-- Pull the number of transactions by store country

select t.transaction\_date, s.store\_country,

COUNT(t.transaction\_date) OVER() AS total\_number\_of\_transctions,

COUNT(t.transaction\_date) OVER(PARTITION BY s.store\_country) AS number\_of\_transactions\_by\_store\_country

from Transaction\_Data t

left join Stores s on s.store\_id = t.store\_id

A screenshot of a computer

Description automatically generated

-- Pull the weekly revenue trending

select SUM(t.revenue) AS weekly\_revenue, DATETRUNC(WEEK, c.Date) AS start\_of\_week from Transaction\_Data t

inner join Calendar c on c.date = t.transaction\_date

group by DATETRUNC(WEEK, c.Date)

A screenshot of a computer

Description automatically generated

-- Pull the top 10 product brands in terms of number of Transactions and Revenue

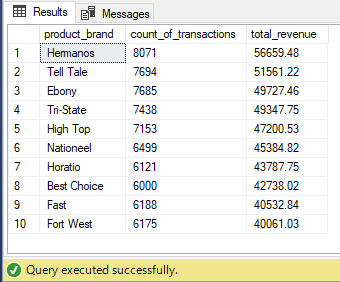
select top 10

p.product\_brand, COUNT(t.transaction\_date) AS count\_of\_transactions, SUM(t.revenue) AS total\_revenue from Transaction\_Data t

inner join products p on p.product\_id = t.product\_id

group by p.product\_brand

order by SUM(t.revenue) DESC



-- Bonus question Rank the total revenue and total transactions by product brand

WITH CTE AS

(select p.product\_brand, SUM(t.revenue) AS total\_revenue, COUNT(t.transaction\_date) AS count\_of\_transactions,

RANK() OVER(ORDER BY SUM(t.revenue) DESC) AS total\_revenue\_ranked,

RANK() OVER(ORDER BY COUNT(t.transaction\_date) DESC) AS total\_transactions\_ranked

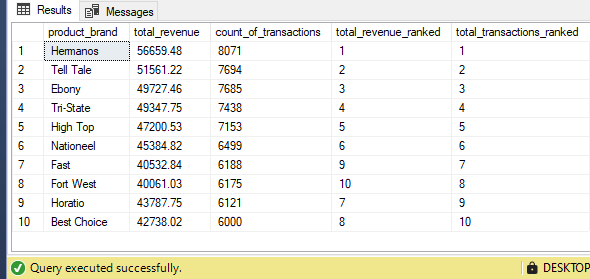
from Transaction\_Data t

inner join products p on p.product\_id = t.product\_id

group by p.product\_brand)

select \* from CTE where (total\_revenue\_ranked >= 1 and total\_revenue\_ranked <=10)

AND (total\_transactions\_ranked >=1 and total\_transactions\_ranked <=10)



-- Pull the weekly number of customer

select count(distinct customer\_id) AS number\_of\_customers, DATETRUNC(week, c.date) as start\_of\_week from Transaction\_Data t

left join Calendar c on t.transaction\_date = c.Date

group by DATETRUNC(week, c.date)

order by DATETRUNC(week, c.date) asc

A screenshot of a computer

Description automatically generated

-- Pull the number of transactions by education excluding the bachelors degree

select count(t.transaction\_date) as number\_of\_transactions, c.education from Transaction\_Data t

inner join Customers c on c.customer\_id = t.customer\_id

where education != 'Bachelors Degree'

group by c.education

A screenshot of a computer

Description automatically generated

-- Additional Question Report the number of transactions in each country broken down by education level (transposing data)

select customer\_country,

count(case when education = 'Partial High School' then t.transaction\_date else null end) as 'partial high school transactions',

count(case when education = 'Partial College' then t.transaction\_date else null end) as 'partial college transactions',

count(case when education = 'Graduate Degree' then t.transaction\_date else null end) as 'Graduate Degree transactions',

count(case when education = 'High School Degree' then t.transaction\_date else null end) as 'High School Degree transactions'

from Transaction\_Data t

inner join Customers c on c.customer\_id = t.customer\_id

group by customer\_country

A screenshot of a computer

Description automatically generated

-- Show the top 3 occupations in terms of number of transactions excluding customers with a bachelors degree

select TOP 3

count(t.transaction\_date) as number\_of\_transactions, c.occupation

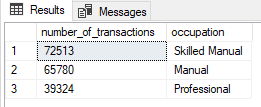
from Transaction\_Data t

inner join Customers c on c.customer\_id = t.customer\_id

where c.education != 'Bachelors Degree'

group by c.occupation

order by number\_of\_transactions desc



-- Using RANKING

WITH CTE AS (select c.occupation, count(t.transaction\_date) as total\_number\_of\_transactions,

RANK() OVER(ORDER BY count(t.transaction\_date) desc) AS ranking

from Transaction\_Data t

inner join Customers c on c.customer\_id = t.customer\_id

where c.education != 'Bachelors Degree'

group by c.occupation)

select \* from CTE

where ranking between 1 and 3

A screenshot of a number

Description automatically generated

-- Optional Question. Let's rank customers within each occupation based on their number of transactions.

-- Which customer\_id has the highest volume of transactions for each occupation?

WITH CTE AS

(select c.customer\_id, c.occupation, count(t.transaction\_date) as 'number of transactions',

RANK() OVER(PARTITION BY c.occupation ORDER BY COUNT(t.transaction\_date) desc) as ranking,

ROW\_NUMBER() OVER(PARTITION BY c.occupation ORDER BY COUNT(t.transaction\_date) desc) as r\_n

from Transaction\_Data t

inner join Customers c on c.customer\_id = t.customer\_id

group by c.customer\_id, c.occupation)

select \* from CTE

where r\_n = 1

A screenshot of a computer

Description automatically generated

-- Pull total number of transactions, revenue grouped by customer id and full\_name. Filter and rank 100 top customers by number of transactions

select TOP 100 c.customer\_id, CONCAT\_WS(' ', c.first\_name, c.last\_name) AS 'full name',

COUNT(t.transaction\_date) AS 'number of transactions', SUM(t.revenue) AS 'total revenue',

RANK() OVER(ORDER BY COUNT(t.transaction\_date) DESC) as ranking

from Transaction\_Data t

inner join Customers c on c.customer\_id = t.customer\_id

group by c.customer\_id, CONCAT\_WS(' ', c.first\_name, c.last\_name)

order by COUNT(t.transaction\_date) DESC

A screenshot of a computer

Description automatically generated

-- Which customer drove the highest revenue? Report its number of transactions too

WITH CTE AS

(select c.customer\_id, c.full\_name, SUM(t.revenue) as 'total revenue', COUNT(t.transaction\_date) as 'count of transactions',

RANK() OVER(ORDER BY SUM(t.revenue) DESC) AS ranking

from Transaction\_Data t

inner join Customers c on c.customer\_id = t.customer\_id

group by c.customer\_id, c.full\_name)

SELECT \* FROM CTE WHERE ranking = 1



-- Total Revenue & Profit Trending

select SUM(t.quantity \* p.product\_retail\_price) AS Total\_revenue,

SUM(t.revenue - t.COGS) AS Total\_profit,

DATEPART(YEAR, c.date) AS year,

DATETRUNC(MONTH, c.Date) AS start\_of\_month,

DATEPART(QUARTER, c.date) AS quarter,

DATETRUNC(WEEK, c.Date) AS start\_of\_week

from Transaction\_Data t

inner join calendar c on c.Date = t.transaction\_date

inner join products p on p.product\_id = t.product\_id

group by DATEPART(YEAR, c.date), DATETRUNC(MONTH, c.Date), DATEPART(QUARTER, c.date), DATETRUNC(WEEK, c.Date)

A screenshot of a computer

Description automatically generated

-- Additional Question. Calculate total revenue relating to product name 'High Top Garlic' and 'Tri State Red Pepper' for each year

select DATEPART(YEAR, c.Date) AS year,

SUM(CASE WHEN p.product\_name = 'High Top Garlic' THEN revenue ELSE NULL END) AS total\_revenue\_HighTopGarlic,

SUM(CASE WHEN p.product\_name = 'Tri-State Sweet Onion' THEN revenue ELSE NULL END) AS total\_revenue\_TriStateSweetOnion

from Transaction\_Data t

inner join products p on p.product\_id = t.product\_id

inner join Calendar c on c.Date = t.transaction\_date

group by DATEPART(YEAR, c.Date)

A screenshot of a computer

Description automatically generated