Learning Amazon Athena

with Uduak Sonuyi

LEARNING

Solution

Step 1

Optionally, create a new S3 bucket.

Create a new folder for your CSV file.

Step 2

First, create a crawler to pull in GamCentr data.

Step 3

View the data in Athena and try out these answers.

To view the data:

AWS Glue > Data Catalog > Databases > Tables > Line item for GamCentr table > Click on Table data

Question 1: Create Tables

Create three tables in Athena with the following data:

1. Customer with customer_id, c_first_name, c_last_name columns
2. Salesperson with salesperson_id, s_first_name, s_last_name columns
3. Gameorder with order_id, order_date, total_due, status,customer_id, salesperson_id columns

In your query, consider the possibility of duplicates.

Answer:

```
CREATE TABLE customer

AS

SELECT DISTINCT  customer_id, c_first_name, c_last_name

FROM GamCentr

CREATE TABLE salesperson

AS

SELECT DISTINCT salesperson_id, s_first_name, s_last_name

FROM GamCentr
```

Note: You may use the DESCRIBE keyword on your crawler-created table to view column names.

```
CREATE TABLE Gameorder

AS

SELECT order_id, order_date, total_due,
status,customer_id,salesperson_id

FROM GamCentr
```

Note: “Order” is a keyword. Either use a different name or surround the name in double quotation marks if you choose to use “Order” in creating/selecting tables in Athena.

Question 2: Create a View

Create a view called orders_past_due with the following considerations:

1. Show total amount due per year to two decimal places.
2. Use the status column to find “past due” order payments.
3. You may need to format your date column so it can be read in date format. Use the date_parse function for this.

Answer:

```
CREATE VIEW orders_past_due

AS

SELECT year(date_parse(order_date, ‘%d-%m-%Y’)) AS year,
ROUND(SUM(total_due),2) AS total_due_per_year

FROM GamCentr

WHERE status like ‘past due’

GROUP BY year(date_parse(order_date, ‘%d-%m-%Y’))
```

Question 3: Insight

Who has made the highest number of sales (without considering the total sale value per salesperson)?

Show the number of sales for such a salesperson.

Answer (version 1 using WITH clause):

```
WITH sales_volume

AS

    (SELECT salesperson_id, COUNT(salesperson_id) AS
no_of_transactions
```

```
    FROM Gameorder
    GROUP BY salesperson_id) SOURCE CODE PRO
SELECT sp.s_first_name, sp.s_last_name, sv.no_of_transactions
FROM sales_volume sv
JOIN salesperson sp
ON sv.salesperson_id = sp.salesperson_id
WHERE sv.no_of_transactions = (SELECT MAX(no_of_transactions) from
sales_volume) SOURCE PRO
```

Answer (version 2 using subqueries):

```
SELECT sp.s_first_name, sp.s_last_name, COUNT(gm.salesperson_id) AS
no_of_transactions
FROM salesperson sp
JOIN Gameorder gm
ON sp.salesperson_id = gm.salesperson_id
GROUP BY sp.s_first_name, sp.s_last_name
HAVING COUNT(gm.salesperson_id) = (SELECT MAX(no_of_transactions)
                                   FROM
                                    (SELECT count(salesperson_id) AS no_of_transactions
                                     FROM Gameorder
                                     GROUP BY salesperson_id))
```