SQL - Capstone Project

Purposes Of The Capstone Project

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

```
-- to avoid spaces in column names it cause issues while writing a sql queries

ALTER TABLE amazon

RENAME COLUMN `Invoice ID` TO Invoice_ID;

ALTER TABLE amazon

RENAME COLUMN `Customer type` TO CustomerType;

ALTER TABLE amazon

RENAME COLUMN `Unit price` TO UnitPrice;

ALTER TABLE amazon

RENAME COLUMN `Tax 5%` TO Tax_5_Percentage;

ALTER TABLE amazon

RENAME COLUMN `gross margin percentage` TO gross_margin_percentage;

ALTER TABLE amazon

RENAME COLUMN `gross income` TO GrossIncome;

ALTER TABLE amazon

RENAME COLUMN `Product line` TO ProductLine;


SET SQL_SAFE_UPDATES = 0;
```

-- 2.1 Add a new column named timeofday 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 timeofday column

ALTER TABLE amazon ADD COLUMN timeofday VARCHAR(10);

UPDATE amazon

SET timeofday  = CASE
```

```
    WHEN HOUR(time) BETWEEN 6 AND 11 THEN 'Morning'

    WHEN HOUR(time) BETWEEN 12 AND 17 THEN 'Afternoon'

    ELSE 'Evening'

END;
```

2.2 Add a new column named dayname 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 dayname column

alter table amazon add column dayname varchar(10);

update amazon

set dayname=dayname(date);
```

2.3 Add a new column named monthname 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.

```
--  Add monthname column

alter table amazon add column monthname varchar(10);

update amazon

set monthname=monthname(date);
```

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

Business Questions To Answer:

```
-- 1. count of distinct cities in the dataset

select count(distinct city) as Distinct_city_count from amazon;


-- 2.For each branch, what is the corresponding city?

select city,branch from amazon

order by city,branch;
```

```
limit 1;


-- 3. Count of distinct product lines
select distinct(ProductLine) from amazon;

-- 4.Most frequent payment method
select payment,count(*) as count
from amazon
group by payment
order by count desc
limit 1;

-- 5.Product line with the highest sales
select ProductLine,sum(Total) as total_sales
from amazon
group by ProductLine
order by total_sales desc
limit 1;

-- 6. How much revenue is generated each month?
select monthname,sum(total) from amazon
group by monthname;

--  7.Month with peak cost of goods sold (cogs)
select monthname,sum(cogs) as total_cogs
from amazon
group by monthname
order by total_cogs desc
limit 1;

-- 8.Product line with highest revenue
```

```
SELECT ProductLine, SUM(total) AS revenue
FROM amazon
GROUP BY ProductLine
ORDER BY revenue DESC
LIMIT 1;

-- 9.City with highest revenue
SELECT city, SUM(total) AS revenue
FROM amazon
GROUP BY city
ORDER BY revenue DESC
LIMIT 1;

-- 10. Product line with highest Value Added Tax (VAT)
SELECT ProductLine, SUM(Tax_5_Percentage) AS total_vat
FROM amazon
GROUP BY ProductLine
ORDER BY total_vat DESC
LIMIT 1;

-- 11.For each product line, add a column indicating "Good"
-- if its sales are above average, otherwise "Bad."
select ProductLine,
case
when sum(total)>(select avg(total) from amazon ) then 'Good'
else 'Bad'
end as Performance
from amazon
group by ProductLine;
```

```sql
-- 12.Identify the branch that exceeded the average number of products sold
select  Branch
from amazon
group by Branch
Having avg(Quantity )>(select avg(Quantity) from amazon);
```

```sql
-- 13. Which product line is most frequently associated with each gender?
SELECT gender, ProductLine, COUNT(*) AS freq
FROM amazon
GROUP BY gender, ProductLine
ORDER BY gender, freq DESC;
```

```sql
-- 14. Average rating for each product line
SELECT ProductLine, ROUND(AVG(rating), 2) AS avg_rating
FROM amazon
GROUP BY ProductLine;
```

```sql
-- 15.Sales occurrences by time of day and weekday
select timeofday,dayname,count(*) as sales_count
from amazon
group by timeofday,dayname
order by dayname,timeofday;
```

```sql
-- 16.Customer type contributing highest revenue
SELECT CustomerType, SUM(total) AS revenue
FROM amazon
GROUP BY CustomerType
ORDER BY revenue DESC
LIMIT 1;
```

```sql
-- 17.City with highest VAT %
SELECT city, SUM(Tax_5_Percentage) / SUM(cogs) * 100 AS vat_percent
FROM amazon
GROUP BY city
ORDER BY vat_percent DESC
LIMIT 1;

-- 18. Identify the customer type with the highest VAT payments.
SELECT CustomerType, SUM(Tax_5_Percentage)  AS total_vat
FROM amazon
GROUP BY CustomerType
ORDER BY total_vat DESC
LIMIT 1;
-- 19. Count of distinct customer types
SELECT COUNT(DISTINCT CustomerType) AS num_customer_types
FROM amazon;

-- 20. Count of distinct payment methods
SELECT COUNT(DISTINCT payment) AS num_payment_methods
FROM amazon;

-- 21.Most frequent customer type
select CustomerType,count(*) as count
from amazon
group by CustomerType
order by count desc
limit 1;

-- 22.Customer type with highest purchase frequency
SELECT CustomerType, COUNT(*) AS purchases
```

```sql
FROM amazon
GROUP BY CustomerType
ORDER BY purchases DESC
LIMIT 1;

-- 23.Predominant gender among customers
SELECT gender, COUNT(*) AS count
FROM amazon
GROUP BY gender
ORDER BY count DESC
LIMIT 1;

--  24.Gender distribution within each branch
SELECT branch, gender, COUNT(*) AS count
FROM amazon
GROUP BY branch, gender
ORDER BY branch;

-- 25.Time of day when customers provide most ratings
SELECT timeofday, COUNT(rating) AS rating_count
FROM amazon
GROUP BY timeofday
ORDER BY rating_count DESC
LIMIT 1;

--  26.Time of day with highest average rating per branch
SELECT branch, timeofday, ROUND(AVG(rating), 2) AS avg_rating
FROM amazon
GROUP BY branch, timeofday
ORDER BY branch, avg_rating DESC;
```

```sql
--  27.Day of the week with highest average rating
SELECT dayname, ROUND(AVG(rating), 2) AS avg_rating
FROM amazon
GROUP BY dayname
ORDER BY avg_rating DESC
LIMIT 1;

-- 28. Determine the day of the week with the highest average ratings for each branch.
SELECT branch, dayname, ROUND(AVG(rating), 2) AS avg_rating
FROM amazon
GROUP BY branch, dayname
ORDER BY branch, avg_rating DESC;
```