```
Create Database Walmart_Sales_Data_Analysis;

Create Table walmart_sales(

invoice_id varchar(30) not null PRIMARY KEY,
branch varchar(5) not null,
city varchar(30) not null,
customer_type varchar(30) not null,
gender varchar(10) not null,
product_line varchar(100) not null,
unit_price decimal(10,2) not null,
quantity int not null,
tax_pct FLOAT not null,
total decimal(12,4) not null,
date datetime not null,
time time not null,
payment varchar(15) not null,
cogs decimal(10,2) not null,
gross_margin_pct FLOAT NOT NULL,
gross_income decimal(12,4),
rating float NOT NULL
);
```

1. 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.

```
ALTER TABLE walmart_sales1
ADD time_of_day VARCHAR(15);

UPDATE walmart_sales1
SET time_of_day =
  CASE
    WHEN CAST(time AS TIME) BETWEEN '06:00:00' AND '11:59:59' THEN 'Morning'
    WHEN CAST(time AS TIME) BETWEEN '12:00:00' AND '17:59:59' THEN 'Afternoon'
```

```
    WHEN CAST(time AS TIME) BETWEEN '18:00:00' AND '23:59:59' THEN 'Evening'
    ELSE 'Late Night'
  END;
```

2. 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_sales1
ADD day_name VARCHAR(15);

UPDATE walmart_sales1
SET day_name = LEFT(DATENAME(WEEKDAY, date),3);
```

3. 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_sales1
ADD month_name VARCHAR(10);

UPDATE walmart_sales1
SET month_name = LEFT(DATENAME(MONTH, date),3);
```

----------Business Questions To Answer----------

---- Generic Questions ------

1. How many unique cities does the data have?

```
SELECT COUNT(Distinct(City))
FROM walmart_sales1;
```

2. In which city is each branch?

```
SELECT DISTINCT City, Branch
FROM walmart_sales1;
```

---------- PRODUCT ----------

1. How many unique product lines does the data have?

```
SELECT
        COUNT(distinct(Product_line))
FROM walmart_sales1;        ------ "6" : six unique product lines
```

2. What is the most common payment method?

```
SELECT TOP 1
        Payment,
        Count(Payment) as payment_method_COUNT
FROM walmart_sales1
GROUP BY Payment
ORDER BY payment_method_COUNT DESC;
                        ----------"E-wallet" : is the most common payment method
```

3. What is the most selling product line?

```
SELECT TOP 1
        Product_line,
        ROUND(SUM(total),2) as total_sales
FROM walmart_sales1
GROUP BY Product_line
ORDER BY total_sales DESC;        ------ ''Food and beverages'' : is the most selling product line
```

4. What is the total revenue by month?

```
SELECT
        month_name,
        ROUND(SUM(total),2) as total_sales
FROM walmart_sales1
GROUP BY month_name
ORDER BY total_sales DESC;
```

5. What month had the largest COGS?

```
SELECT TOP 1
        month_name,
        ROUND(SUM(cogs),2) as total_cogs
FROM walmart_sales1
GROUP BY month_name
ORDER BY total_cogs DESC;
```

------- "January" month has the longest COGS : 110754.16

6. What product line had the largest revenue?

```
SELECT TOP 1
        Product_line,
        ROUND(SUM(Unit_price * Quantity),2) as Total_revenue
FROM walmart_sales1
GROUP BY Product_line
ORDER BY Total_revenue DESC;
```

---------- "Food and beverages" has the highest revenue = 53471.28

7. What is the city with the largest revenue?

```
SELECT TOP 1
        City,
        ROUND(SUM(Unit_price * Quantity),2) as Total_revenue
FROM walmart_sales1
GROUP BY City
```

```
ORDER BY Total_revenue DESC;
```

---------- ""Naypyitaw"" is the city with largest revenue = 105303.53

8. What product line had the largest VAT? (5%)

```
SELECT TOP 1
        Product_line,
        ROUND(SUM(0.05*cogs),2) as largest_VAT
FROM walmart_sales1
GROUP BY Product_line
ORDER BY largest_VAT DESC;
```

--------- "Food and beverages" is the largest VAT product_line with : 2673.56

9. Fetch each product line and add a column to those product line showing "Good", "Bad". Good if its greater than average sales [using CTE function]

```
WITH Product_sale_performance AS (
        SELECT
                Product_line,
                ROUND(SUM(Total),2) AS Total_sales
        FROM walmart_sales1
        GRoUP BY Product_line
)

SELECT
        Product_line,
        Total_sales,
        CASE
                WHEN Total_sales > ( select AVG(Total_sales) from Product_sale_performance ) THEN 'GOOD'
                ELSE 'BAD'
        END AS PERFORMANCE
FROM Product_sale_performance
ORDER BY Product_line;
```

10. Which branch sold more products than average product sold? [using CTE function]

```
WITH BRANCH_QTY_SALES_AVG AS (
  SELECT
    Branch,
    SUM(Quantity) AS Branch_quantity_sales
  FROM walmart_sales1
  GROUP BY Branch
),
AVG_SALES AS (
  SELECT AVG(Branch_quantity_sales) AS Average_quantity_sales
  FROM BRANCH_QTY_SALES_AVG
)
SELECT
  bqa.Branch,  -- 'bqa' is used here as a shorthand for BRANCH_QTY_SALES_AVG
  bqa.Branch_quantity_sales,
  CASE
    WHEN bqa.Branch_quantity_sales > (SELECT Average_quantity_sales FROM AVG_SALES) THEN 'ABOVE'
    ELSE 'BELOW'
  END AS AVERAGE_PRODUCT_SOLD
FROM BRANCH_QTY_SALES_AVG bqa  -- Alias for the BRANCH_QTY_SALES_AVG CTE
ORDER BY bqa.Branch;
```

---------- "Branch-A" sold more products then average products sold by other branches

11. What is the most common product line by gender? [using CTE function]

```
WITH Product_line_Gender_Count AS (
        SELECT
                Product_line,
                Gender,
                COUNT(*) AS Gender_Count
        FROM walmart_sales1
        GROUP BY Product_line, Gender
),
Ranked_Product_Lines AS (
        SELECT
                Product_line,
```

```
                Gender,
                Gender_Count,
                RANK() OVER (PARTITION BY Gender ORDER BY Gender_Count DESC) AS Rank
        FROM Product_line_Gender_Count
)
SELECT
        Gender,
        Product_line AS Most_common_Product_Line,
        Gender_Count
FROM Ranked_Product_Lines
WHERE Rank = 1
ORDER BY Gender;
```

---------- Fashion accessories : Female = 96

----------Health and beauty : Male = 88

12. What is the average rating of each product line?

```
SELECT
        Product_line,
        ROUND(AVG(Rating),2) as average_rating
FROM walmart_sales1
GROUP BY Product_line
ORDER BY average_rating DESC;
```

---------- SALES ----------

1. Number of sales made in each 'time of the day' per weekday

```
SELECT
  time_of_day,
  day_name,
  COUNT(Total) AS number_of_sales
FROM walmart_sales1
WHERE day_name NOT IN ('Sat', 'Sun')
GROUP BY time_of_day, day_name
ORDER BY day_name ASC, time_of_day DESC;
```

2. Which of the customer types brings the most revenue?

```
SELECT
        Customer_type,
        ROUND(SUM(Total),2) AS Total_revenue
FROM walmart_sales1
GROUP BY Customer_type
ORDER BY Customer_type, Total_revenue;
```

---------- 'Member' customer type generating most revenue compared to 'Normal' customers

3. Which city has the largest tax percentage/ VAT (**Value Added Tax**)?

```
WITH Total_Tax AS (
        SELECT
                ROUND(SUM(Tax_5),2) AS TOTAL_TAX
        FROM walmart_sales1
)
SELECT
        City,
        ROUND(SUM(Tax_5),2) AS Total_Tax_contribution_each_city,
        ROUND(SUM(Tax_5) / (SELECT TOTAL_TAX FROM Total_Tax) * 100, 2) AS Tax_Contribution_Percentage
FROM walmart_sales1
GROUP BY City
ORDER BY Tax_Contribution_Percentage DESC;
```

---------- "Naypyitaw : 34.24%" is the city contributing more compared to other two cities (Yangon: 32.88% & Mandalay: 32.88%)

4. Which customer type pays the most in VAT?

```
SELECT
        Customer_type,
        ROUND(SUM(Tax_5),2) AS Total_vat_paid
FROM walmart_sales1
GROUP BY Customer_type
ORDER BY Customer_type, Total_vat_paid DESC;
```

---------- " Member-customer :7820.16 " has paid more VAT then the ' Normal - customer : 7820.16 '

---------- CUSTOMERS ----------

1. How many unique customer types does the data have?

```
SELECT
        COUNT(DISTINCT(Customer_type))
FROM walmart_sales1;
```

---------- "2" there are 2 distinct customer types in this dataset.

2. How many unique payment methods does the data have?

```
SELECT
        COUNT(DISTINCT(Payment))
FROM walmart_sales1;
```

---------- "3" There are 3 distinct payment methods are there in this dataset.

3. What is the most common customer type?

```
SELECT
                Customer_type,
                Count(Quantity) AS most_common_customer
FROM walmart_sales1
GROUP BY Customer_type
ORDER BY Customer_type, most_common_customer;
```

4. What customer type buys the most ?

```
SELECT
        Customer_type,
        ROUND(SUM(Total),2) AS Total_purchase_Amount
FROM walmart_sales1
GROUP BY Customer_type
ORDER BY Customer_type, Total_purchase_Amount;
```

-------- " Member " is the most common customer_type with :164223.44

5. What is the gender of most of the customers?

```
SELECT
        Gender,
        ROUND(SUM(Total),2) AS Total_purchase_Amount
```

```
FROM walmart_sales1
GROUP BY Gender
ORDER BY Gender, Total_purchase_Amount;
```

------- " Female " is the gender of most frequent customers

6. What is the gender distribution per branch?

```
SELECT              ---------- Quert=y for total sales per gender per branch
        Branch,
        Gender,
        ROUND(SUM(Total),2) AS total_sales
FROM walmart_sales1
GROUP BY Gender, Branch
ORDER BY total_sales;

SELECT
        ---------- Query for Count-based Distribution
  Branch,
  Gender,
  COUNT(*) AS transaction_count
FROM walmart_sales1
GROUP BY Branch, Gender
ORDER BY Branch, transaction_count DESC;

SELECT
        ---------- Query for Percentage-based Distribution
  Branch,
  Gender,
  COUNT(*) AS transaction_count,
  ROUND((COUNT(*) * 100.0) / SUM(COUNT(*)) OVER (PARTITION BY Branch), 2) AS percentage_distribution
FROM walmart_sales1
GROUP BY Branch, Gender
ORDER BY Branch, Gender;
```

7. Which time of the day do customers give most ratings?

```
SELECT
        time_of_day,
```

```
        COUNT(Rating) AS number_of_ratings
FROM walmart_sales1
GROUP BY time_of_day
ORDER BY number_of_ratings DESC;
```

---------- " Afternoon " most of the customers preffered afternoon for rating the service..which counts more than 50% out of total 1000 customers

8. Which time of the day do customers give the most ratings per branch?

```
SELECT
        Branch,
        time_of_day,
        COUNT(Rating) AS number_of_ratings
FROM walmart_sales1
GROUP BY time_of_day, Branch
ORDER BY number_of_ratings DESC, Branch ASC;
```

------ For all three branches (A, C, B) most of the customers preferred "Afternoon" for their preferred time to rating the service compared to morning and evening

9. Which day of the week has the best average ratings?

```
SELECT TOP 1
        day_name,
        ROUND(AVG(Rating),2) AS week_day_avg_rating
FROM walmart_sales1
GROUP BY day_name
ORDER BY week_day_avg_rating DESC;
```

------- " MONDAY = 7.15 " is having highest average rating compared to all other days in a week

10. Which day of the week has the best average ratings per branch?

```
WITH Ranked_Ratings AS (
        SELECT
                Branch,
                day_name,
                ROUND(AVG(Rating),2) AS week_day_avg_rating,
                RANK() OVER (PARTITION BY Branch ORDER BY  AVG(Rating) DESC) AS rank_per_branch
        FROM walmart_sales1
        GROUP BY Branch, day_name
```

```
)
SELECT
	Branch,
	day_name AS best_day,
	week_day_avg_rating AS highest_avg_rating
FROM Ranked_Ratings
WHERE rank_per_branch = 1
ORDER BY Branch ASC;
```

---------- " Friday " is the best average rating day for branch 'A'

---------- " Monday " is the best average rating day for branch 'B'

---------- " Friday " is the best average rating day for branch 'C'