create database SalesDataWalmart;

USE SalesDataWalmart

GO

Create table WMsales (

invoice\_id varchar(100) Not null primary Key,

branch varchar (100) not null,

customer\_type varchar (100) not null,

gender varchar(100) not null,

product\_line varchar (100) not null,

unit\_price decimal(10,2) not null,

quantity int not null,

VAT float not null,

total decimal (12,4) not null,

date datetime not null,

time TIME not null,

payment\_method varchar (100) not null,

cogs decimal (10, 2) not null,

gross\_margin\_pct float,

gross\_income decimal(12, 4) not null,

rating float,

);

USE SalesDataWalmart

GO

Create table StoreName (

city varchar (30) not null,

branch varchar(100) not null primary key,

);

Select \*

From WMsales

Select \*

From StoreName

-- Add the foreign key constraint

ALTER TABLE WMsales

ADD CONSTRAINT Branch

FOREIGN KEY (Branch)

REFERENCES StoreName(Branch);

-- Add the column

ALTER TABLE WMsales

ADD timeofday VARCHAR(30);

-- Update the values

UPDATE WMsales

SET timeofday = (

CASE

WHEN [time] BETWEEN '00:00:00.0000000' AND '12:00:00.0000000' THEN 'Morning'

WHEN [time] BETWEEN '12:00:01.0000000' AND '17:59:59.0000000' THEN 'Afternoon'

ELSE 'Evening'

END

);

Select \*

From WMsales

-- Add day\_name column

ALTER TABLE WMsales

ADD day\_name VARCHAR(10);

-- Update day\_name column

UPDATE WMsales

SET day\_name = DATENAME(dw, date);

-- Add month\_name column

ALTER TABLE WMsales

ADD month\_name VARCHAR(15);

-- Update month\_name column

UPDATE WMsales

SET month\_name = FORMAT(date, 'MMMM');

-- Select all columns from wmsales

SELECT \*

FROM WMsales as w

INNER JOIN StoreName as s ON s.branch=w.branch;

----------------------------------- Generic Question -----------------------------------

-- 1. How many unique cities does the data have?

SELECT DISTINCT city

FROM wmsales as w

INNER JOIN StoreName as s ON w.branch=s.branch ;

-- 2. In which city is each branch?

SELECT DISTINCT s.city, w.branch

FROM wmsales as w

INNER JOIN StoreName as s ON w.branch=s.branch ;

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

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

SELECT COUNT(DISTINCT product\_line) as No\_Of\_Products

FROM wmsales;

-- 2. What is the most common payment method?

SELECT payment\_method, COUNT(payment\_method) AS cnt

FROM wmsales

GROUP BY payment\_method

ORDER BY cnt DESC;

-- 3. What is the most selling product line?

SELECT product\_line, COUNT(product\_line) AS cnt

FROM wmsales

GROUP BY product\_line

ORDER BY cnt DESC;

-- 4. What is the total revenue by month?

SELECT month\_name AS Month, SUM(total) AS Total\_Revenue

FROM wmsales

GROUP BY month\_name

ORDER BY Total\_Revenue DESC;

-- 5. What month had the largest COGS?

SELECT month\_name AS month, SUM(cogs) AS cogs

FROM wmsales

GROUP BY month\_name

ORDER BY cogs DESC;

-- 6. What product line had the largest revenue?

SELECT product\_line, SUM(total) AS Total\_revenue

FROM wmsales

GROUP BY product\_line

ORDER BY Total\_revenue DESC;

-- 7. What is the city with the largest revenue?

SELECT s.city, SUM(total) AS Total\_revenue

FROM wmsales as w

INNER JOIN StoreName as s ON w.branch=s.branch

GROUP BY city

ORDER BY Total\_revenue DESC;

-- 8. What product line had the largest VAT?

SELECT product\_line, SUM(VAT) AS Valueable\_Tax

FROM wmsales

GROUP BY product\_line

ORDER BY Valueable\_Tax DESC;

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

-- (Assuming you want to add a column named 'Sales\_Category' with values 'Good' or 'Bad')

SELECT product\_line,

CASE WHEN SUM(total) > (SELECT AVG(total) FROM wmsales) THEN 'Good' ELSE 'Bad' END AS Sales\_Category

FROM wmsales

GROUP BY product\_line;

-- 10. Which branch sold more products than average product sold?

SELECT branch, SUM(quantity) AS qty

FROM wmsales

GROUP BY branch

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

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

SELECT gender, product\_line, COUNT(gender) AS total\_count

FROM wmsales

GROUP BY gender, product\_line

ORDER BY total\_count DESC;

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

SELECT ROUND(AVG(rating), 2) AS avg\_rating, product\_line

FROM wmsales

GROUP BY product\_line

ORDER BY avg\_rating DESC;

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

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

SELECT timeofday, COUNT(\*) AS total\_sales

FROM wmsales

WHERE day\_name = 'Sunday'

GROUP BY timeofday

ORDER BY total\_sales DESC;

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

SELECT customer\_type, ROUND(SUM(total), 2) AS total\_revenue

FROM wmsales

GROUP BY customer\_type

ORDER BY total\_revenue;

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

SELECT s.city, AVG(VAT) AS value\_added\_tax

FROM wmsales as w

INNER JOIN StoreName as s ON s.branch=w.branch

GROUP BY city

ORDER BY value\_added\_tax DESC;

-- 4. Which customer type pays the most in VAT?

SELECT customer\_type, AVG(VAT) AS value\_added\_tax

FROM wmsales

GROUP BY customer\_type

ORDER BY value\_added\_tax DESC;

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

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

SELECT DISTINCT customer\_type

FROM wmsales;

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

SELECT DISTINCT payment\_method

FROM wmsales;

-- 3. What is the most common customer type?

SELECT customer\_type, COUNT(\*) AS total\_count

FROM wmsales

GROUP BY customer\_type

ORDER BY total\_count;

-- 4. Which customer type buys the most?

SELECT customer\_type, COUNT(\*) AS total\_count

FROM wmsales

GROUP BY customer\_type

ORDER BY total\_count;

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

SELECT gender, COUNT(\*) AS gender\_count

FROM wmsales

GROUP BY gender

ORDER BY gender\_count DESC;

-- 6. What is the gender distribution per branch?

SELECT gender, COUNT(\*) AS gender\_count

FROM wmsales

WHERE branch = 'C'

GROUP BY gender

ORDER BY gender\_count DESC;

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

SELECT timeofday, AVG(rating) AS avg\_rating

FROM wmsales

GROUP BY timeofday

ORDER BY avg\_rating DESC;

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

SELECT timeofday, branch, AVG(rating) AS avg\_rating

FROM wmsales

GROUP BY timeofday, branch

ORDER BY avg\_rating;

-- 9. Which day of the week has the best avg ratings?

SELECT day\_name, AVG(rating) AS avg\_rating

FROM wmsales

GROUP BY day\_name

ORDER BY avg\_rating DESC;

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

SELECT day\_name, AVG(rating) AS avg\_rating

FROM wmsales

WHERE branch = 'A'

GROUP BY day\_name

ORDER BY avg\_rating DESC;

----------------------------------- Revenue And Profit Calculation -----------------------------------

-- Total gross sales

SELECT SUM(VAT + cogs) AS total\_grass\_sales

FROM wmsales;

-- Gross profit

SELECT (SUM(VAT + cogs) - SUM(cogs)) AS gross\_profit

FROM wmsales;

-- Another way to calculate gross profit

SELECT (SUM(ROUND(VAT, 2) + COGS) - SUM(COGS)) AS gross\_profit

FROM wmsales;

delete from StoreName

delete from WMsales