Optimized Batch & Stream Data Processing for Enhanced Inventory Management

Submitted in partial fulfilment of the requirements for the degree of

Post Graduate Diploma in Data Engineering

by

Sudarshan P (G23AI1046)

Utkarsh Gupta (G23AI1048)

Under the guidance of

Dr. Pradip Samal

IIT Jodhpur

Indian Institute of Technology Jodhpur
Advance Data Engineering in Cloud
Trimester-3 (July 2024)

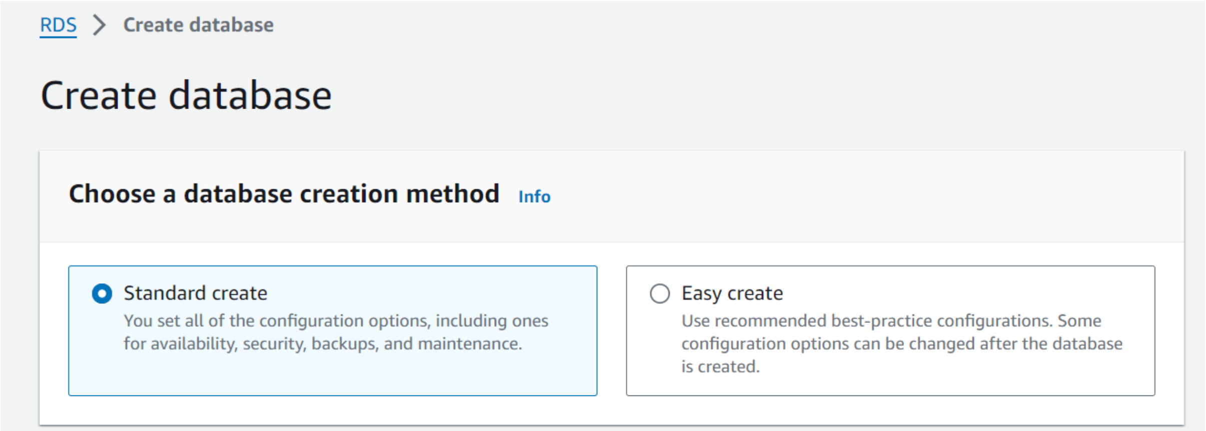
॥ त्वं ज्ञानमयो विज्ञानमयोऽसि ॥

Assignment – 3

https://github.com/utkarshgupta98/advance_data_engineering

Part 5: Create RDS instance

1. Go to RDS console and Create DB instance



RDS > Create database

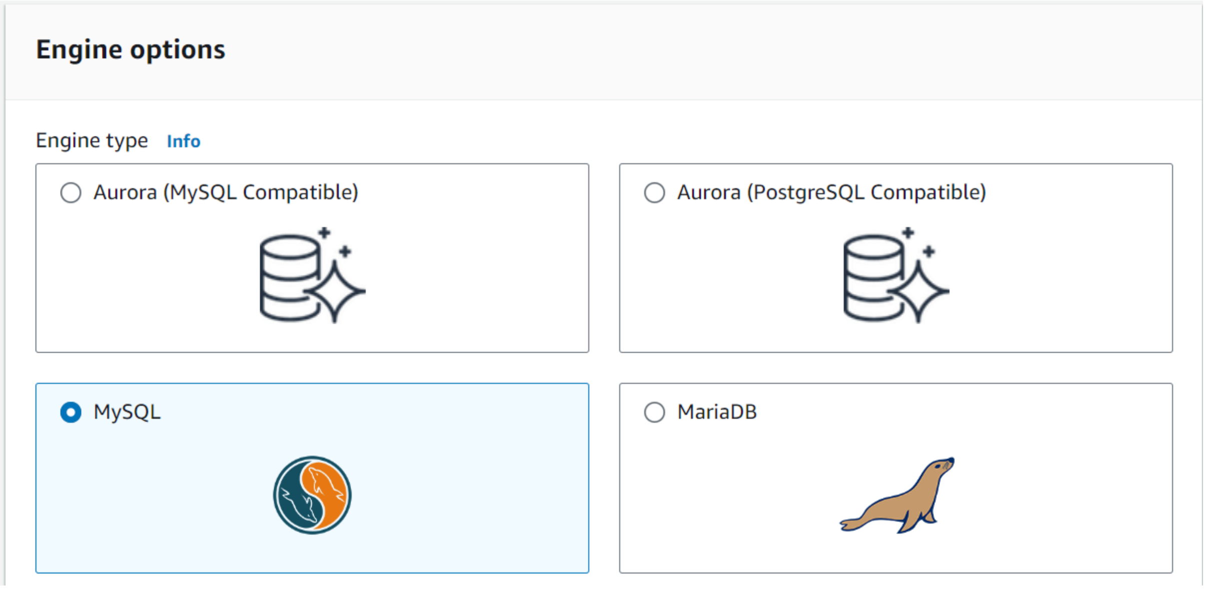
Create database

Choose a database creation method Info

● Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

○ Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

2. Choose MySQL



Engine options

Engine type Info

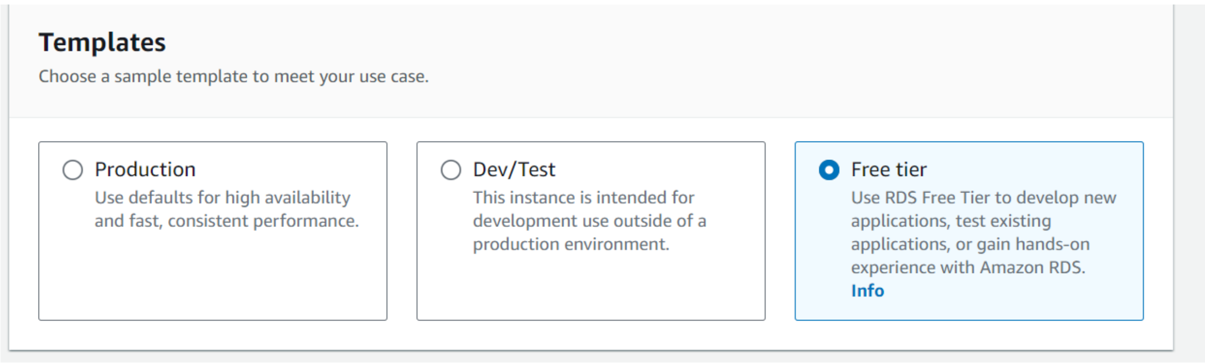
○ Aurora (MySQL Compatible)

○ Aurora (PostgreSQL Compatible)

● MySQL

○ MariaDB

3. Choose Free Tier



Templates

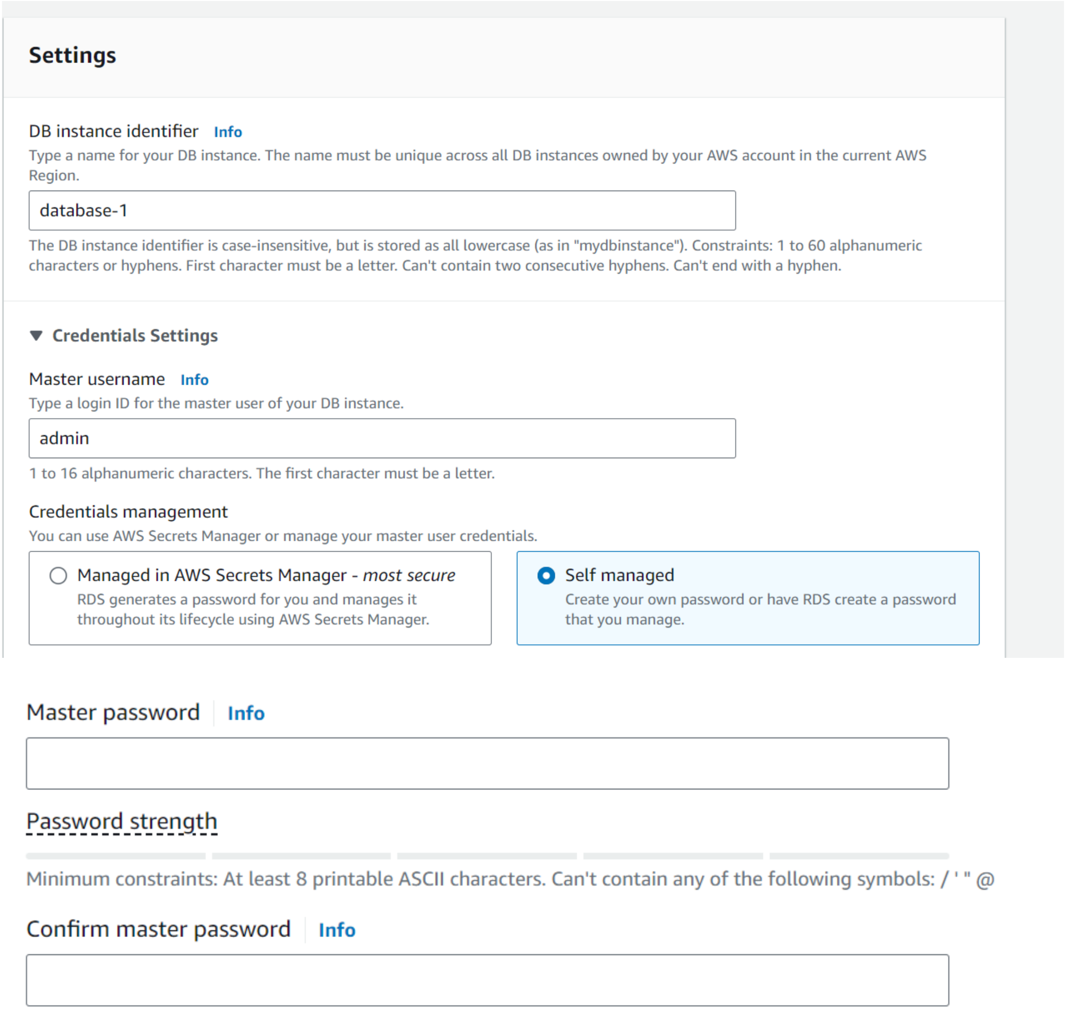
Choose a sample template to meet your use case.

○ Production
Use defaults for high availability and fast, consistent performance.

○ Dev/Test
This instance is intended for development use outside of a production environment.

● Free tier
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.
Info

4. Give DB instance and DB username and password



Settings

DB instance identifier Info
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

database-1

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ Credentials Settings

Master username Info
Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.

○ Managed in AWS Secrets Manager - *most secure*
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

● Self managed
Create your own password or have RDS create a password that you manage.

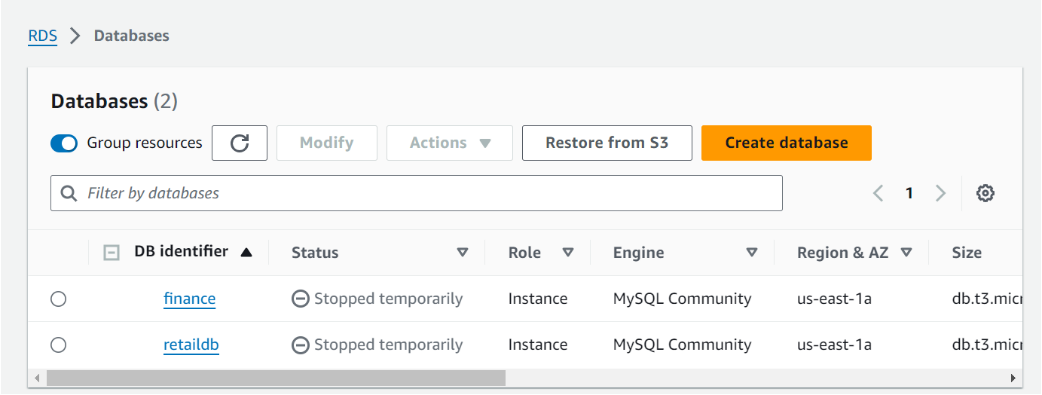
Master password Info

Password strength

Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / ' " @

Confirm master password Info

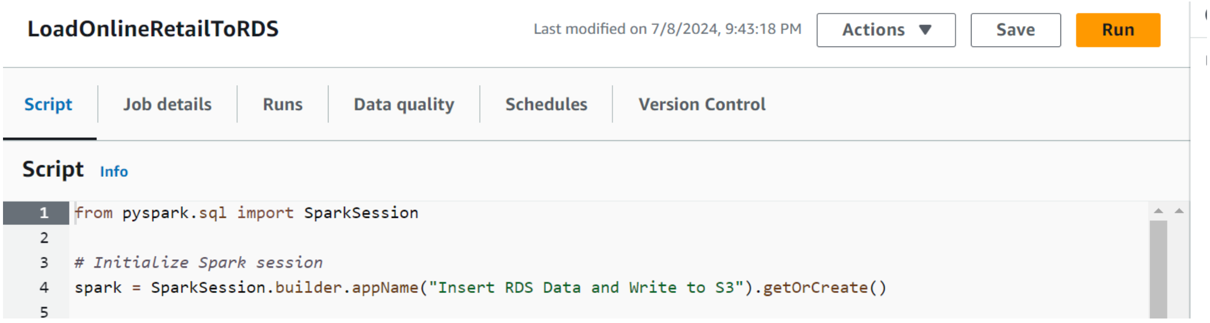
5. Keep everything else as it is.
6. Once the you create it should have status as available (Here I have stopped it for cost purpose)



DB identifier ▲	Status	Role	Engine	Region & AZ	Size
○ finance	Stopped temporarily	Instance	MySQL Community	us-east-1a	db.t3.micr
○ retaildb	Stopped temporarily	Instance	MySQL Community	us-east-1a	db.t3.micr

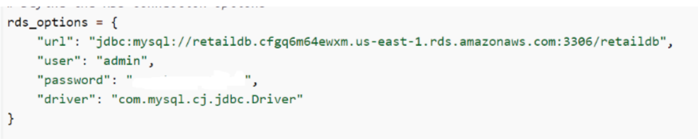
Part 6: Load Transformed Data To RDS

1. Create Spark Glue Job to load s3 data to RDS



```
from pyspark.sql import SparkSession

# Initialize Spark session
spark = SparkSession.builder.appName("Insert RDS Data and Write to S3").getOrCreate()

```

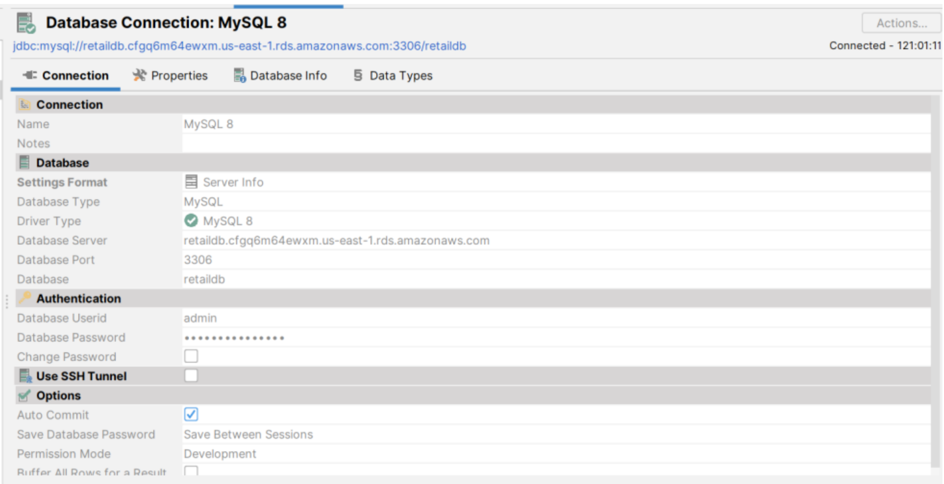
2. Provide connection details



```
rds_options = {
    "url": "jdbc:mysql://retaildb.cfgq6m64ewxm.us-east-1.rds.amazonaws.com:3306/retaildb",
    "user": "admin",
    "password": "[redacted]",
    "driver": "com.mysql.cj.jdbc.Driver"
}
```

3. Once the data is loaded check the data from local as well

    a. Create connection using DBVisualizer



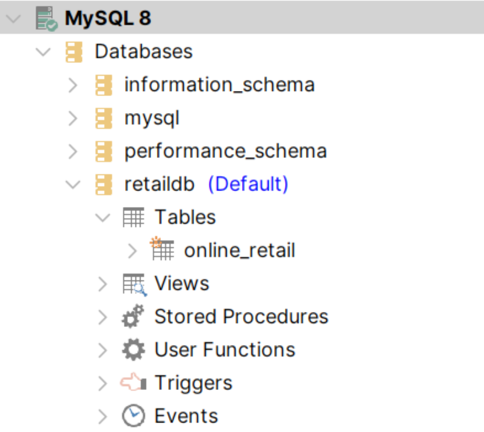
Database Connection: MySQL 8

jdbc:mysql://retaildb.cfgq6m64ewxm.us-east-1.rds.amazonaws.com:3306/retaildb

Connected - 121:01:11

Connection	
Name	MySQL 8
Notes	
Database	
Settings Format	Server Info
Database Type	MySQL
Driver Type	✓ MySQL 8
Database Server	retaildb.cfgq6m64ewxm.us-east-1.rds.amazonaws.com
Database Port	3306
Database	retaildb
Authentication	
Database Userid	admin
Database Password	•••••••••••••••
Change Password	□
Use SSH Tunnel	□
Options	
Auto Commit	☑
Save Database Password	Save Between Sessions
Permission Mode	Development
Buffer All Rows for a Result	□

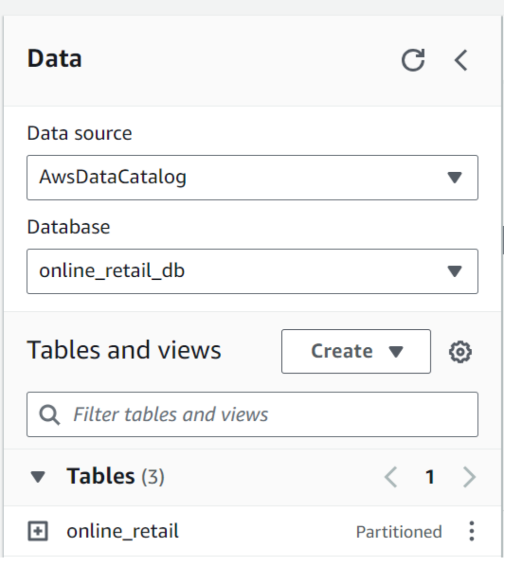
    b. Check if table is created



- **MySQL 8**
  - Databases
    - information_schema
    - mysql
    - performance_schema
    - retaildb (Default)
      - Tables
        - online_retail
      - Views
      - Stored Procedures
      - User Functions
      - Triggers
      - Events

Part 7: Setup Athena and KPIs

1. Go to Athena Console
2. Use the correct database and Table



Data

Data source: AwsDataCatalog

Database: online_retail_db

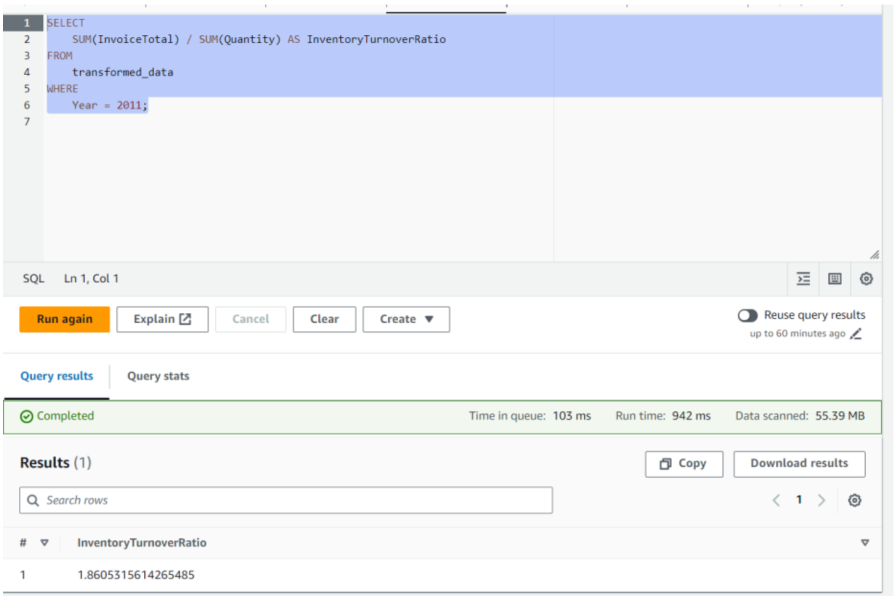
Tables and views

Tables (3)

online_retail	Partitioned

3. Run aggregation KPI queries:

   a. **Inventory Turnover Ratio** measures how efficiently inventory is managed by indicating how many times inventory is sold and replaced over a period.



```
SELECT
    SUM(InvoiceTotal) / SUM(Quantity) AS InventoryTurnoverRatio
FROM
    transformed_data
WHERE
    Year = 2011;
```

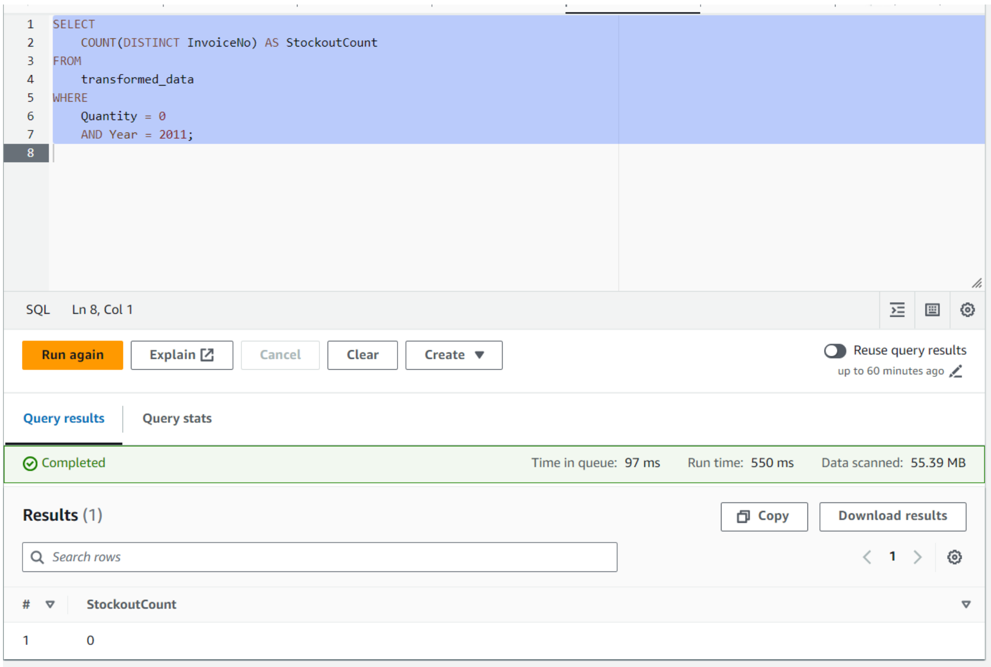
SQL Ln 1, Col 1

Completed Time in queue: 103 ms Run time: 942 ms Data scanned: 55.39 MB

Results (1)

#	InventoryTurnoverRatio
1	1.8605315614265485

b. **Stockout Rate** measures the percentage of time products are out of stock.



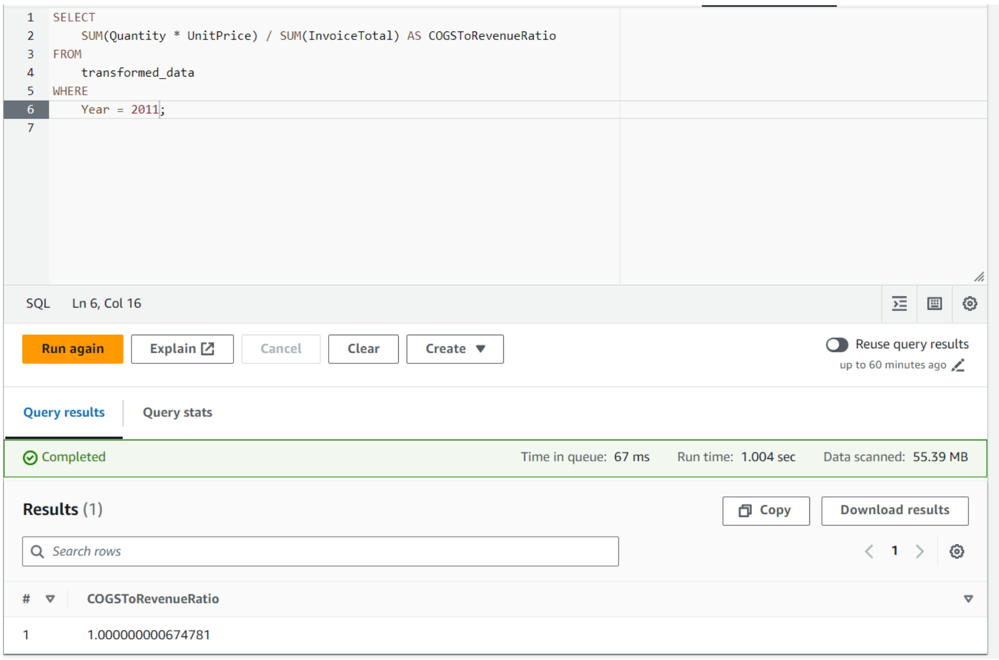
```sql
SELECT
    COUNT(DISTINCT InvoiceNo) AS StockoutCount
FROM
    transformed_data
WHERE
    Quantity = 0
    AND Year = 2011;
```

Completed Time in queue: 97 ms Run time: 550 ms Data scanned: 55.39 MB

Results (1)

#	StockoutCount
1	0

c. **COGS to Revenue Ratio** measures the efficiency of managing inventory costs relative to revenue.



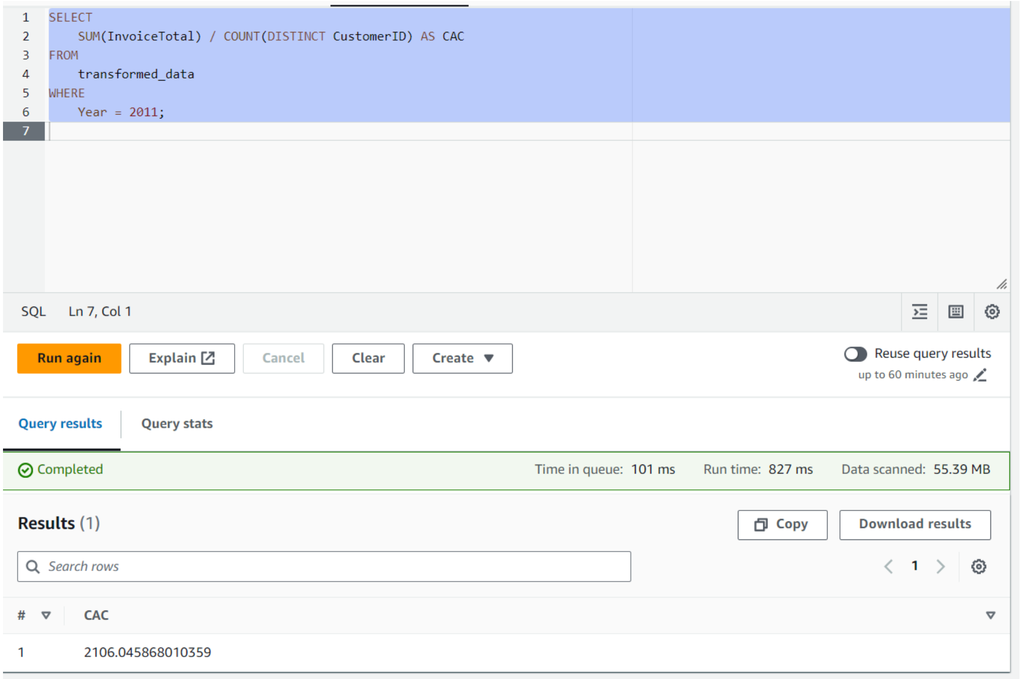
```sql
SELECT
    SUM(Quantity * UnitPrice) / SUM(InvoiceTotal) AS COGSToRevenueRatio
FROM
    transformed_data
WHERE
    Year = 2011;
```

Completed Time in queue: 67 ms Run time: 1.004 sec Data scanned: 55.39 MB

Results (1)

#	COGSToRevenueRatio
1	1.000000000674781

d. **Customer Acquisition Cost (CAC)** measures the average cost of acquiring a new customer.



```
SELECT
    SUM(InvoiceTotal) / COUNT(DISTINCT CustomerID) AS CAC
FROM
    transformed_data
WHERE
    Year = 2011;
```

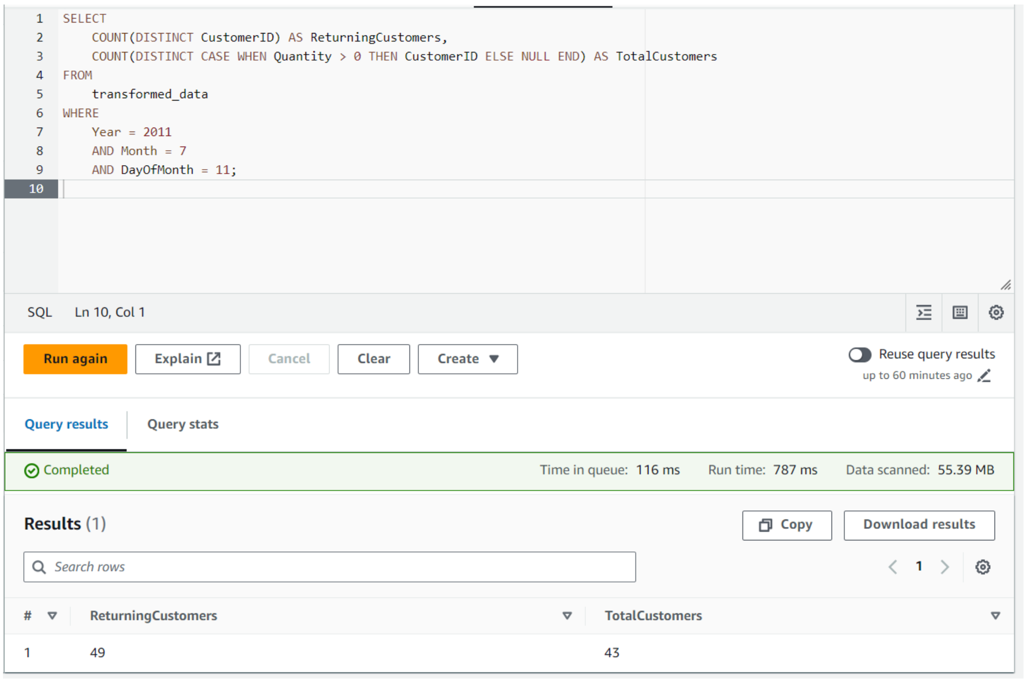
SQL Ln 7, Col 1

Completed Time in queue: 101 ms Run time: 827 ms Data scanned: 55.39 MB

Results (1)

#	CAC
1	2106.045868010359

e. **Returning Customers**: The total number of unique customers who made purchases on July 11, 2011.



```
SELECT
    COUNT(DISTINCT CustomerID) AS ReturningCustomers,
    COUNT(DISTINCT CASE WHEN Quantity > 0 THEN CustomerID ELSE NULL END) AS TotalCustomers
FROM
    transformed_data
WHERE
    Year = 2011
    AND Month = 7
    AND DayOfMonth = 11;
```

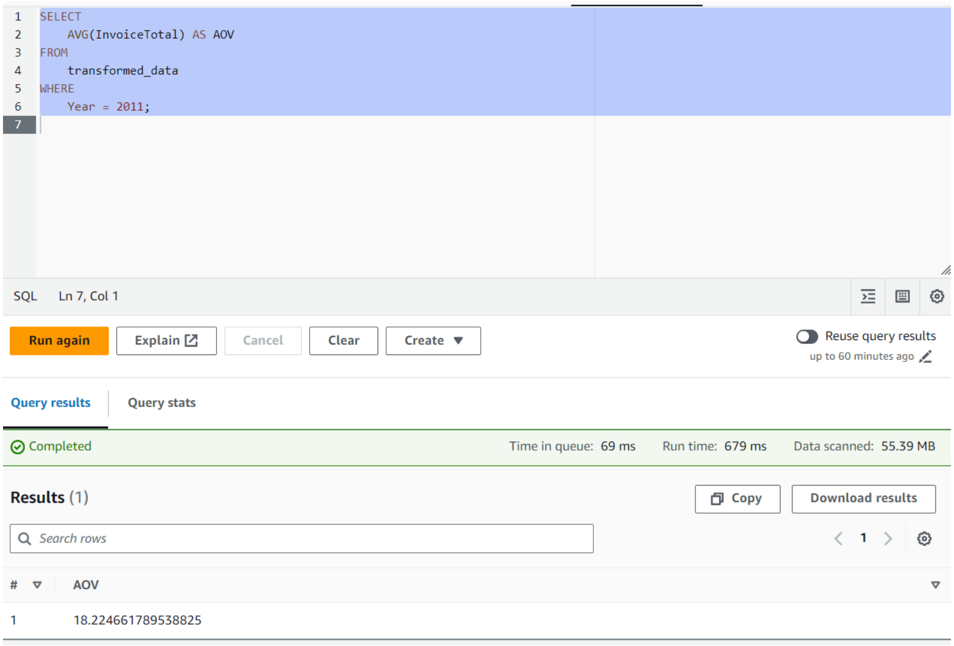
SQL Ln 10, Col 1

Completed Time in queue: 116 ms Run time: 787 ms Data scanned: 55.39 MB

Results (1)

#	ReturningCustomers	TotalCustomers
1	49	43

f. **Average Order Value (AOV)**: Measures the average amount spent per order over the course of the year 2011.

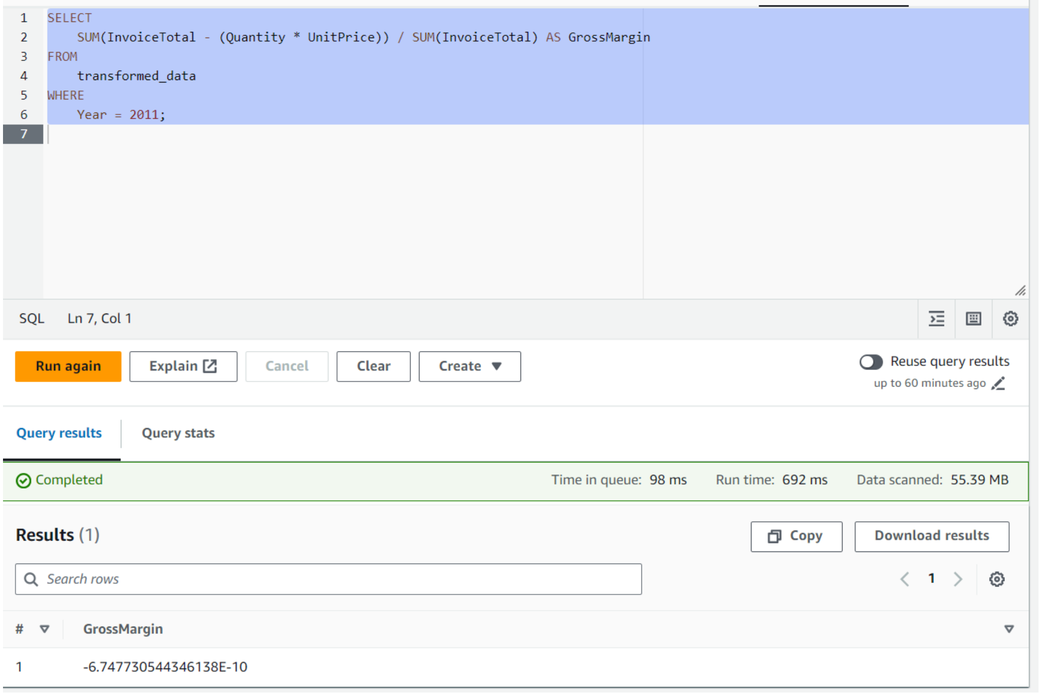


```
SELECT
    AVG(InvoiceTotal) AS AOV
FROM
    transformed_data
WHERE
    Year = 2011;

```

Completed Time in queue: 69 ms Run time: 679 ms Data scanned: 55.39 MB

#	AOV
1	18.224661789538825

g. **Gross Margin**: Measures the profitability of products by comparing the revenue (InvoiceTotal) to the cost of goods sold (Quantity * UnitPrice).



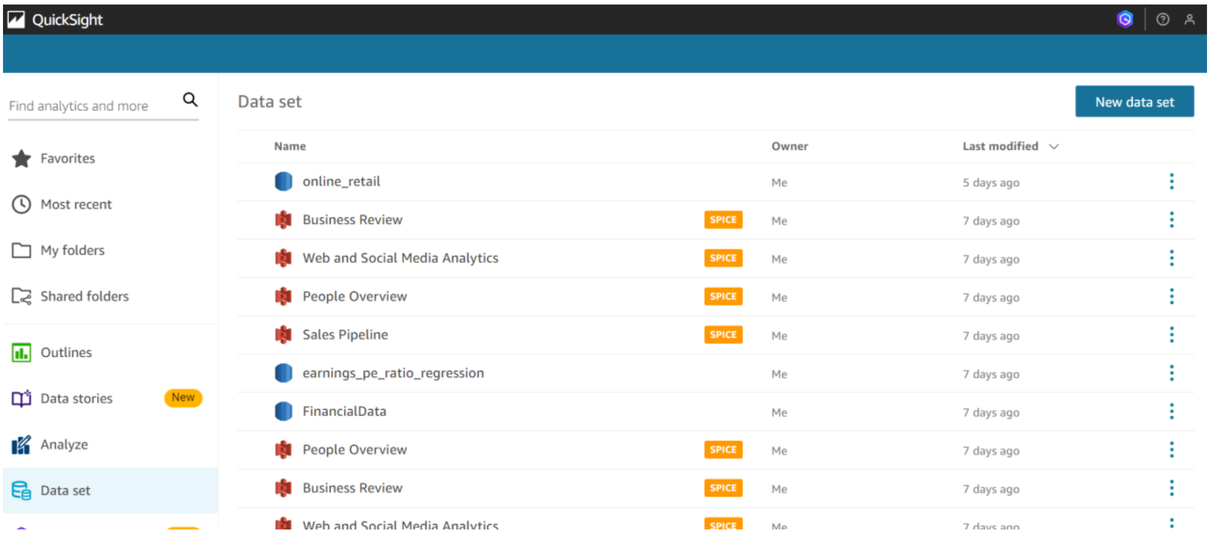
```
SELECT
    SUM(InvoiceTotal - (Quantity * UnitPrice)) / SUM(InvoiceTotal) AS GrossMargin
FROM
    transformed_data
WHERE
    Year = 2011;

```

Completed Time in queue: 98 ms Run time: 692 ms Data scanned: 55.39 MB

#	GrossMargin
1	-6.747730544346138E-10

Part 8: Quicksight and Visualizations

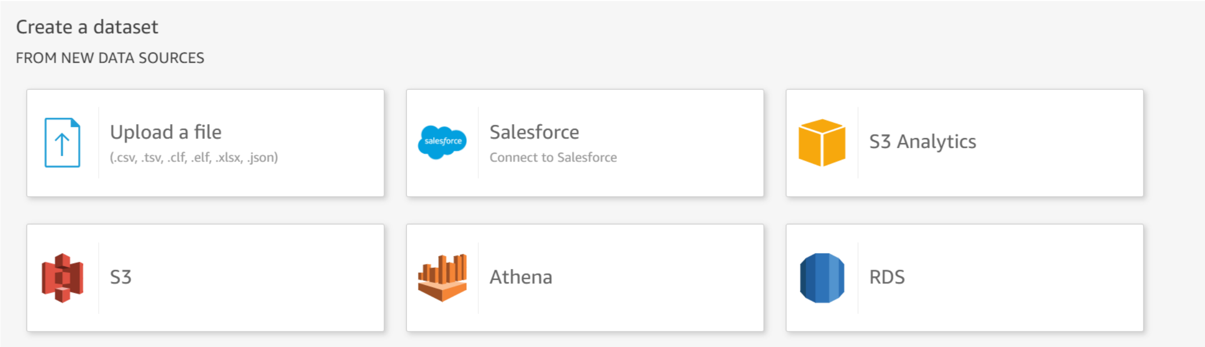
1. Go to Quicksight and login
2. Choose the dataset to have visualization on



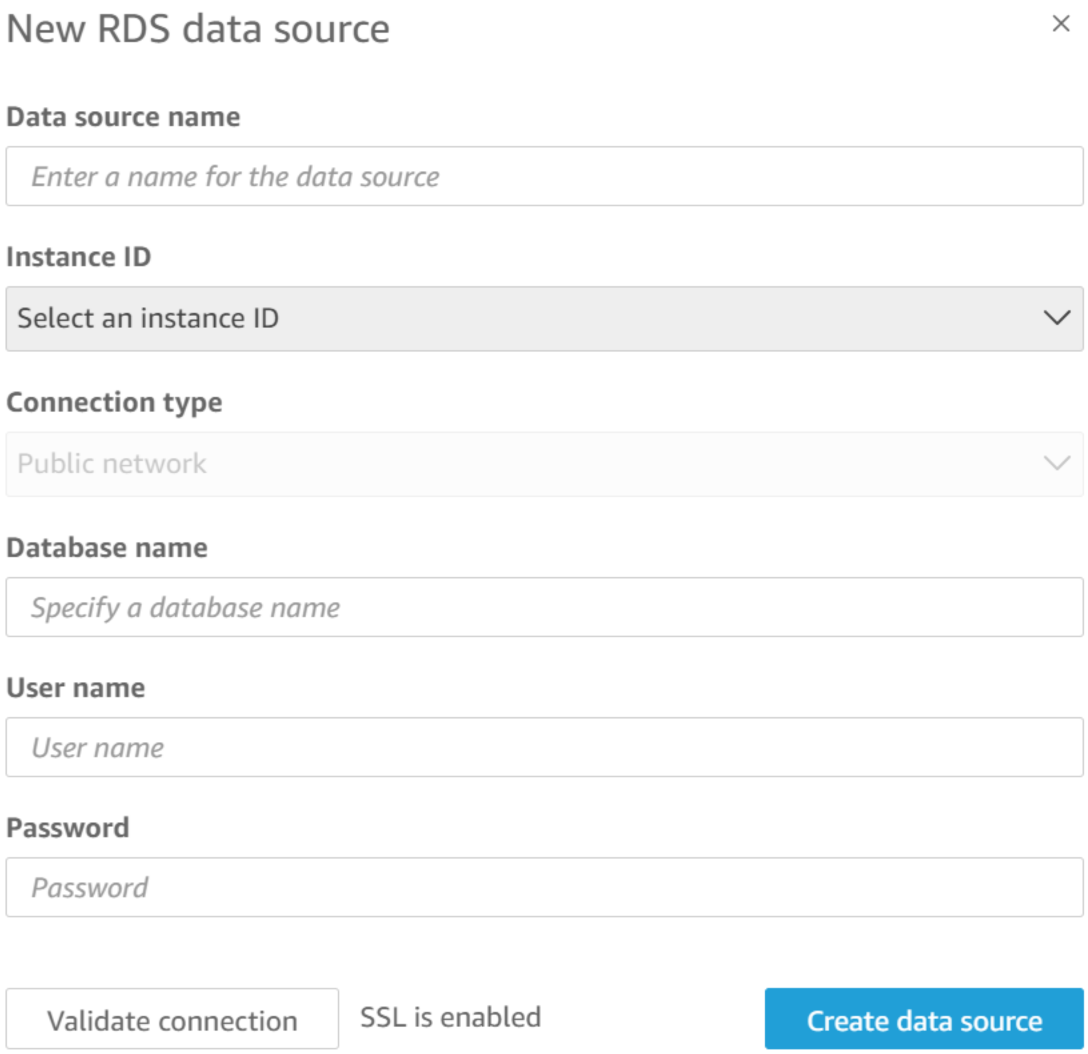
Data set

Name		Owner	Last modified
online_retail		Me	5 days ago
Business Review	SPICE	Me	7 days ago
Web and Social Media Analytics	SPICE	Me	7 days ago
People Overview	SPICE	Me	7 days ago
Sales Pipeline	SPICE	Me	7 days ago
earnings_pe_ratio_regression		Me	7 days ago
FinancialData		Me	7 days ago
People Overview	SPICE	Me	7 days ago
Business Review	SPICE	Me	7 days ago
Web and Social Media Analytics	SPICE	Me	7 days ago

3. Click on new dataset and choose RDS



4. Enter RDS details and test the connection



New RDS data source

Data source name

Enter a name for the data source

Instance ID

Select an instance ID

Connection type

Public network

Database name

Specify a database name

User name

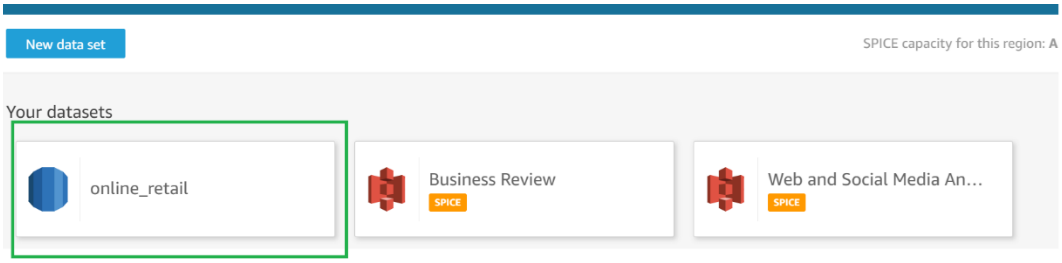
User name

Password

Password

SSL is enabled

5. Once created, it will be listed in the datasets



SPICE capacity for this region: A

Your datasets

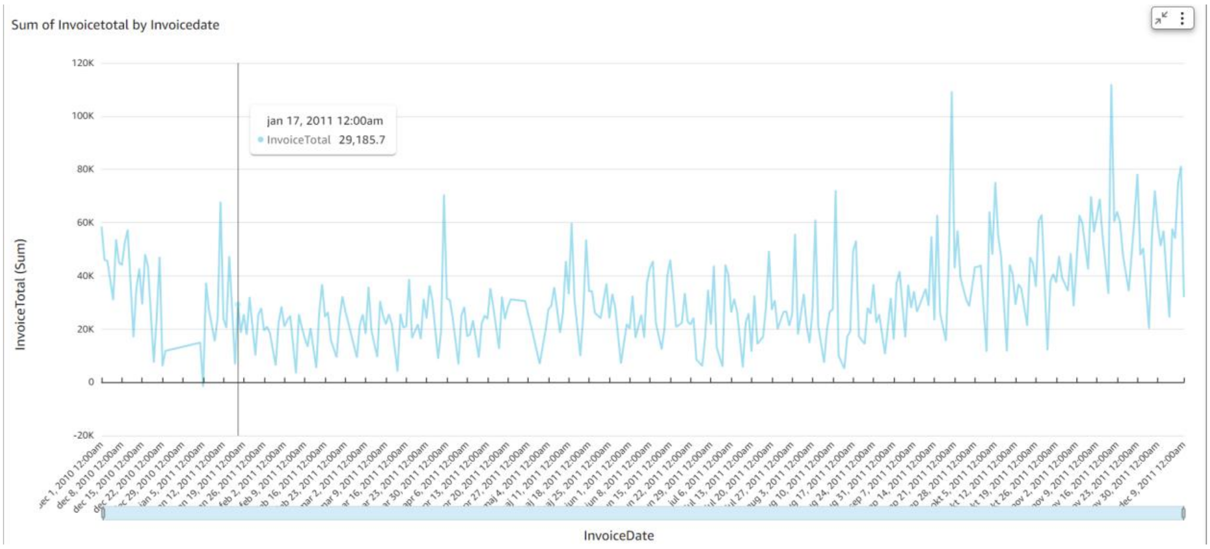
online_retail

Business Review SPICE

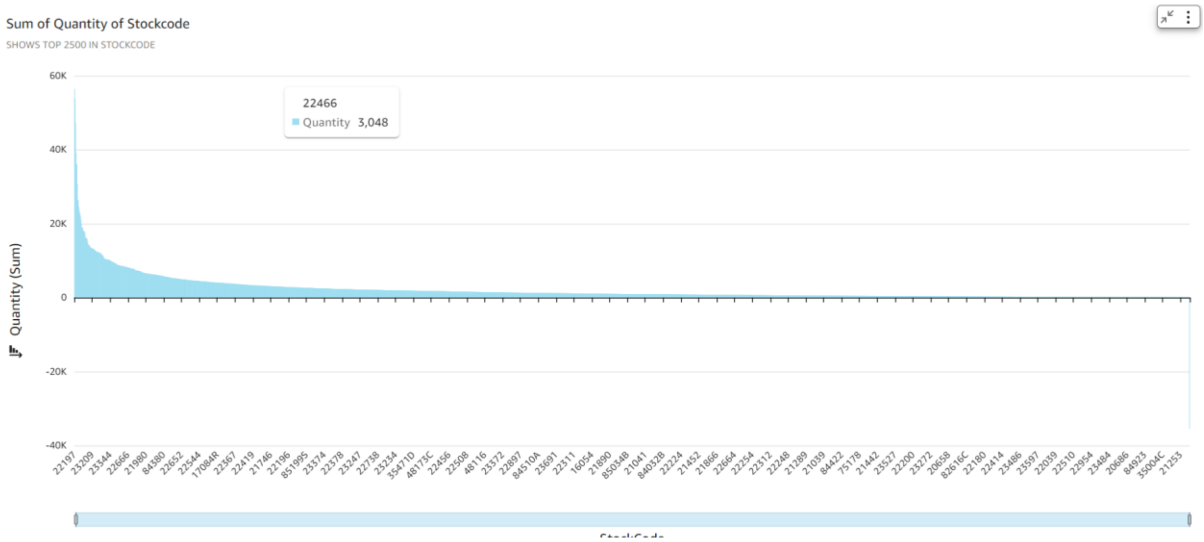
Web and Social Media An... SPICE

6. Create Dashboard

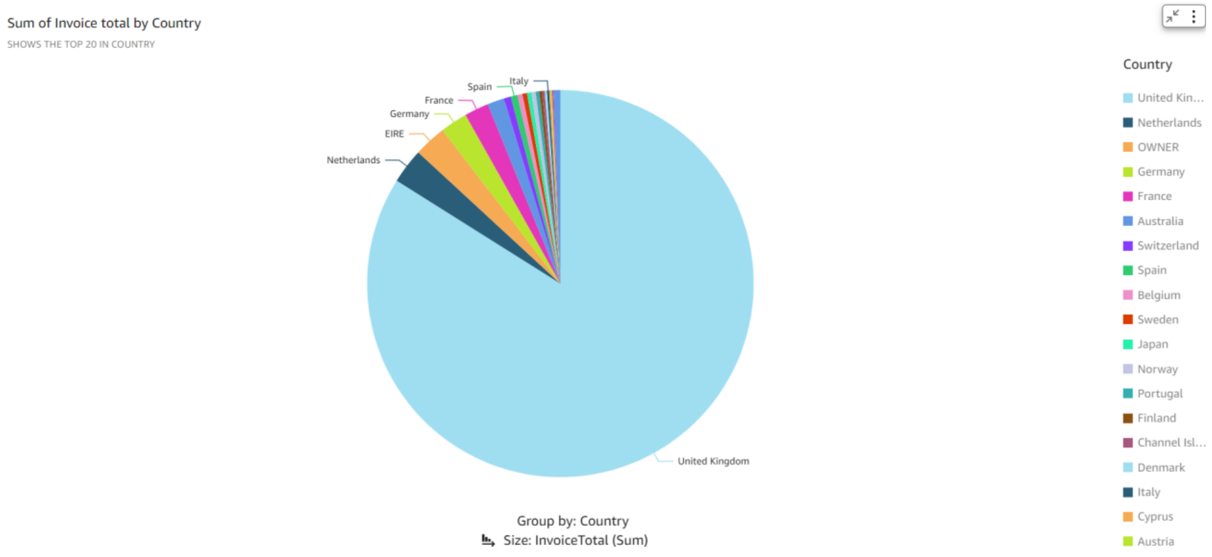
a. Sum of Invoice total by invoice date



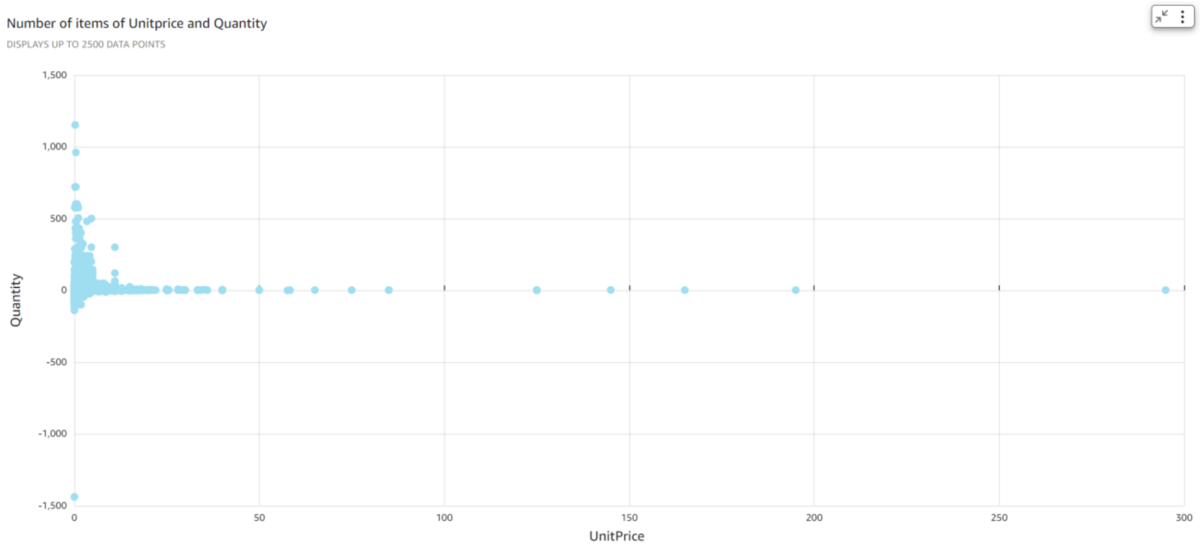
b. Sum of Quantity of Stockcode



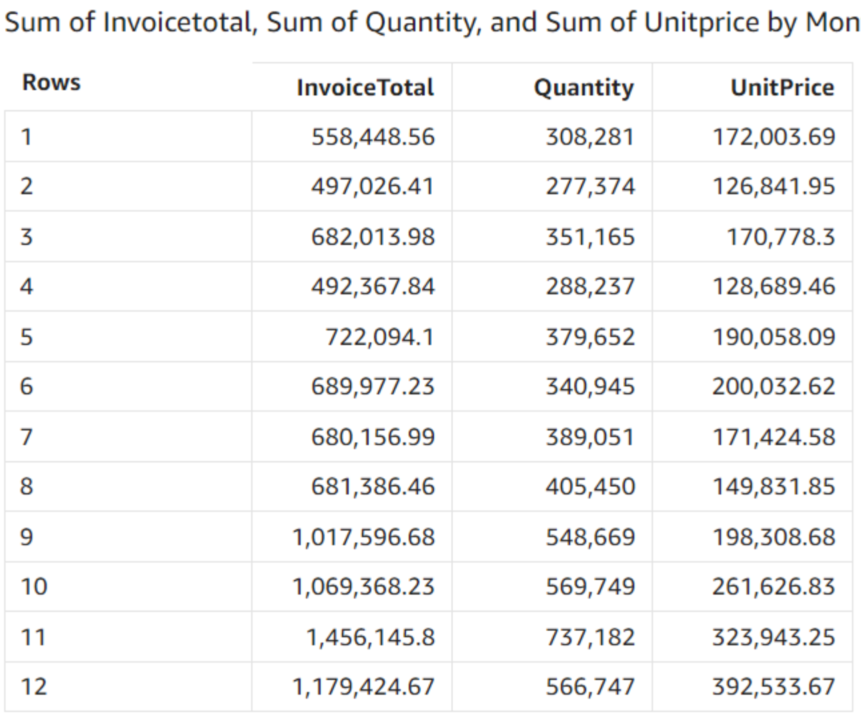
c. Sum of invoice total by country



d. Number of items of Unitprice and Quantity



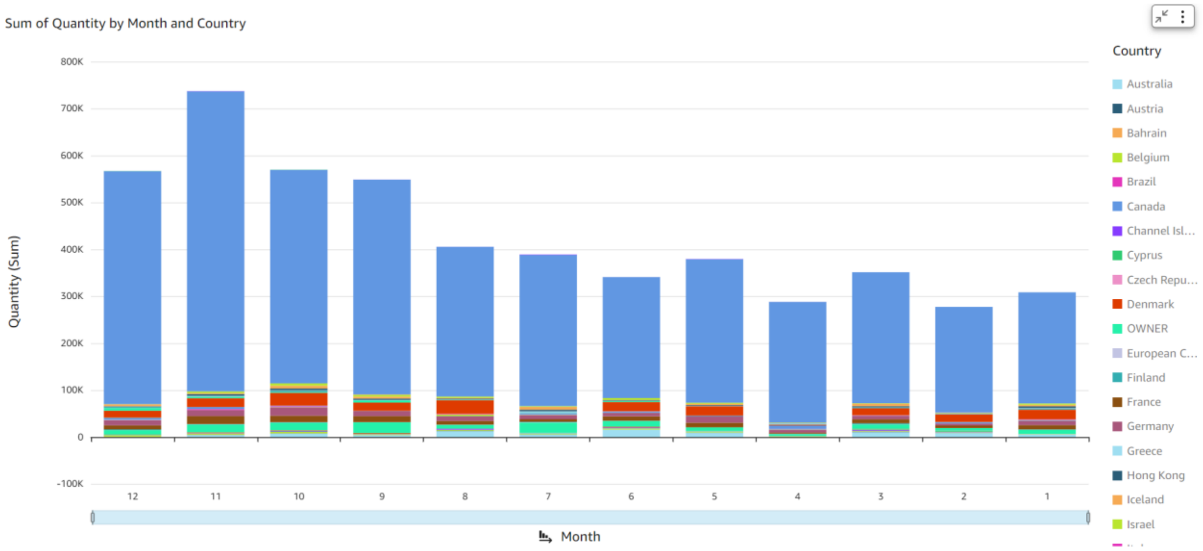
e. Sum of Invoicetotal, Sum of Quantity, and Sum of Unitprice by Month



Sum of Invoicetotal, Sum of Quantity, and Sum of Unitprice by Month

Rows	InvoiceTotal	Quantity	UnitPrice
1	558,448.56	308,281	172,003.69
2	497,026.41	277,374	126,841.95
3	682,013.98	351,165	170,778.3
4	492,367.84	288,237	128,689.46
5	722,094.1	379,652	190,058.09
6	689,977.23	340,945	200,032.62
7	680,156.99	389,051	171,424.58
8	681,386.46	405,450	149,831.85
9	1,017,596.68	548,669	198,308.68
10	1,069,368.23	569,749	261,626.83
11	1,456,145.8	737,182	323,943.25
12	1,179,424.67	566,747	392,533.67

f. Sum of Quantity by Month and Country


Sum of Quantity by Month and Country
800K
700K
600K
500K
400K
300K
200K
100K
-100K
Quantity (Sum)
Month
Country
Australia
Austria
Bahrain
Belgium
Brazil
Canada
Channel Isl...
Cyprus
Czech Repu...
Denmark
OWNER
European C...
Finland
France
Germany
Greece
Hong Kong
Iceland
Israel