Walmart Sales Data Analysis

MySQL

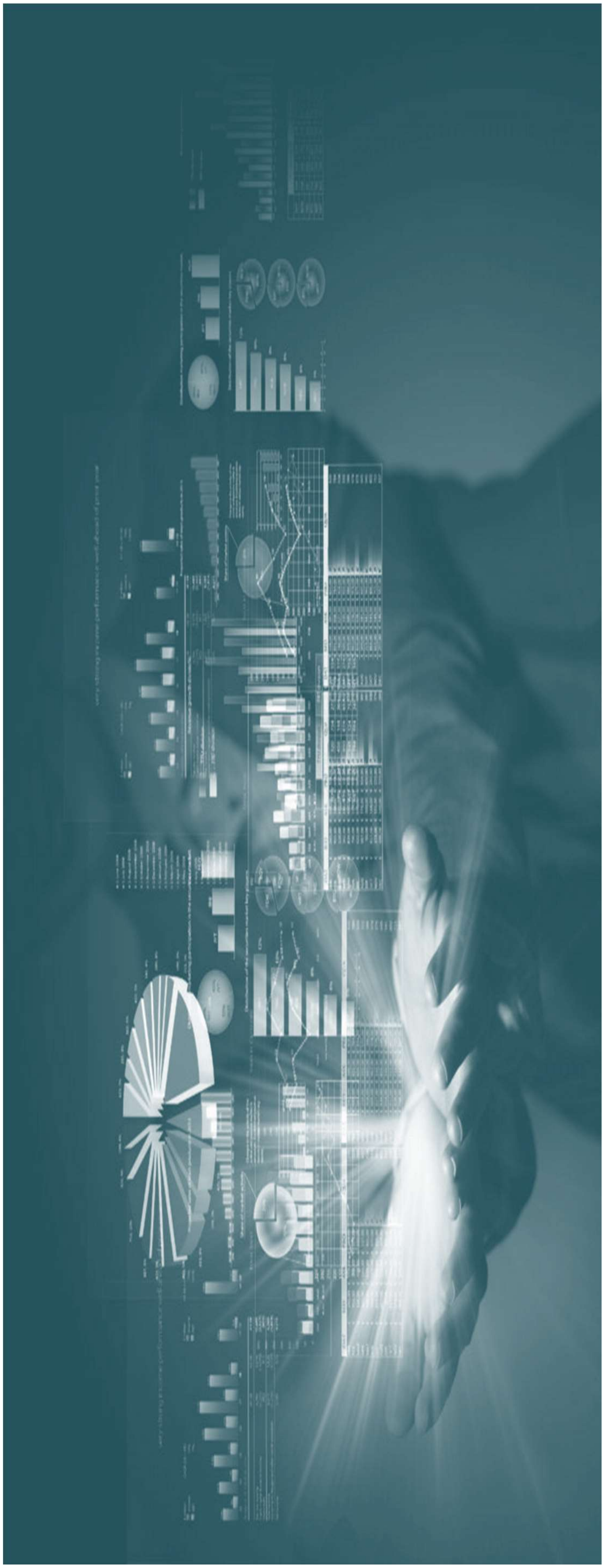


TABLE OF CONTENTS

INTRODUCTION

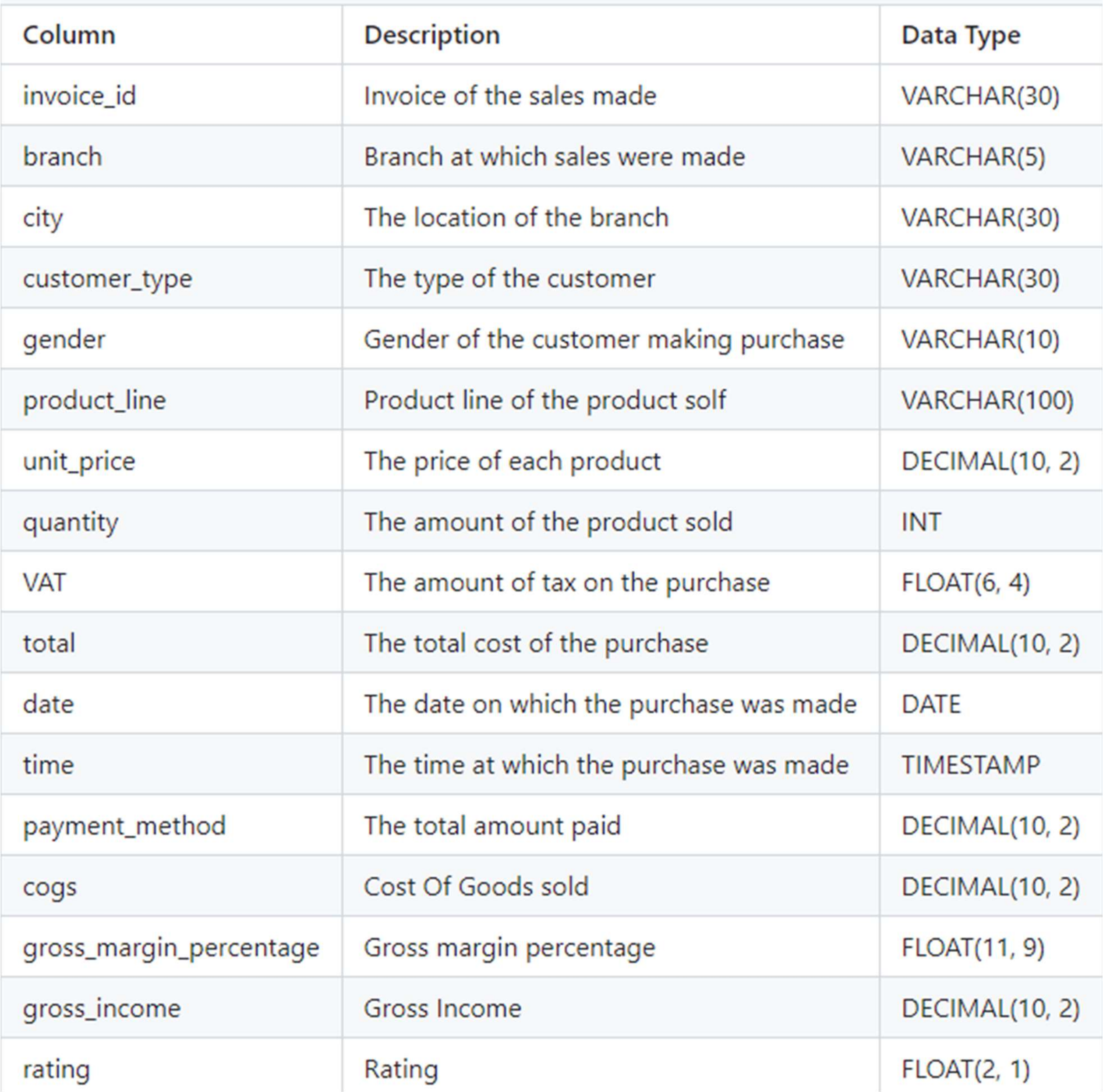
This project aims to explore the Walmart Sales data to understand top performing branches and products, sales trend of different products, customer behaviour. The aims is to study how sales strategies can be improved and optimized.

Purposes Of The Project

The major aim of thie project is to gain insight into the sales data of Walmart to understand the different factors that affect sales of the different branches.

1. ABOUT DATA

This dataset contains sales transactions from a three different branches of Walmart, respectively located in Mandalay, Yangon and Naypyitaw. The data contains 17 columns and 1000 rows.



Column	Description	Data Type
invoice_id	Invoice of the sales made	VARCHAR(30)
branch	Branch at which sales were made	VARCHAR(5)
city	The location of the branch	VARCHAR(30)
customer_type	The type of the customer	VARCHAR(30)
gender	Gender of the customer making purchase	VARCHAR(10)
product_line	Product line of the product solf	VARCHAR(100)
unit_price	The price of each product	DECIMAL(10, 2)
quantity	The amount of the product sold	INT
VAT	The amount of tax on the purchase	FLOAT(6, 4)
total	The total cost of the purchase	DECIMAL(10, 2)
date	The date on which the purchase was made	DATE
time	The time at which the purchase was made	TIMESTAMP
payment_method	The total amount paid	DECIMAL(10, 2)
cogs	Cost Of Goods sold	DECIMAL(10, 2)
gross_margin_percentage	Gross margin percentage	FLOAT(11, 9)
gross_income	Gross Income	DECIMAL(10, 2)
rating	Rating	FLOAT(2, 1)

2. ANALYSIS

1. Product Analysis

Conduct analysis on the data to understand the different product lines, the products lines performing best and the product lines that need to be improved.

2. Sales Analysis

This analysis aims to answer the question of the sales trends of product. The result of this can help use measure the effectiveness of each sales strategy the business applies and what modifications are needed to gain more sales.

3. Customer Analysis

This analysis aims to uncover the different customers segments, purchase trends and the profitability of each customer segment.

4. APPROACH USED

1. Data Wrangling:

This is the first step where inspection of data is done to make sure **NULL** values and missing values are detected and data replacement methods are used to replace, missing or **NULL** values.

- Build a database
- Create table and insert the data.
- Select columns with null values in them. There are no null values in our database as in creating the tables, we set **NOT NULL** for each field, hence null values are filtered out.

2. Feature Engineering:

This will help use generate some new columns from existing ones.

- 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.

- 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.

- 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.

3. Exploratory Data Analysis (EDA):

Exploratory data analysis is done to answer the listed questions and aims of this project.

Generic Question

1. How many unique cities does the data have?

2. In which city is each branch?

Product Analysis

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

2. What is the most common payment method?

3. What is the most selling product line?

4. What is the total revenue by month?

5. What month had the largest COGS?

6. What product line had the largest revenue?

5. What is the city with the largest revenue?

6. What product line had the largest VAT?

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

8. Which branch sold more products than average product sold?

9. What is the most common product line by gender?

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

Customer Analysis

1. How many unique customer types does the data have?
2. How many unique payment methods does the data have?
3. Which customer type buys the most?
4. What is the gender of most of the customers?
5. What is the gender distribution per branch?
6. Which time of the day do customers give most ratings?
7. Which time of the day do customers give most ratings per branch?
8. Which day fo the week has the best avg ratings?
9. Which day of the week has the best average ratings per branch?

Sales Analysis

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

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

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

4. Which customer type pays the most in VAT?

REVENUE AND PROFIT CALCULATIONS

COGS = Units Price * quantity

VAT = 5% * COGS

VAT is added to the COGS and this is what is billed to the customer.

Total (gross_sales) = VAT + COGS

Gross Profit = Total - COGS

Gross Margin is gross profit expressed in percentage of the total(gross profit/revenue)

Gross Margin = gross income / total revenue

Example with the first row in our DB

Data given:

Unite Price = 45.79 $
Quantity = 7 $
COGS = 45.79 * 7 = 320.53 $

VAT = 5% * COGS = 5% * 320.53 = **16.0265 $**

Total = VAT + COGS = 16.0265 + 320.53 = **336.5565 $**

Gross Margin Percentage = gross income / total revenue = 16.0265 / 336.5565
= 0.047619 approx
= **4.7619 %**

BUILD DATABASE

Create the database if it does not exist.

```
CREATE DATABASE IF NOT EXISTS WalmartSales;

USE WalmartSales;
```

Create the Sales table if it does not exist.

```
CREATE TABLE IF NOT EXISTS Sales(

        Invoice_ID VARCHAR(30) NOT NULL,

        Branch VARCHAR(5) NOT NULL,

        City VARCHAR(30) NOT NULL,

        Customer_type VARCHAR(30) NOT NULL,

        Gender VARCHAR(6) NOT NULL,

        Product_line VARCHAR(100) NOT NULL,

        Unit_price DECIMAL(10, 2) NOT NULL,

        Quantity INT NOT NULL,

        Tax DECIMAL(6, 4) NOT NULL,

        Total DECIMAL(12, 4) NOT NULL,

        Date DATE NOT NULL,

        Time TIME NOT NULL, -- Changed from DATETIME to TIME for storing time

        Payment_Method VARCHAR(15) NOT NULL,

        cogs DECIMAL(10, 2) NOT NULL,

        gross_margin_pct DECIMAL(11, 9) NOT NULL,

        gross_income DECIMAL(12, 4) NOT NULL,

        Rating DECIMAL(2, 1) NOT NULL

);
```

Disable strict SQL mode for the current session

```
SET SESSION sql_mode = '';
```

Load data from a CSV file into the Sales table

```
LOAD DATA INFILE 'E:\\WalmartSalesData.csv'
INTO TABLE Sales
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n'
IGNORE 1 ROWS;
```

Select all records from the Sales table to verify the data load

```
SELECT * FROM Sales;
```

FEATURE ENGINEERING

Feature Engineering : This will help use generate some new columns from existing ones.

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.

```
SELECT
time,
(CASE
        WHEN `time` BETWEEN "00:00:00" AND "12:00:00" THEN "Morning"
      WHEN `time` BETWEEN "12:01:00" AND "16:00:00" THEN "Afternoon"
      ELSE "Evening"
   END) AS time_of_day
FROM sales;
```

```
        ALTER TABLE Sales ADD COLUMN Time_of_day varchar(20);
-- For this to work turn off safe mode for update
-- Edit > Preferences > SQL Edito > scroll down and toggle safe mode
-- Reconnect to MySQL: Query > Reconnect to server

UPDATE sales
SET time_of_day = (CASE
                WHEN `time` BETWEEN "00:00:00" AND "12:00:00" THEN "Morning"
      WHEN `time` BETWEEN "12:01:00" AND "16:00:00" THEN "Afternoon"
      ELSE "Evening"
   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.

```
SELECT
    date,
    DAYNAME(date)
from sales;
SELECT * FROM Sales;

ALTER TABLE Sales ADD COLUMN day_name varchar(12);

UPDATE Sales
SET day_name = DAYNAME(date);
```

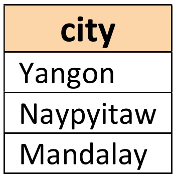
3. Add a new column named **`month_name`** that contains the extracted months of the year on which the given transaction took place (Jan, Feb, Mar). Help determine which month of the year has the most sales and profit.

```
SELECT
    date,
    MONTHNAME(date)
from sales;

ALTER TABLE Sales ADD COLUMN month_name varchar(12);

UPDATE Sales
SET month_name = MONTHNAME(date);

SELECT * FROM Sales;
```

GENERIC QUESTIONS

1. How many unique cities does the data have?

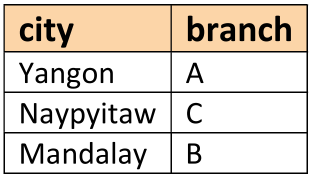
```
SELECT
      DISTINCT city
from sales;
```



city
Yangon
Naypyitaw
Mandalay

2. In which city is each branch?

```
SELECT
           DISTINCT city,
    branch
FROM Sales;
```



city	branch
Yangon	A
Naypyitaw	C
Mandalay	B

PRODUCT ANALYSIS

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

```
SELECT
    COUNT(DISTINCT product_line)
FROM Sales;
```

total_pl
6

2. What is the most selling product line

```
SELECT
    SUM(quantity) AS qty,
    product_line
FROM Sales
GROUP BY product_line
ORDER BY qty DESC;
```

qty	product_line
1942	Electronic accessories
1904	Food and beverages
1840	Sports and travel
1822	Home and lifestyle
1804	Fashion accessories
1708	Health and beauty

Conclusions :- Electronic Accessories has most selling product whereas Health and Beauty has less selling.

3. What is the total revenue by month

```
SELECT
    sum(total) as total_revenue,
    month_name as month
FROM Sales
GROUP BY month
ORDER BY total_revenue DESC;
```

total_revenue	month
116291.9	January
109455.5	March
97219.37	February

Conclusions :- January month generate highest revenue whereas February has lowest revenue.

4. What month had the largest COGS?

```
SELECT
        month_name AS month,
        SUM(cogs) AS cogs
FROM sales
GROUP BY month_name
ORDER BY cogs;
```

month	cogs
February	92589.88
March	104243.3
January	110754.2

Conclusions :- January month generate highest cogs whereas February has lowest cogs.

5. What product line had the largest revenue?

```
SELECT
        product_line,
        SUM(total) as total_revenue
FROM sales
GROUP BY product_line
ORDER BY total_revenue DESC;
```

product_line	total_revenue
Food and beverages	112289.7
Sports and travel	110245.7
Electronic accessories	108675.1
Fashion accessories	108611.8
Home and lifestyle	107723.8
Health and beauty	98387.48

Conclusions :- Food and beverages has highest revenue followed by Sports and Travel whereas Health and Beauty has lowest.

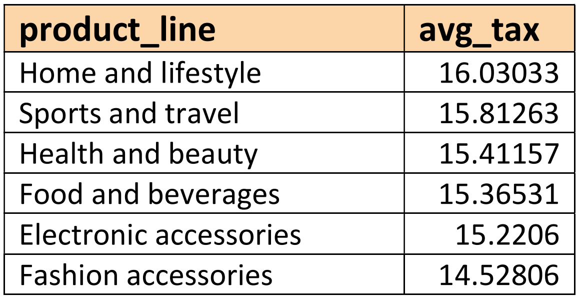
6. What is the city with the largest revenue?

```
SELECT
        branch,
        city,
        SUM(total) AS total_revenue
FROM sales
GROUP BY city, branch
ORDER BY total_revenue;
```

branch	city	total_revenue
B	Mandalay	212395.3
A	Yangon	212400.7
C	Naypyitaw	221137.4

7. What product line had the largest TAX?

```
SELECT
        product_line,
        AVG(tax) as avg_tax
FROM sales
GROUP BY product_line
ORDER BY avg_tax DESC;
```



product_line	avg_tax
Home and lifestyle	16.03033
Sports and travel	15.81263
Health and beauty	15.41157
Food and beverages	15.36531
Electronic accessories	15.2206
Fashion accessories	14.52806

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

```
SELECT
    AVG(quantity) as avg_qnty
FROM Sales;

SELECT
    product_line,
    case
        when AVG(quantity) > 5.5100 then "Good"
        else "Bad"
    end as remark
from Sales
group by product_line;
```

product_line	remark
Health and beauty	Good
Electronic accessories	Good
Home and lifestyle	Good
Sports and travel	Good
Food and beverages	Bad
Fashion accessories	Bad

9. Which branch sold more products than average product sold?

```
SELECT
   branch,
   SUM(quantity) AS qty
FROM Sales
GROUP BY branch
HAVING SUM(quantity) > (SELECT AVG(quantity) AS avg_quantity FROM Sales);
```

branch	qty
A	3718
C	3662
B	3640

10. What is the most common product line by gender

```
SELECT
        gender,
   product_line,
   COUNT(gender) AS total_cnt
FROM sales
GROUP BY gender, product_line
ORDER BY total_cnt DESC;
```

gender	product_line	total_cnt
Female	Fashion accessories	192

Female	Food and beverages	180
Male	Health and beauty	176
Female	Sports and travel	176
Male	Electronic accessories	172
Female	Electronic accessories	168
Male	Food and beverages	168
Male	Fashion accessories	164
Male	Home and lifestyle	162
Female	Home and lifestyle	158
Male	Sports and travel	156
Female	Health and beauty	128

11. What is the average rating of each product line

```
SELECT
        ROUND(AVG(rating), 2) as avg_rating,
   product_line
FROM sales
GROUP BY product_line
ORDER BY avg_rating DESC;
```

avg_rating	product_line
7.11	Food and beverages
7.03	Fashion accessories
7	Health and beauty
6.92	Electronic accessories
6.91	Sports and travel
6.84	Home and lifestyle

Conclusions :- Food and beverages has highest average Rating Home and Lifestyle has lowest.

CUSTOMER ANALYSIS

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

```
SELECT
        DISTINCT customer_type
FROM sales;
```

customer_type
Member
Normal

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

```
SELECT
    DISTINCT payment_method
FROM Sales;
```

payment_method
Ewallet
Cash
Credit card

3. Which customer type buys the most?

```
SELECT
        customer_type,
   COUNT(*)
FROM sales
GROUP BY customer_type;
```

customer_type	COUNT
Member	1002
Normal	998

Conclusions :- There is not a big difference, approximately same.

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

```
SELECT
        gender,
        COUNT(*) as gender_cnt
FROM sales
GROUP BY gender
ORDER BY gender_cnt DESC;
```

gender	gender_cnt
Female	1002
Male	998

Conclusions :- There is not a big difference, approximately same.

5. What is the gender distribution per branch?

```
SELECT
        gender,
        COUNT(*) as gender_cnt
FROM sales
WHERE branch = "C"
GROUP BY gender
ORDER BY gender_cnt DESC;
```

gender	gender_cnt
Female	356
Male	300

Conclusions :- Gender per branch is more or less the same hence, I don't think has an effect of the sales per branch and other factors.

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

```
SELECT
        time_of_day,
        AVG(rating) AS avg_rating
FROM sales
GROUP BY time_of_day
ORDER BY avg_rating DESC;
```

time_of_day	avg_rating
Afternoon	7.03103
Morning	6.96021
Evening	6.92616

Conclusion :- Looks like time of the day does not really affect the rating, its more or less the same rating each time of the day.

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

8.

```
SELECT
    time_of_day,
    branch,
    AVG(rating) AS avg_rating
FROM sales
WHERE branch IN ("A", "B", "C")
```

```
GROUP BY time_of_day, branch
ORDER BY avg_rating DESC;
```

time_of_day	branch	avg_rating
Afternoon	A	7.18889
Evening	C	7.11818
Afternoon	C	7.06667
Morning	A	7.00548
Morning	C	6.97458
Evening	A	6.89291
Morning	B	6.88983
Afternoon	B	6.836
Evening	B	6.7723

Conclusions:- Branch A and C are doing well in ratings, branch B needs to do a little more to get better ratings.

9. Which day fo the week has the best avg ratings?

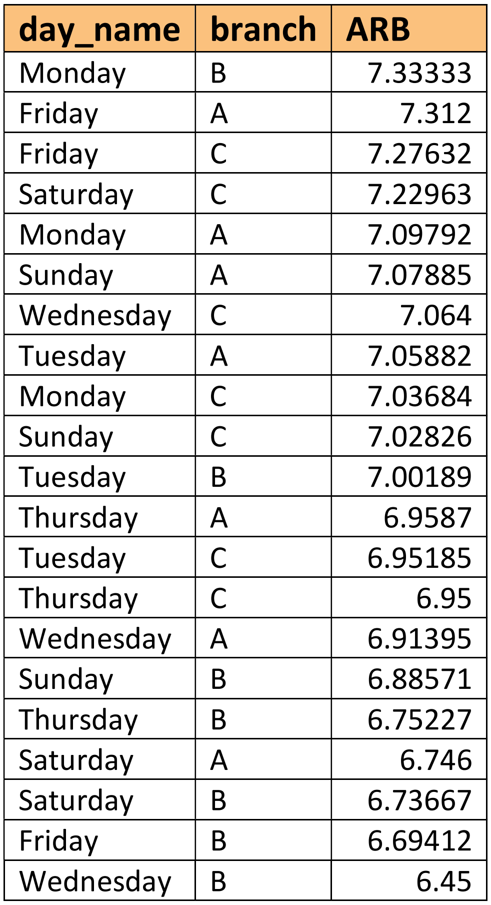
```
SELECT
        day_name,
        AVG(rating) AS avg_rating
FROM sales
GROUP BY day_name
ORDER BY avg_rating DESC;
```

day_name	avg_rating
Wednesday	6.8042
Thursday	6.88986
Saturday	6.90183
Tuesday	7.00316
Sunday	7.01053
Friday	7.07554
Monday	7.1528

Conclusions :- Sunday, Mon, Tue and Friday are the top best days for good ratings

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

```
SELECT
        day_name,
   branch,
        Avg(rating) as ARB
FROM sales
WHERE branch in ("A","B","C")
GROUP BY day_name, branch
ORDER BY ARB DESC;
```



day_name	branch	ARB
Monday	B	7.33333
Friday	A	7.312
Friday	C	7.27632
Saturday	C	7.22963
Monday	A	7.09792
Sunday	A	7.07885
Wednesday	C	7.064
Tuesday	A	7.05882
Monday	C	7.03684
Sunday	C	7.02826
Tuesday	B	7.00189
Thursday	A	6.9587
Tuesday	C	6.95185
Thursday	C	6.95
Wednesday	A	6.91395
Sunday	B	6.88571
Thursday	B	6.75227
Saturday	A	6.746
Saturday	B	6.73667
Friday	B	6.69412
Wednesday	B	6.45

Conclusions :- In Branch B Monday has highest Rating and Wednesday has Lowest Rating.
In Branch A Friday has highest Rating and Saturday has Lowest Rating.
In Branch C Friday has highest Rating and Thursday has Lowest Rating.

SALES ANALYSIS

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

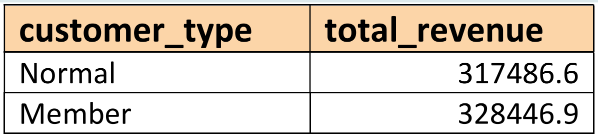
```
SELECT
        time_of_day,
        COUNT(*) AS total_sales
FROM sales
WHERE day_name = "Sunday"
GROUP BY time_of_day
ORDER BY total_sales DESC;
```

time_of_day	total_sales
Evening	58
Afternoon	53
Morning	22

Conclusions :- Evenings experience most sales, the stores are filled during the evening hours, followed by Afternoon and morning has less sales.

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

```
SELECT
        customer_type,
        SUM(total) AS total_revenue
FROM sales
GROUP BY customer_type
ORDER BY total_revenue;
```

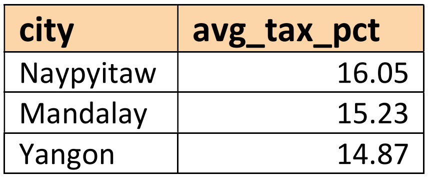


customer_type	total_revenue
Normal	317486.6
Member	328446.9

Conclusions :- Member type generated more revenue as compare to Normal type.

3. Which city has the largest tax/VAT percent?

```
SELECT
        city,
          ROUND(AVG(tax), 2) AS avg_tax_pct
FROM sales
GROUP BY city
ORDER BY avg_tax_pct DESC;
```



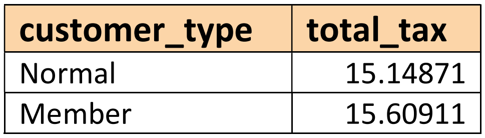
city	avg_tax_pct
Naypyitaw	16.05
Mandalay	15.23
Yangon	14.87

Conclusions :- Naypyitaw city has the largest tax % that is 16.05 and followed by Mandalay(15.23).

4. Which customer type pays the most in VAT?

```
SELECT
        customer_type,
        AVG(tax) AS total_tax
FROM sales
```

```
GROUP BY customer_type
ORDER BY total_tax;
```



customer_type	total_tax
Normal	15.14871
Member	15.60911

Conclusions :- Member pays more tax as compare to Normal.