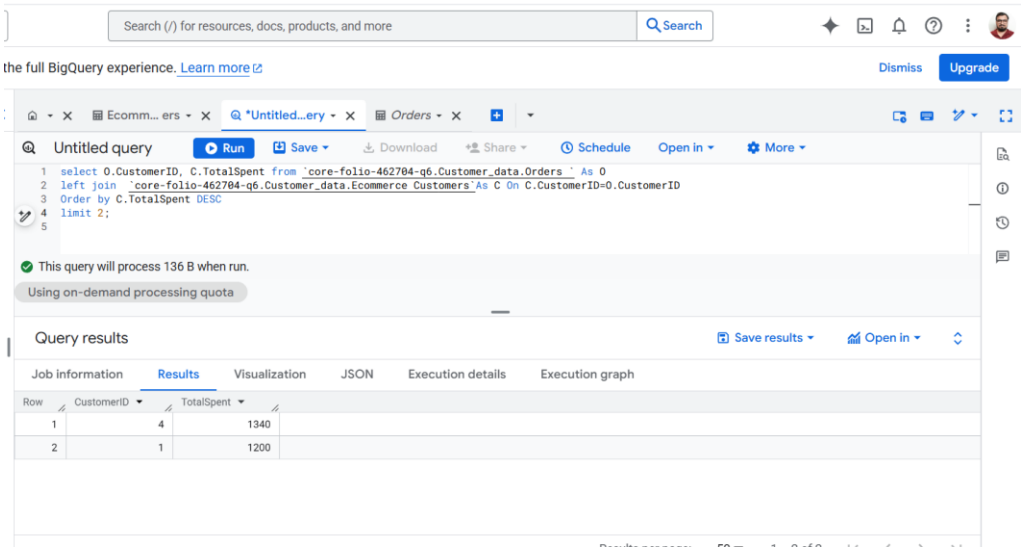
SQL ADVANCE QUESTION ON DATASET

1) Find the top 2 customers by total spent in Orders.

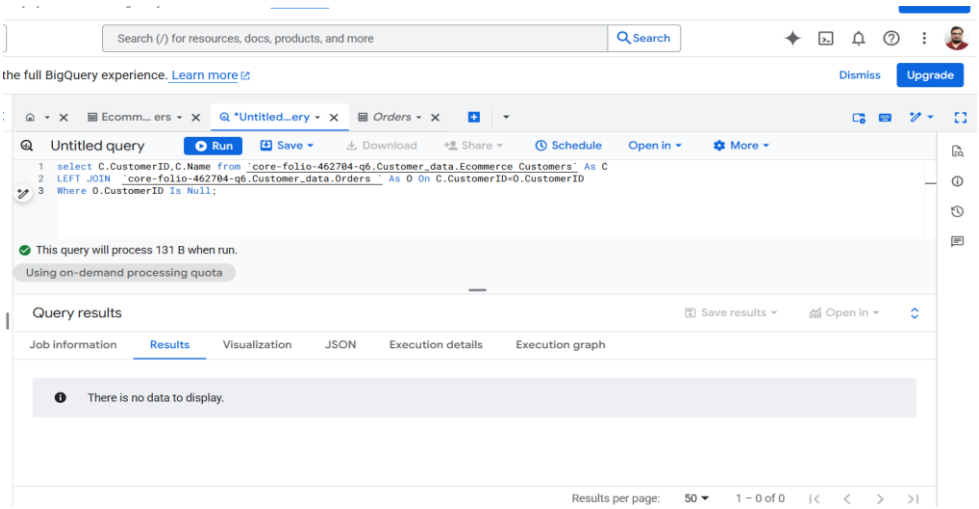
Query:- select O.CustomerID, C.TotalSpent from `core-folio-462704-q6.Customer_data.Orders ` As O
left join `core-folio-462704-q6.Customer_data.Ecommerce Customers`As C On C.CustomerID=O.CustomerID
Order by C.TotalSpent DESC
limit 2;



2) Identify customers who have not ordered any product.

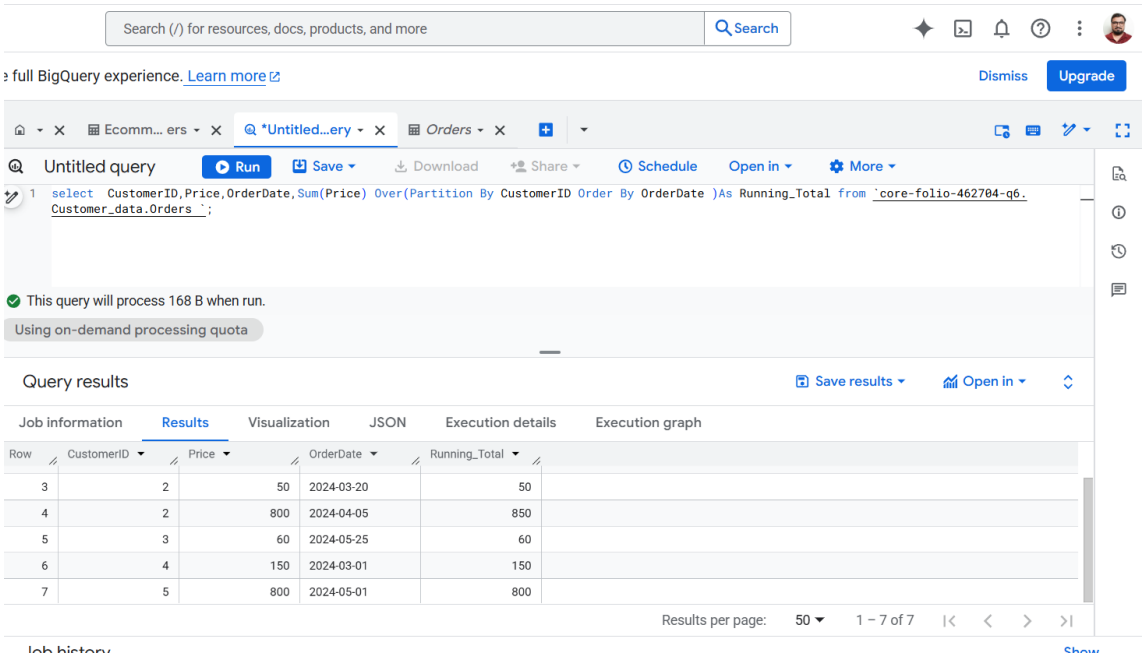
QUERY:- select C.CustomerID,C.Name from `core-folio-462704-q6.Customer_data.Ecommerce Customers` As C
LEFT JOIN `core-folio-462704-q6.Customer_data.Orders ` As O On C.CustomerID=O.CustomerID
Where O.CustomerID Is Null;

NOTE:-There is no null value in both the table that's why the result showing there is no data to display.



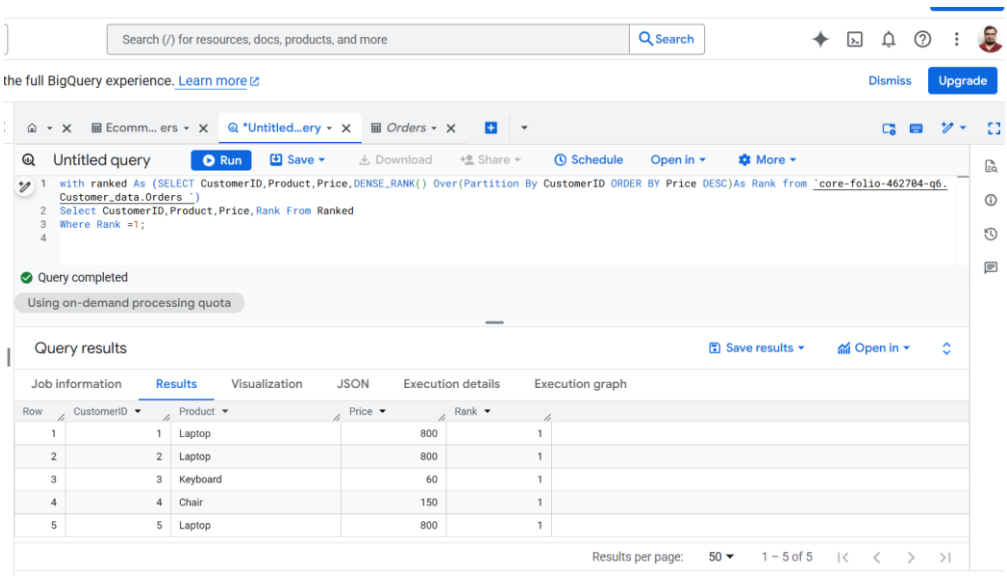
3) Calculate the **running total of price** ordered by each customer by OrderDate.

QUERY :- select CustomerID,Price,OrderDate,Sum(Price) Over(Partition By CustomerID Order By OrderDate)As Running_Total from `core-folio-462704-q6.Customer_data.Orders `;



4) Find the most expensive product ordered by each customer.

QUERY :- with ranked As (SELECT CustomerID,Product,Price,DENSE_RANK() Over(Partition By CustomerID ORDER BY Price DESC)As Expensive from `core-folio-462704-q6.Customer_data.Orders `)
Select CustomerID,Product,Price From Ranked
Where Expensive =1;



5) List customers along with the number of distinct products they purchased.

QUERY :- Select C.Name,O.CustomerID, COUNT(DISTINCT O.Product) AS Distinct_Products_Purchased,CONCAT(COUNT(DISTINCT O.Product), ' (', STRING_AGG(DISTINCT O.Product, ', '), ')') AS Product_List_Display From `core-folio-462704-q6.Customer_data.Orders ` AS O
left join `core-folio-462704-q6.Customer_data.Ecommerce Customers` As C ON C.CustomerID=O.CustomerID
Group By C.Name,O.CustomerID;