

-- Pull the number of transactions by store country

```
select t.transaction_date, s.store_country,
COUNT(t.transaction_date) OVER() AS total_number_of_transactions,
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
```

transaction_date	store_country	total_number_of_transactions	number_of_transactions_by_store_country
1998-01-16	USA	269720	180823
1998-01-15	USA	269720	180823
1998-01-16	USA	269720	180823
1998-01-16	USA	269720	180823
1998-01-16	USA	269720	180823
1998-01-16	USA	269720	180823
1998-01-16	USA	269720	180823
1998-01-15	USA	269720	180823
1998-01-16	USA	269720	180823
1998-01-16	USA	269720	180823
1998-01-15	USA	269720	180823

Query executed successfully. DESKTOP-B

-- 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)
```

weekly_revenue	start_of_week
3347.86	1996-12-29
14579.01	1997-01-05
10640.96	1997-01-12
12339.06	1997-01-19
5485.97	1997-01-26
11368.84	1997-02-02
12762.95	1997-02-09
11095.62	1997-02-16
9205.40	1997-02-23
7775.74	1997-03-02
13083.95	1997-03-09

Query executed successfully.

-- 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
```

product_brand	count_of_transactions	total_revenue
Hermanos	8071	56659.48
Tell Tale	7694	51561.22
Ebony	7685	49727.46
Tri-State	7438	49347.75
High Top	7153	47200.53
Nationeel	6499	45384.82
Horatio	6121	43787.75
Best Choice	6000	42738.02
Fast	6188	40532.84
Fort West	6175	40061.03

Query executed successfully.

-- 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)
```

	product_brand	total_revenue	count_of_transactions	total_revenue_ranked	total_transactions_ranked
1	Hermanos	56659.48	8071	1	1
2	Tell Tale	51561.22	7694	2	2
3	Ebony	49727.46	7685	3	3
4	Tri-State	49347.75	7438	4	4
5	High Top	47200.53	7153	5	5
6	Nationeel	45384.82	6499	6	6
7	Fast	40532.84	6188	9	7
8	Fort West	40061.03	6175	10	8
9	Horatio	43787.75	6121	7	9
10	Best Choice	42738.02	6000	8	10

Query executed successfully.

-- 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
```

	number_of_customers	start_of_week
1	120	1996-12-29
2	511	1997-01-05
3	368	1997-01-12
4	412	1997-01-19
5	194	1997-01-26
6	416	1997-02-02
7	451	1997-02-09
8	384	1997-02-16
9	294	1997-02-23
10	303	1997-03-02
11	466	1997-03-09

Query executed successfully.

-- 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
```

	number_of_transactions	education
1	81536	Partial High School
2	25080	Partial College
3	14471	Graduate Degree
4	79481	High School Degree

-- 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

```

	customer_country	partial high school transactions	partial college transactions	Graduate Degree transactions	High School Degree transactions
1	USA	53553	16946	10261	53712
2	Mexico	23431	6596	3479	20575
3	Canada	4552	1538	731	5194

-- 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

```

	number_of_transactions	occupation
1	72513	Skilled Manual
2	65780	Manual
3	39324	Professional

-- 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

```

	occupation	total_number_of_transactions	ranking
1	Skilled Manual	72513	1
2	Manual	65780	2
3	Professional	39324	3

-- 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
```

	customer_id	occupation	number of transactions	ranking	r_n
1	7836	Clerical	184	1	1
2	5542	Management	265	1	1
3	4676	Manual	301	1	1
4	4727	Professional	297	1	1
5	5295	Skilled Manual	290	1	1

Query executed successfully.

-- 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
```

	customer_id	full name	number of transactions	total revenue	ranking
1	4676	D Laner	301	1995.21	1
2	4727	J Horvat	297	2121.31	2
3	5295	I Rodriguez	290	2235.43	3
4	1787	M Archuleta	279	1763.58	4
5	4021	M Benigar	275	1973.79	5
6	5174	L Flowers	267	1750.14	6
7	5542	B Marschang	265	1713.75	7
8	4397	G Todero	262	1664.45	8
9	1720	A McDonnell	260	1907.90	9
10	4391	E Winters	257	1776.27	10
11	1297	J Mramor	251	1826.31	11

Query executed successfully.

-- 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
```

	customer_id	full_name	total revenue	count of transactions	ranking
1	5295	Ida Rodriguez	2235.43	290	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)
```

Results		Messages				
	Total_revenue	Total_profit	year	start_of_month	quarter	start_of_week
1	3347.86	1995.26	1997	1997-01-01	1	1996-12-29
2	14579.01	8690.67	1997	1997-01-01	1	1997-01-05
3	10640.96	6351.54	1997	1997-01-01	1	1997-01-12
4	12339.06	7362.28	1997	1997-01-01	1	1997-01-19
5	4632.80	2774.08	1997	1997-01-01	1	1997-01-26
6	853.17	510.29	1997	1997-02-01	1	1997-01-26
7	11368.84	6761.63	1997	1997-02-01	1	1997-02-02
8	12762.95	7646.21	1997	1997-02-01	1	1997-02-09
9	11095.62	6573.13	1997	1997-02-01	1	1997-02-16
10	7978.21	4774.70	1997	1997-02-01	1	1997-02-23
11	1227.19	744.65	1997	1997-03-01	1	1997-02-23

✓ Query executed successfully.

-- 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)
```

Results		Messages	
	year	total_revenue_HighTopGarlic	total_revenue_TriStatesweetonion
1	1997	530.40	240.21
2	1998	1217.20	649.98