**PROJECT- SQL- WALMART SALES DATA**

CREATE DATABASE IF NOT EXISTS salesDataWalmart;

CREATE TABLE IF NOT EXISTS 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(30) NOT NULL,

product\_line VARCHAR(100) NOT NULL,

unit\_price DECIMAL(10,2) NOT NULL,

quantity INT NOT NULL,

tax\_pct FLOAT(6,4) 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(11,9),

gross\_income DECIMAL(12, 4),

rating FLOAT(2, 1)

);

SELECT \* FROM salesDataWalmart.sales;

---------------------------------------------------------------------------------------------

-- ------------------------------Feature Engineering------------------------------------------

-- time\_of\_day

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\_date

FROM sales;

ALTER TABLE sales ADD COLUMN time\_of\_day VARCHAR(20);

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

);

-- day\_name

SELECT

date,

DAYNAME(date) AS day\_name

FROM sales;

ALTER TABLE sales ADD COLUMN day\_name VARCHAR(10);

UPDATE sales

SET day\_name = DAYNAME(date);

-- month\_name

SELECT

date,

MONTHNAME(date)

FROM sales;

ALTER TABLE sales ADD COLUMN month\_name VARCHAR(10);

UPDATE sales

SET month\_name = MONTHNAME(date);

---------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------

-- ----------------------------------Generic--------------------------------------------------

-- How many unique cities does the data have?

SELECT

DISTINCT city

FROM sales;

-- There are 3 unique cities in the data.

-- How many unique branches does the data have?

SELECT

DISTINCT branch

FROM sales;

-- There are 3 unique branches in the data.

-- In which city is each branch?

SELECT

DISTINCT city,

branch

FROM sales;

---------------------------------------------------------------------------------------------

-- --------------------------------------Product---------------------------------------------

-- How many unique product lines does the data have?

SELECT

DISTINCT product\_line

FROM sales;

SELECT

COUNT(DISTINCT product\_line)

FROM sales;

-- There are 6 unique product lines.

-- What is the most common payment method?

SELECT payment, COUNT(payment) AS count

FROM sales

GROUP BY payment

ORDER BY count DESC ;

-- Cash is the most common payment method.

-- What is the most selling product line?

SELECT product\_line, COUNT(product\_line) AS PL\_COUNT

FROM sales

GROUP BY product\_line

ORDER BY PL\_COUNT DESC ;

-- Fashion accessories is the most selling product line.

-- What is the total revenue by month?

SELECT

month\_name AS month,

SUM(total) AS total\_revenue

FROM sales

GROUP BY month\_name

ORDER BY total\_revenue DESC;

-- January has the highest revenue produced followed by March and February.

-- What month had the largest COGS?

SELECT

month\_name AS month,

SUM(cogs) AS COGS

FROM sales

GROUP BY month\_name

ORDER BY COGS DESC;

-- January has the highest COGS followed by March and February.

-- 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;

-- Food and beverages have the largest revenue whereas health and beauty have the lowest revenue.

-- What is the city with the largest revenue?

SELECT

city,

branch,

SUM(total) AS total\_revenue

FROM sales

GROUP BY city, branch

ORDER BY total\_revenue DESC;

-- Naypyitaw has the largest revenue.

-- What product line had the largest VAT?

SELECT

product\_line,

AVG(tax\_pct) AS avg\_tax

FROM sales

GROUP BY product\_line

ORDER BY avg\_tax DESC;

-- Home and lifestyle have the largest average VAT whereas fashion accessories have the lowest average 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?

SELECT

AVG(quantity) as avg\_qty

FROM sales;

SELECT

product\_line,

CASE

WHEN AVG(quantity) > 6 THEN "GOOD"

ELSE "Bad"

END AS remark

FROM sales

GROUP BY product\_line;

-- 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) FROM sales);

-- A had sold more products than average products sold.

-- What is the most common product line by gender?

SELECT

gender,

product\_line,

count(gender) AS total\_count

FROM sales

GROUP BY gender, product\_line

ORDER BY total\_count DESC;

-- Fashion accessories are the most common among females while it is health and beauty among males.

-- 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;

-- The highest average rating is for food and beverages and lowest is for home and lifestyle.

-- Which product line has highest and lowest gross income?

SELECT

product\_line,

SUM(gross\_income) AS gross\_income

FROM sales

GROUP BY product\_line

ORDER BY gross\_income DESC;

-- Highest - Food and beverages, Lowest - Health and beauty

-- Which branch has highest and lowest gross income?

SELECT

branch,

SUM(gross\_income) AS gross\_income

FROM sales

GROUP BY branch

ORDER BY gross\_income DESC;

-- Highest - C, Lowest - B

-- Which city has highest and lowest gross income?

SELECT

city,

SUM(gross\_income) AS gross\_income

FROM sales

GROUP BY city

ORDER BY gross\_income DESC;

-- Highest - Naypitaw, Lowest - Mandalay

-- Which gender brings the highest gross income?

SELECT

gender,

SUM(gross\_income) AS gross\_income

FROM sales

GROUP BY gender

ORDER BY gross\_income DESC;

-- Female brings the highest gross income

---------------------------------------------------------------------------------------------

-- ------------------------------------Sales--------------------------------------------------

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

SELECT

time\_of\_day,

COUNT(\*) AS total\_sales

FROM sales

GROUP BY time\_of\_day

ORDER BY total\_sales DESC;

-- 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 DESC;

-- Member brings the most revenue.

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

SELECT

city,

AVG(tax\_pct) AS VAT

FROM sales

GROUP BY city

ORDER BY VAT DESC;

-- Naypyitaw has the largest average VAT.

-- Which customer type pays the most in VAT?

SELECT

customer\_type,

AVG(tax\_pct) AS VAT

FROM sales

GROUP BY customer\_type

ORDER BY VAT DESC;

-- Though the average VAT paid by both member and normal are almost same, member paid more.

-- ----------------------------------------------------------------------------------------------

------ -------------------------------Customer----------------------------------------------------

-- How many unique customer types does the data have?

SELECT

DISTINCT customer\_type

FROM sales;

SELECT

COUNT(DISTINCT customer\_type)

FROM sales;

-- How many unique payment methods does the data have?

SELECT

DISTINCT payment

FROM sales;

SELECT

COUNT(DISTINCT payment)

FROM sales;

-- What is the most common customer type?

SELECT

customer\_type,

count(customer\_type) AS cus\_count

FROM sales

GROUP BY customer\_type

ORDER BY cus\_count DESC;

# OR METHOD:

SELECT

customer\_type,

count(\*) as count

FROM sales

GROUP BY customer\_type

ORDER BY count DESC;

-- Which customer type buys the most?

SELECT

customer\_type,

count(\*) as count

FROM sales

GROUP BY customer\_type

ORDER BY count DESC;

-- What is the gender of most of the customers?

SELECT

gender,

count(\*) as gen\_count

FROM sales

GROUP BY gender

ORDER BY gen\_count DESC;

-- Male-498 and females-497

-- What is the gender distribution per branch?

SELECT

branch,

gender,

count(\*) as gen\_count

FROM sales

GROUP BY branch, gender

ORDER BY gen\_count DESC;

#OR METHOD:

SELECT

gender,

COUNT(\*) as gender\_cnt

FROM sales

WHERE branch = "C" #Change letter accordingly

GROUP BY gender

ORDER BY gender\_cnt DESC;

-- 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;

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

SELECT

branch,

time\_of\_day,

AVG(rating) AS avg\_rating

FROM sales

GROUP BY branch,time\_of\_day

ORDER BY avg\_rating DESC;

#OR METHOD

SELECT

time\_of\_day,

AVG(rating) AS avg\_rating

FROM sales

WHERE branch = "A" #Change letter accordingly

GROUP BY time\_of\_day

ORDER BY avg\_rating DESC;

-- Which day of 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;

-- Monday has the best average ratings whereas Wednesday has the least.

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

SELECT

day\_name,

AVG(rating) AS avg\_rating

FROM sales

WHERE branch ="B"

GROUP BY day\_name

ORDER BY avg\_rating DESC;

-- A - Friday, B - Monday, C- Saturday

-- Which customer type brings highest gross income?

SELECT

customer\_type,

SUM(gross\_income) AS gross\_income

FROM sales

GROUP BY customer\_type

ORDER BY gross\_income DESC;

-- Member brings the highest gross income.