RETAIL SALES ANALYSIS USING SQL

By Maira Nawaz

INTRODUCTION

- **Project Overview**

This project demonstrates SQL techniques used by data analysts to explore, clean, and analyze retail sales data. The goal is to set up a retail sales database and extract meaningful business insights from the data.

PROBLEM STATEMENT

- **Understanding Retail Sales Performance**

Retail businesses generate large volumes of transactional data, which, when analyzed, can reveal valuable insights into customer behavior, sales trends, and product performance.

- **Challenges:**
  - Identifying missing data and cleaning the dataset
  - Understanding customer demographics and sales trends
  - Extracting insights to optimize business decisions

BUSINESS REQUIREMENTS:

- Write a SQL query to retrieve all columns for sales made on '2022-11-05
- Write a SQL query to retrieve all transactions where the category is 'Clothing' and the quantity sold is more than 10 in the month of Nov-2022
- Write a SQL query to calculate the total sales (total_sale) for each category.
- Write a SQL query to find the average age of customers who purchased items from the 'Beauty' category.
- Write a SQL query to find all transactions where the total_sale is greater than 1000.
- Write a SQL query to find the total number of transactions (transaction_id) made by each gender in each category.
- Write a SQL query to calculate the average sale for each month. Find out best selling month in each year
- Write a SQL query to find the top 5 customers based on the highest total sales
- Write a SQL query to find the number of unique customers who purchased items from each category.
- Write a SQL query to create each shift and number of orders (Example Morning <=12, Afternoon Between 12 & 17, Evening >17)

CREATING DATABASE

```
1 •  CREATE DATABASE retail_sales_analysis
2
```

```sql
CREATE TABLE  retail_sales(
    transaction_id INT PRIMARY KEY,
    sale_date DATE,
    sale_time TIME,
    customer_id INT,
    gender VARCHAR(10),
    age INT,
    category VARCHAR(50),
    quantity INT,
    price_per_unit DECIMAL(10,2),
    cogs DECIMAL(10,2),
    total_sale DECIMAL(10,2)
);
```

CREATING TABLE

DATA EXPLORATION

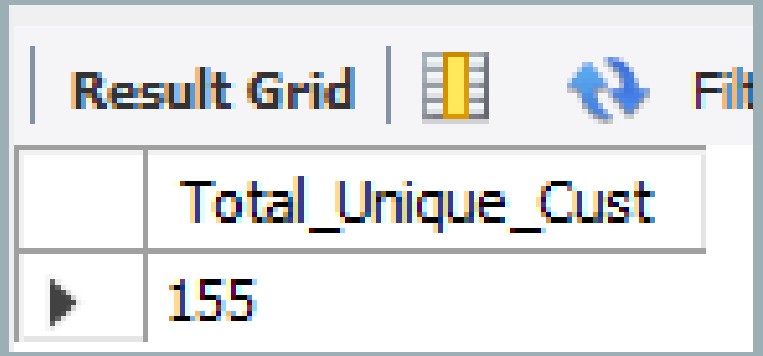
```
3   -- 1. Total Sales
4 • SELECT
5       COUNT(*)
6   FROM
7       retail_sales
```

Result Grid

	COUNT(*)
▶	1987

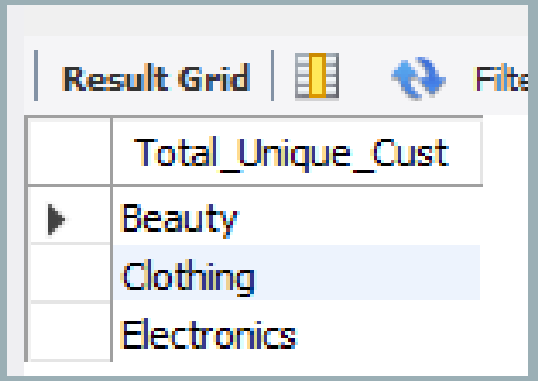
DATA EXPLORATION

```
 9   -- 2. Total Unique Customers
10   SELECT
11       COUNT(DISTINCT customer_id) AS Total_Unique_Cust
12   FROM
13       retail_sales
```



DATA EXPLORATION

```sql
-- 3. Total Distinct Categories
SELECT
    DISTINCT category AS Total_Unique_Cust
FROM
    retail_sales
```



BUSINESS REQUIREMENTS

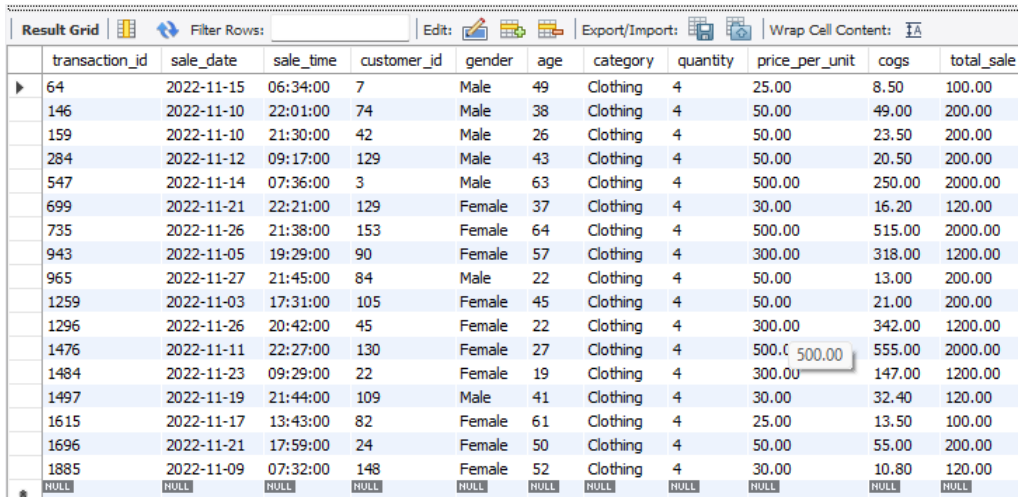
Q.1 Write a SQL query to retrieve all columns for sales made on '2022-11-05

```sql
SELECT
    *
FROM
    retail_sales
WHERE
    sale_date = '2022-11-05'
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

transaction_id	sale_date	sale_time	customer_id	gender	age	category	quantity	price_per_unit	cogs	total_sale
180	2022-11-05	10:47:00	117	Male	41	Clothing	3	300.00	129.00	900.00
214	2022-11-05	16:31:00	53	Male	20	Beauty	2	30.00	8.10	60.00
240	2022-11-05	11:49:00	95	Female	23	Beauty	1	300.00	123.00	300.00
856	2022-11-05	17:43:00	102	Male	54	Electronics	4	30.00	9.30	120.00
943	2022-11-05	19:29:00	90	Female	57	Clothing	4	300.00	318.00	1200.00
1137	2022-11-05	22:34:00	104	Male	46	Beauty	2	500.00	145.00	1000.00
1256	2022-11-05	09:58:00	29	Male	23	Clothing	2	500.00	190.00	1000.00
1265	2022-11-05	14:35:00	86	Male	55	Clothing	3	300.00	111.00	900.00
1587	2022-11-05	20:06:00	140	Female	40	Beauty	4	300.00	105.00	1200.00
1819	2022-11-05	20:44:00	83	Female	35	Beauty	2	50.00	13.50	100.00
1896	2022-11-05	20:19:00	87	Female	30	Electronics	2	25.00	30.75	50.00
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Q.2 Write a SQL query to retrieve all transactions where the category is 'clothing' and the quantity sold is more than 10 in the month of nov-2022



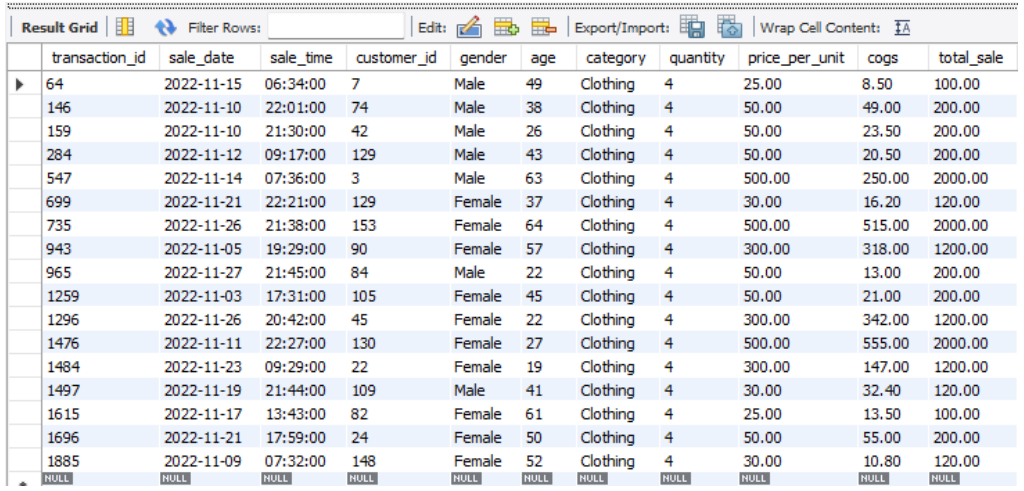
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

transaction_id	sale_date	sale_time	customer_id	gender	age	category	quantity	price_per_unit	cogs	total_sale
64	2022-11-15	06:34:00	7	Male	49	Clothing	4	25.00	8.50	100.00
146	2022-11-10	22:01:00	74	Male	38	Clothing	4	50.00	49.00	200.00
159	2022-11-10	21:30:00	42	Male	26	Clothing	4	50.00	23.50	200.00
284	2022-11-12	09:17:00	129	Male	43	Clothing	4	50.00	20.50	200.00
547	2022-11-14	07:36:00	3	Male	63	Clothing	4	500.00	250.00	2000.00
699	2022-11-21	22:21:00	129	Female	37	Clothing	4	30.00	16.20	120.00
735	2022-11-26	21:38:00	153	Female	64	Clothing	4	500.00	515.00	2000.00
943	2022-11-05	19:29:00	90	Female	57	Clothing	4	300.00	318.00	1200.00
965	2022-11-27	21:45:00	84	Male	22	Clothing	4	50.00	13.00	200.00
1259	2022-11-03	17:31:00	105	Female	45	Clothing	4	50.00	21.00	200.00
1296	2022-11-26	20:42:00	45	Female	22	Clothing	4	300.00	342.00	1200.00
1476	2022-11-11	22:27:00	130	Female	27	Clothing	4	500.(500.00	555.00	2000.00
1484	2022-11-23	09:29:00	22	Female	19	Clothing	4	300.00	147.00	1200.00
1497	2022-11-19	21:44:00	109	Male	41	Clothing	4	30.00	32.40	120.00
1615	2022-11-17	13:43:00	82	Female	61	Clothing	4	25.00	13.50	100.00
1696	2022-11-21	17:59:00	24	Female	50	Clothing	4	50.00	55.00	200.00
1885	2022-11-09	07:32:00	148	Female	52	Clothing	4	30.00	10.80	120.00
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```sql
SELECT
    *
FROM
    retail_sales
WHERE
    category = 'Clothing'
        AND DATE_FORMAT(sale_date, '%Y-%m') = '2022-11'
        AND quantity >= 4
```

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

```sql
SELECT
    category,
    SUM(total_sale) AS net_sales,
    COUNT(*) AS total_order
FROM
    retail_sales
GROUP BY 1
```



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

transaction_id	sale_date	sale_time	customer_id	gender	age	category	quantity	price_per_unit	cogs	total_sale
64	2022-11-15	06:34:00	7	Male	49	Clothing	4	25.00	8.50	100.00
146	2022-11-10	22:01:00	74	Male	38	Clothing	4	50.00	49.00	200.00
159	2022-11-10	21:30:00	42	Male	26	Clothing	4	50.00	23.50	200.00
284	2022-11-12	09:17:00	129	Male	43	Clothing	4	50.00	20.50	200.00
547	2022-11-14	07:36:00	3	Male	63	Clothing	4	500.00	250.00	2000.00
699	2022-11-21	22:21:00	129	Female	37	Clothing	4	30.00	16.20	120.00
735	2022-11-26	21:38:00	153	Female	64	Clothing	4	500.00	515.00	2000.00
943	2022-11-05	19:29:00	90	Female	57	Clothing	4	300.00	318.00	1200.00
965	2022-11-27	21:45:00	84	Male	22	Clothing	4	50.00	13.00	200.00
1259	2022-11-03	17:31:00	105	Female	45	Clothing	4	50.00	21.00	200.00
1296	2022-11-26	20:42:00	45	Female	22	Clothing	4	300.00	342.00	1200.00
1476	2022-11-11	22:27:00	130	Female	27	Clothing	4	500.00	555.00	2000.00
1484	2022-11-23	09:29:00	22	Female	19	Clothing	4	300.00	147.00	1200.00
1497	2022-11-19	21:44:00	109	Male	41	Clothing	4	30.00	32.40	120.00
1615	2022-11-17	13:43:00	82	Female	61	Clothing	4	25.00	13.50	100.00
1696	2022-11-21	17:59:00	24	Female	50	Clothing	4	50.00	55.00	200.00
1885	2022-11-09	07:32:00	148	Female	52	Clothing	4	30.00	10.80	120.00
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

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

```sql
SELECT
    ROUND(AVG(age), 2) AS avg_age
FROM
    retail_sales
WHERE
    category = 'Beauty'
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

transaction_id	sale_date	sale_time	customer_id	gender	age	category	quantity	price_per_unit	cogs	total_sale
64	2022-11-15	06:34:00	7	Male	49	Clothing	4	25.00	8.50	100.00
146	2022-11-10	22:01:00	74	Male	38	Clothing	4	50.00	49.00	200.00
159	2022-11-10	21:30:00	42	Male	26	Clothing	4	50.00	23.50	200.00
284	2022-11-12	09:17:00	129	Male	43	Clothing	4	50.00	20.50	200.00
547	2022-11-14	07:36:00	3	Male	63	Clothing	4	500.00	250.00	2000.00
699	2022-11-21	22:21:00	129	Female	37	Clothing	4	30.00	16.20	120.00
735	2022-11-26	21:38:00	153	Female	64	Clothing	4	500.00	515.00	2000.00
943	2022-11-05	19:29:00	90	Female	57	Clothing	4	300.00	318.00	1200.00
965	2022-11-27	21:45:00	84	Male	22	Clothing	4	50.00	13.00	200.00
1259	2022-11-03	17:31:00	105	Female	45	Clothing	4	50.00	21.00	200.00
1296	2022-11-26	20:42:00	45	Female	22	Clothing	4	300.00	342.00	1200.00
1476	2022-11-11	22:27:00	130	Female	27	Clothing	4	500.00	555.00	2000.00
1484	2022-11-23	09:29:00	22	Female	19	Clothing	4	300.00	147.00	1200.00
1497	2022-11-19	21:44:00	109	Male	41	Clothing	4	30.00	32.40	120.00
1615	2022-11-17	13:43:00	82	Female	61	Clothing	4	25.00	13.50	100.00
1696	2022-11-21	17:59:00	24	Female	50	Clothing	4	50.00	55.00	200.00
1885	2022-11-09	07:32:00	148	Female	52	Clothing	4	30.00	10.80	120.00

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

```sql
SELECT
    *
FROM
    retail_sales
WHERE
    total_sale > 1000
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

transaction_id	sale_date	sale_time	customer_id	gender	age	category	quantity	price_per_unit	cogs	total_sale
13	2023-02-08	17:43:00	106	Male	22	Electronics	3	500.00	245.00	1500.00
15	2022-07-01	11:50:00	75	Female	42	Electronics	4	500.00	210.00	2000.00
16	2022-06-25	10:33:00	82	Male	19	Clothing	3	500.00	180.00	1500.00
31	2023-12-31	17:47:00	3	Male	44	Electronics	4	300.00	129.00	1200.00
46	2022-11-08	17:50:00	54	Female	20	Electronics	4	300.00	84.00	1200.00
47	2022-10-22	17:22:00	96	Female	40	Beauty	3	500.00	600.00	1500.00
54	2022-10-20	10:17:00	142	Female	38	Electronics	3	500.00	200.00	1500.00
58	2023-09-16	19:18:00	53	Male	18	Clothing	4	300.00	75.00	1200.00
65	2022-12-11	20:03:00	84	Male	51	Electronics	4	500.00	160.00	2000.00
67	2023-08-19	20:19:00	119	Female	48	Beauty	4	300.00	129.00	1200.00
72	2023-12-06	19:19:00	5	Female	20	Electronics	4	500.00	195.00	2000.00
74	2023-10-05	19:50:00	56	Female	18	Beauty	4	500.00	205.00	2000.00
78	2023-02-17	21:08:00	68	Female	47	Clothing	3	500.00	265.00	1500.00
89	2023-12-30	21:15:00	117	Female	55	Electronics	4	500.00	590.00	2000.00
93	2022-01-25	20:52:00	148	Female	35	Beauty	4	500.00	140.00	2000.00
99	2023-11-19	15:12:00	71	Female	50	Electronics	4	300.00	132.00	1200.00
107	2022-10-06	09:18:00	75	Female	21	Clothing	4	300.00	78.00	1200.00
109	2023-09-06	19:57:00	94	Female	34	Electronics	4	500.00	560.00	2000.00

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

```sql
SELECT
    category, gender, COUNT(*) AS total_transactions
FROM
    retail_sales
GROUP BY 1 , 2
ORDER BY 3
```

Result Grid | Filter Rows:

category	gender	total_transactions
Beauty	Male	281
Beauty	Female	330
Electronics	Female	335
Electronics	Male	343
Clothing	Female	347
Clothing	Male	351

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

```sql
WITH ranked_sales AS (
    SELECT
        EXTRACT(YEAR FROM sale_date) AS year,
        EXTRACT(MONTH FROM sale_date) AS month,
        AVG(total_sale) AS avg_sale,
        RANK() OVER (PARTITION BY EXTRACT(YEAR FROM sale_date)
        ORDER BY AVG(total_sale) DESC)
        AS rank_value
    FROM retail_sales
    GROUP BY year, month
)
SELECT
    year,
    month,
    avg_sale
FROM ranked_sales
WHERE rank_value = 1;
```

Result Grid | Filter Rows:

year	month	avg_sale
2022	7	541.341463
2023	2	535.531915

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

```sql
SELECT
    customer_id, SUM(total_sale) AS net_sales
FROM
    retail_sales
GROUP BY 1
ORDER BY 2 DESC
LIMIT 5
```

Result Grid | Filter Rows:

customer_id	net_sales
3	38440.00
1	30750.00
5	30405.00
2	25295.00
4	23580.00

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

```sql
SELECT
    category, COUNT(DISTINCT customer_id) AS unique_customers
FROM
    retail_sales
GROUP BY 1
```

Result Grid | Filter Rows:

	category	unique_customers
▶	Beauty	141
	Clothing	149
	Electronics	144

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

```sql
WITH hourly_sale
AS
(
SELECT *,
    CASE
        WHEN EXTRACT(HOUR FROM sale_time) < 12 THEN 'Morning'
        WHEN EXTRACT(HOUR FROM sale_time) BETWEEN 12 AND 17 THEN 'Afternoon'
        ELSE 'Evening'
    END as shift
FROM retail_sales
)
SELECT
    shift,
    COUNT(*) as total_orders
FROM hourly_sale
GROUP BY shift
```

Result Grid | Filter Rows:

shift	total_orders
Evening	1062
Morning	548
Afternoon	377

CONCLUSION

Key Takeaways

- SQL is powerful for retail data analysis
- Data cleaning improves analysis accuracy
- Business insights help optimize sales & customer strategy

Next Steps:

- Automate SQL queries for continuous reporting
- Build a Power BI/Excel dashboard for better visualization

THANK YOU