

Project Overview:

This project explores Walmart sales data to identify top-performing branches and products, analyze sales trends across various product categories, and understand customer behavior. The objective is to uncover insights that can inform sales strategies and optimize performance. The dataset was sourced from the Kaggle Walmart Sales Forecasting Competition.

The competition provided historical sales data for 45 Walmart stores across different regions. Each store contains multiple departments, and participants were tasked with forecasting sales for each department. The dataset includes selected holiday markdown events, which add complexity to the analysis as these markdowns can significantly impact sales, but the extent and affected departments are unpredictable.

Source: Kaggle Walmart Sales Forecasting Competition

Project Objectives:

The primary objective of this project is to analyze Walmart's sales data to identify the key factors influencing sales across various branches and departments. By examining trends and patterns, the goal is to provide actionable insights to refine sales strategies and optimize overall performance.

Feature Engineering:

Q1: Add a new column named `time_of_day` to give insight of sales in the Morning, Afternoon and Evening. This will help answer the question on which part of the day most sales are made.

Step 1: Create a new column in the table:

```
Alter table walmart_sales.walmartsalesdata  
add column time_of_day Varchar(20);
```

Step 2: Use the update statement to add the data into new column.

```
Update walmart_sales.walmartsalesdata  
set time_of_day = (
```

```
case
when time between '00:00:00' and '11:59:59' then 'Morning'
when time between '12:00:00' and '15:59:59' then 'Afternoon'
else 'Evening'
END
);
```

Q2: Add a new column named day_name that contains the extracted days of the week on which the given transaction took place (Mon, Tue, Wed, Thur, Fri). This will help answer the question on which week of the day each branch is busiest.

```
Alter table walmart_sales.walmartsalesdata
add column day_name Varchar(10);
```

```
update walmart_sales.walmartsalesdata
set day_name = (dayname(date));
```

Q3: Add a new column named month_name that contains the extracted months of the year on which the given transaction took place (Jan, Feb, Mar). Help determine which month of the year has the most sales and profit.

```
ALTER TABLE walmart_sales.walmartsalesdata
ADD COLUMN month_name VARCHAR(10);
```

```
update walmart_sales.walmartsalesdata
set month_name = (monthname(date));
```

Business Questions To Answer

* Generic Question

1. How many unique cities does the data have?

```
Select count(distinct city) unique_city_count
from walmart_sales.walmartsalesdata;
```

2. In which city is each branch?

*Select distinct branch, city
from walmart_sales.walmartsalesdata;*

Product→

Problem 1: **How many unique product lines does the data have?**
3.

*select
count(distinct product_line)
from walmart_sales.walmartsalesdata*

Problem 2: **What is the most common payment method?**

Ewallet is the most common payment method with most payments.

*select
Payment, count(1) Total_payments
from walmart_sales.walmartsalesdata
group by 1
order by 2 desc;*

Problem 3: **What is the best-selling product line?**
Electronic accessories (971)

*select
product_line,
sum(quantity) Best_seller
from walmart_sales.walmartsalesdata
group by 1
order by 2 desc
limit 1;*

Problem 4: **What is the total revenue by month?**

*select
month(date) as mnth,*

```
month_name,  
Round(sum(total),2) total_income  
from walmart_sales.walmartsalesdata  
group by 1,2  
order by 1 asc
```

Problem 5: **What month had the largest COGS?**

```
select  
month(date) as mnth,  
month_name,  
Round(sum(cogs),2) COGS  
from walmart_sales.walmartsalesdata  
group by 1,2  
order by 3 desc ;
```

Problem 6: **What product line had the largest revenue?**

```
select  
product_line,  
Round(sum(total),2) Total_Revenue  
from walmart_sales.walmartsalesdata  
group by 1  
order by 2 desc  
limit 1;
```

Problem 7: **What is the city with the largest revenue?**

```
select  
city,  
Round(sum(total),2) Total_Revenue  
from walmart_sales.walmartsalesdata  
group by 1  
order by 2 desc  
limit 1;
```

Problem 8: **What product line had the largest VAT?**
(\$ VAT = 5% * COGS \$)

```
select
product_line,
max(`Tax_5%`) as tax
from walmart_sales.walmartsalesdata
group by 1
order by 2 desc;
OR
```

```
SELECT
    product_line,
    Round( SUM(cogs * 0.05),2) AS total_vat_amount
FROM walmart_sales.walmartsalesdata
GROUP BY product_line
ORDER BY total_vat_amount DESC
```

Problem 9: **Fetch each product line and add a column to those product line showing "Good", "Bad". Good if its greater than average sales**

```
select
    product_line,
    Round(avg(total),2) as product_line_avg,
    Round((select avg(total)from walmart_sales.walmartsalesdata),2) total_avg,
    case
        when avg(total) >
            (select avg(total)from walmart_sales.walmartsalesdata)
        then 'Good'
        else 'Bad'
    end as Category
    from walmart_sales.walmartsalesdata
```

```
group by 1
order by 2 desc ;
```

Problem 10: **Which branch sold more products than average product sold?**

```
select
branch,
sum(Quantity) quantity_sold
from walmart_sales.walmartsalesdata
group by 1
having sum(quantity) > (select
avg(Quantity)
from walmart_sales.walmartsalesdata);
```

Problem 11: **What is the most common product line by gender?**

```
select
gender,
product_line,
count(product_line)
from walmart_sales.walmartsalesdata
group by 1,2
order by 3 desc;
```

Problem 12: **What is the average rating of each product line?**

```
select
product_line,
round(avg(rating),2) Avg_rating
from walmart_sales.walmartsalesdata
group by 1
```

Sales→

Problem1: **Number of sales made in each time of the day per weekday**

```
select
```

```
count(quantity) Total_sales,  
time_of_day  
from walmart_sales.walmartsalesdata  
where day_name not in ('Saturday', 'Sunday')  
group by 2  
order by 1 desc;
```

Problem 2: **Which of the customer types brings the most revenue?**

```
select  
Round(sum(total),2) Total_Sales,  
customer_type  
from walmart_sales.walmartsalesdata  
group by 2  
order by 1 desc  
limit 1;
```

Problem 3: **Which city has the largest tax percent/ VAT (Value Added Tax)?**

```
select  
max(`Tax_5%`) Total_Tax,  
city  
from walmart_sales.walmartsalesdata  
group by 2  
order by 1 desc  
limit 1;
```

Problem 4: **Which customer type pays the most in VAT?**

```
select  
max(`Tax_5%`) Total_Tax,  
customer_type  
from walmart_sales.walmartsalesdata  
group by 2  
order by 1 desc  
limit 1;
```

Customer→

Problem 1: **How many unique customer types does the data have?**

```
SELECT
    count(distinct customer_type) as unique_cx_type
FROM walmart_sales.walmartsalesdata;
```

Problem 2: **How many unique payment methods does the data have?**

```
SELECT
    count(distinct payment) as unique_cx_type
FROM walmart_sales.walmartsalesdata;
```

Problem 3: **What is the most common customer type?**

```
SELECT
    count(1) as cx_count,
    Customer_type
FROM walmart_sales.walmartsalesdata
group by 2
limit 1;
```

Problem 4: **Which customer type buys the most?**

```
SELECT
    sum(total) as total_sale,
    Customer_type
FROM walmart_sales.walmartsalesdata
group by 2
order by 1 desc
limit 1;
```

Problem 5: **What is the gender of most of the customers?**

```
SELECT
    count(1) as total_cx,
    gender
FROM walmart_sales.walmartsalesdata
group by 2
order by 1 desc
limit 1;
```

Problem 6: **What is the gender distribution per branch?**

```
SELECT
    branch,
    gender,
    COUNT(1) AS cnt,
    ROUND((COUNT(1) * 100.0) / SUM(COUNT(1)) OVER(PARTITION BY branch), 2) AS
percentage_distribution
FROM walmart_sales.walmartsalesdata
GROUP BY branch, gender
ORDER BY branch, percentage_distribution DESC;
```

Problem 7: **Which time of the day do customers give most ratings?**

```
SELECT
    time_of_day,
    count(rating) AS rating_count
FROM walmart_sales.walmartsalesdata
GROUP BY time_of_day
ORDER BY rating_count DESC
LIMIT 1;
```

Problem 8: **Which time of the day do customers give most ratings per branch?**

With cte1 as

```
(SELECT
  branch,
  time_of_day,
  COUNT(rating) AS rating_count,
  row_number() over(partition by branch order by count(1) desc ) rk
FROM walmart_sales.walmartsalesdata
GROUP BY branch, time_of_day
ORDER BY branch, rating_count DESC)
```

```
Select branch, time_of_day,
Rating_count
From cte1
where cte1.rk = 1;
```

Problem 9: **Which day of the week has the best avg ratings?**

```
SELECT
avg(Rating),
day_name
FROM walmart_sales.walmartsalesdata
group by 2
order by 1 desc
limit 1;
```

Problem 10: **Which day of the week has the best average ratings per branch?**

```
WITH cte1 AS (
  SELECT
    branch,
    day_name,
    ROUND(AVG(Rating), 2) AS avg_rating,
    ROW_NUMBER() OVER (PARTITION BY branch ORDER BY AVG(Rating) DESC) AS
rank_by_avg_rating
  FROM walmart_sales.walmartsalesdata
  GROUP BY branch, day_name
```

```
)  
SELECT branch, day_name, avg_rating  
FROM cte1  
WHERE rank_by_avg_rating = 1;
```