-- Create database

CREATE DATABASE IF NOT EXISTS walmartSales;

-- Create table

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)

);

-- Data cleaning

SELECT

\*

FROM sales;

-- Add the time\_of\_day column

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

);

-- Add day\_name column

SELECT

date,

DAYNAME(date)

FROM sales;

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

UPDATE sales

SET day\_name = DAYNAME(date);

-- Add month\_name column

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;

-- 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;

-- What is the most selling product line

SELECT

SUM(quantity) as qty,

product\_line

FROM sales

GROUP BY product\_line

ORDER BY qty DESC;

-- What is the most selling product line

SELECT

SUM(quantity) as qty,

product\_line

FROM sales

GROUP BY product\_line

ORDER BY qty DESC;

-- 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;

-- What month had the largest COGS?

SELECT

month\_name AS month,

SUM(cogs) AS cogs

FROM sales

GROUP BY month\_name

ORDER BY cogs;

-- 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;

-- 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;

-- 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;

-- 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) > 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 qnty

FROM sales

GROUP BY branch

HAVING SUM(quantity) > (SELECT AVG(quantity) FROM sales);

-- 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;

-- 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;

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

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

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

-- -------------------------- Customers -------------------------------

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

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

SELECT

DISTINCT customer\_type

FROM sales;

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

SELECT

DISTINCT payment

FROM sales;

-- What is the most common customer type?

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(\*)

FROM sales

GROUP BY customer\_type;

-- 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;

-- 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 per branch is more or less the same hence, I don't think has

-- an effect of the sales per branch and other factors.

-- 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;

-- Looks like time of the day does not really affect the rating, its

-- more or less the same rating each time of the day.alter

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

SELECT

time\_of\_day,

AVG(rating) AS avg\_rating

FROM sales

WHERE branch = "A"

GROUP BY time\_of\_day

ORDER BY avg\_rating DESC;

-- Branch A and C are doing well in ratings, branch B needs to do a

-- little more to get better ratings.

-- 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;

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

-- why is that the case, how many sales are made on these days?

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

SELECT

day\_name,

COUNT(day\_name) total\_sales

FROM sales

WHERE branch = "C"

GROUP BY day\_name

ORDER BY total\_sales DESC;

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

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

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

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

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

-- 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;

-- Evenings experience most sales, the stores are

-- filled during the evening hours

-- 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;

-- Which city has the largest tax/VAT percent?

SELECT

city,

ROUND(AVG(tax\_pct), 2) AS avg\_tax\_pct

FROM sales

GROUP BY city

ORDER BY avg\_tax\_pct DESC;

-- Which customer type pays the most in VAT?

SELECT

customer\_type,

AVG(tax\_pct) AS total\_tax

FROM sales

GROUP BY customer\_type

ORDER BY total\_tax;

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

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