'], record_value['Branch'], record_value['City'],
record_value['Customer type'], record_value['Gender'], record_value['Product line'],
record_value['Unit price'], record_value['Quantity'], record_value['Tax 5%'],
record_value['Total'], record_value['Date'], record_value['Time'],
record_value['Payment'], record_value['Cost of goods sold'],
record_value['Gross margin percentage'], record_value['Gross income'],
record_value['Customer stratification rating']
))
except json.JSONDecodeError as e:
print(f"JSON decoding error: {e}. Message: {msg.value()}")
except Exception as e:
print(f"Error processing message: {e}")
if __name__ == "__main__":
consume_events()
checking for the events
Consumed event: {'Invoice ID': '692-92-5582', 'Branch': 'B', 'City': 'Mandalay', 'Customer type': 'Member', 'Gender': 'Female', 'Product line': 'Food and beverages', 'Unit price': 54.84, 'Quantity': 3, 'Tax 5%': 8.226, 'Total': 172.746, 'Date': '2/20/2019', 'Time': '13:27', 'Payment': 'Credit card', 'Cost of goods sold': 164.52, 'Gross margin percentage': 4.761904762, 'Gross income': 8.226, 'Customer stratification rating': 5.9}
[ ]:


SupermarketCluster — root@348fa6287f94: / — docker exec -it cassandra_locationa cqlsh — 118×76
cqlsh:retailstore> select * from sales_by_branch;
branch | date | time | invoice_id | city | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment | product_line | quantity | tax_5 | total | unit_price
B | 2/20/2019 | 13:27 | 692-92-5582 | Mandalay | 164.52 | 5.9 | Member | Female | 8.226 | 4.7619 | Credit card | Food and beverages | 3 | 8.226 | 172.746 | 54.84
A | 1/10/2019 | 17:15 | 665-32-9167 | Yangon | 72.52 | 7.2 | Member | Female | 3.626 | 4.7619 | Credit card | Health and beauty | 2 | 3.626 | 76.146 | 36.26
A | 1/27/2019 | 20:33 | 123-19-1176 | Yangon | 465.76001 | 8.4 | Member | Male | 23.288 | 4.7619 | Ewallet | Health and beauty | 8 | 23.288 | 489.048 | 58.22
A | 1/5/2019 | 13:08 | 750-67-8428 | Yangon | 522.83002 | 9.1 | Member | Female | 26.1415 | 4.7619 | Ewallet | Health and beauty | 7 | 26.1415 | 548.9715 | 74.69
A | 2/25/2019 | 14:36 | 355-53-5943 | Yangon | 413.04001 | 5.8 | Member | Female | 20.652 | 4.7619 | Ewallet | Electronic accessories | 6 | 20.652 | 433.69199 | 68.84
A | 2/8/2019 | 10:37 | 373-73-7910 | Yangon | 604.16998 | 5.3 | Normal | Male | 30.2085 | 4.7619 | Ewallet | Sports and travel | 7 | 30.2085 | 634.37848 | 86.31
A | 3/3/2019 | 13:23 | 631-41-3108 | Yangon | 324.31 | 7.4 | Normal | Male | 16.2155 | 4.7619 | Credit card | Home and lifestyle | 7 | 16.2155 | 340.52551 | 46.33
(7 rows)
cqlsh:retailstore>

jupyter ConsumerC Last Checkpoint: 15 hours ago

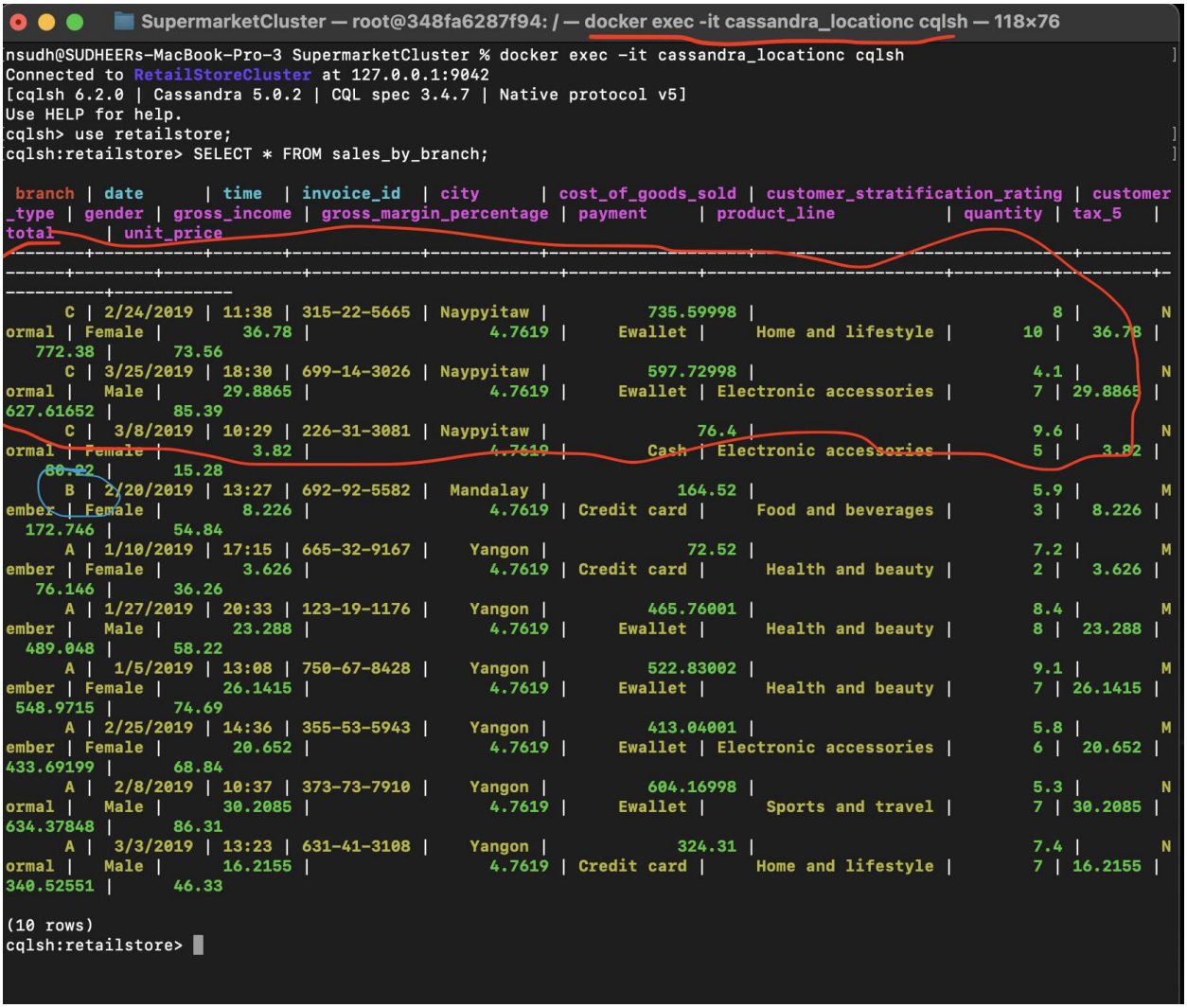
File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python 3 (ipykernel)

[*]:
```
from confluent_kafka import Consumer
import json
from cassandra.cluster import Cluster

# Kafka configuration
kafka_config = {
    'bootstrap.servers': 'localhost:9092',
    'group.id': 'consumer_a_group',
    'auto.offset.reset': 'earliest',
}

# Cassandra configuration
cassandra_cluster = Cluster(['127.0.0.1'])
session = cassandra_cluster.connect()
session.set_keyspace('retailstore')

def consume_events():
    consumer = Consumer(kafka_config)
    print("checking for the events")
    consumer.subscribe(['LocationC'])

    while True:
        msg = consumer.poll(1.0)  # Timeout in seconds
        if msg is None:
            continue
        if msg.error():
            print(f"Consumer error: {msg.error()}")
            continue

        try:
            record_value = json.loads(msg.value().decode('utf-8'))  # Decode and parse JSON
            print(f"Consumed event: {record_value}")

            # Insert into Cassandra
            session.execute("""
                INSERT INTO sales_by_branch(
                    invoice_id, branch, city, customer_type, gender, product_line,
                    unit_price, quantity, tax_5, total, date, time, payment,
                    cost_of_goods_sold, gross_margin_percentage, gross_income, customer_stratification_rating
                ) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
            """, (
                record_value['Invoice ID'], record_value['Branch'], record_value['City'],
                record_value['Customer type'], record_value['Gender'], record_value['Product line'],
                record_value['Unit price'], record_value['Quantity'], record_value['Tax 5%'],
                record_value['Total'], record_value['Date'], record_value['Time'],
                record_value['Payment'], record_value['Cost of goods sold'],
                record_value['Gross margin percentage'], record_value['Gross income'],
                record_value['Customer stratification rating']
            ))
        except json.JSONDecodeError as e:
            print(f"JSON decoding error: {e}. Message: {msg.value()}")
        except Exception as e:
            print(f"Error processing message: {e}")

if __name__ == "__main__":
    consume_events()
```

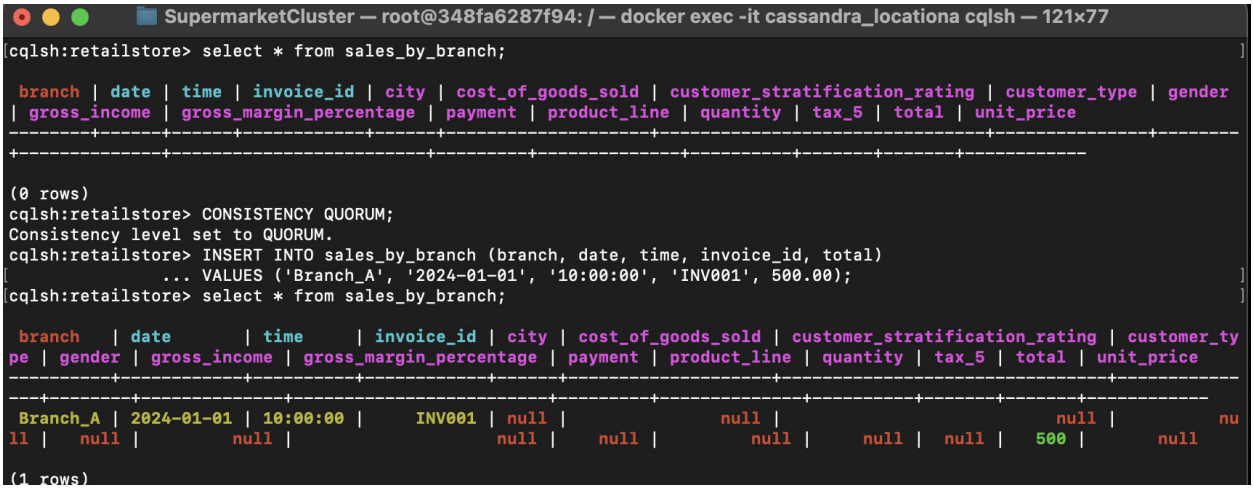
```
checking for the events
Consumed event: {'Invoice ID': '226-31-3081', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Normal', 'Gender': 'Female', 'Product lin
e': 'Electronic accessories', 'Unit price': 15.28, 'Quantity': 5, 'Tax 5%': 3.82, 'Total': 80.22, 'Date': '3/8/2019', 'Time': '10:29', 'Payme
nt': 'Cash', 'Cost of goods sold': 76.4, 'Gross margin percentage': 4.761904762, 'Gross income': 3.82, 'Customer stratification rating': 9.6}
Consumed event: {'Invoice ID': '699-14-3026', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Normal', 'Gender': 'Male', 'Product lin
e': 'Electronic accessories', 'Unit price': 85.39, 'Quantity': 7, 'Tax 5%': 29.8865, 'Total': 627.6165, 'Date': '3/25/2019', 'Time': '18:30',
'Payment': 'Ewallet', 'Cost of goods sold': 597.73, 'Gross margin percentage': 4.761904762, 'Gross income': 29.8865, 'Customer stratification
rating': 4.1}
Consumed event: {'Invoice ID': '315-22-5665', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Normal', 'Gender': 'Female', 'Product lin
e': 'Home and lifestyle', 'Unit price': 73.56, 'Quantity': 10, 'Tax 5%': 36.78, 'Total': 772.38, 'Date': '2/24/2019', 'Time': '11:38', 'Payme
nt': 'Ewallet', 'Cost of goods sold': 735.6, 'Gross margin percentage': 4.761904762, 'Gross income': 36.78, 'Customer stratification rating':
8.0}
```

[ ]:

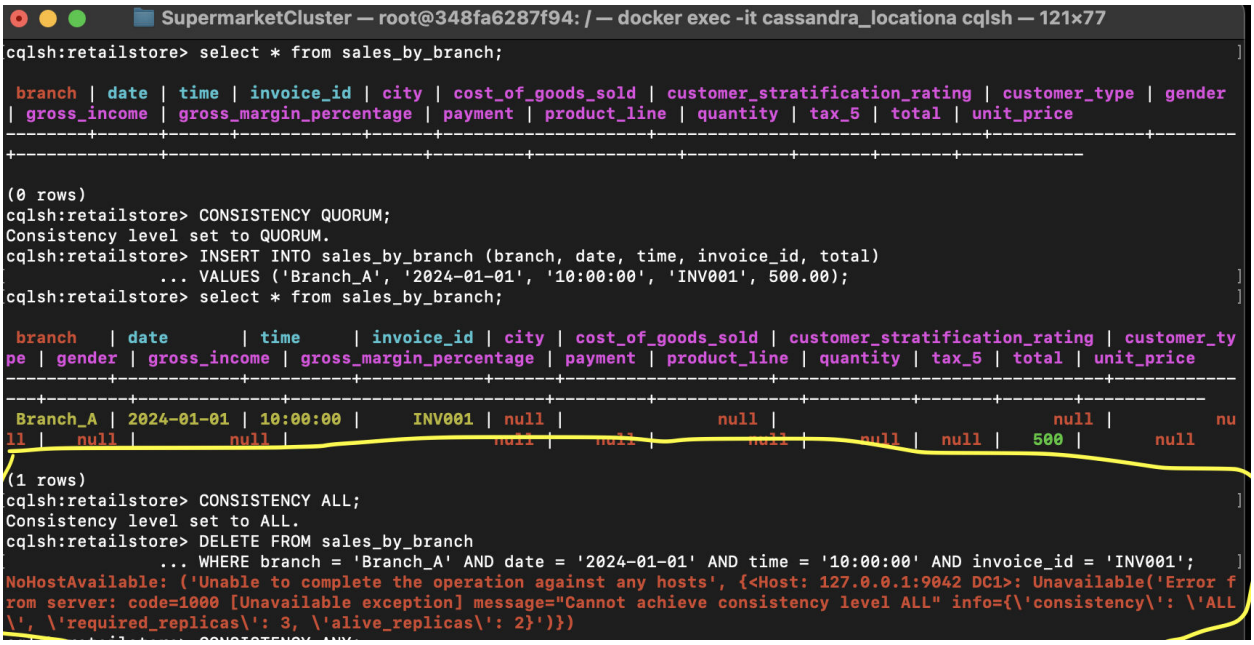

SupermarketCluster — root@348fa6287f94: / — docker exec -it cassandra_locationc cqlsh — 118×76
nsudh@SUDHEERs-MacBook-Pro-3 SupermarketCluster % docker exec -it cassandra_locationc cqlsh
Connected to RetailStoreCluster at 127.0.0.1:9042
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> use retailstore;
cqlsh:retailstore> SELECT * FROM sales_by_branch;
branch | date | time | invoice_id | city | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment | product_line | quantity | tax_5 | total | unit_price
C | 2/24/2019 | 11:38 | 315-22-5665 | Naypyitaw | 735.59998 | 8 | Normal | Female | 36.78 | 4.7619 | Ewallet | Home and lifestyle | 10 | 36.78 | 772.38 | 73.56
C | 3/25/2019 | 18:30 | 699-14-3026 | Naypyitaw | 597.72998 | 4.1 | Normal | Male | 29.8865 | 4.7619 | Ewallet | Electronic accessories | 7 | 29.8865 | 627.61652 | 85.39
C | 3/8/2019 | 10:29 | 226-31-3081 | Naypyitaw | 76.4 | 9.6 | Normal | Female | 3.82 | 4.7619 | Cash | Electronic accessories | 5 | 3.82 | 80.22 | 15.28
B | 2/20/2019 | 13:27 | 692-92-5582 | Mandalay | 164.52 | 5.9 | Member | Female | 8.226 | 4.7619 | Credit card | Food and beverages | 3 | 8.226 | 172.746 | 54.84
A | 1/10/2019 | 17:15 | 665-32-9167 | Yangon | 72.52 | 7.2 | Member | Female | 3.626 | 4.7619 | Credit card | Health and beauty | 2 | 3.626 | 76.146 | 36.26
A | 1/27/2019 | 20:33 | 123-19-1176 | Yangon | 465.76001 | 8.4 | Member | Male | 23.288 | 4.7619 | Ewallet | Health and beauty | 8 | 23.288 | 489.048 | 58.22
A | 1/5/2019 | 13:08 | 750-67-8428 | Yangon | 522.83002 | 9.1 | Member | Female | 26.1415 | 4.7619 | Ewallet | Health and beauty | 7 | 26.1415 | 548.9715 | 74.69
A | 2/25/2019 | 14:36 | 355-53-5943 | Yangon | 413.04001 | 5.8 | Member | Female | 20.652 | 4.7619 | Ewallet | Electronic accessories | 6 | 20.652 | 433.69199 | 68.84
A | 2/8/2019 | 10:37 | 373-73-7910 | Yangon | 604.16998 | 5.3 | Normal | Male | 30.2085 | 4.7619 | Ewallet | Sports and travel | 7 | 30.2085 | 634.37848 | 86.31
A | 3/3/2019 | 13:23 | 631-41-3108 | Yangon | 324.31 | 7.4 | Normal | Male | 16.2155 | 4.7619 | Credit card | Home and lifestyle | 7 | 16.2155 | 340.52551 | 46.33
(10 rows)
cqlsh:retailstore>

Testing consist level all,any,quoram when a node is down by inserting a record into to manually

When Quorum:



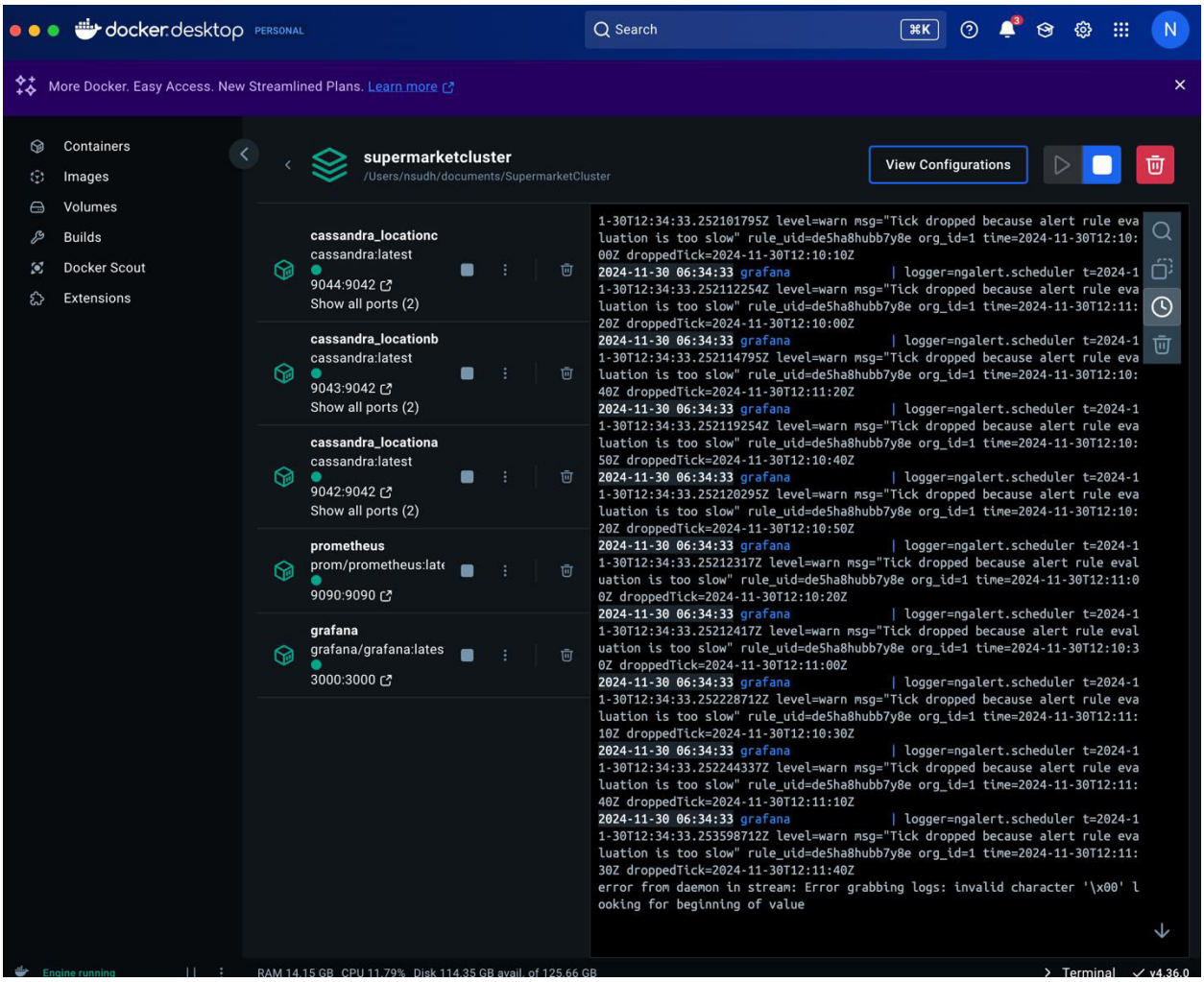
When All:



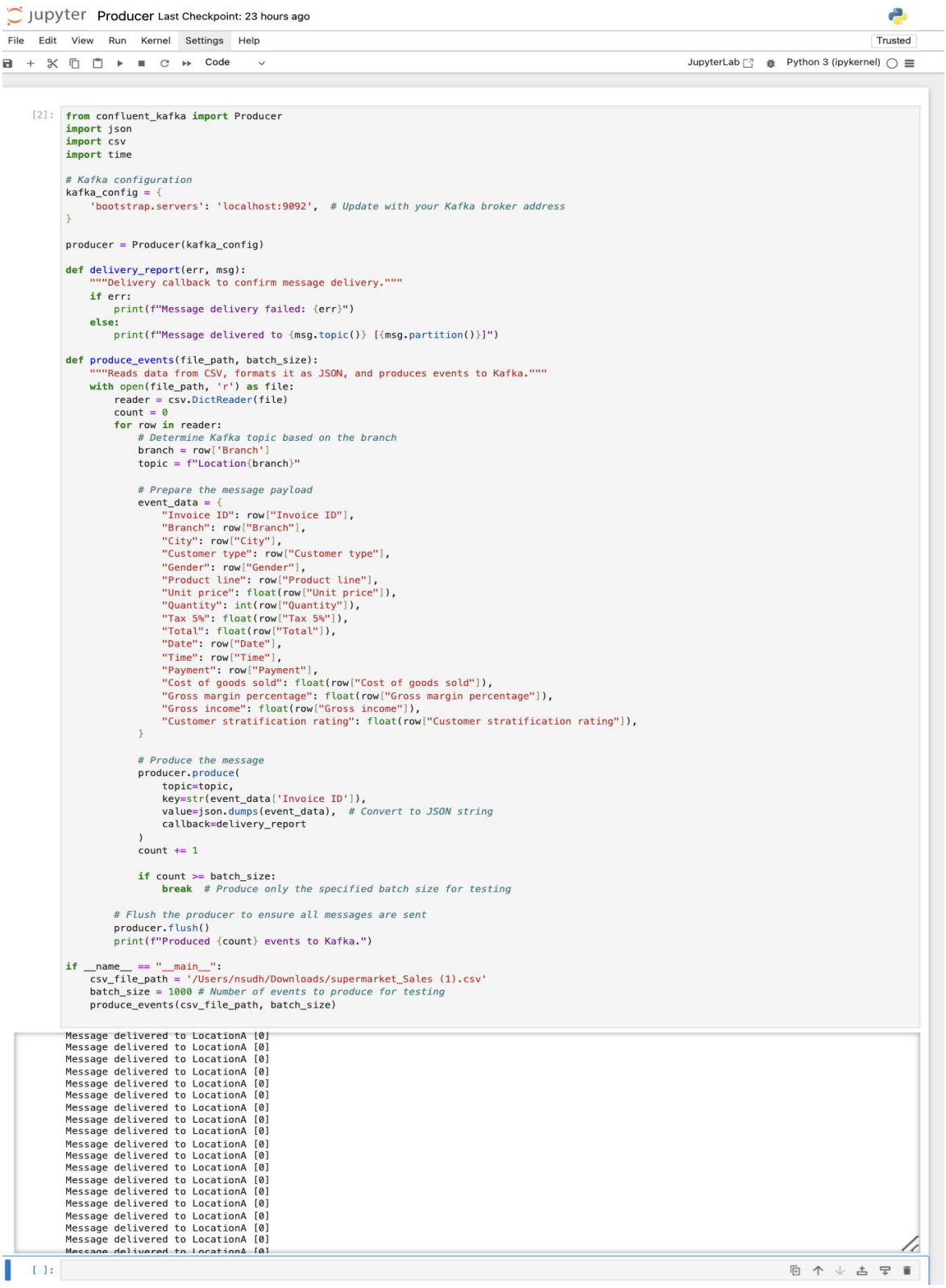
When any:

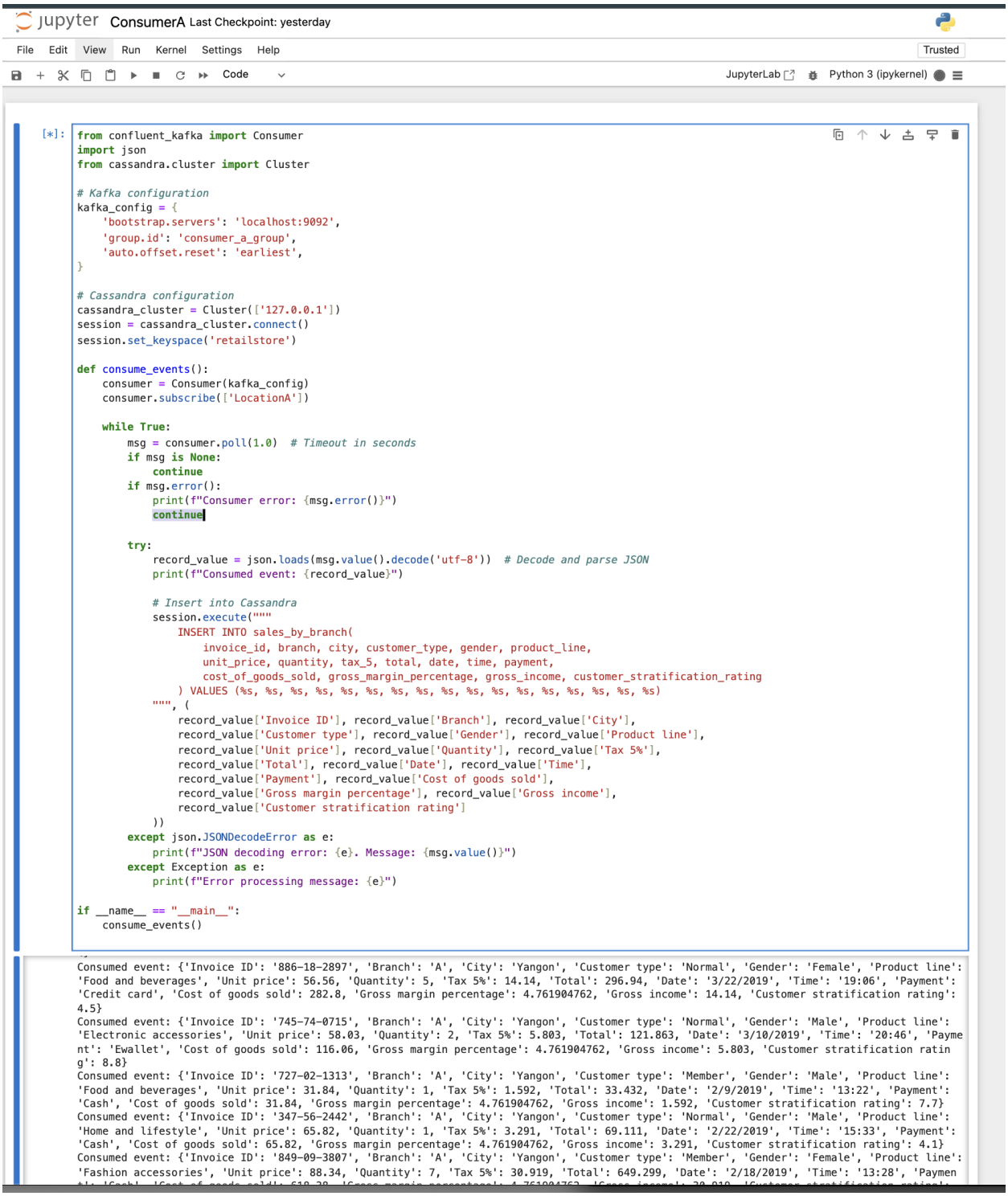
```
cqlsh:retailstore> CONSISTENCY ANY;
Consistency level set to ANY.
cqlsh:retailstore> INSERT INTO sales_by_branch (branch, date, time, invoice_id, total)
               ... VALUES ('Branch_B', '2024-01-01', '11:00:00', 'INV002', 300.00);
cqlsh:retailstore> select * from sales_by_branch;
InvalidRequest: Error from server: code=2200 [Invalid query] message="ANY ConsistencyLevel is only supported for writes"
cqlsh:retailstore> CONSISTENCY ALL;
Consistency level set to ALL.
cqlsh:retailstore> CONSISTENCY QUORUM;
Consistency level set to QUORUM.
cqlsh:retailstore> select * from sales_by_branch;

 branch   | date       | time     | invoice_id | city | cost_of_goods_sold | customer_stratification_rating | customer_ty
pe | gender | gross_income | gross_margin_percentage | payment | product_line | quantity | tax_5 | total | unit_price
----------+------------+----------+------------+------+--------------------+--------------------------------+------------
---+--------+--------------+-------------------------+---------+--------------+----------+-------+-------+------------
 Branch_A | 2024-01-01 | 10:00:00 |     INV001 | null |               null |                           null |          nu
ll |   null |         null |                    null |    null |         null |     null |  null |   500 |       null
 Branch_B | 2024-01-01 | 11:00:00 |     INV002 | null |               null |                           null |          nu
ll |   null |         null |                    null |    null |         null |     null |  null |   300 |       null

(2 rows)
cqlsh:retailstore>
```

Efficient Query Execution:



Loading the Entire Dataset:




jupyter ConsumerA Last Checkpoint: yesterday
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[*]:
from confluent_kafka import Consumer
import json
from cassandra.cluster import Cluster
# Kafka configuration
kafka_config = {
'bootstrap.servers': 'localhost:9092',
'group.id': 'consumer_a_group',
'auto.offset.reset': 'earliest',
}
# Cassandra configuration
cassandra_cluster = Cluster(['127.0.0.1'])
session = cassandra_cluster.connect()
session.set_keyspace('retailstore')
def consume_events():
consumer = Consumer(kafka_config)
consumer.subscribe(['LocationA'])
while True:
msg = consumer.poll(1.0) # Timeout in seconds
if msg is None:
continue
if msg.error():
print(f"Consumer error: {msg.error()}")
continue
try:
record_value = json.loads(msg.value().decode('utf-8')) # Decode and parse JSON
print(f"Consumed event: {record_value}")
# Insert into Cassandra
session.execute("""
INSERT INTO sales_by_branch(
invoice_id, branch, city, customer_type, gender, product_line,
unit_price, quantity, tax_5, total, date, time, payment,
cost_of_goods_sold, gross_margin_percentage, gross_income, customer_stratification_rating
) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
""", (
record_value['Invoice ID'], record_value['Branch'], record_value['City'],
record_value['Customer type'], record_value['Gender'], record_value['Product line'],
record_value['Unit price'], record_value['Quantity'], record_value['Tax 5%'],
record_value['Total'], record_value['Date'], record_value['Time'],
record_value['Payment'], record_value['Cost of goods sold'],
record_value['Gross margin percentage'], record_value['Gross income'],
record_value['Customer stratification rating']
))
except json.JSONDecodeError as e:
print(f"JSON decoding error: {e}. Message: {msg.value()}")
except Exception as e:
print(f"Error processing message: {e}")
if __name__ == "__main__":
consume_events()
Consumed event: {'Invoice ID': '886-18-2897', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Normal', 'Gender': 'Female', 'Product line': 'Food and beverages', 'Unit price': 56.56, 'Quantity': 5, 'Tax 5%': 14.14, 'Total': 296.94, 'Date': '3/22/2019', 'Time': '19:06', 'Payment': 'Credit card', 'Cost of goods sold': 282.8, 'Gross margin percentage': 4.761904762, 'Gross income': 14.14, 'Customer stratification rating': 4.5}
Consumed event: {'Invoice ID': '745-74-0715', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Normal', 'Gender': 'Male', 'Product line': 'Electronic accessories', 'Unit price': 58.03, 'Quantity': 2, 'Tax 5%': 5.803, 'Total': 121.863, 'Date': '3/10/2019', 'Time': '20:46', 'Payment': 'Ewallet', 'Cost of goods sold': 116.06, 'Gross margin percentage': 4.761904762, 'Gross income': 5.803, 'Customer stratification rating': 8.8}
Consumed event: {'Invoice ID': '727-02-1313', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Male', 'Product line': 'Food and beverages', 'Unit price': 31.84, 'Quantity': 1, 'Tax 5%': 1.592, 'Total': 33.432, 'Date': '2/9/2019', 'Time': '13:22', 'Payment': 'Cash', 'Cost of goods sold': 31.84, 'Gross margin percentage': 4.761904762, 'Gross income': 1.592, 'Customer stratification rating': 7.7}
Consumed event: {'Invoice ID': '347-56-2442', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Normal', 'Gender': 'Male', 'Product line': 'Home and lifestyle', 'Unit price': 65.82, 'Quantity': 1, 'Tax 5%': 3.291, 'Total': 69.111, 'Date': '2/22/2019', 'Time': '15:33', 'Payment': 'Cash', 'Cost of goods sold': 65.82, 'Gross margin percentage': 4.761904762, 'Gross income': 3.291, 'Customer stratification rating': 4.1}
Consumed event: {'Invoice ID': '849-09-3807', 'Branch': 'A', 'City': 'Yangon', 'Customer type': 'Member', 'Gender': 'Female', 'Product line': 'Fashion accessories', 'Unit price': 88.34, 'Quantity': 7, 'Tax 5%': 30.919, 'Total': 649.299, 'Date': '2/18/2019', 'Time': '13:28', 'Payment'

jupyter ConsumerB Last Checkpoint: 43 minutes ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python 3 (ipykernel)

[*]:
```python
from confluent_kafka import Consumer
import json
from cassandra.cluster import Cluster

# Kafka configuration
kafka_config = {
    'bootstrap.servers': 'localhost:9092',
    'group.id': 'consumer_a_group',
    'auto.offset.reset': 'earliest',
}


cassandra_cluster = Cluster(['127.0.0.1'])
session = cassandra_cluster.connect()
session.set_keyspace('retailstore')

def consume_events():
    consumer = Consumer(kafka_config)
    print("checking for the events")
    consumer.subscribe(['LocationB'])

    while True:
        msg = consumer.poll(1.0)  # Timeout in seconds
        if msg is None:
            continue
        if msg.error():
            print(f"Consumer error: {msg.error()}")
            continue

        try:
            record_value = json.loads(msg.value().decode('utf-8'))  # Decode and parse JSON
            print(f"Consumed event: {record_value}")

            # Insert into Cassandra
            session.execute("""
                INSERT INTO sales_by_branch(
                    invoice_id, branch, city, customer_type, gender, product_line,
                    unit_price, quantity, tax_5, total, date, time, payment,
                    cost_of_goods_sold, gross_margin_percentage, gross_income, customer_stratification_rating
                ) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
            """, (
                record_value['Invoice ID'], record_value['Branch'], record_value['City'],
                record_value['Customer type'], record_value['Gender'], record_value['Product line'],
                record_value['Unit price'], record_value['Quantity'], record_value['Tax 5%'],
                record_value['Total'], record_value['Date'], record_value['Time'],
                record_value['Payment'], record_value['Cost of goods sold'],
                record_value['Gross margin percentage'], record_value['Gross income'],
                record_value['Customer stratification rating']
            ))
        except json.JSONDecodeError as e:
            print(f"JSON decoding error: {e}. Message: {msg.value()}")
        except Exception as e:
            print(f"Error processing message: {e}")

if __name__ == "__main__":
    consume_events()
```

```
Consumed event: {'Invoice ID': '552-44-5977', 'Branch': 'B', 'City': 'Mandalay', 'Customer type': 'Member', 'Gender': 'Male', 'Product line':
'Health and beauty', 'Unit price': 62.0, 'Quantity': 8, 'Tax 5%': 24.8, 'Total': 520.8, 'Date': '1/3/2019', 'Time': '19:08', 'Payment': 'Cred
it card', 'Cost of goods sold': 496.0, 'Gross margin percentage': 4.761904762, 'Gross income': 24.8, 'Customer stratification rating': 6.2}
Consumed event: {'Invoice ID': '430-53-4718', 'Branch': 'B', 'City': 'Mandalay', 'Customer type': 'Member', 'Gender': 'Male', 'Product line':
'Health and beauty', 'Unit price': 75.37, 'Quantity': 8, 'Tax 5%': 30.148, 'Total': 633.108, 'Date': '1/28/2019', 'Time': '15:46', 'Payment':
'Credit card', 'Cost of goods sold': 602.96, 'Gross margin percentage': 4.761904762, 'Gross income': 30.148, 'Customer stratification ratin
g': 8.4}
Consumed event: {'Invoice ID': '602-16-6955', 'Branch': 'B', 'City': 'Mandalay', 'Customer type': 'Normal', 'Gender': 'Female', 'Product lin
e': 'Sports and travel', 'Unit price': 76.6, 'Quantity': 10, 'Tax 5%': 38.3, 'Total': 804.3, 'Date': '1/24/2019', 'Time': '18:10', 'Payment':
'Ewallet', 'Cost of goods sold': 766.0, 'Gross margin percentage': 4.761904762, 'Gross income': 38.3, 'Customer stratification rating': 6.0}
Consumed event: {'Invoice ID': '690-01-6631', 'Branch': 'B', 'City': 'Mandalay', 'Customer type': 'Normal', 'Gender': 'Male', 'Product line':
'Fashion accessories', 'Unit price': 17.49, 'Quantity': 10, 'Tax 5%': 8.745, 'Total': 183.645, 'Date': '2/22/2019', 'Time': '18:35', 'Paymen
t': 'Ewallet', 'Cost of goods sold': 174.9, 'Gross margin percentage': 4.761904762, 'Gross income': 8.745, 'Customer stratification rating':
6.6}
Consumed event: {'Invoice ID': '303-96-2227', 'Branch': 'B', 'City': 'Mandalay', 'Customer type': 'Normal', 'Gender': 'Female', 'Product lin
e': 'Home and lifestyle', 'Unit price': 97.38, 'Quantity': 10, 'Tax 5%': 48.69, 'Total': 1022.49, 'Date': '3/2/2019', 'Time': '17:16', 'Payme
nt': 'Ewallet', 'Cost of goods sold': 973.8, 'Gross margin percentage': 4.761904762, 'Gross income': 48.69, 'Customer stratification rating':
4.4}
```

jupyter ConsumerC Last Checkpoint: 15 hours ago

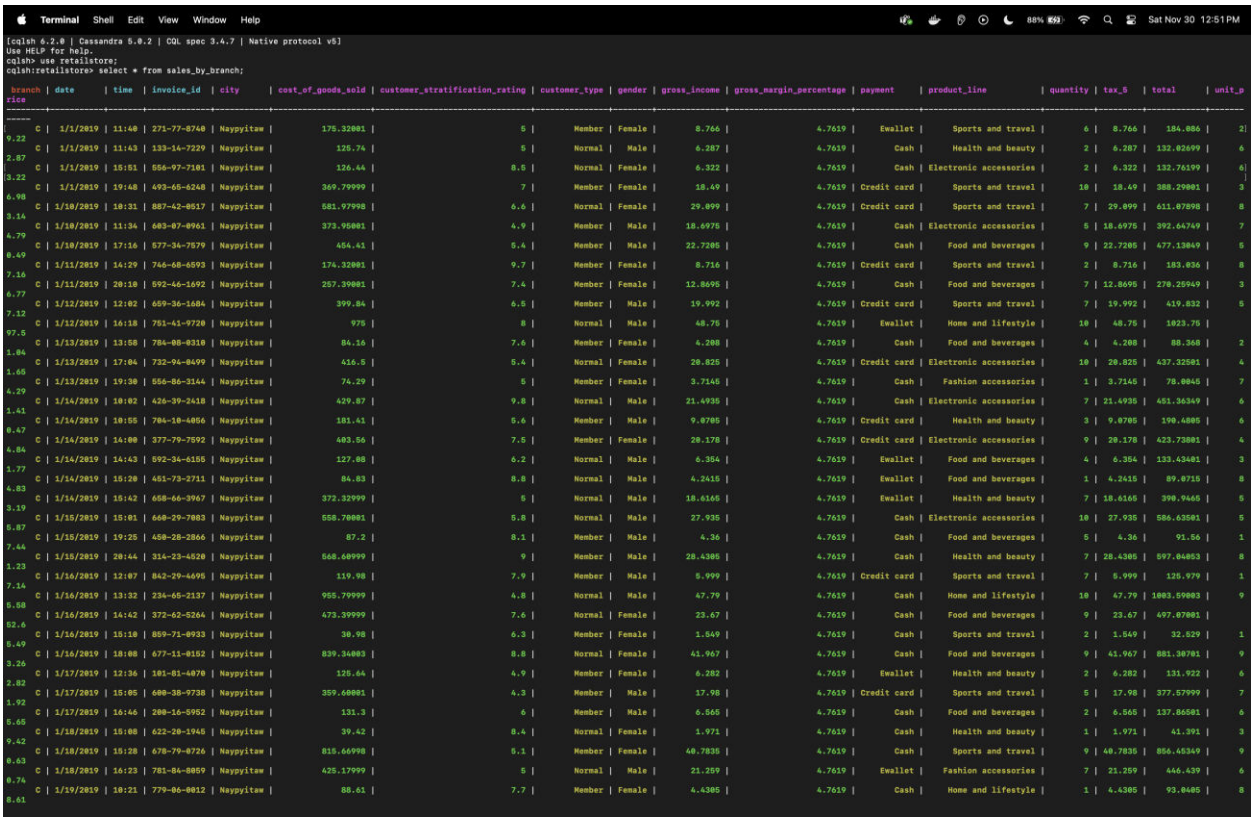
File Edit View Run Kernel Settings Help Trusted

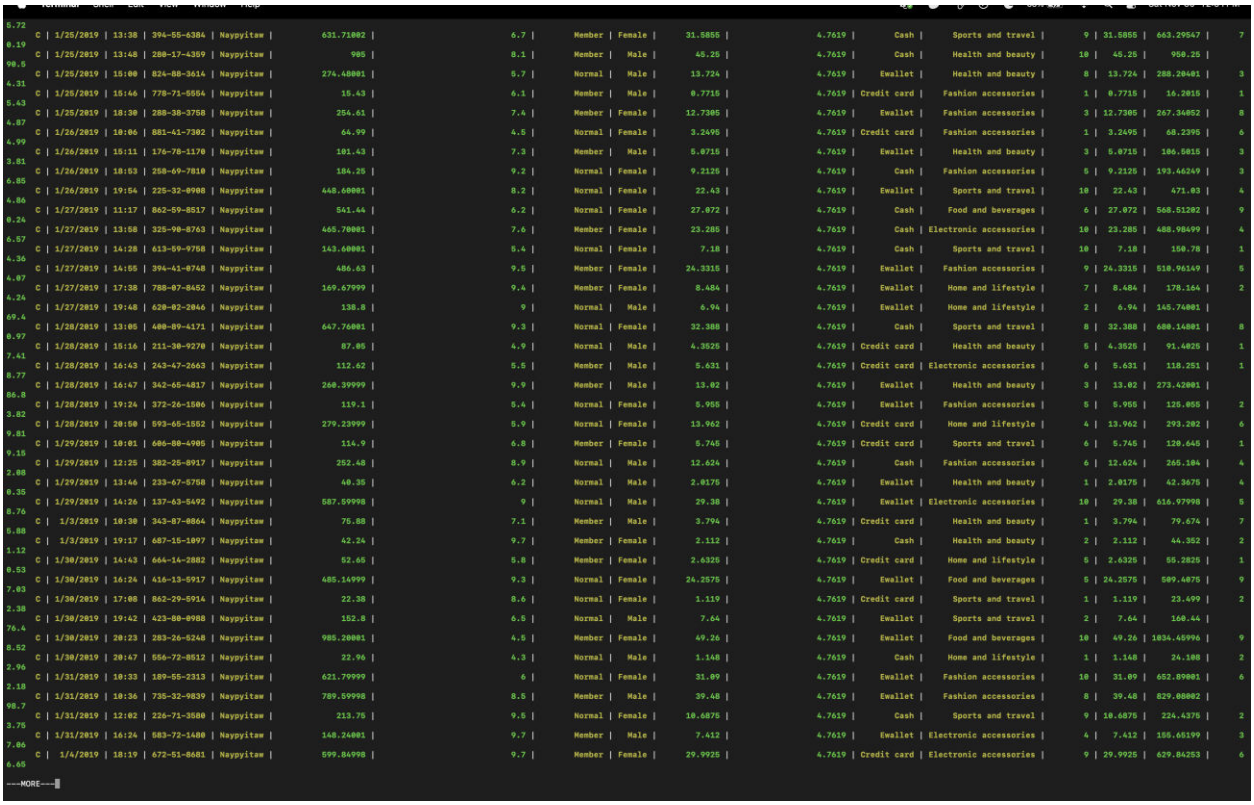
Code JupyterLab Python 3 (ipykernel)

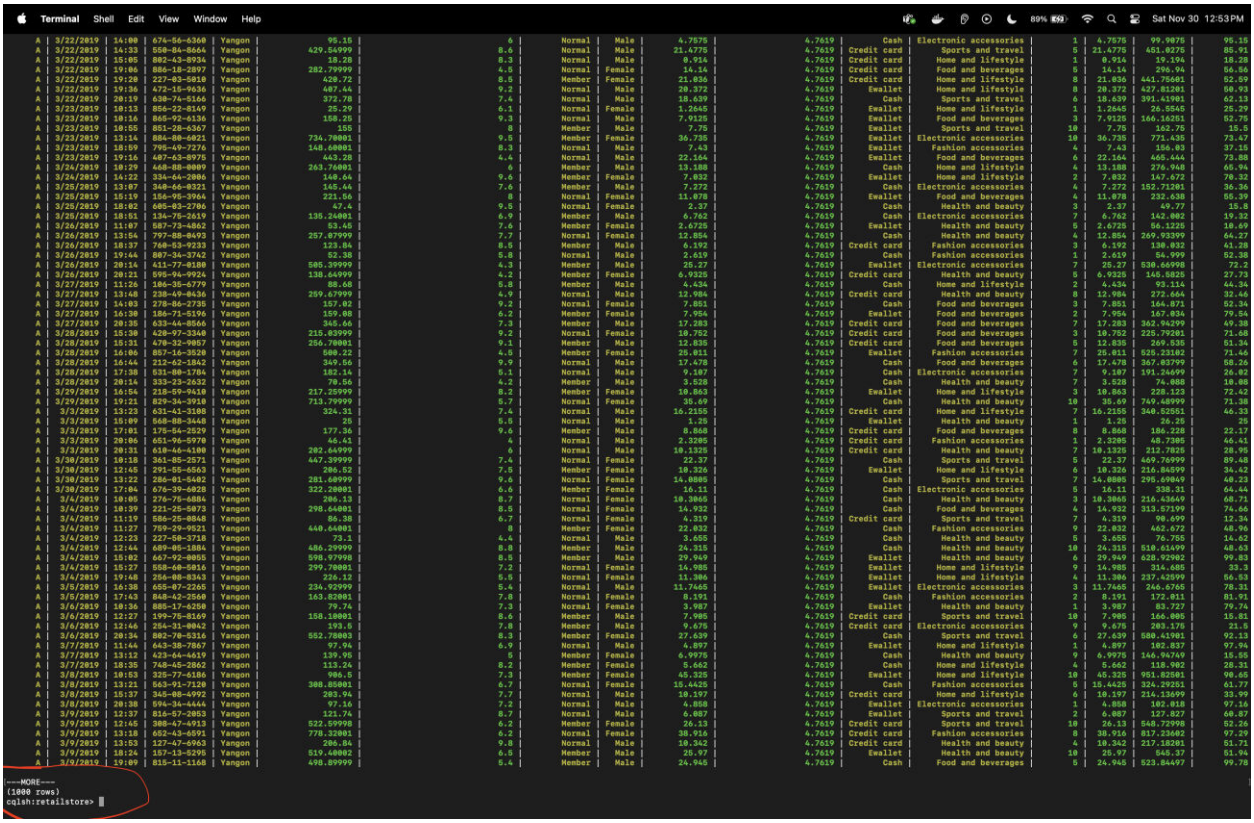
[*]:
```
from confluent_kafka import Consumer
import json
from cassandra.cluster import Cluster

# Kafka configuration
kafka_config = {
    'bootstrap.servers': 'localhost:9092',
    'group.id': 'consumer_a_group',
    'auto.offset.reset': 'earliest',
}

# Cassandra configuration
cassandra_cluster = Cluster(['127.0.0.1'])
session = cassandra_cluster.connect()
session.set_keyspace('retailstore')

def consume_events():
    consumer = Consumer(kafka_config)
    print("checking for the events")
    consumer.subscribe(['LocationC'])

    while True:
        msg = consumer.poll(1.0)  # Timeout in seconds
        if msg is None:
            continue
        if msg.error():
            print(f"Consumer error: {msg.error()}")
            continue

        try:
            record_value = json.loads(msg.value().decode('utf-8'))  # Decode and parse JSON
            print(f"Consumed event: {record_value}")

            # Insert into Cassandra
            session.execute("""
                INSERT INTO sales_by_branch(
                    invoice_id, branch, city, customer_type, gender, product_line,
                    unit_price, quantity, tax_5, total, date, time, payment,
                    cost_of_goods_sold, gross_margin_percentage, gross_income, customer_stratification_rating
                ) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
            """, (
                record_value['Invoice ID'], record_value['Branch'], record_value['City'],
                record_value['Customer type'], record_value['Gender'], record_value['Product line'],
                record_value['Unit price'], record_value['Quantity'], record_value['Tax 5%'],
                record_value['Total'], record_value['Date'], record_value['Time'],
                record_value['Payment'], record_value['Cost of goods sold'],
                record_value['Gross margin percentage'], record_value['Gross income'],
                record_value['Customer stratification rating']
            ))
        except json.JSONDecodeError as e:
            print(f"JSON decoding error: {e}. Message: {msg.value()}")
        except Exception as e:
            print(f"Error processing message: {e}")

if __name__ == "__main__":
    consume_events()
```

```
nt': 'Cash', 'Cost of goods sold': 699.72, 'Gross margin percentage': 4.761904762, 'Gross income': 34.986, 'Customer stratification rating':
6.1}
Consumed event: {'Invoice ID': '189-40-5216', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Normal', 'Gender': 'Male', 'Product lin
e': 'Electronic accessories', 'Unit price': 96.37, 'Quantity': 7, 'Tax 5%': 33.7295, 'Total': 708.3195, 'Date': '1/9/2019', 'Time': '11:40',
'Payment': 'Cash', 'Cost of goods sold': 674.59, 'Gross margin percentage': 4.761904762, 'Gross income': 33.7295, 'Customer stratification ra
ting': 6.0}
Consumed event: {'Invoice ID': '267-62-7380', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Member', 'Gender': 'Male', 'Product lin
e': 'Electronic accessories', 'Unit price': 82.34, 'Quantity': 10, 'Tax 5%': 41.17, 'Total': 864.57, 'Date': '3/29/2019', 'Time': '19:12', 'P
ayment': 'Ewallet', 'Cost of goods sold': 823.4, 'Gross margin percentage': 4.761904762, 'Gross income': 41.17, 'Customer stratification rati
ng': 4.3}
Consumed event: {'Invoice ID': '652-49-6720', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Member', 'Gender': 'Female', 'Product lin
e': 'Electronic accessories', 'Unit price': 60.95, 'Quantity': 1, 'Tax 5%': 3.0475, 'Total': 63.9975, 'Date': '2/18/2019', 'Time': '11:40',
'Payment': 'Ewallet', 'Cost of goods sold': 60.95, 'Gross margin percentage': 4.761904762, 'Gross income': 3.0475, 'Customer stratification r
ating': 5.9}
Consumed event: {'Invoice ID': '233-67-5758', 'Branch': 'C', 'City': 'Naypyitaw', 'Customer type': 'Normal', 'Gender': 'Male', 'Product lin
e': 'Health and beauty', 'Unit price': 40.35, 'Quantity': 1, 'Tax 5%': 2.0175, 'Total': 42.3675, 'Date': '1/29/2019', 'Time': '13:46', 'Payme
nt': 'Ewallet', 'Cost of goods sold': 40.35, 'Gross margin percentage': 4.761904762, 'Gross income': 2.0175, 'Customer stratification ratin
g': 6.2}
```

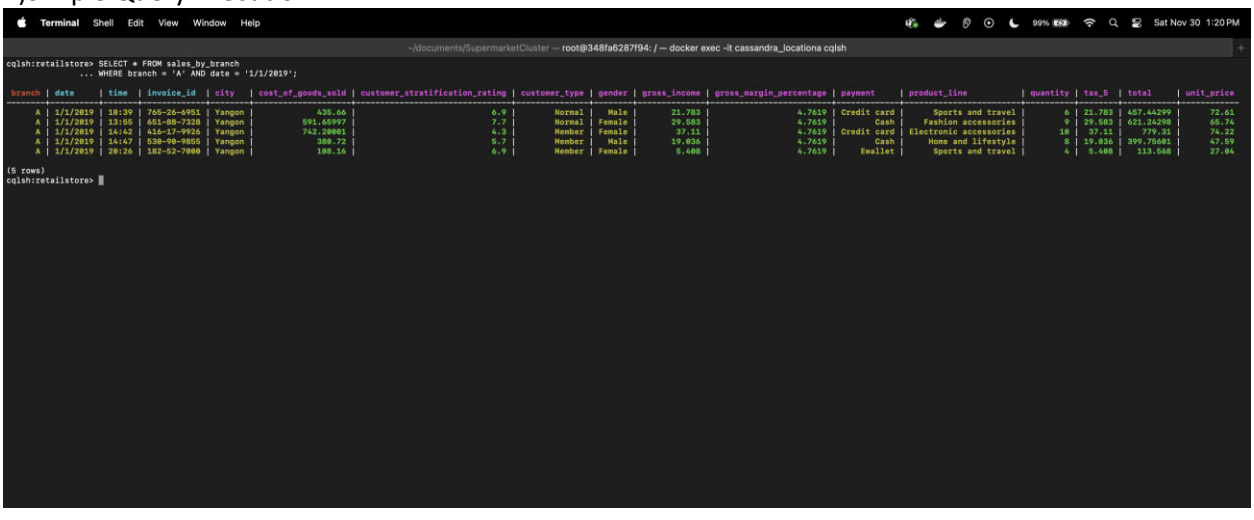
[ ]:

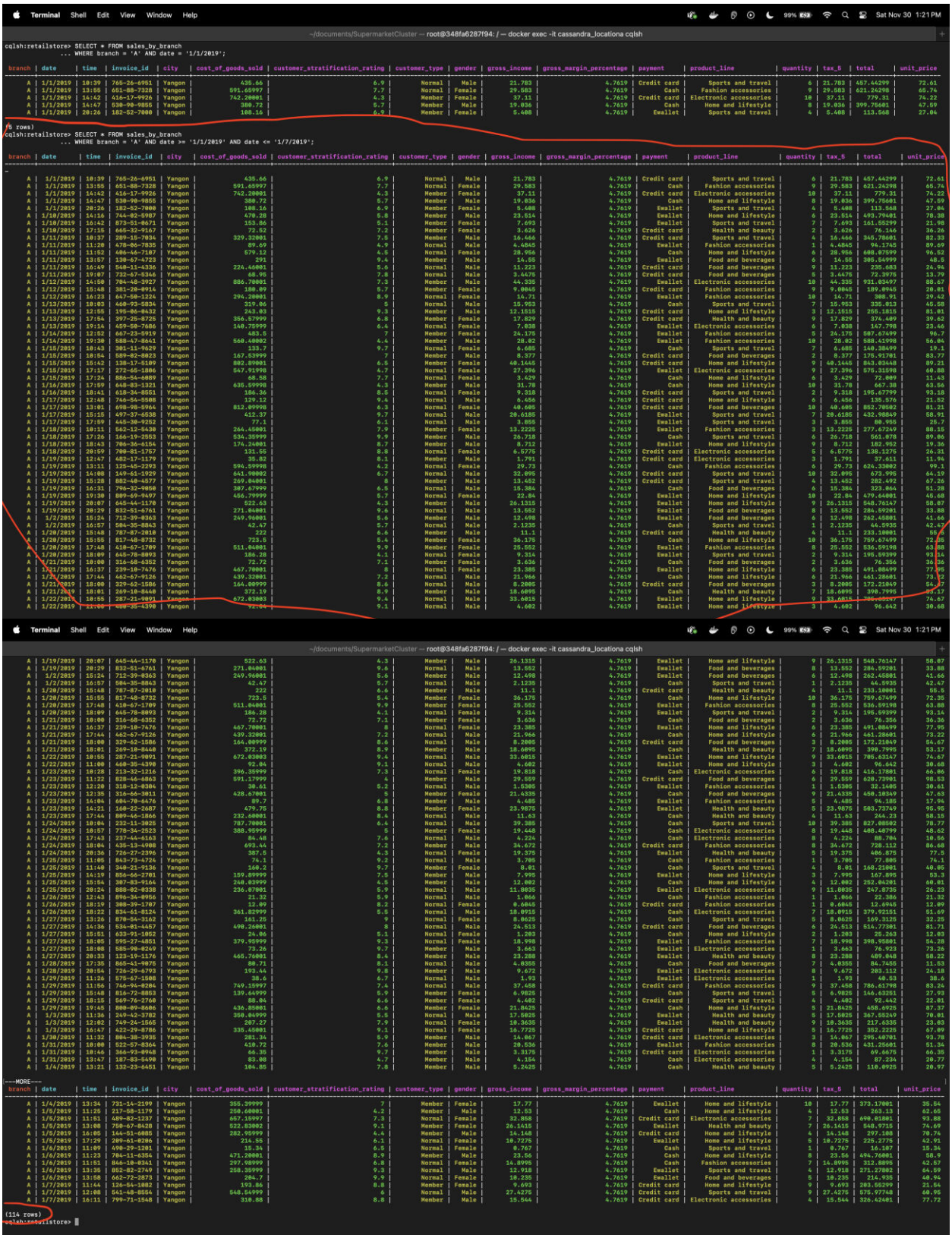

Terminal Shell Edit View Window Help
Sat Nov 30 12:51 PM
[cqlsh 6.2.0 | Cassandra 5.0.2 | CQL spec 3.4.7 | Native protocol v5]
Use HELP for help.
cqlsh> use retailstore;
cqlsh:retailstore> select * from sales_by_branch;
branch | date | time | invoice_id | city | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment | product_line | quantity | tax_5 | total | unit_price
C | 1/1/2019 | 11:40 | 271-77-8740 | Naypyitaw | 175.32001 | 5 | Member | Female | 8.766 | 4.7619 | Ewallet | Sports and travel | 6 | 8.766 | 184.086 | 29.22
C | 1/1/2019 | 11:43 | 133-14-7229 | Naypyitaw | 125.74 | 5 | Normal | Male | 6.287 | 4.7619 | Cash | Health and beauty | 2 | 6.287 | 132.02699 | 62.87
C | 1/1/2019 | 15:51 | 556-97-7101 | Naypyitaw | 126.44 | 8.5 | Normal | Female | 6.322 | 4.7619 | Cash | Electronic accessories | 2 | 6.322 | 132.76199 | 63.22
C | 1/1/2019 | 19:48 | 493-65-6248 | Naypyitaw | 369.79999 | 7 | Member | Female | 18.49 | 4.7619 | Credit card | Sports and travel | 10 | 18.49 | 388.29001 | 36.98
C | 1/10/2019 | 10:31 | 887-42-8517 | Naypyitaw | 581.97998 | 6.6 | Normal | Female | 29.099 | 4.7619 | Credit card | Sports and travel | 7 | 29.099 | 611.07898 | 83.14
C | 1/10/2019 | 11:34 | 603-07-0961 | Naypyitaw | 373.95001 | 4.9 | Member | Male | 18.6975 | 4.7619 | Cash | Electronic accessories | 5 | 18.6975 | 392.64749 | 74.79
C | 1/10/2019 | 17:16 | 577-34-7579 | Naypyitaw | 454.41 | 5.4 | Member | Male | 22.7205 | 4.7619 | Cash | Food and beverages | 9 | 22.7205 | 477.13049 | 50.49
C | 1/11/2019 | 14:29 | 746-68-6593 | Naypyitaw | 174.32001 | 9.7 | Member | Female | 8.716 | 4.7619 | Credit card | Sports and travel | 2 | 8.716 | 183.036 | 87.16
C | 1/11/2019 | 20:18 | 592-46-1692 | Naypyitaw | 257.39001 | 7.4 | Member | Female | 12.8695 | 4.7619 | Cash | Food and beverages | 7 | 12.8695 | 270.25949 | 36.77
C | 1/12/2019 | 12:02 | 659-36-1684 | Naypyitaw | 399.84 | 6.8 | Member | Male | 19.992 | 4.7619 | Credit card | Sports and travel | 7 | 19.992 | 419.832 | 57.12
C | 1/12/2019 | 16:18 | 751-41-9720 | Naypyitaw | 975 | 8 | Normal | Male | 48.75 | 4.7619 | Ewallet | Home and lifestyle | 10 | 48.75 | 1023.75 | 97.5
C | 1/13/2019 | 13:58 | 784-08-0310 | Naypyitaw | 84.16 | 7.6 | Member | Female | 4.208 | 4.7619 | Cash | Food and beverages | 4 | 4.208 | 88.368 | 21.04
C | 1/13/2019 | 17:04 | 732-94-8499 | Naypyitaw | 416.5 | 5.4 | Normal | Female | 20.825 | 4.7619 | Credit card | Electronic accessories | 10 | 20.825 | 437.32501 | 41.65
C | 1/13/2019 | 19:30 | 556-86-3144 | Naypyitaw | 74.29 | 5 | Member | Female | 3.7145 | 4.7619 | Cash | Fashion accessories | 1 | 3.7145 | 78.0046 | 74.29
C | 1/14/2019 | 10:02 | 426-39-2418 | Naypyitaw | 429.87 | 9.8 | Normal | Male | 21.4935 | 4.7619 | Cash | Electronic accessories | 7 | 21.4935 | 451.36349 | 61.41
C | 1/14/2019 | 10:55 | 704-10-4056 | Naypyitaw | 181.41 | 5.6 | Member | Male | 9.0705 | 4.7619 | Credit card | Health and beauty | 3 | 9.0705 | 190.4805 | 60.47
C | 1/14/2019 | 14:00 | 377-79-7592 | Naypyitaw | 403.56 | 7.5 | Member | Female | 20.178 | 4.7619 | Credit card | Electronic accessories | 9 | 20.178 | 423.73801 | 44.84
C | 1/14/2019 | 14:43 | 592-34-6155 | Naypyitaw | 127.08 | 6.2 | Normal | Male | 6.354 | 4.7619 | Ewallet | Food and beverages | 4 | 6.354 | 133.43401 | 31.77
C | 1/14/2019 | 15:20 | 451-73-2711 | Naypyitaw | 84.83 | 8.8 | Normal | Male | 4.2415 | 4.7619 | Ewallet | Food and beverages | 1 | 4.2415 | 89.0715 | 84.83
C | 1/14/2019 | 15:42 | 658-66-3967 | Naypyitaw | 372.32999 | 5 | Normal | Male | 18.6165 | 4.7619 | Ewallet | Health and beauty | 7 | 18.6165 | 390.9465 | 53.19
C | 1/15/2019 | 15:01 | 660-29-7083 | Naypyitaw | 558.70001 | 5.8 | Normal | Male | 27.935 | 4.7619 | Cash | Electronic accessories | 10 | 27.935 | 586.63501 | 55.87
C | 1/15/2019 | 19:25 | 450-28-2866 | Naypyitaw | 87.2 | 8.1 | Member | Male | 4.36 | 4.7619 | Cash | Food and beverages | 5 | 4.36 | 91.56 | 17.44
C | 1/16/2019 | 20:44 | 314-23-4520 | Naypyitaw | 568.60999 | 9 | Member | Male | 28.4305 | 4.7619 | Cash | Health and beauty | 7 | 28.4305 | 597.04053 | 81.23
C | 1/16/2019 | 12:07 | 842-29-4695 | Naypyitaw | 119.98 | 7.9 | Member | Male | 5.999 | 4.7619 | Credit card | Sports and travel | 7 | 5.999 | 125.979 | 17.14
C | 1/16/2019 | 13:32 | 234-65-2137 | Naypyitaw | 955.79999 | 4.8 | Normal | Male | 47.79 | 4.7619 | Cash | Home and lifestyle | 10 | 47.79 | 1003.59003 | 95.58
C | 1/16/2019 | 14:42 | 372-62-5264 | Naypyitaw | 473.39999 | 7.6 | Normal | Female | 23.67 | 4.7619 | Cash | Food and beverages | 9 | 23.67 | 497.07001 | 52.6
C | 1/16/2019 | 15:10 | 859-71-0933 | Naypyitaw | 30.98 | 6.3 | Member | Female | 1.549 | 4.7619 | Cash | Sports and travel | 2 | 1.549 | 32.529 | 15.49
C | 1/16/2019 | 18:00 | 677-11-8152 | Naypyitaw | 839.34003 | 8.8 | Normal | Female | 41.967 | 4.7619 | Cash | Food and beverages | 9 | 41.967 | 881.30701 | 93.26
C | 1/17/2019 | 12:36 | 101-81-4070 | Naypyitaw | 125.64 | 4.9 | Member | Female | 6.282 | 4.7619 | Ewallet | Health and beauty | 2 | 6.282 | 131.922 | 62.82
C | 1/17/2019 | 15:05 | 600-38-9738 | Naypyitaw | 359.60001 | 4.3 | Member | Male | 17.98 | 4.7619 | Credit card | Sports and travel | 5 | 17.98 | 377.57999 | 71.92
C | 1/17/2019 | 16:46 | 200-16-5952 | Naypyitaw | 131.3 | 6 | Member | Male | 6.565 | 4.7619 | Cash | Food and beverages | 2 | 6.565 | 137.86501 | 65.65
C | 1/18/2019 | 15:08 | 622-20-1948 | Naypyitaw | 39.42 | 8.4 | Normal | Female | 1.971 | 4.7619 | Cash | Health and beauty | 1 | 1.971 | 41.391 | 39.42
C | 1/18/2019 | 15:28 | 678-79-8726 | Naypyitaw | 815.66998 | 5.1 | Member | Female | 40.7835 | 4.7619 | Cash | Sports and travel | 9 | 40.7835 | 856.46349 | 90.63
C | 1/18/2019 | 16:23 | 781-84-8859 | Naypyitaw | 425.17999 | 5 | Normal | Male | 21.259 | 4.7619 | Ewallet | Fashion accessories | 7 | 21.259 | 446.439 | 60.74
C | 1/19/2019 | 10:21 | 779-06-0012 | Naypyitaw | 88.61 | 7.7 | Member | Female | 4.4305 | 4.7619 | Cash | Home and lifestyle | 1 | 4.4305 | 93.0405 | 88.61


C | 1/25/2019 | 13:38 | 394-55-6384 | Naypyitaw | 631.71002 | 6.7 | Member | Female | 31.5855 | 4.7619 | Cash | Sports and travel | 9 | 31.5855 | 663.29547 | 70.19
C | 1/25/2019 | 13:48 | 288-17-4359 | Naypyitaw | 905 | 8.1 | Member | Male | 45.25 | 4.7619 | Cash | Health and beauty | 10 | 45.25 | 950.25 | 90.5
C | 1/25/2019 | 15:00 | 824-68-3614 | Naypyitaw | 274.48001 | 5.7 | Normal | Male | 13.724 | 4.7619 | Ewallet | Health and beauty | 8 | 13.724 | 288.20401 | 34.31
C | 1/25/2019 | 15:46 | 778-71-5554 | Naypyitaw | 15.43 | 6.1 | Member | Male | 0.7715 | 4.7619 | Credit card | Fashion accessories | 1 | 0.7715 | 16.2015 | 15.43
C | 1/25/2019 | 18:30 | 288-38-3758 | Naypyitaw | 254.61 | 7.4 | Member | Female | 12.7305 | 4.7619 | Ewallet | Fashion accessories | 3 | 12.7305 | 267.34052 | 84.87
C | 1/26/2019 | 10:06 | 881-41-7302 | Naypyitaw | 64.99 | 4.8 | Normal | Female | 3.2495 | 4.7619 | Credit card | Fashion accessories | 1 | 3.2495 | 68.2395 | 64.99
C | 1/26/2019 | 15:11 | 176-78-1170 | Naypyitaw | 101.43 | 7.3 | Member | Male | 5.0715 | 4.7619 | Ewallet | Health and beauty | 3 | 5.0715 | 106.5015 | 33.81
C | 1/26/2019 | 18:53 | 258-69-7810 | Naypyitaw | 184.25 | 9.2 | Normal | Female | 9.2125 | 4.7619 | Cash | Fashion accessories | 5 | 9.2125 | 193.46249 | 36.85
C | 1/26/2019 | 19:54 | 225-32-0998 | Naypyitaw | 448.60001 | 8.2 | Normal | Female | 22.43 | 4.7619 | Ewallet | Sports and travel | 10 | 22.43 | 471.03 | 44.86
C | 1/27/2019 | 11:17 | 862-59-8517 | Naypyitaw | 541.44 | 6.2 | Normal | Female | 27.072 | 4.7619 | Cash | Food and beverages | 6 | 27.072 | 568.51202 | 90.24
C | 1/27/2019 | 13:58 | 325-90-8763 | Naypyitaw | 465.78001 | 7.6 | Member | Female | 23.285 | 4.7619 | Cash | Electronic accessories | 10 | 23.285 | 488.98499 | 46.57
C | 1/27/2019 | 14:28 | 613-59-9758 | Naypyitaw | 143.60001 | 5.4 | Normal | Female | 7.18 | 4.7619 | Cash | Sports and travel | 10 | 7.18 | 150.78 | 14.36
C | 1/27/2019 | 14:55 | 394-41-0748 | Naypyitaw | 486.63 | 9.6 | Member | Female | 24.3315 | 4.7619 | Ewallet | Fashion accessories | 9 | 24.3315 | 510.96149 | 54.07
C | 1/27/2019 | 17:38 | 788-07-8452 | Naypyitaw | 169.67999 | 9.4 | Member | Female | 8.484 | 4.7619 | Ewallet | Home and lifestyle | 7 | 8.484 | 178.164 | 24.24
C | 1/27/2019 | 19:48 | 620-02-2046 | Naypyitaw | 138.8 | 9 | Normal | Male | 6.94 | 4.7619 | Ewallet | Home and lifestyle | 2 | 6.94 | 145.74001 | 69.4
C | 1/28/2019 | 13:05 | 480-89-4171 | Naypyitaw | 647.76001 | 9.3 | Normal | Female | 32.388 | 4.7619 | Cash | Sports and travel | 8 | 32.388 | 680.14801 | 80.97
C | 1/28/2019 | 15:16 | 211-30-9270 | Naypyitaw | 87.05 | 4.9 | Normal | Male | 4.3525 | 4.7619 | Credit card | Health and beauty | 5 | 4.3525 | 91.4025 | 17.41
C | 1/28/2019 | 16:43 | 243-47-2663 | Naypyitaw | 112.62 | 5.5 | Member | Male | 5.631 | 4.7619 | Credit card | Electronic accessories | 6 | 5.631 | 118.251 | 18.77
C | 1/28/2019 | 16:47 | 342-65-4817 | Naypyitaw | 260.39999 | 9.9 | Member | Male | 13.02 | 4.7619 | Ewallet | Health and beauty | 3 | 13.02 | 273.42001 | 86.8
C | 1/28/2019 | 19:24 | 372-26-1506 | Naypyitaw | 119.1 | 5.4 | Normal | Female | 5.955 | 4.7619 | Ewallet | Fashion accessories | 5 | 5.955 | 125.055 | 23.82
C | 1/28/2019 | 20:50 | 593-65-1552 | Naypyitaw | 279.23999 | 5.9 | Normal | Female | 13.962 | 4.7619 | Credit card | Home and lifestyle | 4 | 13.962 | 293.202 | 69.81
C | 1/29/2019 | 10:01 | 606-80-4985 | Naypyitaw | 114.9 | 6.8 | Member | Female | 5.745 | 4.7619 | Credit card | Sports and travel | 6 | 5.745 | 120.645 | 19.15
C | 1/29/2019 | 12:25 | 382-25-8917 | Naypyitaw | 252.48 | 8.9 | Normal | Male | 12.624 | 4.7619 | Cash | Fashion accessories | 6 | 12.624 | 265.104 | 42.08
C | 1/29/2019 | 13:46 | 233-67-5758 | Naypyitaw | 40.35 | 6.2 | Normal | Male | 2.0175 | 4.7619 | Ewallet | Health and beauty | 1 | 2.0175 | 42.3675 | 40.35
C | 1/29/2019 | 14:26 | 137-63-5492 | Naypyitaw | 587.59998 | 9 | Normal | Male | 29.38 | 4.7619 | Ewallet | Electronic accessories | 10 | 29.38 | 616.97998 | 58.76
C | 1/3/2019 | 10:30 | 343-87-0864 | Naypyitaw | 75.88 | 7.1 | Member | Male | 3.794 | 4.7619 | Credit card | Health and beauty | 1 | 3.794 | 79.674 | 75.88
C | 1/3/2019 | 19:17 | 687-15-1097 | Naypyitaw | 42.24 | 9.7 | Member | Female | 2.112 | 4.7619 | Cash | Health and beauty | 2 | 2.112 | 44.352 | 21.12
C | 1/30/2019 | 14:43 | 664-14-2882 | Naypyitaw | 52.65 | 5.8 | Member | Female | 2.6325 | 4.7619 | Credit card | Home and lifestyle | 5 | 2.6325 | 55.2825 | 10.53
C | 1/30/2019 | 16:24 | 416-13-5917 | Naypyitaw | 485.14999 | 9.3 | Normal | Female | 24.2575 | 4.7619 | Ewallet | Food and beverages | 5 | 24.2575 | 509.4075 | 97.03
C | 1/30/2019 | 17:08 | 862-29-5914 | Naypyitaw | 22.38 | 8.6 | Normal | Female | 1.119 | 4.7619 | Credit card | Sports and travel | 1 | 1.119 | 23.499 | 22.38
C | 1/30/2019 | 19:42 | 423-80-6988 | Naypyitaw | 152.8 | 6.8 | Normal | Male | 7.64 | 4.7619 | Ewallet | Sports and travel | 2 | 7.64 | 160.44 | 76.4
C | 1/30/2019 | 20:23 | 283-26-5248 | Naypyitaw | 985.20001 | 4.5 | Member | Female | 49.26 | 4.7619 | Ewallet | Food and beverages | 10 | 49.26 | 1034.46996 | 98.52
C | 1/30/2019 | 20:47 | 556-72-8612 | Naypyitaw | 22.96 | 4.3 | Normal | Male | 1.148 | 4.7619 | Cash | Home and lifestyle | 1 | 1.148 | 24.108 | 22.96
C | 1/31/2019 | 10:33 | 189-55-2313 | Naypyitaw | 621.79999 | 6 | Normal | Female | 31.09 | 4.7619 | Ewallet | Fashion accessories | 10 | 31.09 | 652.89001 | 62.18
C | 1/31/2019 | 10:36 | 735-32-9839 | Naypyitaw | 789.59998 | 8.5 | Member | Male | 39.48 | 4.7619 | Ewallet | Fashion accessories | 8 | 39.48 | 829.08002 | 98.7
C | 1/31/2019 | 12:02 | 226-71-3580 | Naypyitaw | 213.75 | 9.6 | Normal | Female | 10.6875 | 4.7619 | Cash | Sports and travel | 9 | 10.6875 | 224.4375 | 23.75
C | 1/31/2019 | 16:24 | 583-72-1480 | Naypyitaw | 148.24001 | 9.7 | Member | Male | 7.412 | 4.7619 | Ewallet | Electronic accessories | 4 | 7.412 | 155.65199 | 37.06
C | 1/4/2019 | 18:19 | 672-51-8681 | Naypyitaw | 599.84998 | 9.7 | Member | Female | 29.9925 | 4.7619 | Credit card | Electronic accessories | 9 | 29.9925 | 629.84253 | 66.65
---MORE---



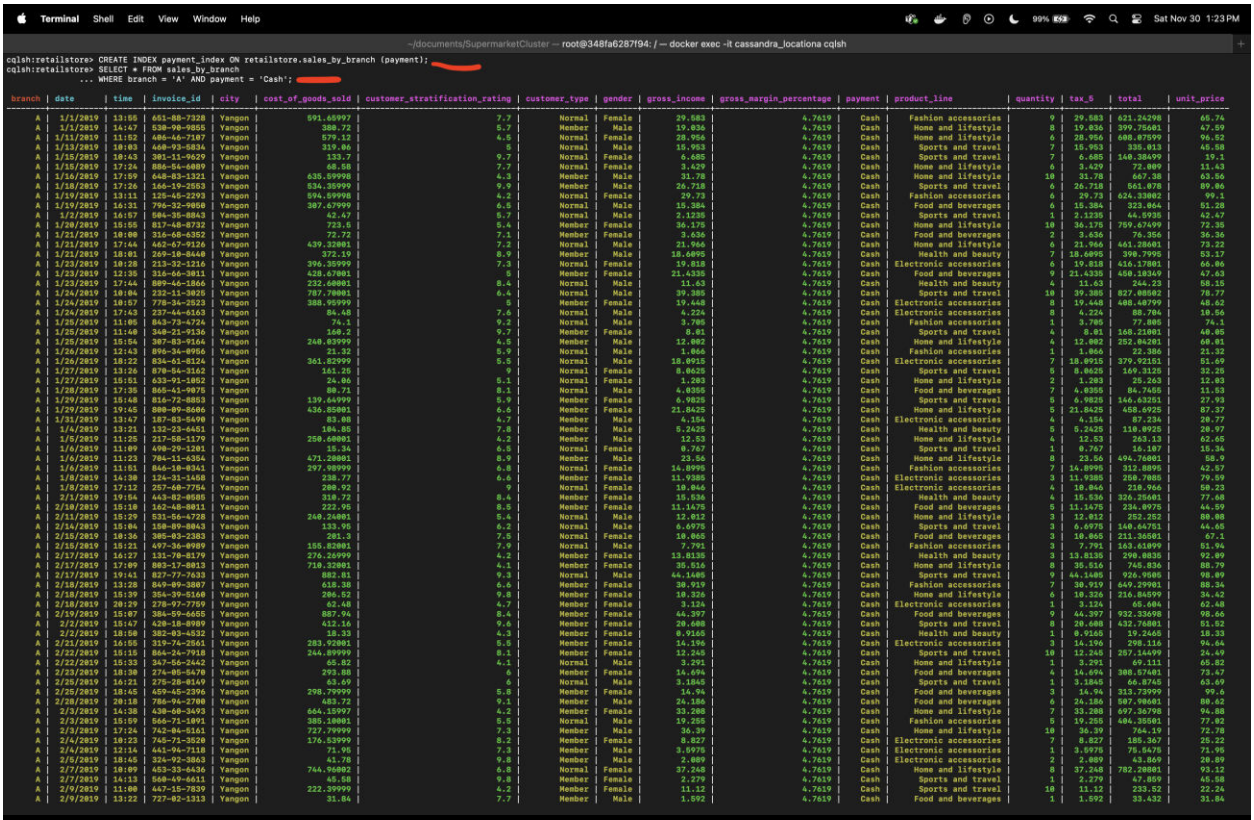
1)Simple Query Execution:



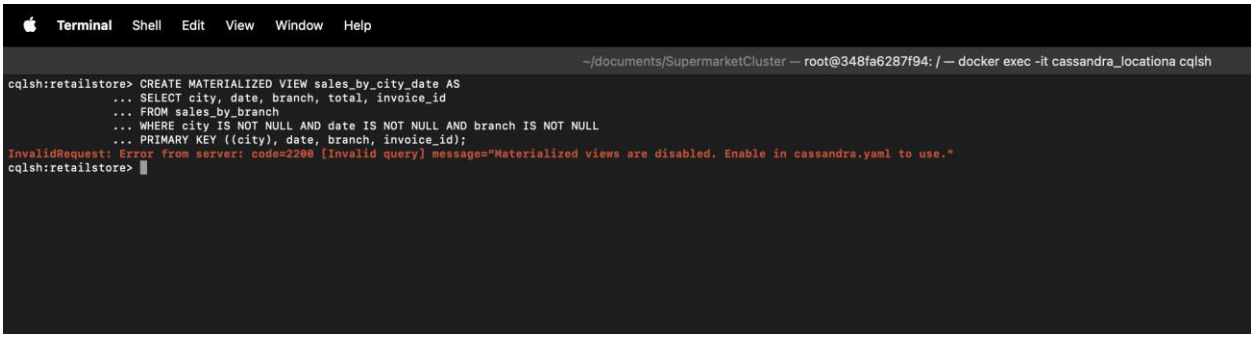


```
cqlsh:retailstore> SELECT * FROM sales_by_branch
               ... WHERE branch = 'A' AND date = '1/1/2019';

 branch | date     | time  | invoice_id  | city   | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment     | product_line           | quantity | tax_5  | total     | unit_price
--------+----------+-------+-------------+--------+--------------------+--------------------------------+---------------+--------+--------------+-------------------------+-------------+------------------------+----------+--------+-----------+------------
      A | 1/1/2019 | 10:39 | 765-26-6951 | Yangon |             435.66 |                            6.9 |        Normal |   Male |      21.783 |                  4.7619 | Credit card |  Sports and travel |        6 | 21.783 | 457.44299 |      72.61
      A | 1/1/2019 | 13:55 | 651-88-7320 | Yangon |             591.6597 |                            7.7 |        Normal | Female |      29.583 |                  4.7619 |        Cash | Fashion accessories |        9 | 29.583 | 621.24298 |      65.74
      A | 1/1/2019 | 14:42 | 416-17-9926 | Yangon |             742.20001 |                            4.3 |        Member | Female |       37.11 |                  4.7619 | Credit card | Electronic accessories |       10 |  37.11 |    779.31 |      74.22
      A | 1/1/2019 | 14:47 | 530-90-9855 | Yangon |             388.72 |                            5.7 |        Member |   Male |      19.036 |                  4.7619 |        Cash |  Home and lifestyle |        8 | 19.036 | 399.75601 |      47.59
      A | 1/1/2019 | 20:26 | 182-52-7000 | Yangon |             108.16 |                            6.9 |        Member | Female |       5.408 |                  4.7619 |     Ewallet |  Sports and travel |        4 |  5.408 |   113.568 |      27.04

(5 rows)
cqlsh:retailstore> SELECT * FROM sales_by_branch
               ... WHERE branch = 'A' AND date >= '1/1/2019' AND date <= '1/7/2019';

 branch | date     | time  | invoice_id  | city   | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment     | product_line           | quantity | tax_5  | total     | unit_price
--------+----------+-------+-------------+--------+--------------------+--------------------------------+---------------+--------+--------------+-------------------------+-------------+------------------------+----------+--------+-----------+------------
-
      A | 1/1/2019 | 10:39 | 765-26-6951 | Yangon |             435.66 |                            6.9 |        Normal |   Male |      21.783 |                  4.7619 | Credit card |  Sports and travel |        6 | 21.783 | 457.44299 |      72.61
      A | 1/1/2019 | 13:55 | 651-88-7320 | Yangon |             591.6597 |                            7.7 |        Normal | Female |      29.583 |                  4.7619 |        Cash | Fashion accessories |        9 | 29.583 | 621.24298 |      65.74
      A | 1/1/2019 | 14:42 | 416-17-9926 | Yangon |             742.20001 |                            4.3 |        Member | Female |       37.11 |                  4.7619 | Credit card | Electronic accessories |       10 |  37.11 |    779.31 |      74.22
      A | 1/1/2019 | 14:47 | 530-90-9855 | Yangon |             388.72 |                            5.7 |        Member |   Male |      19.036 |                  4.7619 |        Cash |  Home and lifestyle |        8 | 19.036 | 399.75601 |      47.59
      A | 1/1/2019 | 20:26 | 182-52-7000 | Yangon |             108.16 |                            6.9 |        Member | Female |       5.408 |                  4.7619 |     Ewallet |  Sports and travel |        4 |  5.408 |   113.568 |      27.04
      A | 1/10/2019 | 14:16 | 744-02-5987 | Yangon |             470.28 |                            5.8 |        Member |   Male |      23.514 |                  4.7619 |     Ewallet |  Home and lifestyle |        6 | 23.514 | 493.79401 |      78.38
      A | 1/10/2019 | 16:42 | 873-51-8671 | Yangon |             153.86 |                            5.1 |        Member | Female |       7.693 |                  4.7619 |     Ewallet |  Sports and travel |        7 |  7.693 | 161.55299 |      21.98
      A | 1/10/2019 | 17:15 | 665-32-9167 | Yangon |              72.52 |                            7.2 |        Member | Female |       3.626 |                  4.7619 | Credit card |  Health and beauty |        2 |  3.626 |    76.146 |      36.26
      A | 1/11/2019 | 10:37 | 289-15-7034 | Yangon |             329.52001 |                            7.5 |        Member |   Male |      16.466 |                  4.7619 | Credit card |  Sports and travel |        4 | 16.466 | 345.78601 |      82.33
      A | 1/11/2019 | 11:28 | 478-66-7835 | Yangon |              89.69 |                            4.9 |        Normal |   Male |       4.4845 |                  4.7619 |     Ewallet | Fashion accessories |        1 | 4.4845 |    94.1745 |      89.69
      A | 1/11/2019 | 11:52 | 406-46-7107 | Yangon |             579.12 |                            4.6 |        Normal | Female |      28.956 |                  4.7619 |        Cash |  Home and lifestyle |        6 | 28.956 | 608.07599 |      96.52
      A | 1/11/2019 | 13:57 | 130-67-4723 | Yangon |                291 |                            9.4 |        Normal | Female |       14.55 |                  4.7619 |     Ewallet |  Food and beverages |        6 |  14.55 |    305.55 |      48.5
      A | 1/11/2019 | 16:49 | 540-11-4336 | Yangon |             224.46001 |                            5.6 |        Normal |   Male |      11.223 |                  4.7619 | Credit card |  Food and beverages |        9 | 11.223 |   235.683 |      24.94
      A | 1/11/2019 | 19:07 | 732-67-5346 | Yangon |              68.96 |                            7.8 |        Normal |   Male |       3.4475 |                  4.7619 | Credit card |  Food and beverages |        5 |  3.4475 |   72.3975 |      13.79
      A | 1/12/2019 | 13:40 | 784-68-3927 | Yangon |             886.70801 |                            7.3 |        Member |   Male |      44.335 |                  4.7619 |     Ewallet | Electronic accessories |       10 |  44.335 |  931.0347 |      88.67
      A | 1/12/2019 | 16:48 | 381-20-0914 | Yangon |             180.09 |                            5.7 |        Member | Female |       9.0045 |                  4.7619 | Credit card | Fashion accessories |        9 | 9.0045 |  189.0945 |      20.01
      A | 1/12/2019 | 16:23 | 647-50-1224 | Yangon |             294.28001 |                            8.9 |        Normal | Female |       14.71 |                  4.7619 |     Ewallet | Fashion accessories |       10 |  14.71 |    308.91 |      29.42
      A | 1/13/2019 | 10:03 | 448-34-5834 | Yangon |              21.06 |                            5 |        Member | Female |       1.053 |                  4.7619 |        Cash |  Sports and travel |        3 |  1.053 |    22.113 |      21.06
      A | 1/13/2019 | 12:55 | 196-06-8432 | Yangon |             243.03 |                            9.3 |        Member |   Male |      12.1515 |                  4.7619 | Credit card |  Home and lifestyle |        3 | 12.1515 |  255.1815 |      81.01
      A | 1/13/2019 | 17:54 | 397-25-8725 | Yangon |             356.67999 |                            6.8 |        Member | Female |      17.829 |                  4.7619 | Credit card |  Health and beauty |        9 |  17.829 |   374.409 |      39.62
      A | 1/13/2019 | 19:14 | 459-50-7686 | Yangon |             140.75999 |                            6.4 |        Normal | Female |       7.038 |                  4.7619 |     Ewallet | Electronic accessories |        6 |  7.038 |   147.798 |      23.46
      A | 1/14/2019 | 12:52 | 667-23-5919 | Yangon |                483.5 |                            7 |        Normal | Female |      24.175 |                  4.7619 |     Ewallet | Fashion accessories |        5 | 24.175 | 507.67499 |       96.7
      A | 1/14/2019 | 19:30 | 588-47-8641 | Yangon |             560.40002 |                            4.4 |        Member |   Male |       28.02 |                  4.7619 |     Ewallet | Fashion accessories |       10 |  28.02 | 588.41998 |      56.04
      A | 1/15/2019 | 10:43 | 301-11-9629 | Yangon |              133.7 |                            9.7 |        Normal | Female |       6.685 |                  4.7619 |        Cash |  Sports and travel |        7 |  6.685 | 140.38499 |       19.1
      A | 1/15/2019 | 10:54 | 509-62-8823 | Yangon |             167.63999 |                            7 |        Member | Female |       8.377 |                  4.7619 | Credit card |  Food and beverages |        2 |  8.377 | 175.91791 |      83.77
      A | 1/15/2019 | 15:42 | 138-17-5109 | Yangon |             882.89001 |                            6.5 |        Member | Female |     40.1445 |                  4.7619 | Credit card |  Home and lifestyle |        9 | 40.1445 | 843.03448 |      89.21
      A | 1/15/2019 | 17:17 | 272-65-1806 | Yangon |             547.91998 |                            4.7 |        Normal | Female |      27.396 |                  4.7619 |     Ewallet | Electronic accessories |        9 |  27.396 | 575.31998 |      60.88
      A | 1/16/2019 | 17:24 | 886-54-4089 | Yangon |              68.58 |                            7.7 |        Normal | Female |       3.429 |                  4.7619 |        Cash |  Home and lifestyle |        1 |  3.429 |    72.009 |      11.43
      A | 1/16/2019 | 17:59 | 648-83-1321 | Yangon |             635.69998 |                            4.3 |        Member |   Male |      31.78 |                  4.7619 |        Cash |  Home and lifestyle |       10 |  31.78 |    667.38 |      63.56
      A | 1/16/2019 | 18:41 | 618-34-8551 | Yangon |             186.36 |                            8.6 |        Normal | Female |       9.318 |                  4.7619 | Credit card |  Sports and travel |        2 |  9.318 |  195.6779 |      93.18
      A | 1/17/2019 | 12:48 | 746-51-4586 | Yangon |             129.12 |                            9.4 |        Normal |   Male |       6.456 |                  4.7619 | Credit card |  Home and lifestyle |        4 |  6.456 |   135.576 |      21.52
      A | 1/17/2019 | 13:01 | 698-96-5964 | Yangon |             812.69998 |                            6.3 |        Normal | Female |      40.605 |                  4.7619 | Credit card |  Food and beverages |       10 |  40.605 | 852.70502 |      81.21
      A | 1/17/2019 | 15:15 | 497-37-6538 | Yangon |              412.37 |                            9.7 |        Normal |   Male |     20.6185 |                  4.7619 |     Ewallet |  Sports and travel |        7 | 20.6185 | 432.98849 |      58.91
      A | 1/17/2019 | 17:59 | 445-30-9252 | Yangon |               77.1 |                            6.1 |        Normal |   Male |       3.855 |                  4.7619 |     Ewallet |  Sports and travel |        3 |  3.855 |    80.955 |       25.7
      A | 1/18/2019 | 10:11 | 562-12-5438 | Yangon |             264.45001 |                            7.9 |        Member |   Male |     13.2225 |                  4.7619 |     Ewallet | Fashion accessories |        3 | 13.2225 | 277.67249 |      88.15
      A | 1/18/2019 | 17:26 | 166-19-2553 | Yangon |             534.35999 |                            9.9 |        Member |   Male |      26.718 |                  4.7619 |        Cash |  Sports and travel |        6 |  26.718 |   561.078 |      89.06
      A | 1/18/2019 | 18:43 | 706-36-6154 | Yangon |             174.24001 |                            8.7 |        Member |   Male |       8.712 |                  4.7619 |     Ewallet |  Home and lifestyle |        9 |  8.712 |   182.952 |      19.36
      A | 1/18/2019 | 20:59 | 708-81-1757 | Yangon |              131.55 |                            8.8 |        Normal | Female |      6.5775 |                  4.7619 | Credit card | Electronic accessories |        5 | 6.5775 |  138.1275 |      26.31
      A | 1/19/2019 | 12:47 | 482-17-1179 | Yangon |              35.82 |                            8.1 |        Member |   Male |       1.791 |                  4.7619 | Credit card | Electronic accessories |        3 |  1.791 |    37.611 |      11.94
      A | 1/19/2019 | 13:11 | 125-45-2293 | Yangon |             594.59998 |                            4.2 |        Normal | Female |      29.73 |                  4.7619 |        Cash | Fashion accessories |        6 |  29.73 |  624.33002 |      99.1
      A | 1/19/2019 | 16:08 | 149-61-1929 | Yangon |             641.98002 |                            6.7 |        Member | Female |      32.099 |                  4.7619 | Credit card |  Sports and travel |       10 |  32.099 |  673.995 |      64.19
      A | 1/19/2019 | 15:28 | 882-40-4577 | Yangon |             269.04001 |                            8 |        Member | Female |      13.452 |                  4.7619 | Credit card |  Sports and travel |        4 |  13.452 |   282.492 |      67.26
      A | 1/19/2019 | 16:31 | 796-32-9850 | Yangon |             387.67999 |                            6.9 |        Normal |   Male |      19.384 |                  4.7619 |        Cash |  Food and beverages |        6 |  19.384 |   323.864 |      81.28
      A | 1/19/2019 | 19:30 | 889-69-8497 | Yangon |             456.79999 |                            8.7 |        Normal | Female |      22.84 |                  4.7619 |     Ewallet |  Home and lifestyle |       10 |  22.84 |  479.6484 |      68.68
      A | 1/19/2019 | 20:07 | 645-44-1170 | Yangon |              522.63 |                            4.3 |        Member |   Male |     26.1315 |                  4.7619 |     Ewallet |  Home and lifestyle |        9 | 26.1315 | 548.76147 |      58.07
      A | 1/19/2019 | 20:29 | 832-51-6761 | Yangon |             271.04001 |                            9.6 |        Normal |   Male |      13.552 |                  4.7619 |     Ewallet |  Food and beverages |        8 |  13.552 | 284.59201 |      33.88
      A | 1/2/2019 | 15:24 | 712-39-8363 | Yangon |             249.96001 |                            5.4 |        Normal |   Male |      12.498 |                  4.7619 |     Ewallet |  Food and beverages |        8 |  12.498 | 262.45801 |      41.66
      A | 1/2/2019 | 16:57 | 584-35-8843 | Yangon |              42.47 |                            6.7 |        Normal |   Male |      2.1235 |                  4.7619 |        Cash |  Sports and travel |        1 |  2.1235 |   44.5935 |      42.47
      A | 1/20/2019 | 15:48 | 787-87-2010 | Yangon |                222 |                            6.6 |        Member |   Male |       11.1 |                  4.7619 | Credit card |  Health and beauty |        4 |  11.1 |  233.1001 |      55.5
      A | 1/20/2019 | 15:55 | 817-48-8732 | Yangon |              723.5 |                            5.4 |        Member | Female |      36.175 |                  4.7619 |        Cash |  Home and lifestyle |       10 |  36.175 |  759.6749 |      72.35
      A | 1/20/2019 | 17:48 | 418-67-1709 | Yangon |             511.84001 |                            9.9 |        Member | Female |      25.592 |                  4.7619 |     Ewallet | Fashion accessories |        8 |  25.592 | 536.59198 |      63.98
      A | 1/20/2019 | 18:09 | 645-78-8093 | Yangon |              186.28 |                            4.1 |        Normal | Female |       9.314 |                  4.7619 |     Ewallet |  Sports and travel |        2 |  9.314 | 195.59399 |      93.14
      A | 1/21/2019 | 10:00 | 316-68-5352 | Yangon |              72.72 |                            7.1 |        Member | Female |       3.636 |                  4.7619 |        Cash |  Food and beverages |        2 |  3.636 |    76.356 |      36.36
      A | 1/21/2019 | 16:37 | 239-18-7476 | Yangon |             467.70001 |                            8 |        Normal | Female |      23.385 |                  4.7619 |     Ewallet |  Home and lifestyle |        7 | 23.385 | 491.08499 |      77.95
      A | 1/21/2019 | 17:44 | 462-67-9126 | Yangon |             439.32001 |                            7.2 |        Normal |   Male |      21.966 |                  4.7619 |        Cash |  Home and lifestyle |        6 |  21.966 | 461.28601 |      73.22
      A | 1/21/2019 | 18:00 | 329-62-1586 | Yangon |             164.09999 |                            8.6 |        Normal |   Male |       8.2005 |                  4.7619 | Credit card |  Food and beverages |        3 |  8.2005 |  172.2105 |      54.67
      A | 1/21/2019 | 18:01 | 269-10-8440 | Yangon |              372.19 |                            8.9 |        Member |   Male |      18.6095 |                  4.7619 |        Cash |  Health and beauty |        7 | 18.6095 |  390.7995 |      53.17
      A | 1/22/2019 | 10:55 | 287-21-9891 | Yangon |             672.03003 |                            9.4 |        Member |   Male |      33.6015 |                  4.7619 |     Ewallet |  Home and lifestyle |        9 | 33.6015 | 705.63147 |      74.67
      A | 1/22/2019 | 11:00 | 460-35-4398 | Yangon |              92.04 |                            9.1 |        Normal |   Male |       4.602 |                  4.7619 |     Ewallet |  Home and lifestyle |        3 |  4.602 |   96.642 |      30.68
      A | 1/22/2019 | 10:28 | 213-32-1216 | Yangon |             396.35999 |                            7.3 |        Member |   Male |      19.818 |                  4.7619 | Credit card | Electronic accessories |        6 |  19.818 | 416.17801 |      66.06
      A | 1/23/2019 | 11:22 | 828-46-6863 | Yangon |             591.17999 |                            4 |        Member |   Male |      29.559 |                  4.7619 | Credit card |  Food and beverages |        6 |  29.559 | 620.73901 |      98.53
      A | 1/23/2019 | 12:20 | 318-12-8304 | Yangon |              30.61 |                            5.2 |        Normal |   Male |      1.5305 |                  4.7619 |     Ewallet | Fashion accessories |        1 |  1.5305 |   32.1405 |      30.61
      A | 1/23/2019 | 12:35 | 316-66-3011 | Yangon |             428.67001 |                            5 |        Member | Female |     21.4335 |                  4.7619 |        Cash |  Food and beverages |        9 | 21.4335 |  450.1839 |      47.63
      A | 1/23/2019 | 14:04 | 684-78-5470 | Yangon |              89.7 |                            6.8 |        Member |   Male |       4.485 |                  4.7619 |     Ewallet | Fashion accessories |        1 |  4.485 |    94.185 |       89.7
      A | 1/23/2019 | 14:21 | 160-22-2687 | Yangon |             479.75 |                            8.8 |        Member | Female |     23.9875 |                  4.7619 |     Ewallet |  Health and beauty |        5 | 23.9875 |  503.7375 |      95.95
      A | 1/23/2019 | 17:44 | 809-46-1866 | Yangon |             232.60001 |                            8.4 |        Normal |   Male |      11.63 |                  4.7619 |        Cash |  Health and beauty |        1 |  11.63 |    244.23 |      58.15
      A | 1/24/2019 | 10:04 | 232-11-3025 | Yangon |             787.78901 |                            6.4 |        Normal |   Male |     39.385 |                  4.7619 |        Cash |  Sports and travel |       10 |  39.385 |  827.8085 |      78.77
      A | 1/24/2019 | 10:57 | 778-34-2523 | Yangon |             388.95999 |                            5 |        Member | Female |      19.448 |                  4.7619 |        Cash | Electronic accessories |        8 |  19.448 | 408.48799 |      48.62
      A | 1/24/2019 | 17:43 | 237-44-6163 | Yangon |              84.48 |                            7.6 |        Normal |   Male |       4.224 |                  4.7619 |        Cash | Electronic accessories |        8 |  4.224 |    88.704 |      10.56
      A | 1/24/2019 | 18:04 | 435-13-4980 | Yangon |             693.44 |                            7.2 |        Member | Female |      34.672 |                  4.7619 | Credit card | Fashion accessories |        8 |  34.672 |  728.112 |      86.68
      A | 1/24/2019 | 20:36 | 726-27-2396 | Yangon |              387.5 |                            4.3 |        Normal | Female |      19.375 |                  4.7619 |     Ewallet |  Health and beauty |        5 |  19.375 |  406.875 |       77.5
      A | 1/25/2019 | 11:05 | 843-73-4724 | Yangon |               74.1 |                            9.2 |        Normal |   Male |       3.705 |                  4.7619 |        Cash | Fashion accessories |        1 |  3.705 |    77.805 |       74.1
      A | 1/25/2019 | 11:40 | 348-21-9136 | Yangon |              160.2 |                            9.7 |        Member | Female |       8.01 |                  4.7619 |        Cash |  Sports and travel |        4 |  8.01 |  168.2101 |      40.05
      A | 1/25/2019 | 14:19 | 854-66-2781 | Yangon |             159.89999 |                            7.5 |        Member |   Male |      7.995 |                  4.7619 |     Ewallet |  Home and lifestyle |        3 |  7.995 |   167.895 |      53.3
      A | 1/25/2019 | 15:54 | 387-83-9164 | Yangon |             240.03999 |                            4.5 |        Member |   Male |      12.002 |                  4.7619 |        Cash |  Home and lifestyle |        4 |  12.002 |  252.0421 |      60.01
      A | 1/25/2019 | 20:24 | 888-02-8338 | Yangon |             236.67001 |                            5.9 |        Normal |   Male |     11.8035 |                  4.7619 |     Ewallet | Electronic accessories |        9 | 11.8035 |  247.8735 |      26.23
      A | 1/26/2019 | 12:43 | 894-26-4956 | Yangon |              21.32 |                            6.9 |        Normal |   Male |       1.066 |                  4.7619 | Credit card | Fashion accessories |        1 |  1.066 |    22.386 |      21.32
      A | 1/26/2019 | 18:19 | 388-39-1707 | Yangon |              12.09 |                            8.2 |        Normal |   Male |      0.6045 |                  4.7619 | Credit card |  Food and beverages |        1 | 0.6045 |   12.6945 |      12.09
      A | 1/26/2019 | 18:22 | 834-61-8124 | Yangon |             361.82999 |                            5.8 |        Normal |   Male |     18.0915 |                  4.7619 |        Cash | Electronic accessories |        7 | 18.0915 | 379.92151 |      51.69
      A | 1/27/2019 | 13:26 | 870-84-3160 | Yangon |              165.26 |                            9 |        Member | Female |      8.0625 |                  4.7619 |        Cash |  Sports and travel |        3 | 8.0625 | 169.3125 |      55.09
      A | 1/27/2019 | 14:36 | 534-01-4457 | Yangon |             490.26001 |                            8 |        Member |   Male |      24.513 |                  4.7619 | Credit card |  Food and beverages |        6 |  24.513 | 514.77301 |      81.71
      A | 1/27/2019 | 15:51 | 633-91-1052 | Yangon |              24.06 |                            5.1 |        Normal | Female |       1.203 |                  4.7619 |        Cash |  Home and lifestyle |        2 |  1.203 |    25.263 |      12.03
      A | 1/27/2019 | 18:05 | 595-27-4851 | Yangon |             379.95999 |                            9.3 |        Member | Female |      18.998 |                  4.7619 |     Ewallet |  Home and lifestyle |        7 |  18.998 | 398.95801 |      54.28
      A | 1/27/2019 | 18:08 | 585-90-8249 | Yangon |              73.26 |                            9.7 |        Member |   Male |       3.663 |                  4.7619 |     Ewallet | Electronic accessories |        1 |  3.663 |    76.923 |      73.26
      A | 1/27/2019 | 20:33 | 123-19-1176 | Yangon |             445.76001 |                            8.4 |        Member |   Male |      23.288 |                  4.7619 |     Ewallet |  Health and beauty |        8 |  23.288 |  489.048 |      58.22
      A | 1/28/2019 | 17:35 | 865-41-9875 | Yangon |              80.71 |                            8.1 |        Normal | Female |       4.0355 |                  4.7619 |        Cash |  Food and beverages |        7 |  4.0355 |  84.7455 |      11.53
      A | 1/28/2019 | 20:54 | 726-29-1793 | Yangon |             193.44 |                            9.8 |        Member | Female |       9.672 |                  4.7619 | Credit card | Electronic accessories |        3 |  9.672 |  203.112 |      64.48
      A | 1/29/2019 | 11:26 | 575-67-1508 | Yangon |              38.6 |                            6.7 |        Normal |   Male |       1.93 |                  4.7619 |     Ewallet | Electronic accessories |        1 |  1.93 |    40.53 |       38.6
      A | 1/29/2019 | 11:56 | 746-94-8204 | Yangon |             749.15997 |                            7.4 |        Normal |   Male |     37.458 |                  4.7619 | Credit card | Fashion accessories |        9 |  37.458 | 786.61798 |      83.24
      A | 1/29/2019 | 16:48 | 816-72-8853 | Yangon |             139.64999 |                            5.9 |        Member |   Male |      6.9825 |                  4.7619 |        Cash |  Sports and travel |        1 |  6.9825 | 146.63251 |      27.93
      A | 1/29/2019 | 18:15 | 569-76-2760 | Yangon |              88.04 |                            6.6 |        Member | Female |      4.402 |                  4.7619 | Credit card |  Sports and travel |        4 |  4.402 |   92.442 |      22.01
      A | 1/29/2019 | 19:45 | 800-09-8606 | Yangon |             436.85001 |                            6.6 |        Member | Female |     21.8425 |                  4.7619 |        Cash |  Home and lifestyle |        5 | 21.8425 |  458.6925 |      87.37
      A | 1/3/2019 | 11:36 | 244-62-3782 | Yangon |             350.04999 |                            5.5 |        Member | Female |     17.5025 |                  4.7619 |     Ewallet |  Health and beauty |        5 | 17.5025 | 367.55249 |      70.01
      A | 1/3/2019 | 12:02 | 749-24-1565 | Yangon |              207.27 |                            7.9 |        Normal | Female |     10.3635 |                  4.7619 |     Ewallet |  Health and beauty |        9 | 10.3635 |  217.6335 |      23.03
      A | 1/3/2019 | 16:47 | 422-29-8786 | Yangon |             335.65001 |                            9.1 |        Normal | Female |     16.7725 |                  4.7619 |        Cash |  Home and lifestyle |        5 | 16.7725 |  352.2225 |      67.09
      A | 1/30/2019 | 11:32 | 884-30-3936 | Yangon |              281.34 |                            5.9 |        Member | Female |      14.067 |                  4.7619 | Credit card | Electronic accessories |        3 |  14.067 | 295.4007 |      93.78
      A | 1/31/2019 | 10:00 | 522-57-8364 | Yangon |              410.72 |                            7.6 |        Member |   Male |      20.536 |                  4.7619 |     Ewallet | Fashion accessories |        8 |  20.536 | 431.25601 |      51.34
      A | 1/31/2019 | 10:46 | 366-93-8948 | Yangon |              66.36 |                            9.7 |        Member |   Male |      3.3175 |                  4.7619 | Credit card | Electronic accessories |        1 |  3.3175 |  69.6675 |      66.36
      A | 1/31/2019 | 13:47 | 187-83-5498 | Yangon |              83.08 |                            4.7 |        Member |   Male |       4.154 |                  4.7619 |        Cash | Electronic accessories |        4 |  4.154 |    87.234 |      20.77
      A | 1/4/2019 | 13:21 | 132-23-6451 | Yangon |             104.85 |                            7.8 |        Member |   Male |      5.2425 |                  4.7619 |        Cash |  Health and beauty |        5 | 5.2425 | 110.0925 |      20.97

---MORE---
 branch | date     | time  | invoice_id  | city   | cost_of_goods_sold | customer_stratification_rating | customer_type | gender | gross_income | gross_margin_percentage | payment     | product_line           | quantity | tax_5  | total     | unit_price
--------+----------+-------+-------------+--------+--------------------+--------------------------------+---------------+--------+--------------+-------------------------+-------------+------------------------+----------+--------+-----------+------------
      A | 1/4/2019 | 13:34 | 731-14-2199 | Yangon |             358.39999 |                              7 |        Member | Female |       17.77 |                  4.7619 |     Ewallet |  Home and lifestyle |       10 |  17.77 | 373.17001 |      35.84
      A | 1/5/2019 | 11:26 | 217-60-1179 | Yangon |             250.68001 |                            4.2 |        Member |   Male |      12.63 |                  4.7619 |        Cash |  Home and lifestyle |        4 |  12.63 |   265.13 |      62.67
      A | 1/5/2019 | 11:51 | 489-82-1237 | Yangon |             657.15997 |                            7.3 |        Normal | Female |      32.858 |                  4.7619 | Credit card | Electronic accessories |        7 | 32.858 | 690.01801 |      93.88
      A | 1/5/2019 | 13:08 | 758-67-8428 | Yangon |             522.83002 |                            9.1 |        Member | Female |     26.1415 |                  4.7619 |     Ewallet |  Health and beauty |        7 | 26.1415 |  548.9715 |      74.69
      A | 1/5/2019 | 16:05 | 154-51-6085 | Yangon |             282.95999 |                            4.4 |        Member |   Male |      14.148 |                  4.7619 | Credit card |  Home and lifestyle |        6 |  14.148 |   297.108 |      70.74
      A | 1/5/2019 | 17:29 | 209-61-8286 | Yangon |              214.55 |                            6.1 |        Normal |   Male |      10.7275 |                  4.7619 |     Ewallet |  Home and lifestyle |        5 | 10.7275 |  225.2775 |      42.91
      A | 1/6/2019 | 11:09 | 498-29-1201 | Yangon |              15.34 |                            6.5 |        Normal | Female |       0.767 |                  4.7619 |        Cash |  Sports and travel |        1 |  0.767 |    16.107 |      15.34
      A | 1/6/2019 | 11:23 | 784-11-6354 | Yangon |             471.20001 |                            8.9 |        Member |   Male |      23.56 |                  4.7619 |        Cash |  Home and lifestyle |        8 |  23.56 | 494.76001 |      58.9
      A | 1/6/2019 | 11:51 | 846-10-8241 | Yangon |             297.98997 |                            6.8 |        Normal | Female |     14.8995 |                  4.7619 |        Cash | Fashion accessories |        7 | 14.8995 |  312.8895 |      42.57
      A | 1/6/2019 | 13:35 | 852-82-2749 | Yangon |             258.35999 |                            9.3 |        Normal |   Male |      12.918 |                  4.7619 |     Ewallet |  Sports and travel |        4 |  12.918 | 271.27882 |      64.59
      A | 1/6/2019 | 13:58 | 662-72-2873 | Yangon |              204.7 |                            9.9 |        Normal | Female |      10.235 |                  4.7619 |     Ewallet |  Food and beverages |        5 |  10.235 |  214.935 |      40.94
      A | 1/7/2019 | 11:44 | 126-54-1882 | Yangon |             193.56 |                            8.8 |        Member | Female |       9.678 |                  4.7619 | Credit card |  Home and lifestyle |        3 |  9.678 | 203.05799 |      31.56
      A | 1/7/2019 | 12:08 | 541-48-8554 | Yangon |             548.54999 |                              6 |        Normal |   Male |      27.4275 |                  4.7619 | Credit card |  Sports and travel |        9 | 27.4275 | 575.97748 |      60.95
      A | 1/7/2019 | 16:11 | 799-71-1548 | Yangon |              310.88 |                            8.8 |        Member |   Male |       15.544 |                  4.7619 | Credit card | Electronic accessories |        4 |  15.544 | 326.42401 |      77.72

(114 rows)
cqlsh:retailstore>
```

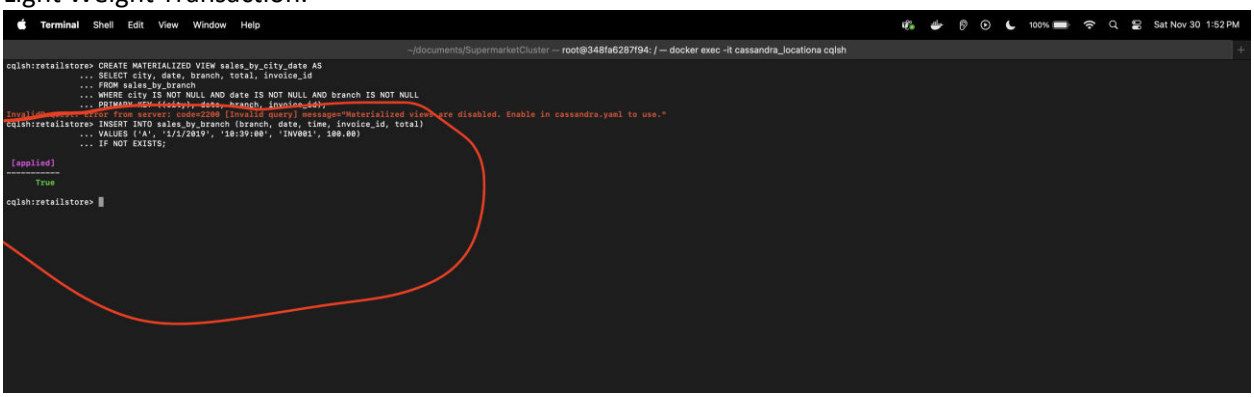
Indexing:


Terminal Shell Edit View Window Help
Sat Nov 30 1:23 PM
~/documents/SupermarketCluster — root@348fa6287f94: / — docker exec -it cassandra_locationa cqlsh
cqlsh:retailstore> CREATE INDEX payment_index ON retailstore.sales_by_branch (payment);
cqlsh:retailstore> SELECT * FROM sales_by_branch
... WHERE branch = 'A' AND payment = 'Cash';
---MORE---
(110 rows)
cqlsh:retailstore>

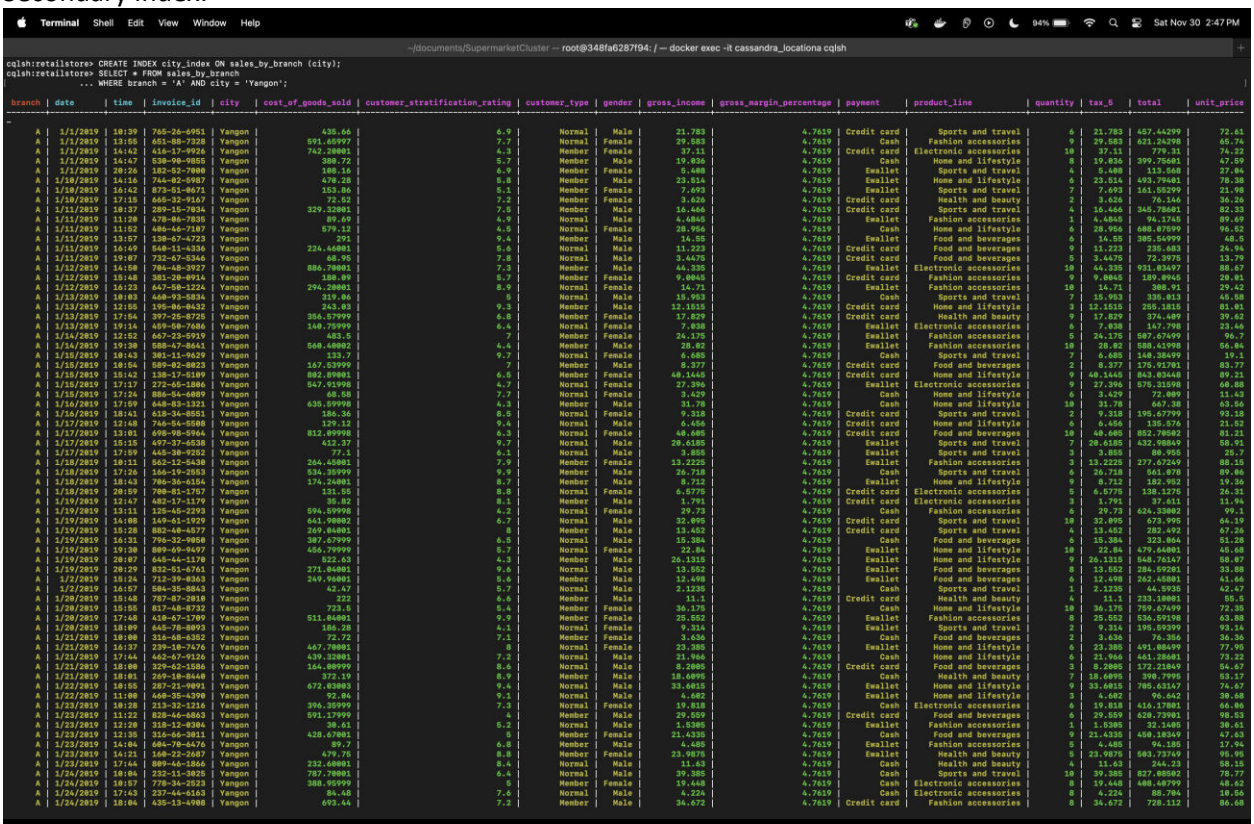
Metalized views:

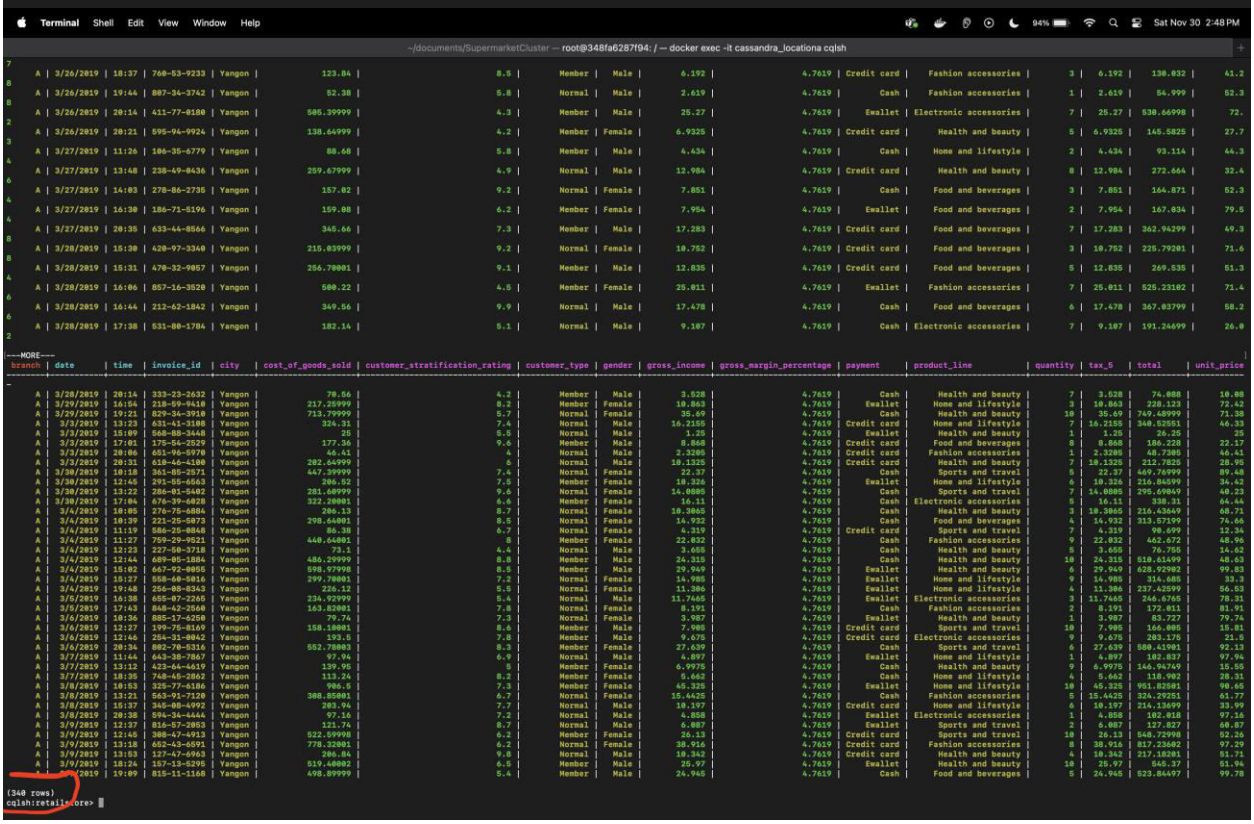


Light Weight Transaction:

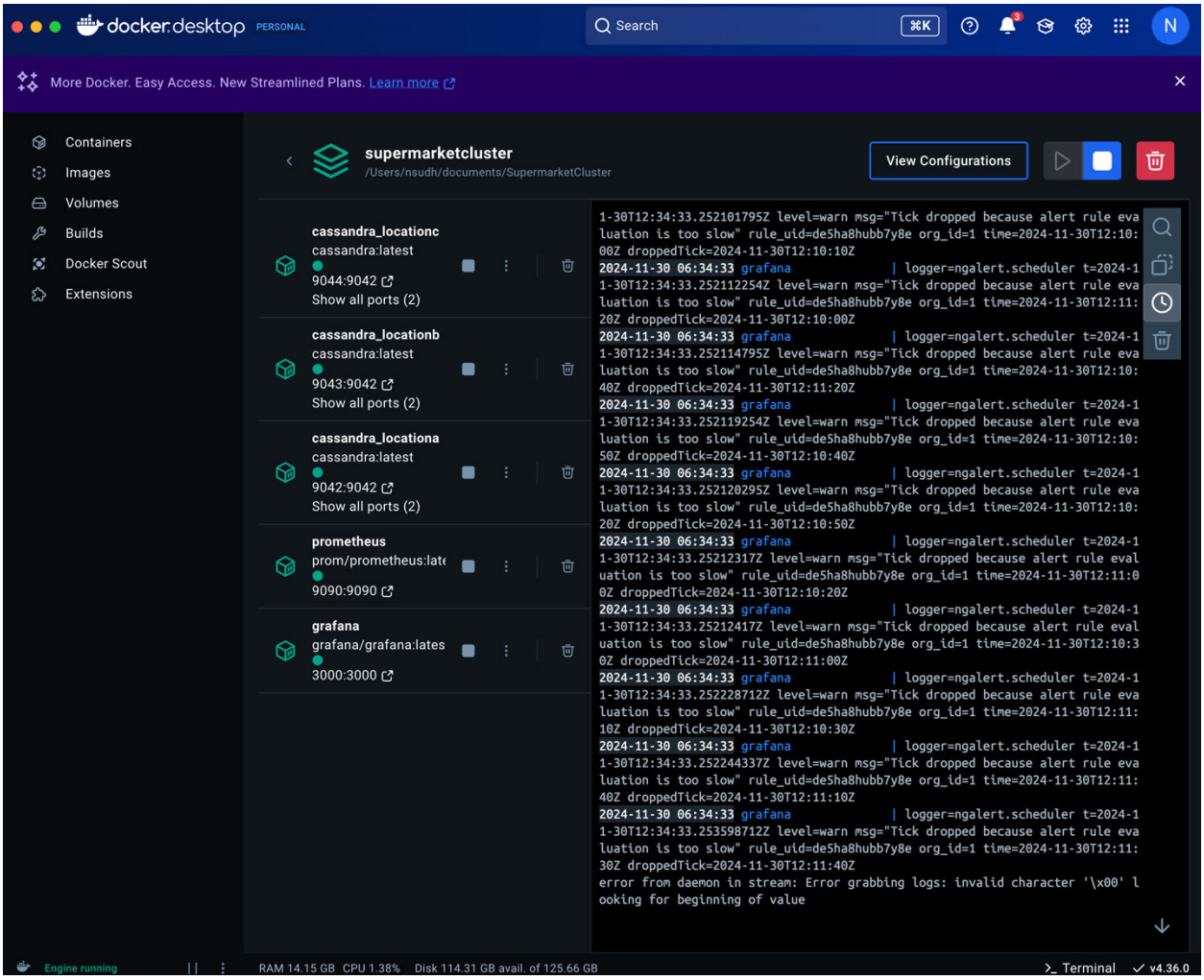


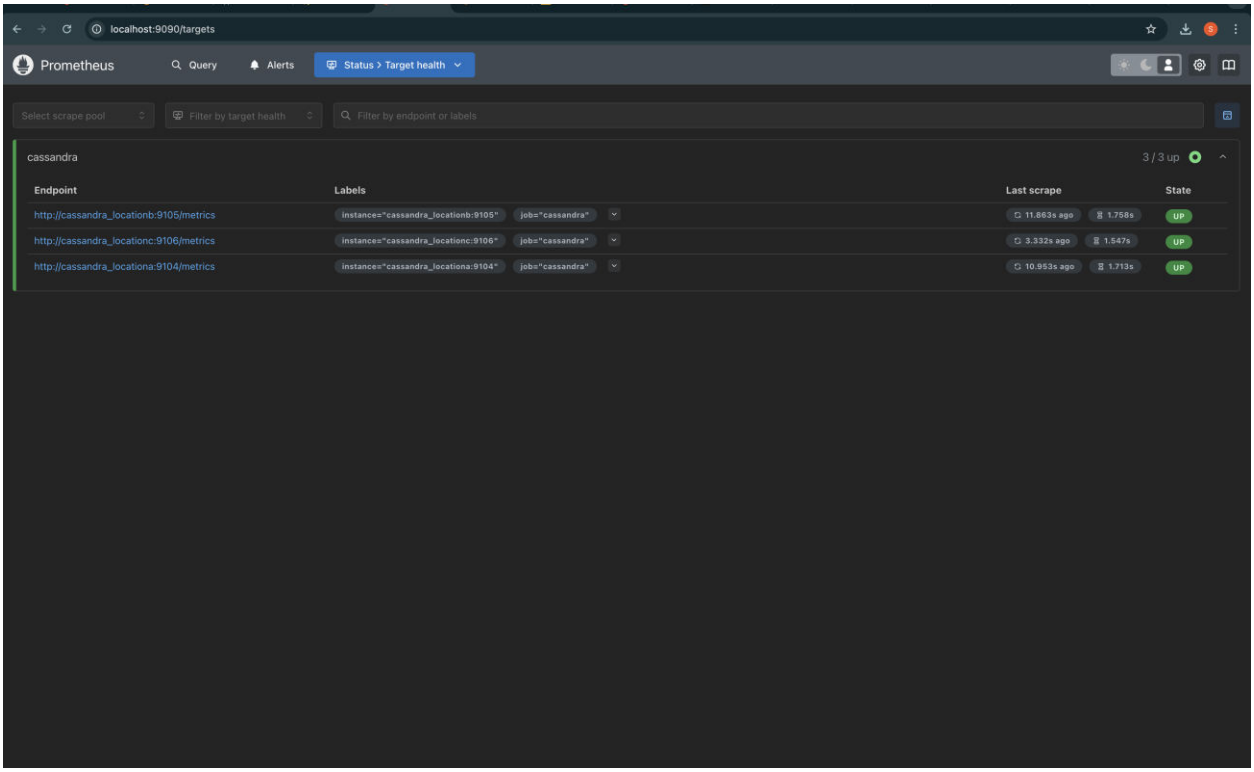
Secondary Index:

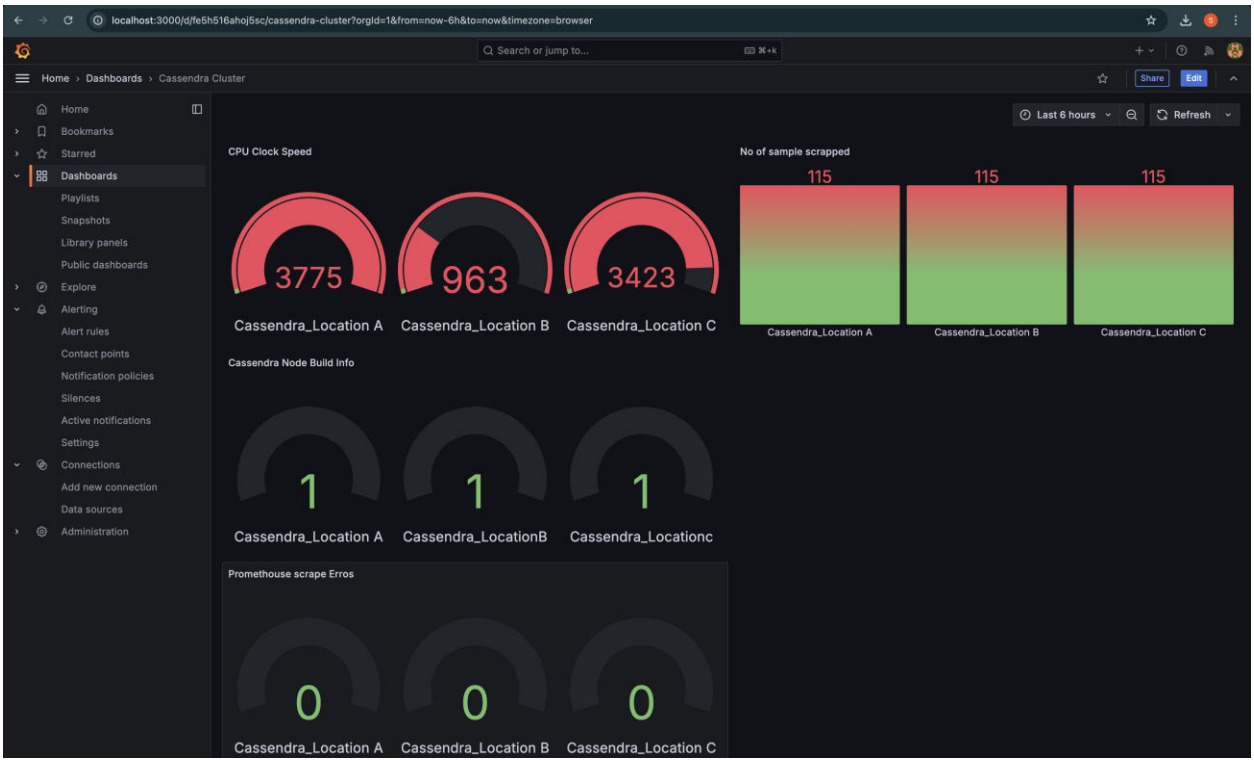


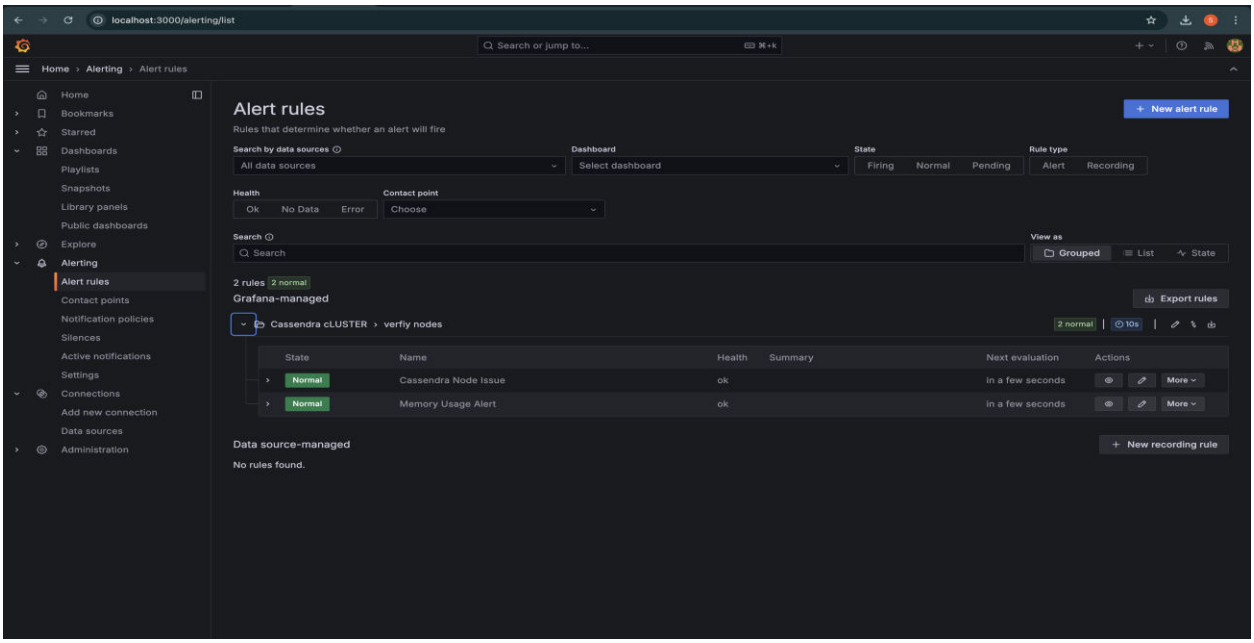


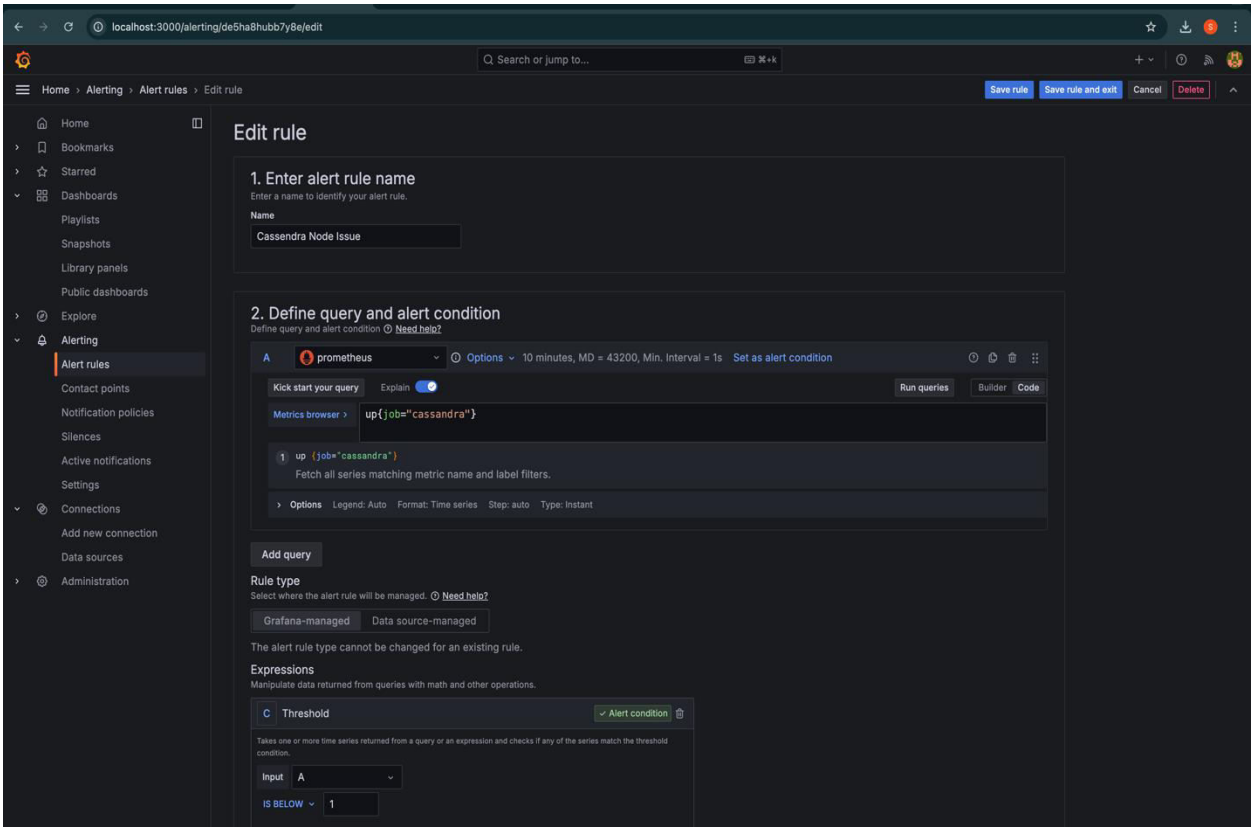
Scalability And Monitoring

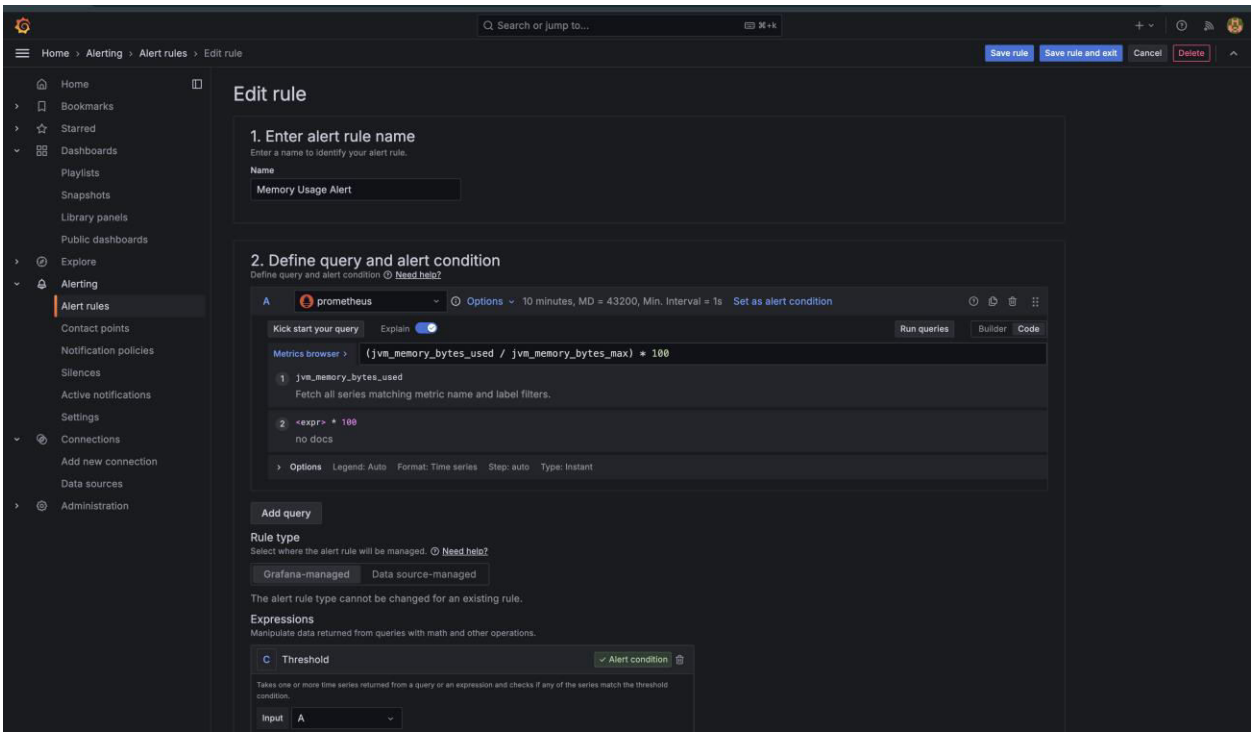


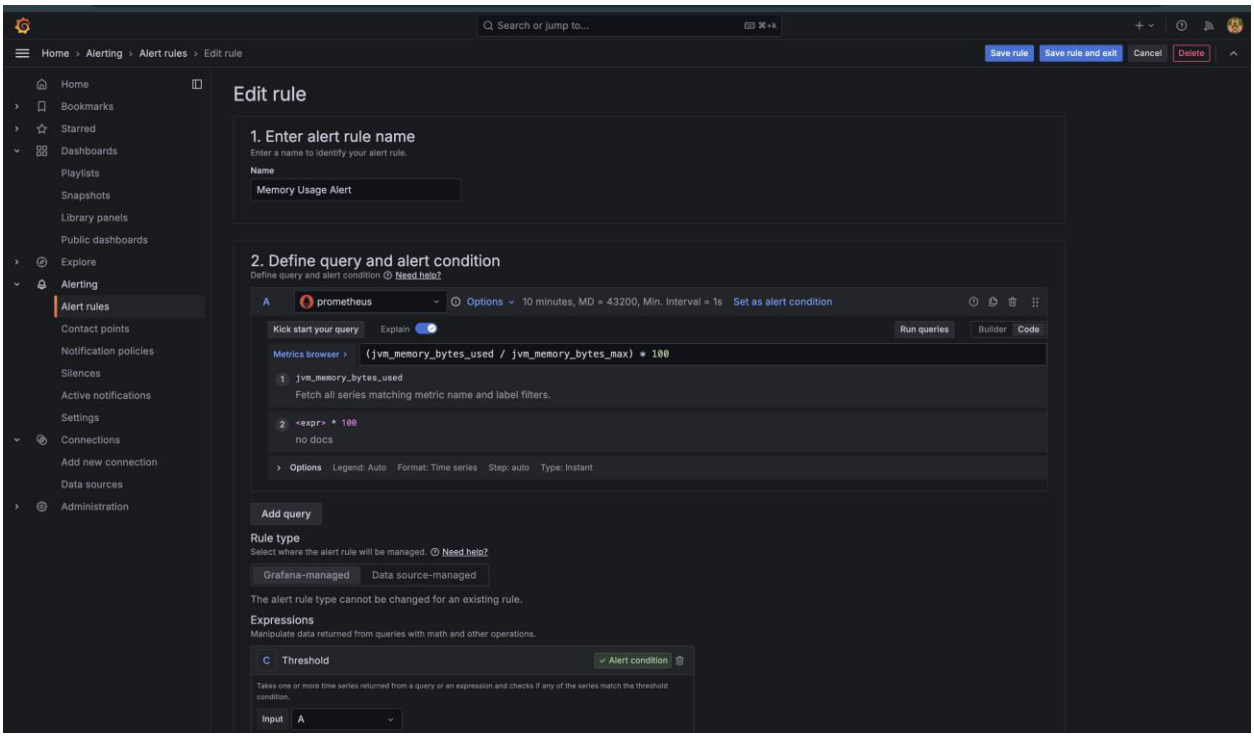

localhost:9090/targets
Prometheus
Query
Alerts
Status > Target health
Select scrape pool
Filter by target health
Filter by endpoint or labels
cassandra
3 / 3 up
Endpoint
Labels
Last scrape
State
http://cassandra_locationb:9105/metrics
instance="cassandra_locationb:9105"
job="cassandra"
11.863s ago
1.758s
UP
http://cassandra_locationc:9106/metrics
instance="cassandra_locationc:9106"
job="cassandra"
3.332s ago
1.547s
UP
http://cassandra_locationa:9104/metrics
instance="cassandra_locationa:9104"
job="cassandra"
10.953s ago
1.713s
UP


localhost:3000/d/fe5h516ahoj5sc/cassendra-cluster?orgId=1&from=now-6h&to=now&timezone=browser
Search or jump to...
Home › Dashboards › Cassendra Cluster
Share
Edit
Home
Bookmarks
Starred
Dashboards
Playlists
Snapshots
Library panels
Public dashboards
Explore
Alerting
Alert rules
Contact points
Notification policies
Silences
Active notifications
Settings
Connections
Add new connection
Data sources
Administration
Last 6 hours
Refresh
CPU Clock Speed
3775
963
3423
Cassendra_Location A
Cassendra_Location B
Cassendra_Location C
No of sample scrapped
115
115
115
Cassendra_Location A
Cassendra_Location B
Cassendra_Location C
Cassendra Node Build Info
1
1
1
Cassendra_Location A
Cassendra_LocationB
Cassendra_Locationc
Promethouse scrape Erros
0
0
0
Cassendra_Location A
Cassendra_Location B
Cassendra_Location C


localhost:3000/alerting/list
Search or jump to...
Home › Alerting › Alert rules
Home
Bookmarks
Starred
Dashboards
Playlists
Snapshots
Library panels
Public dashboards
Explore
Alerting
Alert rules
Contact points
Notification policies
Silences
Active notifications
Settings
Connections
Add new connection
Data sources
Administration
Alert rules
Rules that determine whether an alert will fire
+ New alert rule
Search by data sources
All data sources
Dashboard
Select dashboard
State
Firing
Normal
Pending
Rule type
Alert
Recording
Health
Ok
No Data
Error
Contact point
Choose
Search
Search
View as
Grouped
List
State
2 rules 2 normal
Grafana-managed
Export rules
Cassendra cLUSTER › verfiy nodes
2 normal
10s
State
Name
Health
Summary
Next evaluation
Actions
Normal
Cassendra Node Issue
ok
in a few seconds
More
Normal
Memory Usage Alert
ok
in a few seconds
More
Data source-managed
+ New recording rule
No rules found.

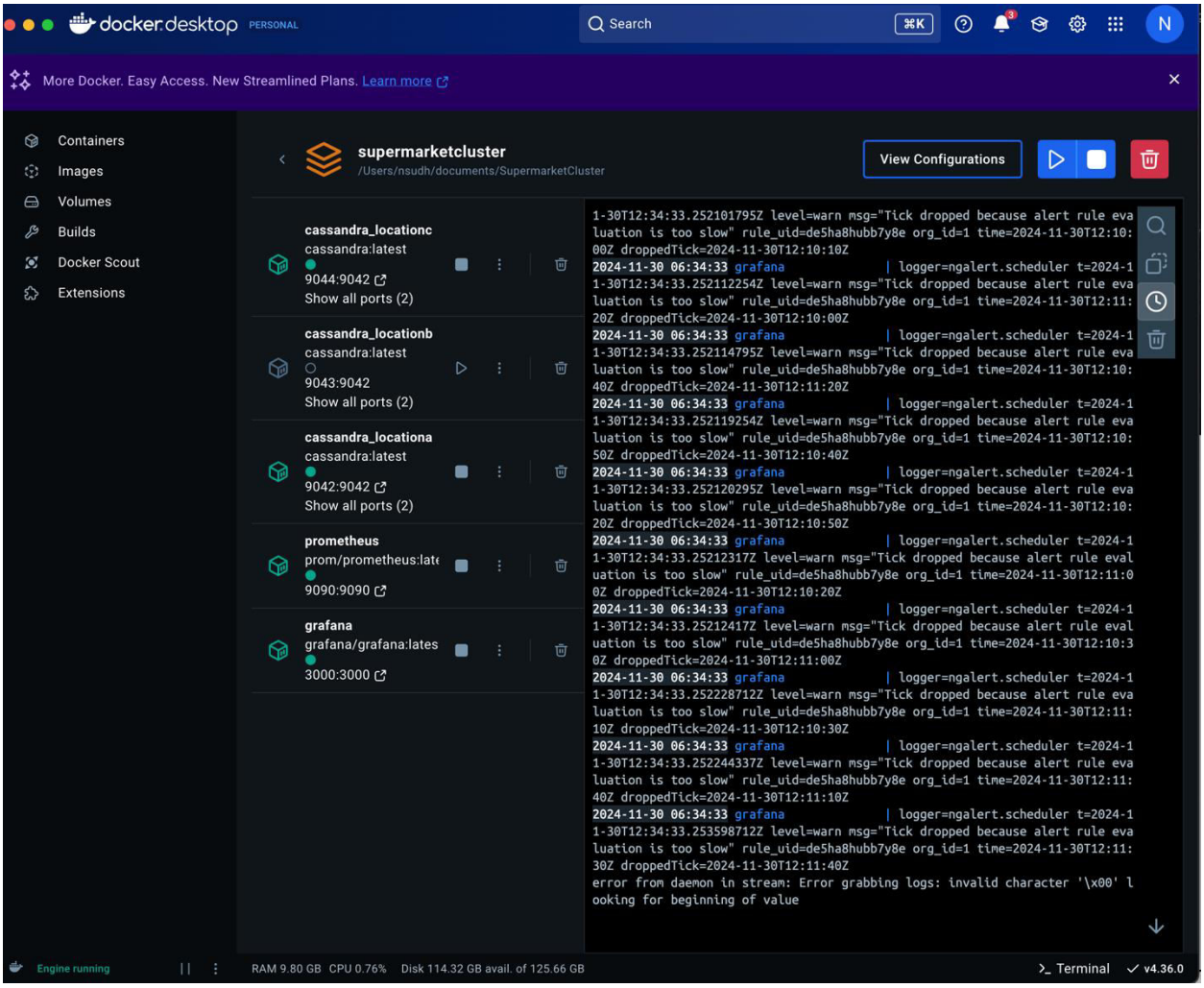

localhost:3000/alerting/de5ha8hubb7y8e/edit
Search or jump to...
Home › Alerting › Alert rules › Edit rule
Save rule
Save rule and exit
Cancel
Delete
Home
Bookmarks
Starred
Dashboards
Playlists
Snapshots
Library panels
Public dashboards
Explore
Alerting
Alert rules
Contact points
Notification policies
Silences
Active notifications
Settings
Connections
Add new connection
Data sources
Administration
Edit rule
1. Enter alert rule name
Enter a name to identify your alert rule.
Name
Cassendra Node Issue
2. Define query and alert condition
Define query and alert condition Need help?
A
prometheus
Options
10 minutes, MD = 43200, Min. Interval = 1s
Set as alert condition
Kick start your query
Explain
Run queries
Builder
Code
Metrics browser ›
up{job="cassandra"}
1 up {job="cassandra"}
Fetch all series matching metric name and label filters.
Options
Legend: Auto
Format: Time series
Step: auto
Type: Instant
Add query
Rule type
Select where the alert rule will be managed. Need help?
Grafana-managed
Data source-managed
The alert rule type cannot be changed for an existing rule.
Expressions
Manipulate data returned from queries with math and other operations.
C
Threshold
Alert condition
Takes one or more time series returned from a query or an expression and checks if any of the series match the threshold condition.
Input
A
IS BELOW
1

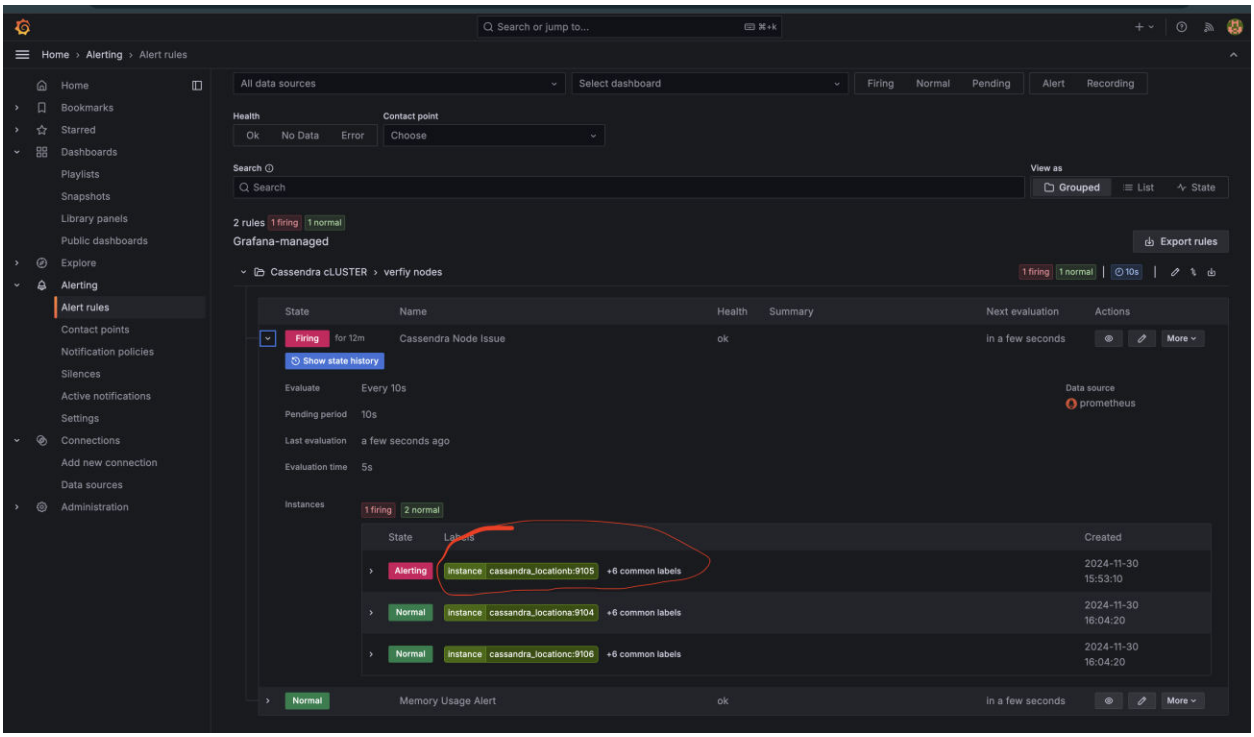

Search or jump to...
Home › Alerting › Alert rules › Edit rule
Save rule
Save rule and exit
Cancel
Delete
Home
Bookmarks
Starred
Dashboards
Playlists
Snapshots
Library panels
Public dashboards
Explore
Alerting
Alert rules
Contact points
Notification policies
Silences
Active notifications
Settings
Connections
Add new connection
Data sources
Administration
Edit rule
1. Enter alert rule name
Enter a name to identify your alert rule.
Name
Memory Usage Alert
2. Define query and alert condition
Define query and alert condition Need help?
A
prometheus
Options
10 minutes, MD = 43200, Min. Interval = 1s
Set as alert condition
Kick start your query
Explain
Run queries
Builder
Code
Metrics browser ›
(jvm_memory_bytes_used / jvm_memory_bytes_max) * 100
1 jvm_memory_bytes_used
Fetch all series matching metric name and label filters.
2 <expr> * 100
no docs
Options
Legend: Auto
Format: Time series
Step: auto
Type: Instant
Add query
Rule type
Select where the alert rule will be managed. Need help?
Grafana-managed
Data source-managed
The alert rule type cannot be changed for an existing rule.
Expressions
Manipulate data returned from queries with math and other operations.
C
Threshold
Alert condition
Takes one or more time series returned from a query or an expression and checks if any of the series match the threshold condition.
Input
A


Search or jump to...
Home › Alerting › Alert rules › Edit rule
Save rule
Save rule and exit
Cancel
Delete
Home
Bookmarks
Starred
Dashboards
Playlists
Snapshots
Library panels
Public dashboards
Explore
Alerting
Alert rules
Contact points
Notification policies
Silences
Active notifications
Settings
Connections
Add new connection
Data sources
Administration
Edit rule
1. Enter alert rule name
Enter a name to identify your alert rule.
Name
Memory Usage Alert
2. Define query and alert condition
Define query and alert condition Need help?
A
prometheus
Options
10 minutes, MD = 43200, Min. Interval = 1s
Set as alert condition
Kick start your query
Explain
Run queries
Builder
Code
Metrics browser ›
(jvm_memory_bytes_used / jvm_memory_bytes_max) * 100
1 jvm_memory_bytes_used
Fetch all series matching metric name and label filters.
2 <expr> * 100
no docs
Options
Legend: Auto
Format: Time series
Step: auto
Type: Instant
Add query
Rule type
Select where the alert rule will be managed. Need help?
Grafana-managed
Data source-managed
The alert rule type cannot be changed for an existing rule.
Expressions
Manipulate data returned from queries with math and other operations.
C
Threshold
Alert condition
Takes one or more time series returned from a query or an expression and checks if any of the series match the threshold condition.
Input
A

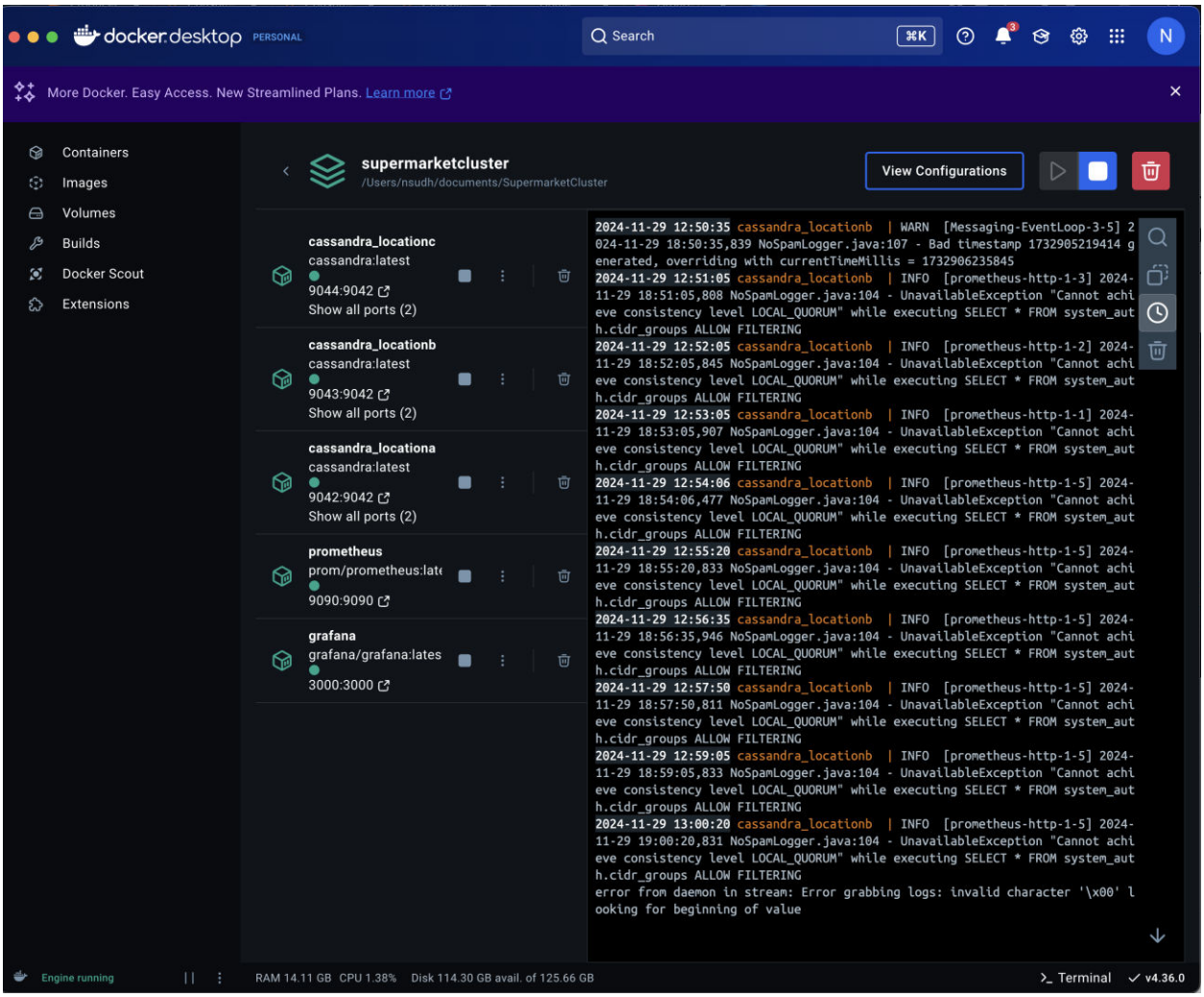
Testing Alrest Cassendra Node down:

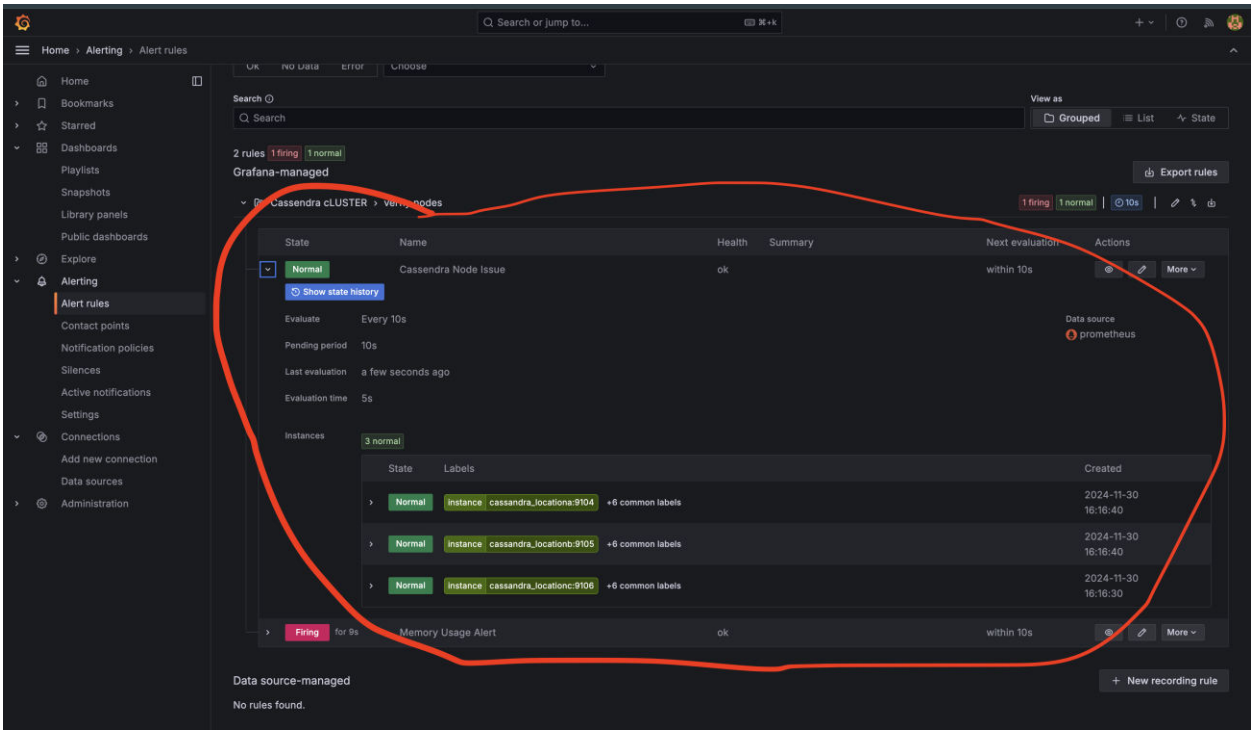
Dowing location B;



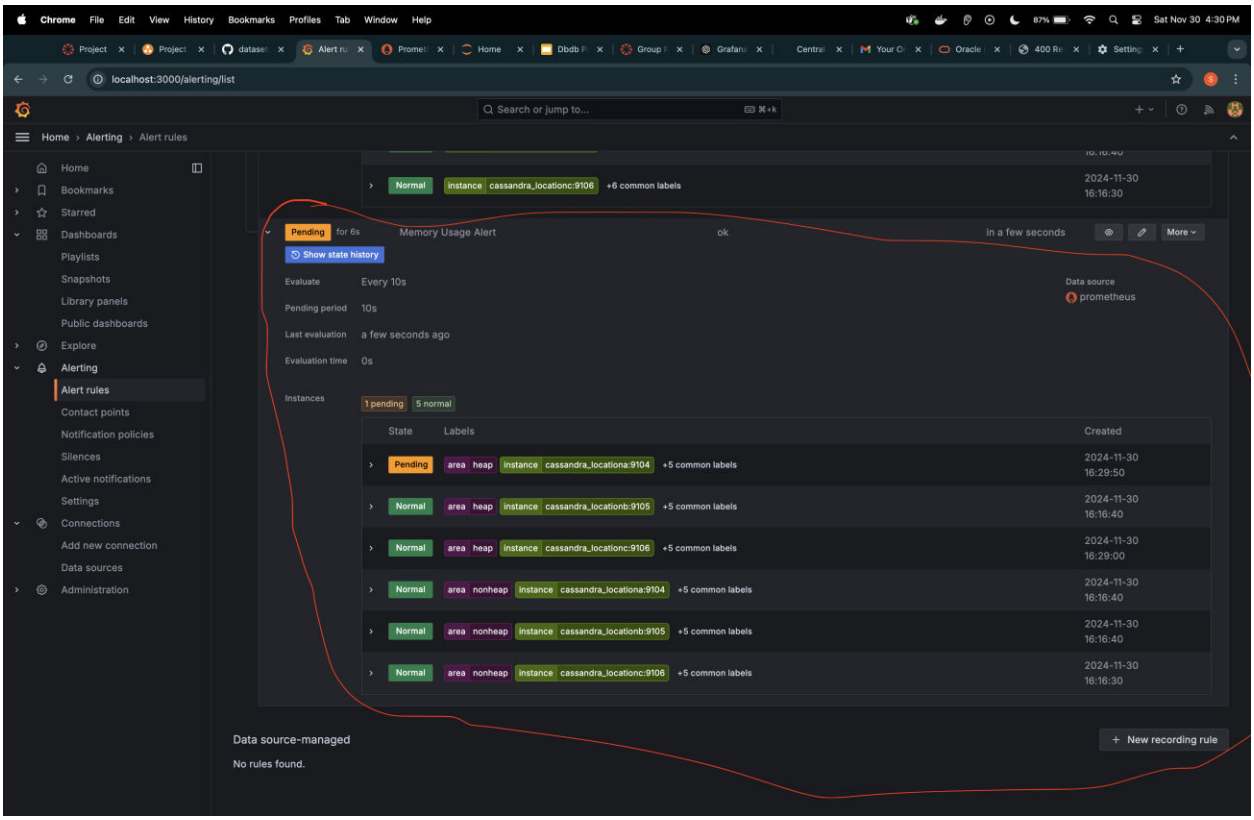

Search or jump to...
Home › Alerting › Alert rules
Home
Bookmarks
Starred
Dashboards
Playlists
Snapshots
Library panels
Public dashboards
Explore
Alerting
Alert rules
Contact points
Notification policies
Silences
Active notifications
Settings
Connections
Add new connection
Data sources
Administration
All data sources
Select dashboard
Firing
Normal
Pending
Alert
Recording
Health
Ok
No Data
Error
Contact point
Choose
Search
Search
View as
Grouped
List
State
2 rules 1 firing 1 normal
Grafana-managed
Export rules
Cassendra cLUSTER › verfiy nodes
1 firing 1 normal
10s
State
Name
Health
Summary
Next evaluation
Actions
Firing
for 12m
Cassendra Node Issue
ok
in a few seconds
More
Show state history
Evaluate
Every 10s
Pending period
10s
Last evaluation
a few seconds ago
Evaluation time
5s
Data source
prometheus
Instances
1 firing
2 normal
State
Labels
Created
Alerting
instance cassandra_locationb:9105
+6 common labels
2024-11-30 15:53:10
Normal
instance cassandra_locationa:9104
+6 common labels
2024-11-30 16:04:20
Normal
instance cassandra_locationc:9106
+6 common labels
2024-11-30 16:04:20
Normal
Memory Usage Alert
ok
in a few seconds
More

Restoring Back:



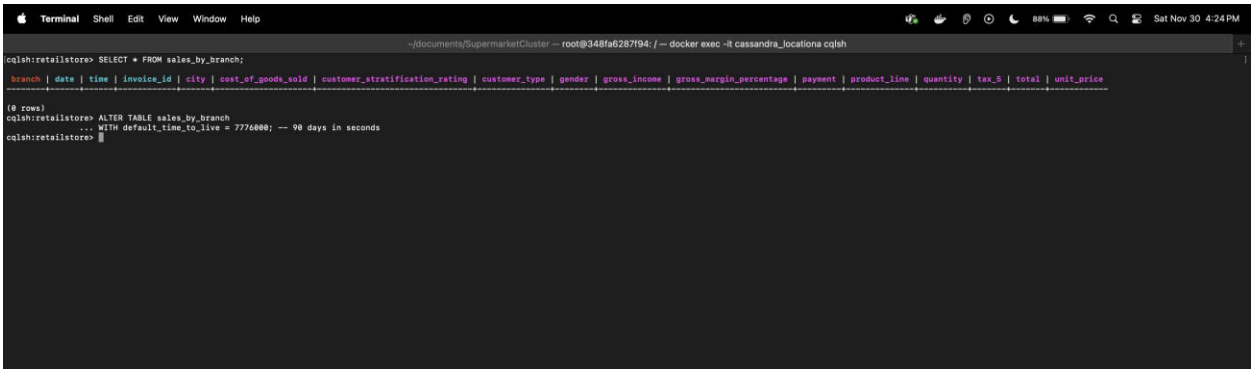


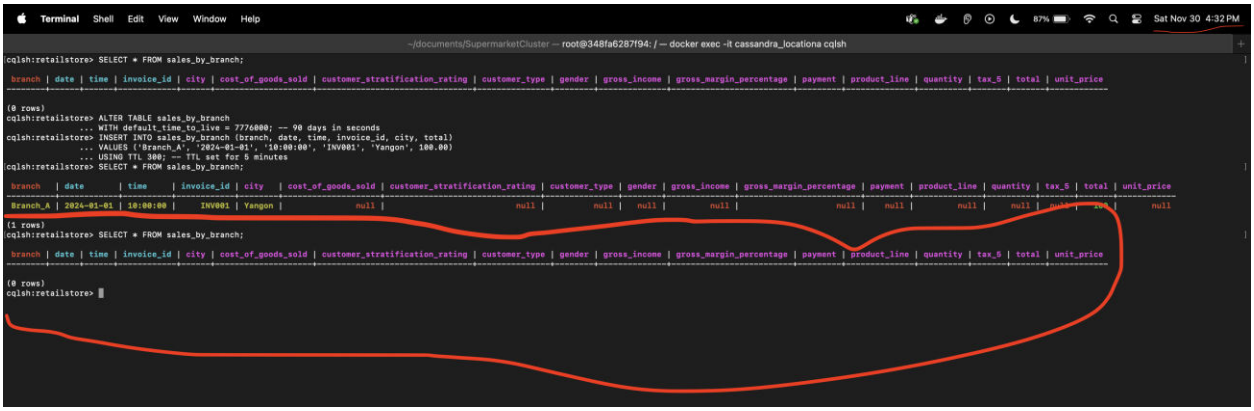
Memory Usgae Alert:


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Sat Nov 30 4:30 PM
localhost:3000/alerting/list
Search or jump to...
Home › Alerting › Alert rules
Home
Bookmarks
Starred
Dashboards
Playlists
Snapshots
Library panels
Public dashboards
Explore
Alerting
Alert rules
Contact points
Notification policies
Silences
Active notifications
Settings
Connections
Add new connection
Data sources
Administration
Normal instance cassandra_locationc:9106 +6 common labels 2024-11-30 16:16:30
Pending for 6s
Memory Usage Alert
ok
in a few seconds
More
Show state history
Evaluate Every 10s
Pending period 10s
Last evaluation a few seconds ago
Evaluation time 0s
Data source
prometheus
Instances
1 pending 5 normal
State Labels Created
Pending area heap instance cassandra_locationa:9104 +5 common labels 2024-11-30 16:29:50
Normal area heap instance cassandra_locationb:9105 +5 common labels 2024-11-30 16:16:40
Normal area heap instance cassandra_locationc:9106 +5 common labels 2024-11-30 16:29:00
Normal area nonheap instance cassandra_locationa:9104 +5 common labels 2024-11-30 16:16:40
Normal area nonheap instance cassandra_locationb:9105 +5 common labels 2024-11-30 16:16:40
Normal area nonheap instance cassandra_locationc:9106 +5 common labels 2024-11-30 16:16:30
Data source-managed
No rules found.
+ New recording rule

Implementing TTL Feature:

Setting up TTL for entire table for 90 days.





References:

Dataset:
https://github.com/plotly/datasets/blob/master/supermarket_Sales.csv