Retail-Sales-Analysis-SQL-Project--P1

```
18  );
19
20  SELECT TOP (10)* FROM [dbo].[Retail Sales ]
21
```

121 % 5 0

Resultados Mensajes

	transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale
1	1	2022-12-16	19:10:00.0000000	50	Male	34	Beauty	3	50.00	16.00	150.00
2	2	2022-06-24	10:07:00.0000000	104	Female	26	Clothing	2	500.00	135.00	1000.00
3	3	2022-06-14	07:08:00.0000000	114	Male	50	Electronics	1	30.00	8.10	30.00
4	4	2023-08-27	18:12:00.0000000	3	Male	37	Clothing	1	500.00	200.00	500.00
5	5	2023-09-05	22:10:00.0000000	3	Male	30	Beauty	2	50.00	24.00	100.00
6	6	2023-11-15	22:16:00.0000000	2	Female	45	Beauty	1	30.00	15.00	30.00
7	7	2023-07-06	06:24:00.0000000	38	Male	46	Clothing	2	25.00	13.25	50.00
8	8	2022-12-27	11:19:00.0000000	148	Male	30	Electronics	4	25.00	11.00	100.00
9	9	2022-12-02	13:12:00.0000000	85	Male	63	Electronics	2	300.00	78.00	600.00
10	10	2022-10-24	22:55:00.0000000	81	Female	52	Clothing	4	50.00	62.50	200.00

```
21  SELECT COUNT(*) FROM [dbo].[Retail Sales ]
22
```

121 % 4 0

Resultados Mensajes

	(Sin nombre de columna)
1	2000

We want to see how many customers we have and we have gender we have age we have category we have quantity we have price per unit we have cost of the good source and total sales.
This data have some NULLs values

How to deal with nulls values.
We we can identify if we have nulls values.

```
22
23  SELECT * FROM [dbo].[Retail Sales ]
24  WHERE [transactions_id] IS NULL
```

121 % 7 0

Resultados Mensajes

transactions_id	sale_date	sale_time	customer_id	gender	a

We don't have any nulls in the transaction Id column

```sql
SELECT * FROM [dbo].[Retail Sales ]
WHERE
    [transactions_id] IS NULL
    OR [sale_date] IS NULL
    OR [sale_time] IS NULL
    OR [customer_id] IS NULL
```

```
    OR [gender] IS NULL
    OR [age] IS NULL
    OR [category] IS NULL
    OR [quantiy] IS NULL
    OR [price_per_unit] IS NULL
    OR [cogs] IS NULL
    OR [total_sale] IS NULL;
```

Resultados | Mensajes

	transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale
1	150	2022-04-13	08:25:00.0000000	89	Female	NULL	Electronics	4	30.00	16.20	120.00
2	432	2022-03-10	11:31:00.0000000	17	Female	NULL	Electronics	2	500.00	190.00	1000.00
3	679	2022-08-26	08:59:00.0000000	64	Female	18	Beauty	NULL	NULL	NULL	NULL
4	746	2022-07-05	11:33:00.0000000	42	Female	33	Clothing	NULL	NULL	NULL	NULL
5	797	2022-09-16	06:38:00.0000000	116	Male	NULL	Clothing	3	25.00	10.75	75.00
6	845	2022-10-27	10:12:00.0000000	25	Male	NULL	Clothing	1	500.00	145.00	500.00
7	921	2022-09-28	09:34:00.0000000	101	Male	NULL	Electronics	3	25.00	8.00	75.00
8	1150	2022-08-22	10:04:00.0000000	77	Female	NULL	Electronics	4	30.00	10.20	120.00
9	1225	2022-02-02	09:51:00.0000000	137	Female	57	Beauty	NULL	NULL	NULL	NULL
10	1367	2022-04-15	11:38:00.0000000	16	Female	NULL	Electronics	1	50.00	15.50	50.00
11	1391	2022-03-01	11:29:00.0000000	130	Male	NULL	Beauty	2	25.00	9.25	50.00
12	1432	2022-12-25	06:24:00.0000000	67	Female	NULL	Electronics	2	500.00	245.00	1000.00
13	1845	2022-05-24	07:06:00.0000000	94	Male	NULL	Clothing	1	500.00	185.00	500.00

We come some nulls so we are going to see how we can delete them.

```
 DELETE FROM [dbo].[Retail Sales ]
WHERE
    [transactions_id] IS NULL
    OR [sale_date] IS NULL
    OR [sale_time] IS NULL
    OR [customer_id] IS NULL
    OR [gender] IS NULL
    OR [age] IS NULL
    OR [category] IS NULL
    OR [quantiy] IS NULL
    OR [price_per_unit] IS NULL
    OR [cogs] IS NULL
    OR [total_sale] IS NULL;
```

Resultados | Mensajes

transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale

Done! with that we have deleted all the nulls values.

Now let's explore the data and then we can solve all the business problems.

Now we can see how many customers we have

```
--How many sales we have
SELECT COUNT (*) AS [total_sale] FROM [dbo].[Retail Sales ]
```

	total_sale
1	1987

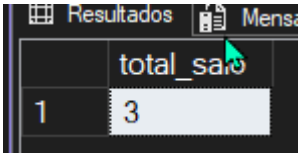
```
SELECT COUNT ([transactions_id]) AS [total_sale] FROM [dbo].[Retail Sales ]
```

	total_sale
1	1987

```
--How many unique customers we have?
SELECT COUNT (DISTINCT [customer_id]) AS [total_sale]
FROM [dbo].[Retail Sales ];
```

	total_sale
1	155

```
--How many unique categories we have?
SELECT COUNT (DISTINCT [category]) AS [total_sale]
FROM [dbo].[Retail Sales ];
```



	total_sale
1	3

We can do more explorations. So let's go ahead and solve some business problems.

Data analysis & business key problems

```
-- Q.1 Write a SQL query to retrieve all columns for sales made on '2022-11-05
SELECT *
FROM [dbo].[Retail Sales ]
WHERE [sale_date] = '2022-11-05'
```

	transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale
1	180	2022-11-05	10:47:00.0000000	117	Male	41	Clothing	3	300.00	129.00	900.00
2	214	2022-11-05	16:31:00.0000000	53	Male	20	Beauty	2	30.00	8.10	60.00
3	240	2022-11-05	11:49:00.0000000	95	Female	23	Beauty	1	300.00	123.00	300.00
4	856	2022-11-05	17:43:00.0000000	102	Male	54	Electronics	4	30.00	9.30	120.00
5	943	2022-11-05	19:29:00.0000000	90	Female	57	Clothing	4	300.00	318.00	1200.00
6	1137	2022-11-05	22:34:00.0000000	104	Male	46	Beauty	2	500.00	145.00	1000.00
7	1256	2022-11-05	09:58:00.0000000	29	Male	23	Clothing	2	500.00	190.00	1000.00
8	1265	2022-11-05	14:35:00.0000000	86	Male	55	Clothing	3	300.00	111.00	900.00
9	1587	2022-11-05	20:06:00.0000000	140	Female	40	Beauty	4	300.00	105.00	1200.00
10	1819	2022-11-05	20:44:00.0000000	83	Female	35	Beauty	2	50.00	13.50	100.00
11	1896	2022-11-05	20:19:00.0000000	87	Female	30	Electronics	2	25.00	30.75	50.00

```sql
-- Q.2 Write a SQL query to retrieve all transactions where the category is
'Clothing'
-- and the quantity sold is more than 4 in the month of Nov-2022
SELECT
    [category]
    ,SUM([quantiy])
FROM [dbo].[Retail Sales ]
WHERE [category] = 'Clothing'
GROUP BY [category]
```

Resultados | Mensajes

	category	(Sin nombre de columna)
1	Clothing	1780

```sql
SELECT
    [category]
    ,SUM([quantiy])
FROM [dbo].[Retail Sales ]
WHERE [category] = 'Clothing'
AND YEAR([sale_date]) = '2022'
AND MONTH([sale_date])='11'
GROUP BY category
```

```
category
---------------------------------------------------- -----------
Clothing                                             134
```

```sql
SELECT
    *
FROM [dbo].[Retail Sales ]
WHERE [category] = 'Clothing'
AND YEAR([sale_date]) = '2022'
AND MONTH([sale_date])='11'
```

Resultados | Mensajes

	transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale
1	59	2022-11-14	18:28:00.0000000	61	Male	62	Clothing	1	50.00	27.00	50.00
2	64	2022-11-15	06:34:00.0000000	7	Male	49	Clothing	4	25.00	8.50	100.00
3	95	2022-11-06	19:57:00.0000000	103	Female	32	Clothing	2	30.00	7.50	60.00
4	103	2022-11-12	10:05:00.0000000	33	Female	59	Clothing	1	25.00	10.75	25.00
5	110	2022-11-18	17:28:00.0000000	149	Male	27	Clothing	3	300.00	99.00	900.00
6	126	2022-11-01	18:16:00.0000000	63	Female	28	Clothing	3	30.00	28.80	90.00
7	145	2022-11-06	19:21:00.0000000	64	Female	39	Clothing	3	25.00	27.50	75.00
8	146	2022-11-10	22:01:00.0000000	74	Male	38	Clothing	4	50.00	49.00	200.00
9	159	2022-11-10	21:30:00.0000000	42	Male	26	Clothing	4	50.00	23.50	200.00
10	180	2022-11-05	10:47:00.0000000	117	Male	41	Clothing	3	300.00	129.00	900.00

So we have the data where we have all this information. An we have the data only for the month of November.

Now we are going to see the quantity We have to use the filter sold more than 10 In the month of November.

```sql
SELECT
    *
```

```
FROM [dbo].[Retail Sales ]
WHERE [category] = 'Clothing'
AND YEAR([sale_date]) = '2022'
AND MONTH([sale_date])='11'
AND [quantiy] >= 4
```

Resultados | Mensajes

	transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale
1	64	2022-11-15	06:34:00.0000000	7	Male	49	Clothing	4	25.00	8.50	100.00
2	146	2022-11-10	22:01:00.0000000	74	Male	38	Clothing	4	50.00	49.00	200.00
3	159	2022-11-10	21:30:00.0000000	42	Male	26	Clothing	4	50.00	23.50	200.00
4	284	2022-11-12	09:17:00.0000000	129	Male	43	Clothing	4	50.00	20.50	200.00
5	547	2022-11-14	07:36:00.0000000	3	Male	63	Clothing	4	500.00	250.00	2000.00
6	699	2022-11-21	22:21:00.0000000	129	Female	37	Clothing	4	30.00	16.20	120.00
7	735	2022-11-26	21:38:00.0000000	153	Female	64	Clothing	4	500.00	515.00	2000.00
8	943	2022-11-05	19:29:00.0000000	90	Female	57	Clothing	4	300.00	318.00	1200.00
9	965	2022-11-27	21:45:00.0000000	84	Male	22	Clothing	4	50.00	13.00	200.00
10	1259	2022-11-03	17:31:00.0000000	105	Female	45	Clothing	4	50.00	21.00	200.00
11	1296	2022-11-26	20:42:00.0000000	45	Female	22	Clothing	4	300.00	342.00	1200.00
12	1476	2022-11-11	22:27:00.0000000	130	Female	27	Clothing	4	500.00	555.00	2000.00
13	1484	2022-11-23	09:29:00.0000000	22	Female	19	Clothing	4	300.00	147.00	1200.00
14	1497	2022-11-19	21:44:00.0000000	109	Male	41	Clothing	4	30.00	32.40	120.00
15	1615	2022-11-17	13:43:00.0000000	82	Female	61	Clothing	4	25.00	13.50	100.00
16	1696	2022-11-21	17:59:00.0000000	24	Female	50	Clothing	4	50.00	55.00	200.00
17	1885	2022-11-09	07:32:00.0000000	148	Female	52	Clothing	4	30.00	10.80	120.00

Now we can see that we have all the transactions where we have the quantity of four or more than four. Category is closing and quantity is 4 or more than four.

```
 -- Q.3 Write a SQL query to calculate the total sales (total_sale) for each
category.
```

Now we need to find the total sales for each category. Now we want to find each category on their sells.

```
SELECT
    [category]
    ,SUM([total_sale]) AS net_sale
    ,COUNT(*) AS total_orders
 FROM [dbo].[Retail Sales ]
GROUP BY [category]
ORDER BY net_sale DESC
```

Resultados | Mensajes

	category	net_sale	total_orders
1	Electronics	311445.00	678
2	Clothing	309995.00	698
3	Beauty	286790.00	611

```
-- Q.4 Write a SQL query to find the average age of customers who purchased items
from the 'Beauty' category.

SELECT
    ROUND(AVG(age), 2)  AS avg_age
    ,MIN(age) AS min_age
    ,MAX(age) AS max_age
```

```
    ,COUNT(DISTINCT [customer_id]) AS total_customers
FROM [dbo].[Retail Sales ]
WHERE [category] = 'Beauty'
```

	avg_age	min_age	max_age	total_customers
1	40	18	64	141

-- Q.5 Write a SQL query to find all transactions where the total_sale is greater than 1000.

```
-- Q.5 Write a SQL query to find all transactions where the total_sale is greater
than 1000.

SELECT * FROM [dbo].[Retail Sales ]
WHERE [total_sale]> 1000
```

	transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale
1	13	2023-02-08	17:43:00.0000000	106	Male	22	Electronics	3	500.00	245.00	1500.00
2	15	2022-07-01	11:50:00.0000000	75	Female	42	Electronics	4	500.00	210.00	2000.00
3	16	2022-06-25	10:33:00.0000000	82	Male	19	Clothing	3	500.00	180.00	1500.00
4	31	2023-12-31	17:47:00.0000000	3	Male	44	Electronics	4	300.00	129.00	1200.00
5	46	2022-11-08	17:50:00.0000000	54	Female	20	Electronics	4	300.00	84.00	1200.00
6	47	2022-10-22	17:22:00.0000000	96	Female	40	Beauty	3	500.00	600.00	1500.00

```
-- Q.6 Write a SQL query to find the total number of transactions (transaction_id)
-- made by each gender in each category.

SELECT
 [category]
 ,[gender]
 ,COUNT (*) AS total_trans
 FROM [dbo].[Retail Sales ]
 GROUP BY [category]
 ,[gender]
```

Resultados Mensajes

	category	gender	total_trans
1	Beauty	Female	330
2	Beauty	Male	281
3	Electronics	Female	335
4	Electronics	Male	343
5	Clothing	Male	351
6	Clothing	Female	347

```
-- Q.6 Write a SQL query to find the total number of transactions (transaction_id)
-- made by each gender in each category.

SELECT
     [category]
     ,[gender]
```

```
     ,COUNT (*) AS total_trans
     ,SUM([total_sale]) AS total_revenue
     ,CAST (AVG([total_sale]) AS DECIMAL (10,2)) AS avg_transaction_value
 FROM [dbo].[Retail Sales ]
 GROUP BY [category] ,[gender]
 ORDER BY [category], total_trans DESC;
```

	category	gender	total_trans	total_revenue	avg_transaction_value
1	Beauty	Female	330	149470.00	452.94
2	Beauty	Male	281	137320.00	488.68
3	Clothing	Male	351	147535.00	420.33
4	Clothing	Female	347	162460.00	468.18
5	Electronics	Male	343	160265.00	467.24
6	Electronics	Female	335	151180.00	451.28

We have to find the average sales for each month on each year

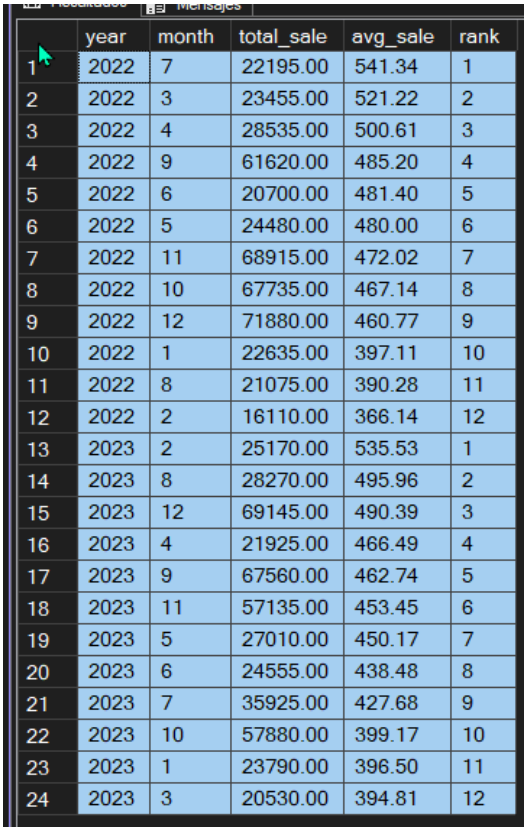
```
-- Q.7 Write a SQL query to calculate the average sale for each month. Find out best
selling month in each year

 SELECT
    YEAR([sale_date]) AS year
    ,MONTH([sale_date]) AS month
    ,SUM([total_sale]) AS total_sale
    ,AVG([total_sale]) AS avg_sale
    FROM [dbo].[Retail Sales ]
GROUP BY YEAR([sale_date])
    ,MONTH([sale_date])
ORDER BY YEAR, MONTH,  avg_sale DESC
```

	year	month	total_sale	avg_sale
1	2022	1	22635.00	397.11
2	2022	2	16110.00	366.14
3	2022	3	23455.00	521.22
4	2022	4	28535.00	500.61
5	2022	5	24480.00	480.00
6	2022	6	20700.00	481.40
7	2022	7	22195.00	541.34
8	2022	8	21075.00	390.28
9	2022	9	61620.00	485.20
10	2022	10	67735.00	467.14
11	2022	11	68915.00	472.02
12	2022	12	71880.00	460.77
13	2023	1	23790.00	396.50
14	2023	2	25170.00	535.53
15	2023	3	20530.00	394.81
16	2023	4	21925.00	466.49
17	2023	5	27010.00	450.17
18	2023	6	24555.00	438.48
19	2023	7	35925.00	427.68
20	2023	8	28270.00	495.96
21	2023	9	67560.00	462.74
22	2023	10	57880.00	399.17
23	2023	11	57135.00	453.45
24	2023	12	69145.00	490.39

Now we need for each year one month that has the highest average selling

```
 SELECT
    YEAR([sale_date]) AS year
    ,MONTH([sale_date]) AS month
    ,SUM([total_sale]) AS total_sale
    ,CAST(AVG([total_sale]) AS DECIMAL(10,2)) AS avg_sale
    ,RANK() OVER(PARTITION BY (YEAR([sale_date])) ORDER BY AVG([total_sale]) DESC) AS
rank
    FROM [dbo].[Retail Sales ]
GROUP BY YEAR([sale_date])
    ,MONTH([sale_date])
--ORDER BY YEAR, MONTH,  avg_sale DESC
```

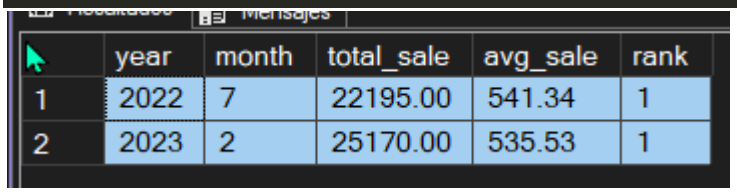


	year	month	total_sale	avg_sale	rank
1	2022	7	22195.00	541.34	1
2	2022	3	23455.00	521.22	2
3	2022	4	28535.00	500.61	3
4	2022	9	61620.00	485.20	4
5	2022	6	20700.00	481.40	5
6	2022	5	24480.00	480.00	6
7	2022	11	68915.00	472.02	7
8	2022	10	67735.00	467.14	8
9	2022	12	71880.00	460.77	9
10	2022	1	22635.00	397.11	10
11	2022	8	21075.00	390.28	11
12	2022	2	16110.00	366.14	12
13	2023	2	25170.00	535.53	1
14	2023	8	28270.00	495.96	2
15	2023	12	69145.00	490.39	3
16	2023	4	21925.00	466.49	4
17	2023	9	67560.00	462.74	5
18	2023	11	57135.00	453.45	6
19	2023	5	27010.00	450.17	7
20	2023	6	24555.00	438.48	8
21	2023	7	35925.00	427.68	9
22	2023	10	57880.00	399.17	10
23	2023	1	23790.00	396.50	11
24	2023	3	20530.00	394.81	12

We can see that for the year 2022 the moon 7 we have the average sale of 541 we got ranking number one and for the month for the year 2023 in the month second we have an average of 535 and we rock it number one.

now we can simply filter by run by saying that give me rank number one so we get it so I'm just going to call it a rank now I'm just going to call it rank.

```
SELECT * FROM
    (
        SELECT
        YEAR([sale_date]) AS year
        ,MONTH([sale_date]) AS month
        ,SUM([total_sale]) AS total_sale
        ,CAST(AVG([total_sale]) AS DECIMAL(10,2)) AS avg_sale
        ,RANK() OVER(PARTITION BY (YEAR([sale_date])) ORDER BY AVG([total_sale])
DESC) AS rank
        FROM [dbo].[Retail Sales ]
    GROUP BY YEAR([sale_date])
        ,MONTH([sale_date])
    --ORDER BY YEAR, MONTH,  avg_sale DESC
    ) as t1
WHERE rank= 1
```



	year	month	total_sale	avg_sale	rank
1	2022	7	22195.00	541.34	1
2	2023	2	25170.00	535.53	1

```sql
SELECT * FROM
    (
        SELECT
            YEAR([sale_date]) AS year
            ,MONTH([sale_date]) AS month
            ,SUM([total_sale]) AS total_sale
            ,CAST(AVG([total_sale]) AS DECIMAL(10,2)) AS avg_sale
            ,RANK() OVER(PARTITION BY YEAR([sale_date]) ORDER BY AVG([total_sale])
DESC) AS rank
        FROM [dbo].[Retail Sales ]
    GROUP BY YEAR([sale_date]) ,MONTH([sale_date])
    --ORDER BY YEAR, MONTH,  avg_sale DESC
    ) as t1
WHERE rank= 1
ORDER BY YEAR;
```

Resultados | Mensajes

	year	month	total_sale	avg_sale	rank
1	2022	7	22195.00	541.34	1
2	2023	2	25170.00	535.53	1

```sql
-- Q.8 Write a SQL query to find the top 5 customers based on the highest total sales

SELECT * FROM [dbo].[Retail Sales ]
```

Resultados | Mensajes

	transactions_id	sale_date	sale_time	customer_id	gender	age	category	quantiy	price_per_unit	cogs	total_sale
1	1	2022-12-16	19:10:00.0000000	50	Male	34	Beauty	3	50.00	16.00	150.00
2	2	2022-06-24	10:07:00.0000000	104	Female	26	Clothing	2	500.00	135.00	1000.00
3	3	2022-06-14	07:08:00.0000000	114	Male	50	Electronics	1	30.00	8.10	30.00
4	4	2023-08-27	18:12:00.0000000	3	Male	37	Clothing	1	500.00	200.00	500.00
5	5	2023-09-05	22:10:00.0000000	3	Male	30	Beauty	2	50.00	24.00	100.00
6	6	2023-11-15	22:16:00.0000000	2	Female	45	Beauty	1	30.00	15.00	30.00
7	7	2023-07-06	06:24:00.0000000	38	Male	46	Clothing	2	25.00	13.25	50.00
8	8	2022-12-27	11:19:00.0000000	148	Male	30	Electronics	4	25.00	11.00	100.00
9	9	2022-12-02	13:12:00.0000000	85	Male	63	Electronics	2	300.00	78.00	600.00
10	10	2022-10-24	22:55:00.0000000	81	Female	52	Clothing	4	50.00	62.50	200.00
11	11	2022-02-27	10:30:00.0000000	151	Male	23	Clothing	2	50.00	23.50	100.00

```sql
-- Q.8 Write a SQL query to find the top 5 customers based on the highest total sales

SELECT TOP 5
   [customer_id]
    ,SUM([total_sale]) AS total_sales
FROM [dbo].[Retail Sales ]
GROUP BY [customer_id]
ORDER BY total_sales
```

	customer_id	total_sales
1	123	200.00
2	17	620.00
3	118	640.00
4	150	675.00
5	23	845.00

```sql
-- Q.9 Write a SQL query to find the number of unique customers who purchased items from each category.

SELECT
    [category]
    ,COUNT(DISTINCT [customer_id]) AS cnt_unique_customer
FROM [dbo].[Retail Sales ]
GROUP BY [category]
```

Resultados Mensajes

	category	cnt_unique_customer
1	Beauty	141
2	Clothing	149
3	Electronics	144

```sql
-- Q.10 Write a SQL query to create each shift and number of orders (Example Morning <=12, Afternoon Between 12 & 17, Evening >17)
```

```sql
SELECT
    CASE
        WHEN DATEPART(HOUR, sale_time) < 12 THEN 'Morning'
        WHEN DATEPART(HOUR, sale_time) BETWEEN 12 AND 17 THEN 'Afternoon'
        ELSE 'Evening'
    END AS shift
    ,COUNT(DISTINCT transactions_id) AS total_orders
FROM dbo.[Retail Sales ]
GROUP BY
    CASE
        WHEN DATEPART(HOUR, sale_time) < 12 THEN 'Morning'
        WHEN DATEPART(HOUR, sale_time) BETWEEN 12 AND 17 THEN 'Afternoon'
        ELSE 'Evening'
    END;
```

	shift	total_orders
1	Morning	548
2	Evening	1062
3	Afternoon	377