

# Walmart SQL Project Vivek Chauhan

```
#we are working with csv files, here are the steps to import csv files in sql  
# 1) create a new database in mysql  
# 2) Go to import section & select the file you want to upload  
# 3) Turn on this setting at the bottom part "The first line of the file contains the table column names (if this is unchecked,  
the first line will become part of the data)"
```

## **About :-**

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

## **Purposes Of The Project :-**

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

## **About Data :-**

The dataset was obtained from the Kaggle Walmart Sales Forecasting Competition. This dataset contains sales transactions from three different branches of Walmart, respectively located in Mandalay, Yangon and Naypyitaw. The data contains 17 columns and 1000 rows:

## **Analysis List :-**

### **1) Product Analysis**

Conduct analysis on the data to understand the different product lines, the product 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 products. The result of this can help 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 customer segments, purchase trends and the profitability of each customer segment.

## **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 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

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.

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.

**3) Exploratory Data Analysis (EDA):** Exploratory data analysis is done to answer the listed questions and aims of this project.

#### **Business Questions To Answer :-**

##### **Generic Question :-**

How many unique cities does the data have?

In which city is each branch?

##### **Product :-**

How many unique product lines does the data have?

What is the most common payment method?

What is the most selling product line?

What is the total revenue by month?

What month had the largest COGS?

What product line had the largest revenue?

What is the city with the largest revenue?

What product line had the largest VAT?

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

Which branch sold more products than average product sold?

What is the most common product line by gender?

What is the average rating of each product line?

##### **Sales :-**

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

Which of the customer types brings the most revenue?

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

Which customer type pays the most in VAT?

##### **Customer :-**

How many unique customer types does the data have?

How many unique payment methods does the data have?

What is the most common customer type?

Which customer type buys the most?

What is the gender of most of the customers?

What is the gender distribution per branch?

Which time of the day do customers give most ratings?

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

Which day of the week has the best avg ratings?

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

### **Revenue And Profit Calculations :-**

\$ COGS = unitsPrice \* quantity \$

\$ VAT = 5% \* COGS \$

V

A

T

is added to the

C

O

G

S

and this is what is billed to the customer.

\$ total(gross\_sales) = VAT + COGS \$

\$ grossProfit(grossIncome) = total(gross\_sales) - COGS \$

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

\$ \text{Gross Margin} = \frac{\text{gross income}}{\text{total revenue}} \$

**Example with the first row in our DB:**

**Data given:**

\$ \text{Unite Price} = 45.79 \$

\$ \text{Quantity} = 7 \$

\$ COGS = 45.79 \* 7 = 320.53 \$

\$ \text{VAT} = 5\% \* COGS = 5\% 320.53 = 16.0265 \$

\$ total = VAT + COGS = 16.0265 + 320.53 =

336.5565

\$ \text{Gross Margin Percentage} = \frac{\text{gross income}}{\text{total revenue}} = \frac{16.0265}{336.5565} = 0.047619 \approx 4.7619\% \$

**→Project Starts from here :-**

## Import the .csv format file in phpMyAdmin then after your project Query is start.

The screenshot shows the phpMyAdmin interface with the following details:

- Server:** MySQL-3306
- Database:** wmvivek
- Table:** `1\_walmartsalesdata\_csv`
- Current selection:** `1\_walmartsalesdata\_csv`
- Query:** `SELECT \* FROM `1\_walmartsalesdata\_csv`;
- Rows:** 24 (1000 total)
- Time taken:** 0.0007 seconds
- Extra options:** Grid edit, checkbox, Edit, Copy and Delete features are not available.
- Buttons:** Profiling, Edit inline, Explain SQL, Create PHP code, Refresh.
- Table Headers:** Invoice ID, Branch, City, Customer type, Gender, Product line, Unit price, Quantity, Tax %, Total, Date, Time, Payment, cogs, gross margin percentage, gross income, Rating.
- Table Data:** A large table containing 24 rows of sales data from various branches (A, B, C) across different cities (Yangon, Naypyitaw, Mandalay) and product categories (Health and beauty, Electronic accessories, Sports and travel, Food and beverages). The data includes details like unit price, quantity, tax percentage, total amount, payment method, and rating.
- Bottom Status Bar:** Shows the number of rows (25), search bar, and system status (10:32, 10-05-2023).

localhost / MySQL | phpMyAdmin

Show query box

MySQL returned an empty result set (i.e. zero rows) (Query took 0.0022 seconds)

```
CREATE DATABASE vivek_walmart;
```

[Edit inline] [Edit] [Create PHP code]

Server: MySQL 3.36.0 Databases SQL Status User accounts Export Import Settings Variables Charsets Engines Plugins

New testsql thisisand thusday try tuesday userssweb userv vivayak vivaan vivek vivek\_coding vivek\_food vivek\_garment vivek\_hos vivek\_hospmanagement vivek\_library vivek\_restaurant vivek\_salesdata vivek\_salesman vivek\_schoolmanagement vivek\_softdrink vivek\_sportclub vivek\_teachers vivek\_vehicle vivek\_walmart walmarthouseproject watches wednesday windowswap wm Consola

Type here to search

localhost / MySQL / vivek\_walmart

localhost/phpmyadmin/index.php?route=/database/sql&db=vivek\_walmart

Show query box

MySQL returned an empty result set (i.e. zero rows) (Query took 0.0001 seconds)

```
USE vivek_walmart;
```

[Edit inline] [Edit] [Create PHP code]

Server: MySQL 3.36.0 Database: vivek\_walmart Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Designer

Show query box

MySQL returned an empty result set (i.e. zero rows) (Query took 0.0001 seconds)

vivek vivek\_coding vivek\_food vivek\_garment vivek\_hos vivek\_hospmanagement vivek\_library vivek\_restaurant vivek\_salesdata vivek\_salesman vivek\_schoolmanagement vivek\_softdrink vivek\_sportclub vivek\_teachers vivek\_vehicle vivek\_walmart walmarthouseproject watches wednesday windowswap wm wmmvivek New 1\_walmartsalesdata\_csv Columns yeah yeahschoolsystem yeahfaver yumstop zomatoesproject

Type here to search

localhost / MySQL / vivek\_walmart

localhost/phpmyadmin/index.php?route=/import

Show query box

Import has been successfully finished. 2 queries executed.

The following structures have either been created or altered. Here you can:

- View a structure's contents by clicking on its name.
- Change any of its settings by clicking the corresponding "Options" link.
- Edit structure by following the "Structure" link.

vivek\_walmart (Options) **walmart.csv (Structure) (Options)**

(walmart.csv)

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0059 seconds)

```
CREATE TABLE IF NOT EXISTS `vivek_walmart`(`Invoice ID` varchar(11), `Branch` varchar(1), `City` varchar(9), `Customer type` varchar(6), `Gender` varchar(6), `Product line` varchar(22), `Unit price` decimal(4,2), `Quantity` int(2), `Tax %` decimal(6,4), `Total` decimal(7,4), `Date` varchar(10), `Time` varchar(8), `Payment` varchar(11), `cogs` decimal(5,2), `gross margin percentage` decimal(10,9), `gross Income` decimal(6,4), `Rating` decimal(3,1)) DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
```

[Edit inline] [Edit] [Create PHP code]

Warning: #1681 Integer display width is deprecated and will be removed in a future release.

Warning: #3719 utf8 is currently an alias for the character set UTF8MB3, but will be an alias for UTF8MB4 in a future release. Please consider using UTF8MB4 in order to be unambiguous.

Warning: #3778 utf8mb3\_general\_ci is a collation of the deprecated character set UTF8MB3. Please consider using UTF8MB4 with an appropriate collation instead.

1000 rows inserted (Query took 0.0110 seconds.)

```
INSERT INTO `vivek_walmart`(`walmart.csv`)(`Invoice ID`, `Branch`, `City`, `Customer type`, `Gender`, `Product line`, `Unit price`, `Quantity`, `Tax %`, `Total`, `Date`, `Time`, `Payment`, `cogs`, `gross margin percentage`, `gross Income`, `Rating`) VALUES('750-67-4828', 'A', 'Yangon', 'Member', 'Female', 'Health and beauty', 74.69, 7, 26.1415, 548.9715, '2019-01-05', '13:08:00', 'Ewallet', 522.63, 4.761984762, 26.1415, 9.1), ('226-31-3081', 'C', 'Navyataw', 'Normal', 'Female', 'Electronics accessories', 15.28, 5, 3.82, 80.22, '2019-03-08', '10:29:00', 'Cash', 76.4, 4.761984762, 3.82, 9.6), ('631-41-3108', 'A', 'Yangon', 'Normal', 'Male', 'Home and lifestyle', 46.33, 7, 16.2155, 340.5255, '2019-01-01', '11:23:00', 'Credit card', 124.31, 4.761984762, 16.2155, 7.4), ('123-19-1176', 'A', 'Yangon', 'Member', 'Male', 'Health and beauty', 58.22, 8, 23.288, 489.048, '2019-01-27', '2033:00', 'Ewallet', 465.76, 4.761984762, 23.288, 8.4), ('373-73-7910', 'A', 'Yangon', 'Normal', 'Male', [...] )
```

[Edit]

Warning: #1264 Out of range value for column 'Total' at row 168

Warning: #1264 Out of range value for column 'Total' at row 168

Warning: #1264 Out of range value for column 'Total' at row 168

Console

**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.**

→With this we get the data classified into time of day

→Create the table first to apply the query.

→Your Query is :-

```
ALTER TABLE walmart_csv ADD COLUMN time_of_day varchar(20);
```

→Second Step is :-

```
UPDATE walmart_csv 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 );
```

→Third Step is :-

```
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 walmart.csv;
```

Time	time_of_day
13:08:00	Afternoon
10:29:00	Morning
13:23:00	Afternoon
20:33:00	Evening
10:37:00	Morning
18:30:00	Evening
14:36:00	Afternoon
11:38:00	Morning
17:15:00	Evening
13:27:00	Afternoon
18:07:00	Evening
17:03:00	Evening
10:25:00	Morning
16:48:00	Evening
19:21:00	Evening
16:19:00	Evening
11:03:00	Morning
10:39:00	Morning
18:00:00	Evening
15:30:00	Afternoon
11:24:00	Morning
10:40:00	Morning
12:20:00	Afternoon
11:15:00	Morning
17:36:00	Evening

**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.**

from this we had bring the name of the days from the date column

Showing rows 0 - 24 (1000 total, Query took 0.0014 seconds.)

SELECT Date, DAYNAME(Date) FROM walmart\_csv;

Date	DAYNAME(Date)
2019-01-05	Saturday
2019-03-08	Friday
2019-03-09	Sunday
2019-01-27	Sunday
2019-02-08	Friday
2019-03-25	Monday
2019-02-25	Monday
2019-02-24	Sunday
2019-01-10	Thursday
2019-02-20	Wednesday
2019-02-06	Wednesday
2019-03-09	Saturday
2019-02-12	Tuesday
2019-02-07	Thursday
2019-03-29	Friday
2019-01-15	Tuesday
2019-03-11	Monday
2019-01-01	Tuesday
2019-01-21	Monday
2019-03-11	Monday
2019-02-25	Monday
2019-03-05	Tuesday
2019-03-15	Friday
2019-02-17	Sunday
2019-03-02	Saturday

now we want to add new column in our dataset named day\_name

→ Your Query is :-

ALTER TABLE walmart\_csv ADD COLUMN day\_name varchar(10);

this query will update the column in the data set

→ Second Query is :-

UPDATE walmart\_csv SET day\_name = DAYNAME(Date);

if you check a column had been added at last named day\_name with all the data inserted

1000 rows affected (Query took 0.0131 seconds.)

UPDATE walmart\_csv 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.**

**now we need to add another new column named month\_name in our dataset**

→Your Query is :-

```
ALTER TABLE walmart_csv ADD COLUMN month_name varchar(10);
```

The screenshot shows the phpMyAdmin interface. In the left sidebar, the database structure is visible, including a table named 'walmart\_csv'. In the main area, a query has been run:

```
ALTER TABLE walmart_csv ADD COLUMN month_name varchar(10);
```

The status bar at the bottom right indicates the date and time: 10-05-2025 11:00.

this will add a new column in our dataset, now we need to add data into the column

now this will add data in our dataset

→Second Query is :-

```
UPDATE walmart_csv SET month_name = MONTHNAME(Date);
```

The screenshot shows the phpMyAdmin interface on a Windows desktop. A query was run: `UPDATE walmart_csv SET month_name = MONTHNAME(Date);`. The result message indicates "1000 rows affected. (Query took 0.0132 seconds.)". The database structure on the left includes a tree view of databases like vivek, vivek\_walmart, and tables such as walmart\_csv, vivaan, vivek\_food, etc.

#### 4) How many unique cities does the data have?

→ Your Query is :-

```
SELECT DISTINCT City from walmart_csv;
```

we have total 3 different cities in our dataset

The screenshot shows the phpMyAdmin interface displaying the results of the query `SELECT DISTINCT City from walmart_csv;`. The results show three distinct cities: Yangon, Naypyitaw, and Mandalay. The interface includes a sidebar with a tree view of databases and tables, and a bottom navigation bar with various icons.

City
Yangon
Naypyitaw
Mandalay

#### 5) In which city is each brand ?

→ Your Query is :-

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

localhost / MySQL / vivek\_walmart > Database: vivek\_walmart > Table: walmart\_csv

**Show query box**

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 2 (3 total). Query took 0.00078 seconds.

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

**Extra options**

City	Branch
Yangon	A
Naypyitaw	C
Mandalay	B

**Query results operations**

[Print](#) [Copy to clipboard](#) [Export](#) [Display chart](#) [Create view](#)

## 6) How many unique product lines does the data have ?

→ Your Query is :-

ALTER TABLE walmart\_csv RENAME COLUMN `Product line` TO Product\_line;

localhost / MySQL / vivek\_walmart > ALTER (RENAME) in SQL | Gee... > Rename Column MySQL

**Show query box**

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0123 seconds.)

```
ALTER TABLE walmart_csv RENAME COLUMN `Product line` TO Product_line;
```

**Operations**

→ Second Query is :-

SELECT COUNT(DISTINCT Product\_line) from walmart\_csv;

the business have total 6 unique product lines

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. A query has been run: `SELECT COUNT(DISTINCT Product_line) from walmart_csv;`. The result is displayed in the 'Query results operations' section, showing a single row with the value '6'.

## 7) What is the most common payment method ?

as we can see that e wallet & cash are the most common payment method

→Your Query is :-

```
SELECT Payment,COUNT(Payment) FROM walmart_csv GROUP BY Payment;
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. A query has been run: `SELECT Payment,COUNT(Payment) FROM walmart_csv GROUP BY Payment;`. The result is displayed in the 'Query results operations' section, showing a table with three rows:

Payment	COUNT(Payment)
Ewallet	345
Cash	344
Credit card	311

## 8) What is the most selling product line ?

As we can see that Fashion accessories & Food beverages are the most selling product line

→Your Query is :-

```
SELECT Product_line,COUNT(Product_line) as product FROM walmart_csv GROUP BY Product_line ORDER BY product DESC;
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. The query executed was:

```
SELECT Product_line,COUNT(Product_line) as product FROM walmart_csv GROUP BY Product_line ORDER BY product DESC;
```

The results are displayed in a table:

Product_line	product
Fashion accessories	178
Food and beverages	174
Electronic accessories	170
Sports and travel	166
Home and lifestyle	160
Health and beauty	152

## 9) What is the total revenue by the month ?

as we can see that janauary had the highest revenue

→Your Query is :-

```
SELECT month_name AS month,SUM(Total) AS total_revenue FROM walmart_csv GROUP BY month_name ORDER BY total_revenue DESC;
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. The query executed was:

```
SELECT month_name AS month,SUM(Total) AS total_revenue FROM walmart_csv GROUP BY month_name ORDER BY total_revenue DESC;
```

The results are displayed in a table:

month	total_revenue
January	116205.5525
March	105433.0169
February	97116.7287

## 10) What month had the largest COGS ?

As you can see that january month had the highest cogs

→Your Query is :-

```
SELECT month_name AS month,SUM(cogs) AS COGS FROM walmart_csv GROUP BY month_name ORDER BY cogs DESC;
```

The screenshot shows the phpMyAdmin interface with the database 'vivek\_walmart' selected. In the left sidebar, the 'Tables' section lists 'walmart\_csv'. The main query results table displays the following data:

month	COGS
January	110754.16
March	104243.34
February	92599.88

## 11) What product line had the largest revenue ?

As from the data we can say that food & beverages had the highest revenue.

→Your Query is :-

```
SELECT Product_line,SUM(Total) AS total_revenue from walmart_csv GROUP BY Product_line ORDER BY total_revenue;
```

The screenshot shows the phpMyAdmin interface with the database 'vivek\_walmart' selected. In the left sidebar, the 'Tables' section lists 'walmart\_csv'. The main query results table displays the following data:

Product_line	total_revenue
Health and beauty	49193.7390
Home and lifestyle	53789.6976
Fashion accessories	54203.2497
Electronic accessories	54337.5315
Sports and travel	55120.7064
Food and beverages	56110.3839

## 12) What is the city with the largest revenue ?

As we can see that city Naypyitaw had the largest revenue.

→ Your Query is :-

```
SELECT City,SUM(Total) AS total_revenue from walmart_csv GROUP BY City ORDER BY total_revenue DESC;
```

City	total_revenue
Naypyitaw	110441.4309
Yangon	106161.0804
Mandalay	106152.7968

## 13) What product lines had the largest VAT (TAX) ?

As we can see that food & beverages had the largest tax

→ Your Query is :-

```
ALTER TABLE walmart_csv RENAME COLUMN `Tax 5%` TO Tax;
```

```
ALTER TABLE walmart_csv RENAME COLUMN `Tax 5%` TO Tax;
```

→ Second Query is :-

```
SELECT Product_line,SUM(Tax) AS total_tax from walmart_csv GROUP BY Product_line ORDER BY total_tax DESC;
```

Product_line	total_tax
Food and beverages	2673.5640
Sports and travel	2624.8965
Electronic accessories	2587.5015
Fashion accessories	2585.9950
Home and lifestyle	2564.8530
Health and beauty	2342.5590

#### 14) Which branch sold more products than average product sold ?

As we can see that branch A had sold more products than average product sold

→ Your Query is :-

```
SELECT Branch,SUM(Quantity) AS total_quantity FROM walmart_csv GROUP BY Branch HAVING SUM(Quantity)>(SELECT AVG(Quantity) FROM walmart_csv);
```

Branch	total_quantity
A	1859
C	1831
B	1820

#### 15) What is the most common product line by gender ?

As we can see that Gender female & product line fashion accessories have the highest count

→Your Query is :-

```
SELECT Gender,Product_line,COUNT(Gender) AS total_gen FROM walmart_csv GROUP BY Gender,Product_line  
ORDER BY total_gen DESC;
```

Gender	Product_line	total_gen
Female	Fashion accessories	96
Female	Food and beverages	90
Male	Health and beauty	88
Female	Sports and travel	88
Male	Electronic accessories	86
Female	Electronic accessories	84
Male	Food and beverages	84
Male	Fashion accessories	82
Male	Home and lifestyle	81
Female	Home and lifestyle	79
Male	Sports and travel	78
Female	Health and beauty	64

## 16) What is the average rating of each product line ?

→Your Query is :-

```
SELECT AVG(Rating) AS avg_rating,Product_line FROM walmart_csv GROUP BY Product_line ORDER BY avg_rating  
DESC;
```

avg_rating	Product_line
7.11322	Food and beverages
7.02921	Fashion accessories
7.00329	Health and beauty
6.92471	Electronic accessories
6.91627	Sports and travel
6.83750	Home and lifestyle

## 17) Number of sales made in each time of the day per weekday ?

**we had counted for monday, you can change the day as per your requirement**

→Your Query is :-

```
SELECT time_of_day,COUNT(*) AS total_sales FROM walmart_csv WHERE day_name = "Monday"
```

The screenshot shows the phpMyAdmin interface connected to a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. A SQL query has been run:

```
SELECT time_of_day,COUNT(*) AS total_sales FROM walmart_csv WHERE day_name = "Monday";
```

The results show a single row:

time_of_day	total_sales
Evening	125

Below the results, there are options to Print, Copy to clipboard, Export, Display chart, and Create view.

## 18) Which of the customer types brings the most revenue ?

As we can see that customer type member brings the most revenue

→Your Query is :-

```
ALTER TABLE walmart_csv RENAME COLUMN `Customer type` TO Customer_type;
```

The screenshot shows the phpMyAdmin interface connected to a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. A SQL query has been run:

```
ALTER TABLE walmart_csv RENAME COLUMN `Customer type` TO Customer_type;
```

A message indicates that MySQL returned an empty result set (Query took 0.0160 seconds):

MySQL returned an empty result set (i.e. zero rows) (Query took 0.0160 seconds.)

Below the message, there are links for Edit inline, Edit, and Create PHP code.

→ Second Query is :-

```
SELECT Customer_type,SUM(Total) AS total_rev FROM walmart_csv GROUP BY Customer_type ORDER BY total_rev DESC;
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The left sidebar lists various databases and tables under the 'Current server: MySQL' section. The main panel displays the results of the following SQL query:

```
SELECT Customer_type,SUM(Total) AS total_rev FROM walmart_csv GROUP BY Customer_type ORDER BY total_rev DESC;
```

The results table shows two rows:

Customer_type	total_rev
Member	164125.6287
Normal	158629.6794

Below the results, there are several operation buttons: Print, Copy to clipboard, Export, Display chart, and Create view.

19) Which city has the largest tax/vat percent ?

→ Your Query is :-

```
SELECT City,AVG(Tax) AS avg_tax FROM walmart_csv GROUP BY City ORDER BY avg_tax DESC;
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. The query executed was:

```
SELECT City, AVG(Tax) AS avg_tax FROM walmart_csv GROUP BY City ORDER BY avg_tax DESC;
```

The results are:

City	avg_tax
Naypyitaw	16.05236738
Mandalay	15.23202410
Yangon	14.87400147

## 20) Which customer type pays the most tax/vat ?

→ Your Query is :-

```
SELECT Customer_type, AVG(Tax) AS Tax FROM walmart_csv GROUP BY Customer_type ORDER BY Tax DESC;
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. The query executed was:

```
SELECT Customer_type, AVG(Tax) AS Tax FROM walmart_csv GROUP BY Customer_type ORDER BY Tax DESC;
```

The results are:

Customer_type	Tax
Member	15.60910978
Normal	15.14870741

## 21) How many unique customer types does the data have?

→ Your Query is :-

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

The screenshot shows the phpMyAdmin interface connected to a MySQL server. The database selected is 'vivek\_walmart' and the table is 'walmart\_csv'. A query has been run:

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

The result of the query is displayed in the 'Query results operations' section:

COUNT(DISTINCT(Customer_type))
2

## 22) How many unique payment methods does the data have?

→ Your Query is :-

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

The screenshot shows the phpMyAdmin interface connected to a MySQL server. The database selected is 'vivek\_walmart' and the table is 'walmart\_csv'. A query has been run:

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

The result of the query is displayed in the 'Query results operations' section:

COUNT(DISTINCT(Payment))
3

## 23) What is the most common customer type?

→ Your Query is :-

```
SELECT Customer_type,COUNT(*) AS Customer_count FROM walmart_csv GROUP BY Customer_type;
```

The screenshot shows the phpMyAdmin interface for a database named 'vivek\_walmart'. The left sidebar lists various databases and tables. The main area displays the results of a SQL query:

```
SELECT Customer_type,COUNT(*) AS Customer_count FROM walmart_csv GROUP BY Customer_type;
```

The results table shows two rows:

Customer_type	Customer_count
Member	501
Normal	499

## 24) What is the gender of most of the customers?

→ Your Query is :-

```
SELECT Gender,COUNT(*) AS gender_count FROM walmart_csv GROUP BY Gender ORDER BY gender_count DESC;
```

The screenshot shows the phpMyAdmin interface for a database named 'vivek\_walmart'. The left sidebar lists various databases and tables. The main area displays the results of a SQL query:

```
SELECT Gender,COUNT(*) AS gender_count FROM walmart_csv GROUP BY Gender ORDER BY gender_count DESC;
```

The results table shows two rows:

Gender	gender_count
Female	501
Male	499

## 25) What is the gender distribution per branch?

→ Your Query is :-

```
SELECT Gender,COUNT(*) AS gender_count FROM walmart_csv WHERE Branch = "C";
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. A query has been run:

```
SELECT Gender,COUNT(*) AS gender_count FROM walmart_csv WHERE Branch = "C";
```

The results show a single row:

Gender	gender_count
Female	328

Below the results, there are options to Print, Copy to clipboard, Export, Display chart, and Create view.

## 26) Which time of the day do customers give most ratings?

→ Your Query is :-

```
SELECT Gender,COUNT(*) AS gender_count FROM walmart_csv GROUP BY Gender ORDER BY gender_count DESC;
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. A query has been run:

```
SELECT Gender,COUNT(*) AS gender_count FROM walmart_csv GROUP BY Gender ORDER BY gender_count DESC;
```

The results show two rows:

Gender	gender_count
Female	501
Male	499

Below the results, there are options to Show all, Number of rows (set to 25), Filter rows, and Search this table.

## 27) Which time of the day do customers give most ratings per branch?

→ Your Query is :-

```
SELECT time_of_day,SUM(Rating) AS total_rating FROM walmart_csv GROUP BY time_of_day ORDER BY total_rating;
```

Showing rows 0 - 2 (3 total). Query took 0.0072 seconds.

```
SELECT time_of_day, SUM(Rating) AS total_rating FROM walmart_csv GROUP BY time_of_day ORDER BY total_rating;
```

time_of_day	total_rating
Morning	1329.5
Afternoon	2650.8
Evening	2992.4

→ Second Query is :-

`SELECT time_of_day, SUM(Rating) AS total_rating from walmart_csv WHERE Branch = "C" GROUP BY time_of_day ORDER BY total_rating;`

Showing rows 0 - 2 (3 total). Query took 0.0073 seconds.

```
SELECT time_of_day, SUM(Rating) AS total_rating from walmart_csv WHERE Branch = "C" GROUP BY time_of_day ORDER BY total_rating;
```

time_of_day	total_rating
Morning	411.5
Afternoon	890.4
Evening	1018.0

## 28) Which day of the week has the best avg ratings?

→ Your Query is :-

`SELECT day_name, AVG(Rating) as avg_rating from walmart_csv GROUP BY day_name ORDER BY avg_rating DESC;`

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. The query executed was:

```
SELECT day_name, AVG(Rating) AS avg_rating from walmart_csv GROUP BY day_name ORDER BY avg_rating DESC;
```

The results are:

day_name	avg_rating
Monday	7.15360
Friday	7.07626
Sunday	7.01128
Tuesday	7.00316
Saturday	6.90183
Thursday	6.88986
Wednesday	6.80559

→ Second Query is :-

SELECT day\_name, AVG(Rating) AS avg\_rating from walmart\_csv WHERE Branch = "C" GROUP BY day\_name ORDER BY avg\_rating DESC;

The screenshot shows the phpMyAdmin interface for a MySQL database named 'vivek\_walmart'. The current table is 'walmart\_csv'. The query executed was:

```
SELECT day_name, AVG(Rating) AS avg_rating from walmart_csv WHERE Branch = "C" GROUP BY day_name ORDER BY avg_rating DESC;
```

The results are:

day_name	avg_rating
Friday	7.27895
Saturday	7.22963
Wednesday	7.06400
Monday	7.03684
Sunday	7.02826
Tuesday	6.95185
Thursday	6.95000