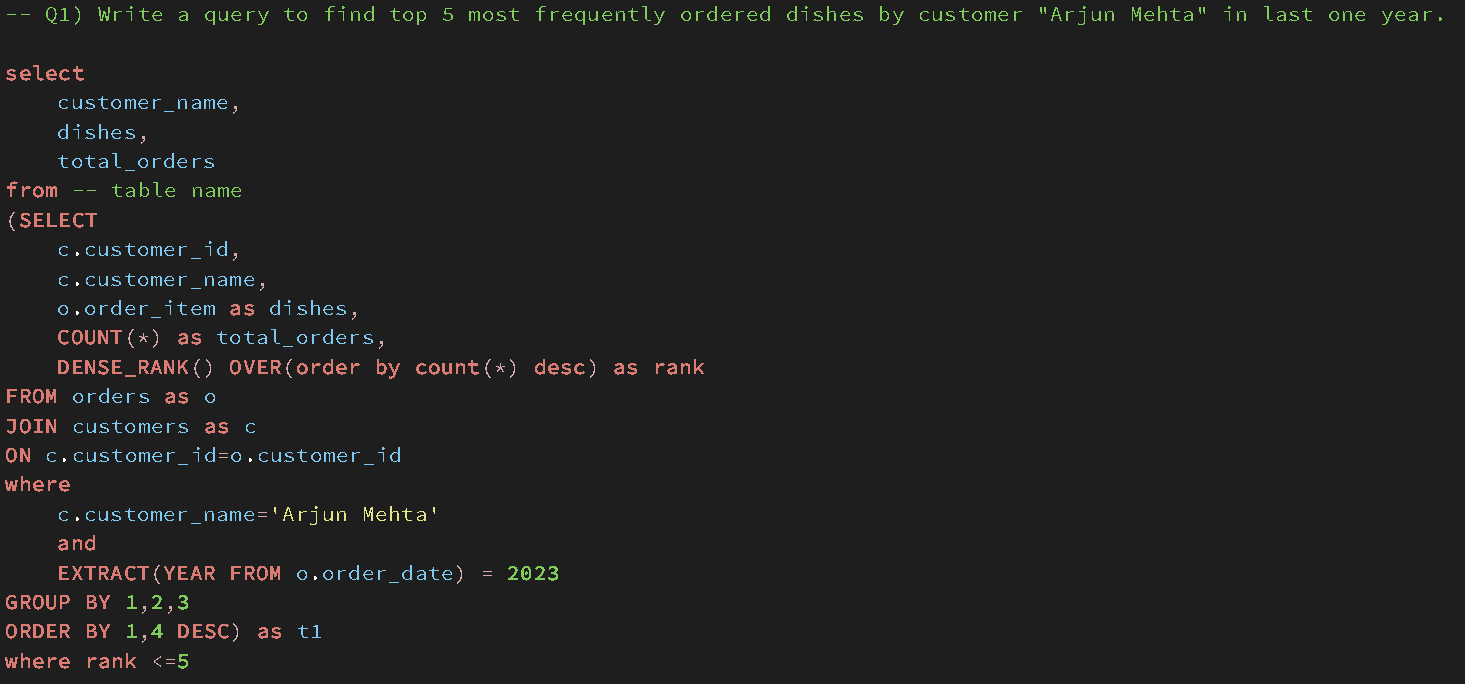
Q1) **Query to find the top 5 most ordered dishes by “Arjun Mehta”**

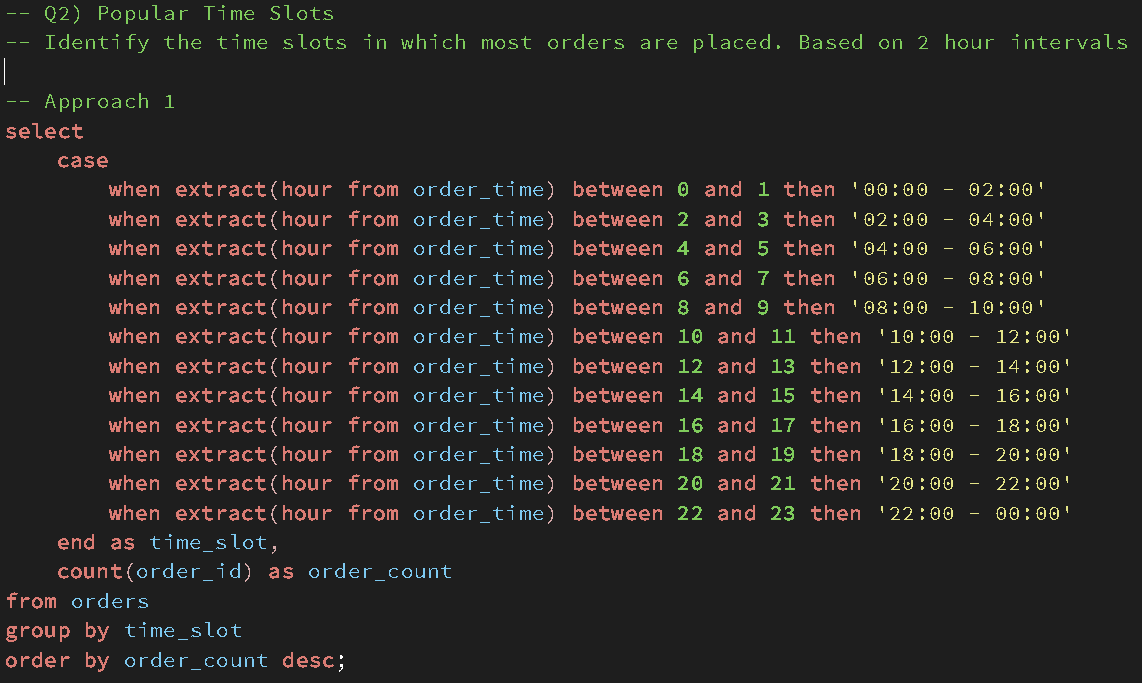


**Answer**



Q2) **Popular Time Slots - Identify the time slots in which most orders are placed. Based on 2 hour intervals**

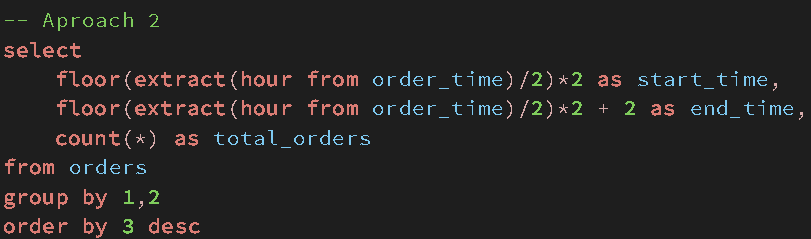
**Approach 1**

****

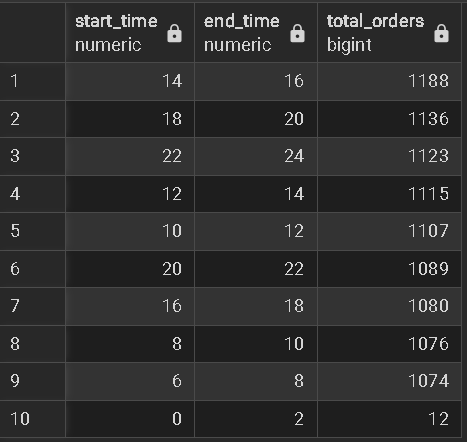
**Output**

****

**Approach 2**

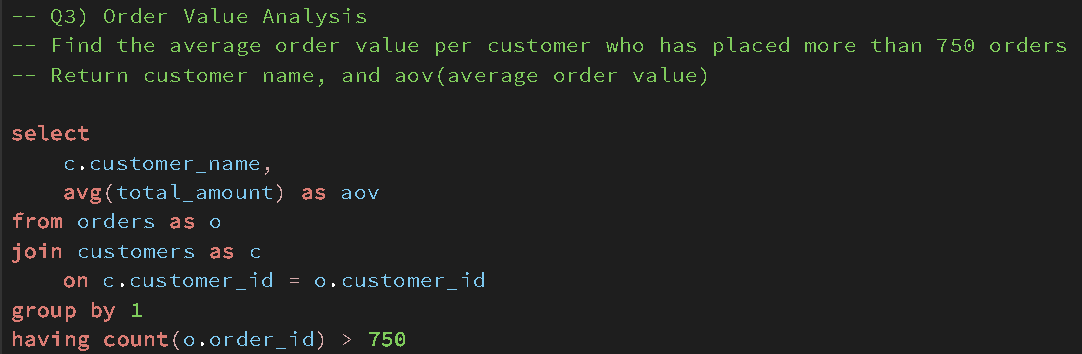
****

**Output**

****

**Q3) Order Value Analysis - Find the average order value per customer who has placed more than 750 orders**

**Return customer name, and aov(average order value)**

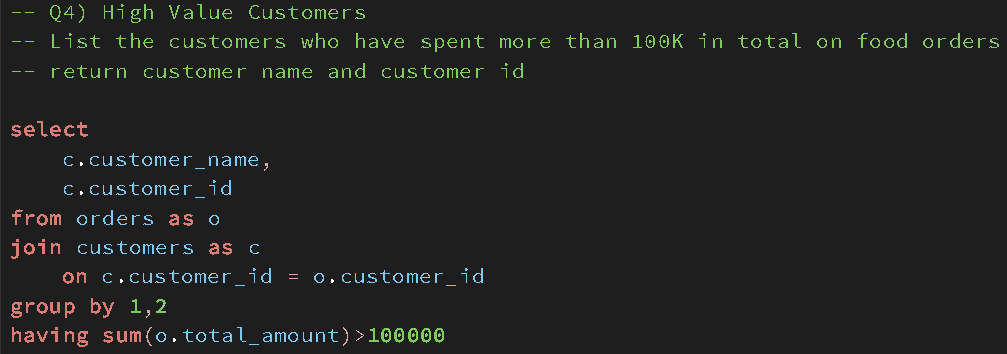
****

**Output**

****

**Q4) Q4) High Value Customers - List the customers who have spent more than 100K in total on food orders**

**Return customer name and customer id**

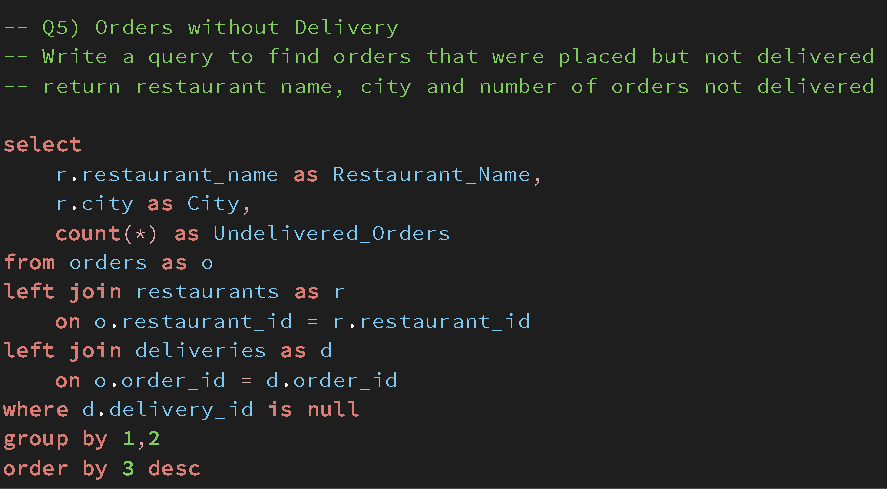
****

**Output**

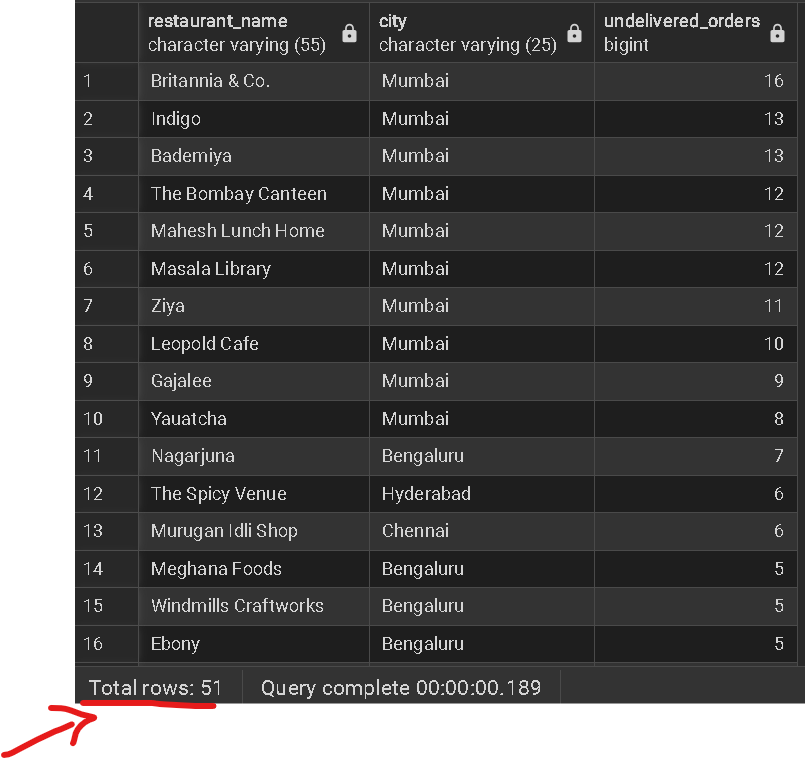
****

**Q5) Orders without Delivery - Write a query to find orders that were placed but not delivered**

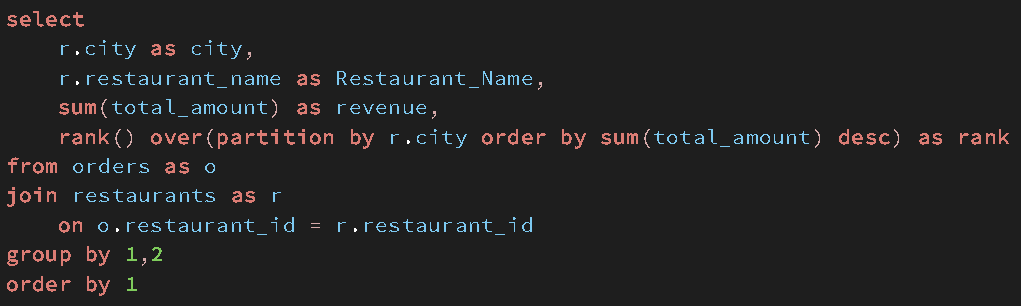
**return restaurant name, city and number of orders not delivered**



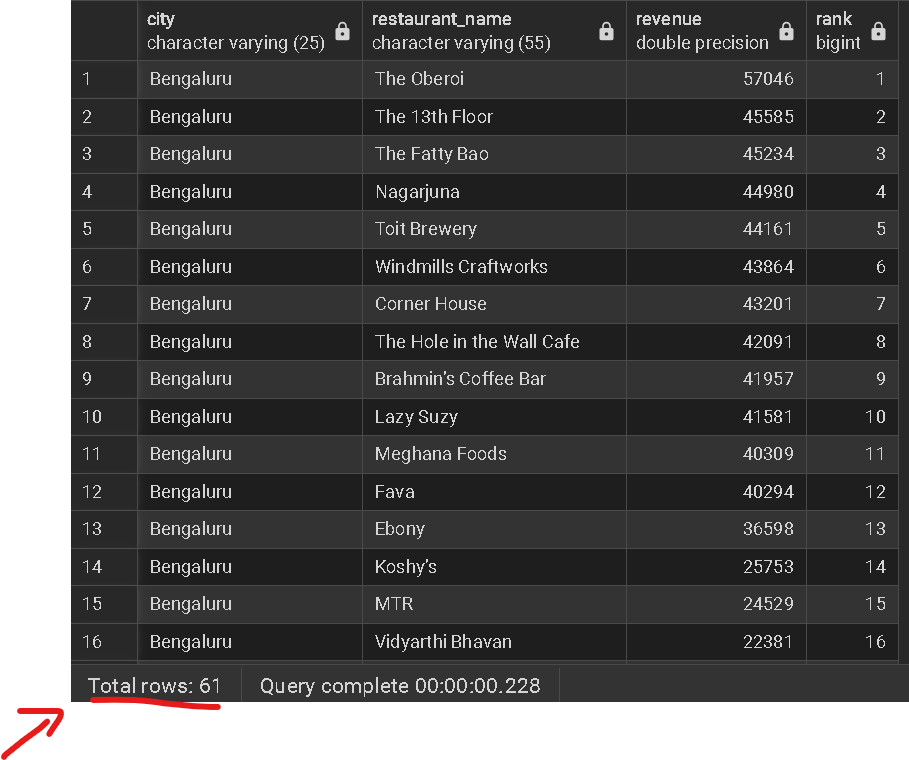
**Output**



**Q6) Restaurant Revenue Ranking - Rank the restaurants by their total revenue last year, including their name, total revenue and rank within their city**

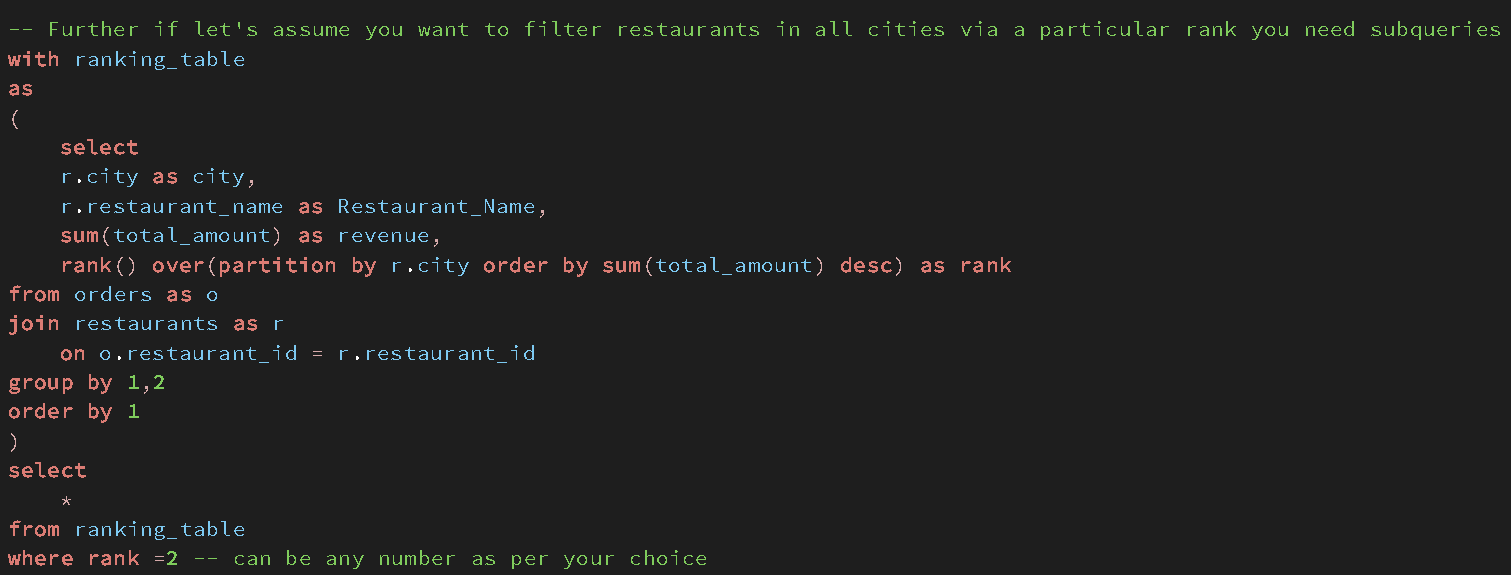
****

**Output**

****

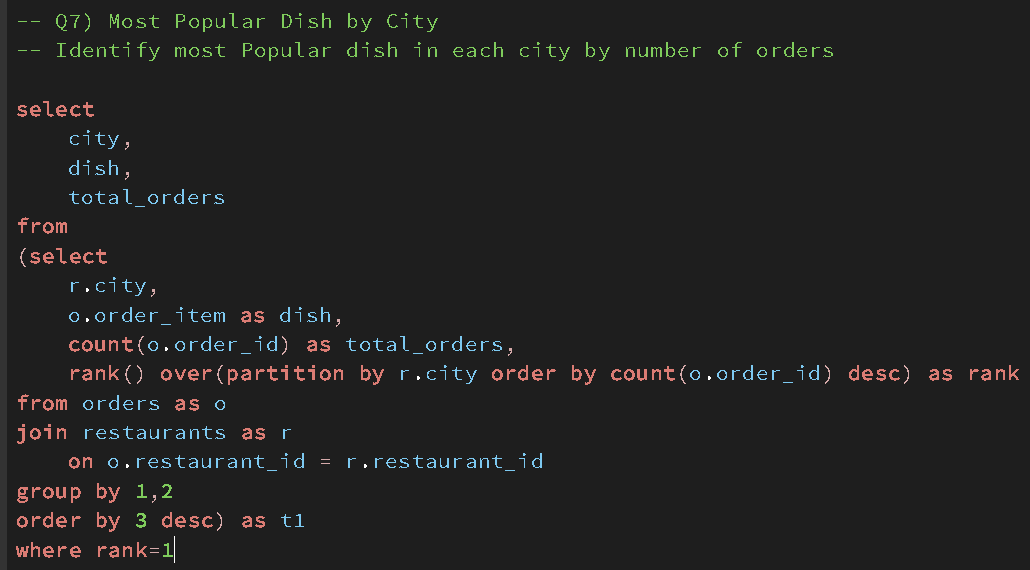
**Another modification to the question.**

**Further if let's assume you want to filter restaurants in all cities via a particular rank you need subqueries**

****

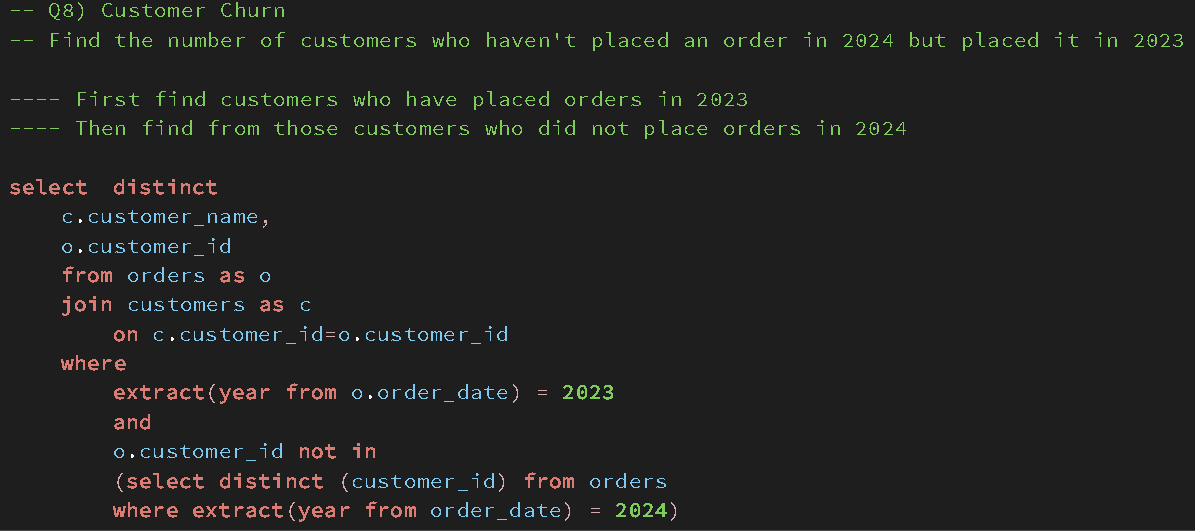
**Output  
**

**Q7) Most Popular Dish by City - Identify most Popular dish in each city by number of orders**

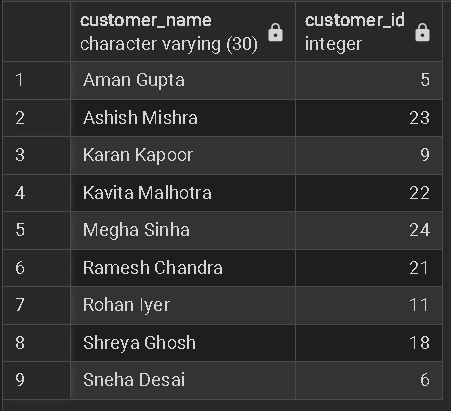
****

**Output**

****

**Q8) **

**Output**

****

**Q9) Cancellation Rate Comparison**

**-- Calculate and Compare order cancellation rate for each restaurant**

**-- in the current year(2024) and previous year(2023)**

**--- A cancelled order will have delivery id as "NUll" in the delivery column**

**-- CTE 1**

**with cancellation\_ratio\_23**

**as**

**(select**

**o.restaurant\_id,**

**count(o.order\_id) as total\_orders,**

**count(case when d.delivery\_id is null then 1 end) as not\_delivered**

**from orders as o**

**left join deliveries as d**

**on o.order\_id = d.order\_id**

**where extract (year from order\_date) = 2023**

**group by 1),**

**-- CTE 2**

**cancellation\_ratio\_24**

**as**

**(select**

**o.restaurant\_id,**

**count(o.order\_id) as total\_orders,**

**count(case when d.delivery\_id is null then 1 end) as not\_delivered**

**from orders as o**

**left join deliveries as d**

**on o.order\_id = d.order\_id**

**where extract (year from order\_date) = 2024**

**group by 1),**

**-- CTE 3**

**last\_year\_data**

**as**

**(select**

**restaurant\_id,**

**total\_orders,**

**not\_delivered,**

**round((not\_delivered::numeric/total\_orders::numeric)\*100,2) as ratio**

**from cancellation\_ratio\_23),**

**-- CTE 4**

**current\_year\_data**

**as**

**(select**

**restaurant\_id,**

**total\_orders,**

**not\_delivered,**

**round((not\_delivered::numeric/total\_orders::numeric)\*100,2) as ratio**

**from cancellation\_ratio\_24)**

**select**

**cyd.restaurant\_id,**

**cyd.ratio as current\_year,**

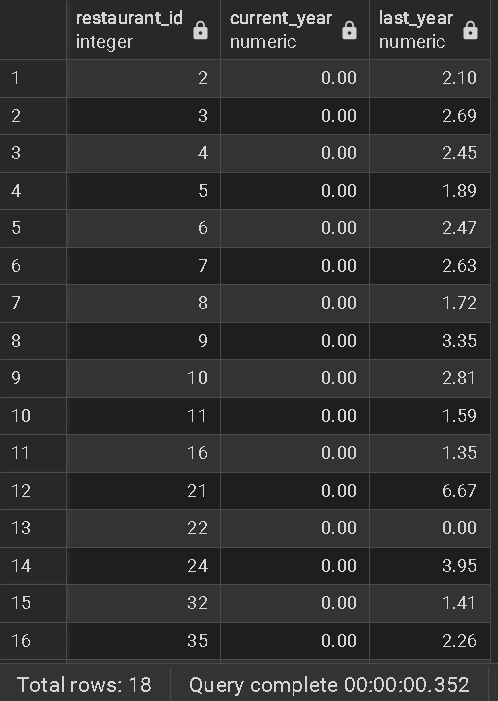
**lyd.ratio as last\_year**

**from current\_year\_data as cyd**

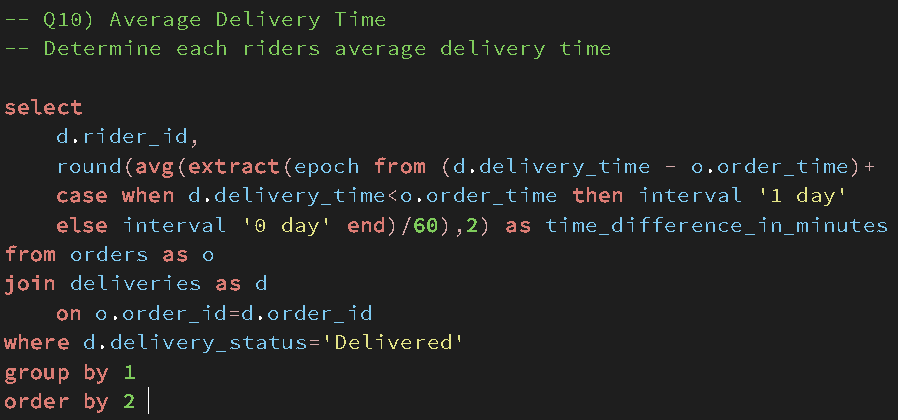
**join last\_year\_data as lyd**

**on cyd.restaurant\_id=lyd.restaurant\_id**

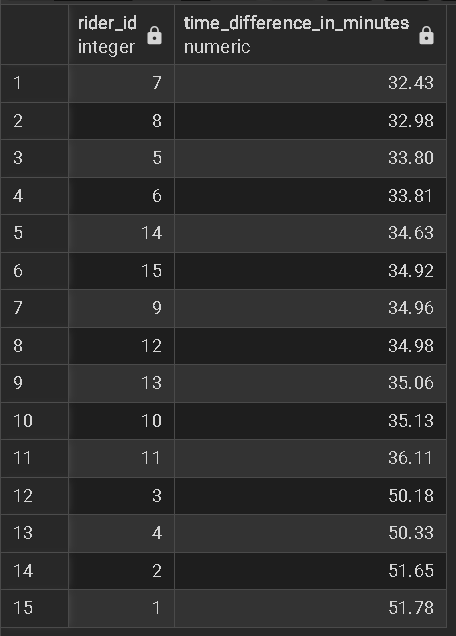
**Output**

****

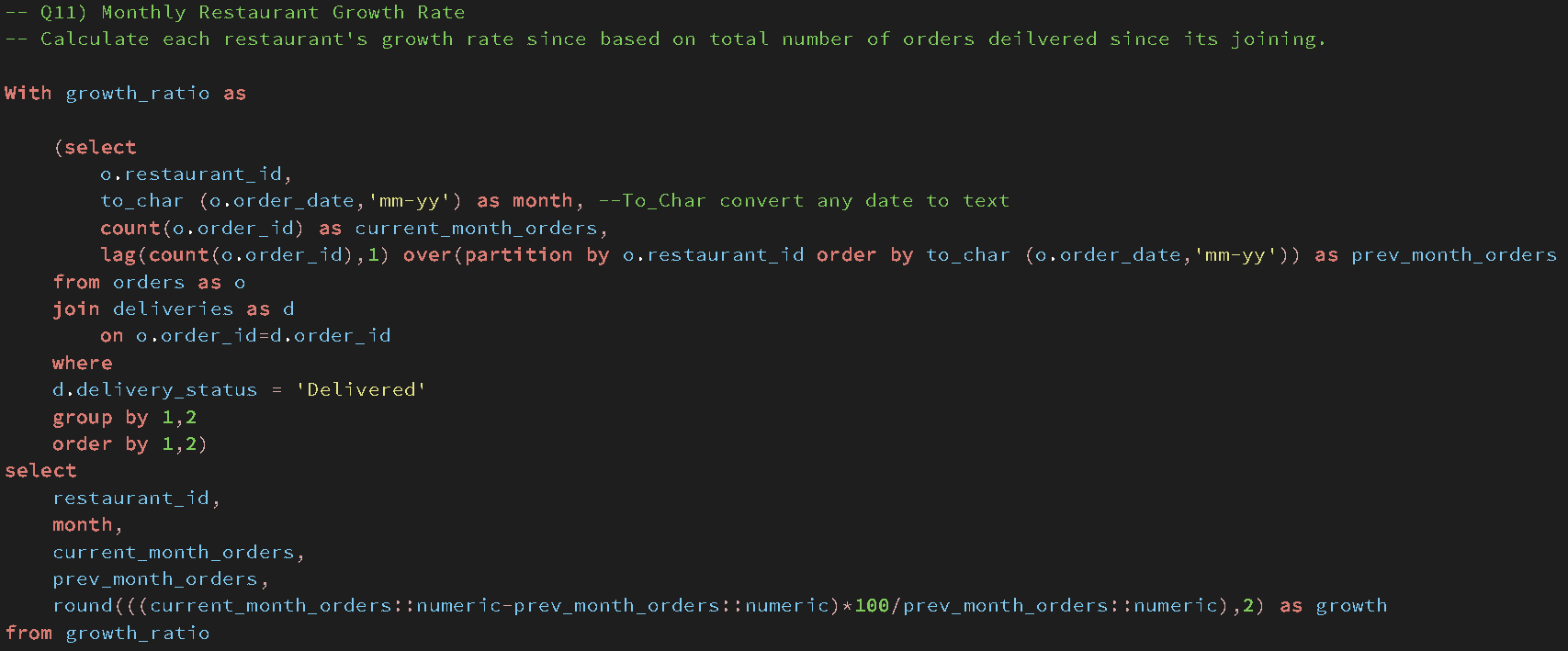
**Q10) Average Delivery time – Determine each rider’s delivery time**

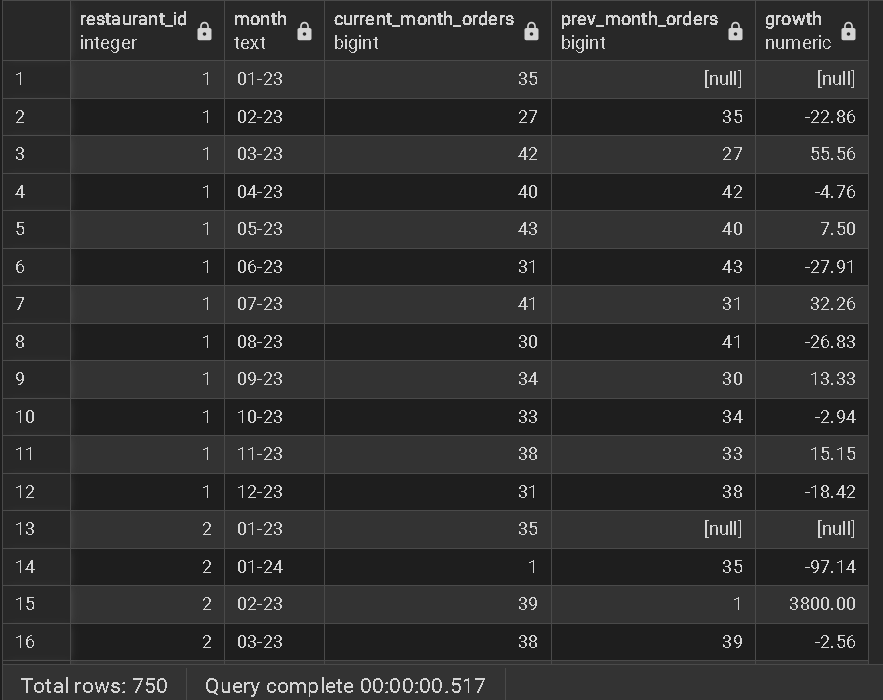
****

**Output**

****

**Q11) Monthly Restaurant Growth Rate -**Calculate each restaurant's growth rate since based on total number of orders deilvered since its joining.

****

****

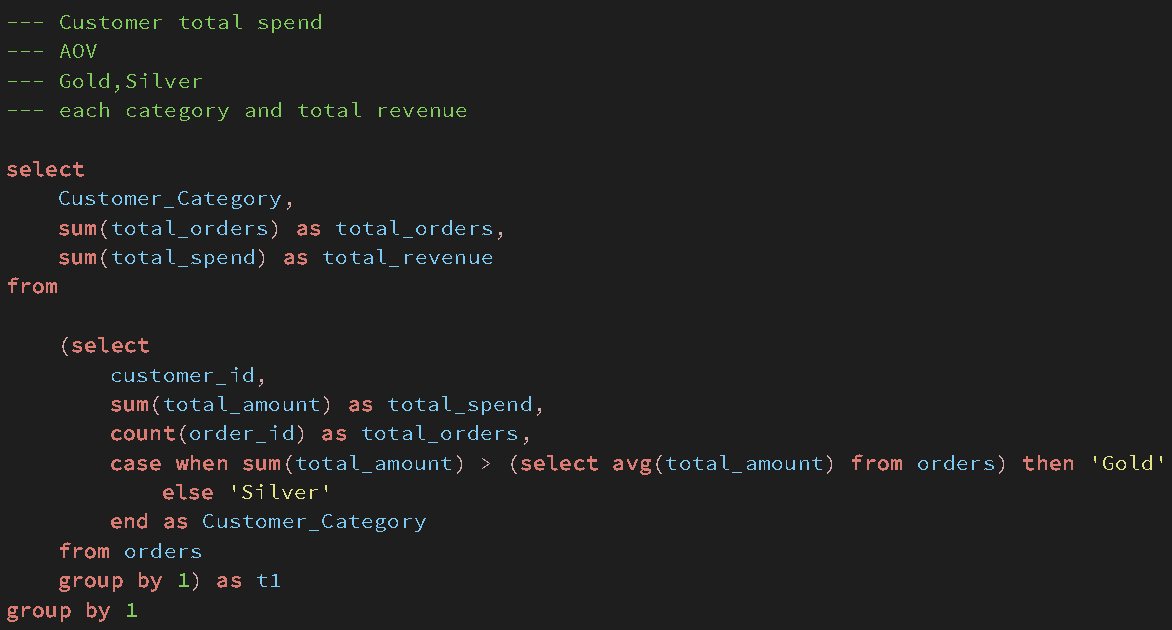
**Q12) Customer Segmentation**

**-- Customer Segmentation - Segment customers into gold or silver groups based on their total spending**

**-- compared to Average order Value(AOV), If customers spending exceeds AOV**

**-- label them as 'Gold'; otherwise 'Silver'. Write SQL query to determine each segment's**

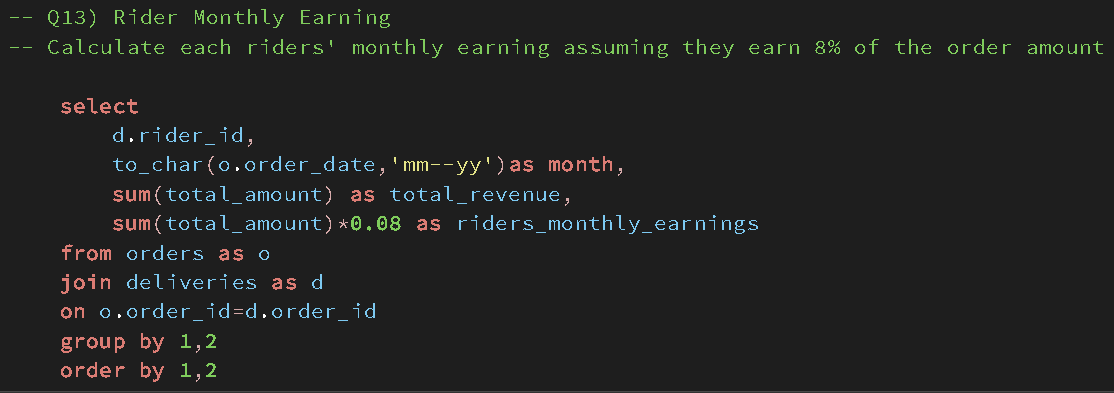
**-- total number of orders and total revenue**

****

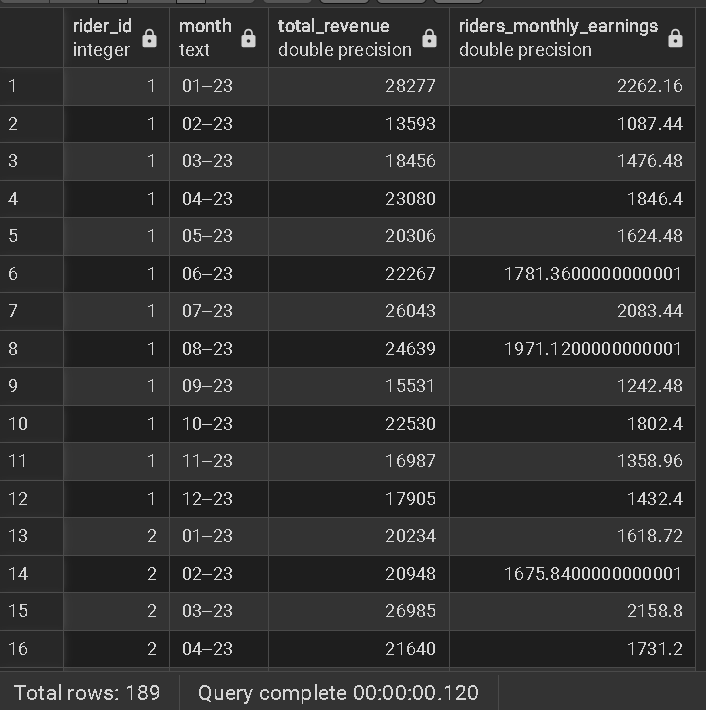
**Output**

****

**Q13) Rider Monthly Earning - Calculate each riders' monthly earning assuming they earn 8% of the order amount**

****

**Output**

****

**Q14) Rider Rating Analysis**

**-- Find the numbers of 5,4 and 3 star rating each rider has**

**-- riders receive this rating based on delivery times**

**-- if order delivered in less than 15 minutes - 5 star**

**-- if order delivered between 15 and 20 minutes - 4 star**

**-- if order delivered after 20 minutes - 3 star**

**select**

**rider\_id,**

**ratings,**

**count(\*) as total\_ratings**

**from**

**(**

**select**

**rider\_id,**

**delivery\_process\_time,**

**case when delivery\_process\_time < 15 then '5 star'**

**when delivery\_process\_time between 15 and 20 then '4 star'**

**else '3 star'**

**end as ratings**

**from**

**(select**

**o.order\_id,**

**o.order\_date,**

**o.order\_time,**

**d.delivery\_time,**

**extract (epoch from (d.delivery\_time-o.order\_time +**

**case when d.delivery\_time<o.order\_time then interval'1 day'**

**else interval '0 day' end**

**))/60 as delivery\_process\_time,**

**d.rider\_id**

**from orders as o**

**join deliveries as d**

**on o.order\_id = d.order\_id**

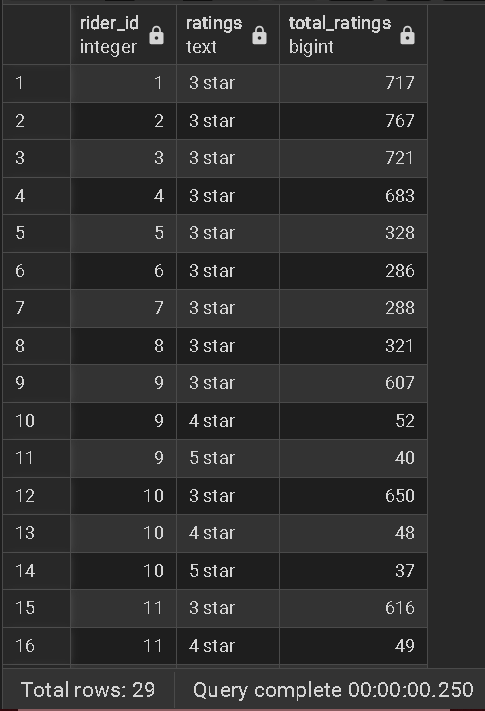
**where d.delivery\_status = 'Delivered') as t1**

**) as t2**

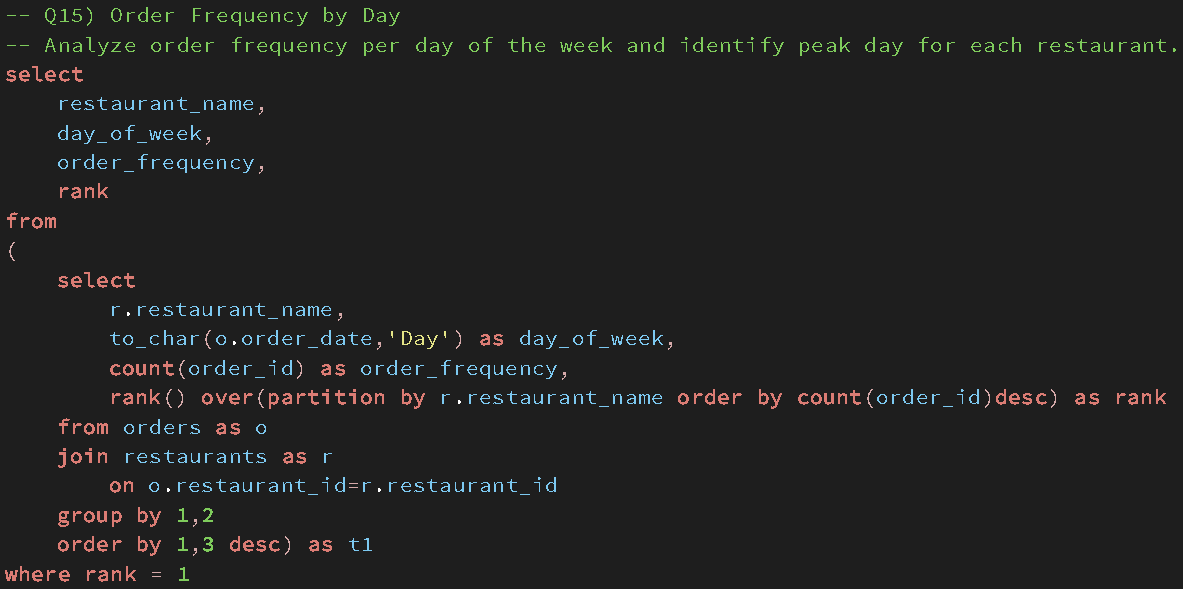
**group by 1,2**

**order by 1,3 desc**

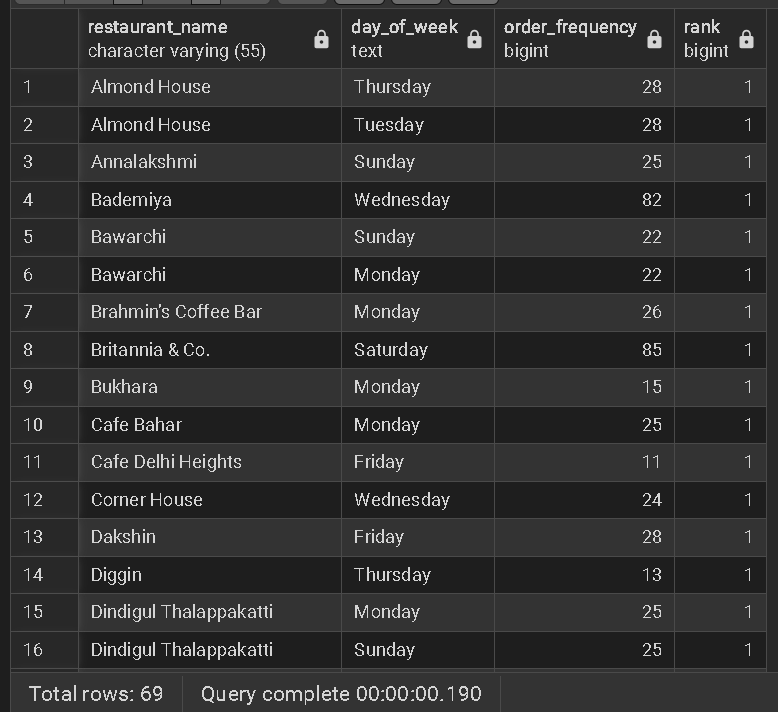
**Output**

****

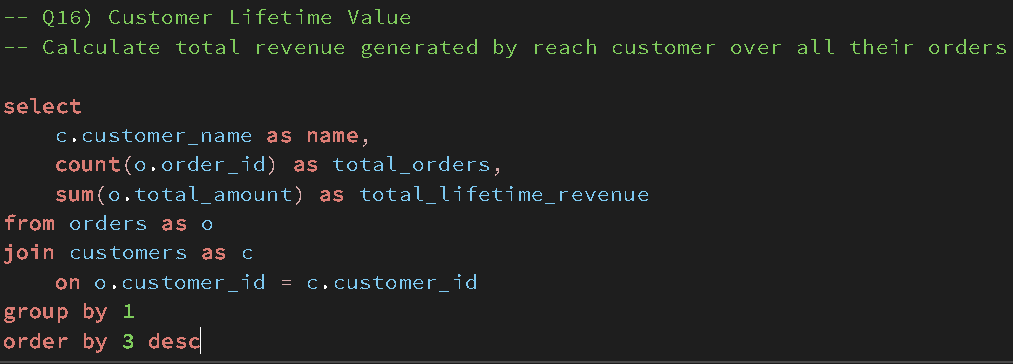
**Q15) Order Frequency by Day - Analyze order frequency per day of the week and identify peak day for each restaurant.**

****

**Output**

****

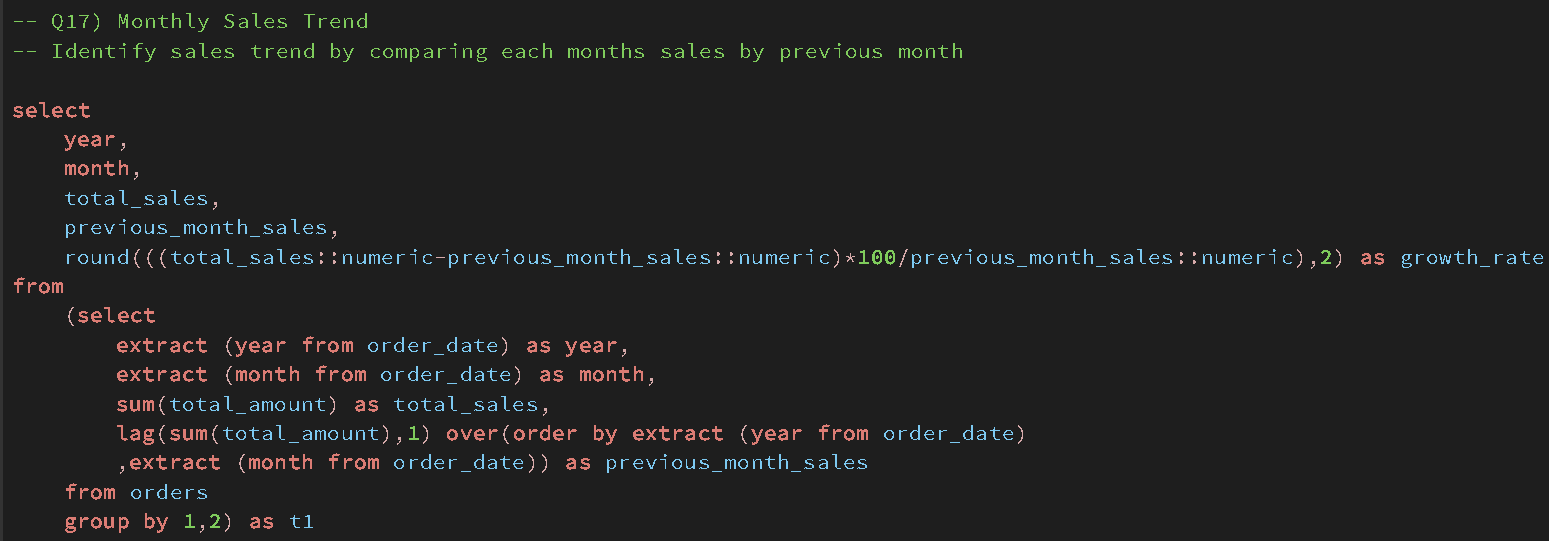
**Q16) Customer Lifetime Value - Calculate total revenue generated by reach customer over all their orders**

****

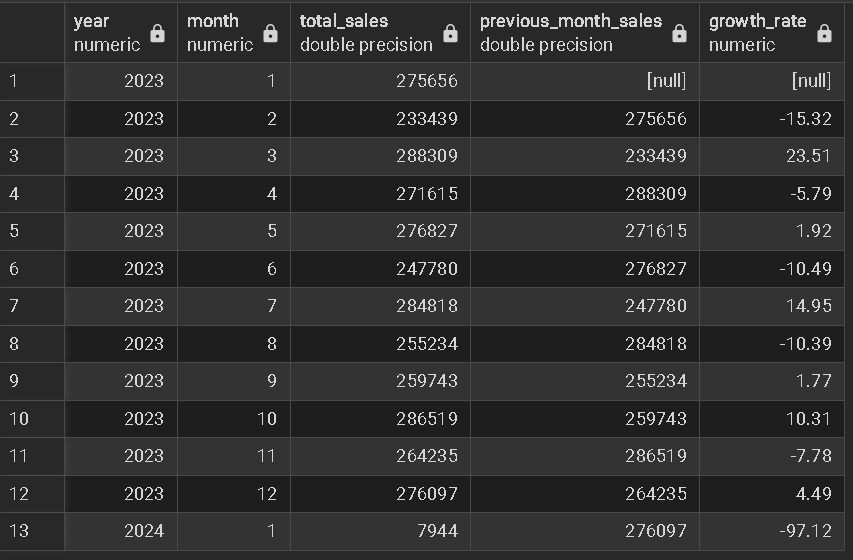
**Output**

****

**Q17) Monthly Sales Trend - Identify sales trend by comparing each months sales by previous month**

****

**Output**

****

**Q18) Rider Efficiency - Evaluate Rider efficiency by determining average delivery times and identifying the lowest and highest averages**

**with delivery\_table as**

**(**

**select**

**d.rider\_id as rider\_id,**

**round(avg(extract(epoch from (d.delivery\_time - o.order\_time)+**

**case when d.delivery\_time<o.order\_time then interval '1 day'**

**else interval '0 day' end)/60),2) as time\_to\_deliver**

**from orders as o**

**join deliveries as d**

**on o.order\_id = d.order\_id**

**where d.delivery\_status = 'Delivered'**

**group by 1),**

**riders\_time as(**

**select**

**rider\_id,**

**round(avg(time\_to\_deliver::numeric),2) as avg\_time**

**from delivery\_table**

**group by 1)**

**select**

**min(avg\_time),**

**max(avg\_time)**

**from riders\_time**

**Output**

****

**Q19)**