Amazon Sales Data Analysis
SQL CAPSTONE PROJECT

BY ANJALI RANI

ABOUT PROJECT

This dataset contains sales transactions from three different branches of Amazon, respectively located in Mandalay, Yangon and Naypyitaw. The data contains 17 columns and 1000 rows:

Column	Description	Data Type
invoice_id	Invoice of the sales made	VARCHAR(30)
branch	Branch at which sales were made	VARCHAR(5)
city	The location of the branch	VARCHAR(30)
customer_type	The type of the customer	VARCHAR(30)
gender	Gender of the customer making purchase	VARCHAR(10)
product_line	Product line of the product sold	VARCHAR(100)
unit_price	The price of each product	DECIMAL(10, 2)
quantity	The amount of the product sold	INT
VAT	The amount of tax on the purchase	FLOAT(6, 4)
total	The total cost of the purchase	DECIMAL(10, 2)
date	The date on which the purchase was made	DATE
time	The time at which the purchase was made	TIMESTAMP
payment_method	The total amount paid	DECIMAL(10, 2)
cogs	Cost Of Goods sold	DECIMAL(10, 2)
gross_margin_percentage	Gross margin percentage	FLOAT(11, 9)
gross_income	Gross Income	DECIMAL(10, 2)
rating	Rating	FLOAT(2, 1)

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.

Analysis List

1. **Product Analysis**

Conduct analysis on the data to understand the different product lines, the products line 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 product. The result of this can help us 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.

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 a 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.**

Data preprocessing

- First, load the data from the CSV file into a SQL environment. This could be done using tools like MySQL workbench
- Creating DB
- Write table structure and check null values

Feature Engineering

This will help us generate some new columns from existing ones.

- **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.**

```
ALTER TABLE sales_data.amazon
ADD COLUMN timeofday VARCHAR(10);
update sales_data.amazon
set timeofday = case
when hour(time) between 6  and 11 or (hour(time) = 12 and minute(time) = 0) then 'morning'
when hour(time) between 12 and 17 or (hour(time) = 11 and minute(time) > 0) then 'afternoon'
else 'evening'
end;
```

- **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.**

```
alter table sales_data.amazon
add column dayname varchar(20);
UPDATE sales_data.amazon
SET dayname = DAYNAME(STR_TO_DATE(Date, '%d-%m-%Y'));
```

- **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.**

```
alter table sales_data.amazon
add column dayname varchar(20);
UPDATE sales_data.amazon
SET month_name = monthname(STR_TO_DATE(Date, '%d-%m-%Y'));
```

TABLE STRUCTURE

Table: amazon

Columns:

ID	text
Branch	text
City	text
Customer	text
Gender	text
Product	text
price	double
Quantity	int
VAT	double
Total	double
Date	text
Time	text
Payment	text
cogs	double
margin	double
income	double
Rating	double
timeofday	varchar(10)
dayname	varchar(20)
monthname	varchar(20)

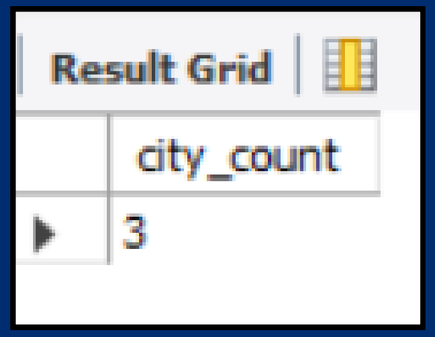
Exploratory Data Analysis (EDA)

Business Questions To Answer:

QUERY -->

```
-- Q.1) -- What is the count of distinct cities in the dataset?
        select count(distinct city) as city_count from sales_data.amazon;
```

Output -->

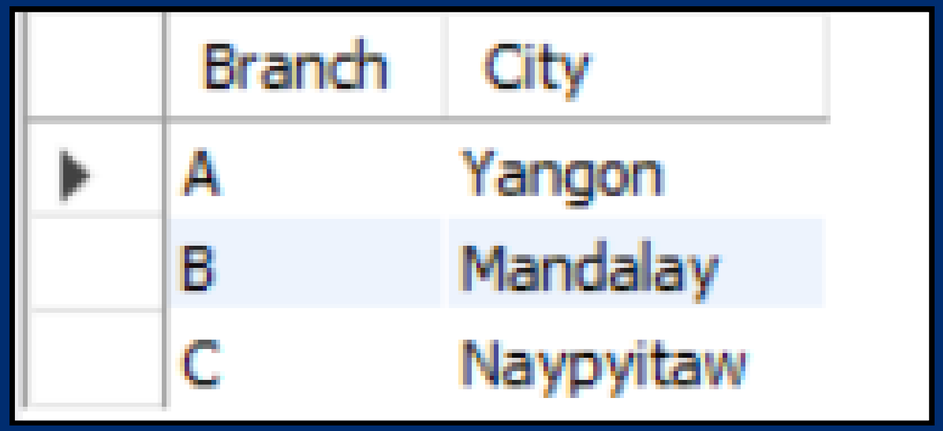


Insight: **The count of distinct cities in a dataset is crucial for understanding the geographic diversity of the data.**

QUERY -->

```
-- Q.2) -- For each branch, what is the corresponding city?
        SELECT DISTINCT Branch, City FROM SALES_DATA.AMAZON
        ORDER BY Branch;
```

Output -->



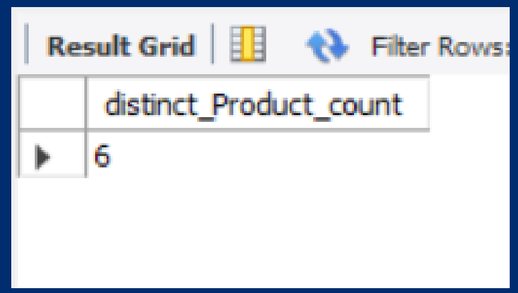
Branch	City
A	Yangon
B	Mandalay
C	Naypyitaw

Insight: **Branch-City Relationship: This query provides a mapping between each branch and its city. It's essential for understanding the geographical distribution of branches, which can be valuable for logistics, marketing, and customer service strategies.**

QUERY -->

```sql
Q.3) -- What is the count of distinct product lines in the dataset?
  SELECT COUNT(DISTINCT Product ) AS distinct_Product_count FROM SALES_DATA.AMAZON;
```

Output -->



Insight: The count of distinct product lines helps in understanding the variety of products or services offered by the business.

QUERY -->

```sql
-- Q.4) -- Which payment method occurs most frequently?
select payment,count(payment) as payment_count from sales_data.amazon
group by payment
order by payment desc;
```

Output -->

Result Grid | Filter Rows:

payment	payment_count
Ewallet	345
Credit card	311
Cash	344

Insight: The most frequently used payment Ewallet method reveals customer payment preferences This can help the business tailor its services to support and promote popular payment options.

QUERY -->

```
-- Q.5) -- Which product line has the highest sales?
  select product,sum(total) as total_sales from sales_data.amazon
  group by product
  order by total_sales desc;
```

Output -->

Result Grid | Filter Rows:

product	total_sales
Food and beverages	56144.844000000005
Sports and travel	55122.826499999996
Electronic accessories	54337.531500000005
Fashion accessories	54305.895
Home and lifestyle	53861.91300000001

Insight: Identifying the product line with the highest sales provides insight into food and beverages category generates the most revenue. This information can guide inventory management, marketing strategies, and future product development.

QUERY -->

```sql
-- Q.6) -- How much revenue is generated each month?
    select month_name,sum(price * quantity) as total_revenue from sales_data.amazon
    group by month_name
    order by total_revenue;
```

Output -->

Result Grid | Filter Rows:

month_name	total_revenue
February	92589.88
March	104243.33999999997
January	110754.16

Insight: Jan month Revenue insights can guide strategic decisions, such as product launches, pricing strategies, and expansion plans.

QUERY -->

```sql
-- Q.7) -- In which month did the cost of goods sold reach its peak?
    select month_name,sum(cogs) as cost_of_goods from sales_data.amazon
    group by month_name
    order by cost_of_goods desc;
```

Output -->

month_name	cost_of_goods
January	110754.16000000002
March	104243.33999999997
February	92589.88

Insight: If JAN month peaks in COGS are consistent and predictable, consider strategies to manage or reduce these costs, such as negotiating better supplier terms or improving inventory management.

QUERY -->

```
-- Q.8) -- Which product line generated the highest revenue?
    select product,sum(price * quantity) as total_revenue from sales_data.amazon
    group by product
    order by total_revenue desc;
```

Output -->

Result Grid | Filter Rows:

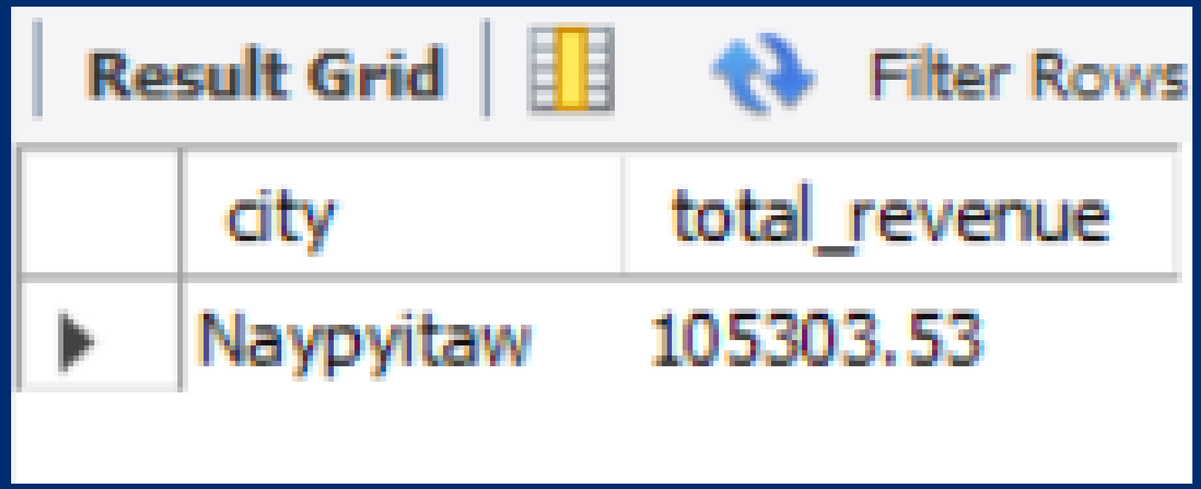
product	total_revenue
Food and beverages	53471.28000000006
Sports and travel	52497.93000000002
Electronic accessories	51750.02999999984
Fashion accessories	51719.89999999997
Home and lifestyle	51297.05999999998

Insight: product lines food and beverages based on the performance of the top line to drive overall growth and improve other areas of the business.

QUERY -->

```sql
-- Q.9) -- In which city was the highest revenue recorded?
    select city,sum(price * quantity) as total_revenue from sales_data.amazon
    group by city
    order by total_revenue desc
    limit 1;
```

Output -->



	city	total_revenue
▶	Naypyitaw	105303.53

Insight: The top Naypyitaw city to understand why it performs well and use this knowledge to improve performance in other locations.

QUERY -->

```
-- Q.10) -- Which product line incurred the highest Value Added Tax?
  select product,sum(VAT) as highest_VAT from sales_data.amazon
  group by product
  order by highest_VAT;
```

Output -->

product	highest_VAT
Health and beauty	2342.5589999999993
Home and lifestyle	2564.853000000002
Fashion accessories	2585.995
Electronic accessories	2587.5015000000017
Sports and travel	2624.896499999994

Insight: A product line with a high VAT contribution might also indicate its significance to overall revenue. If a certain product line is generating high VAT, it likely means it's also a major contributor to total sales, which could be important for financial planning and inventory management.

QUERY -->

```sql
-- Q.11) -- For each product line, add a column indicating "Good" if its sales are above average, otherwise "Bad."
    select product,total,
    case
    when total >= 500 then 'good'
    when total < 200 then 'bad'
    else 'average'
    end as sales
    from sales_data.amazon;
```

Output -->

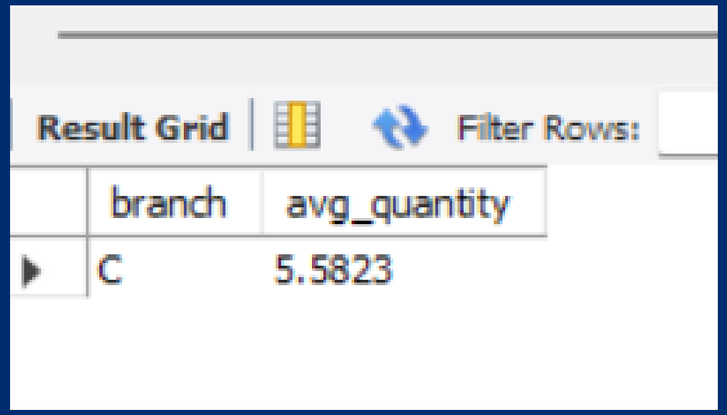
product	total	sales
Health and beauty	548.9715	good
Electronic accessories	80.22	bad
Home and lifestyle	340.5255	average
Health and beauty	489.048	average
Sports and travel	634.3785	good

Insight: By labeling product lines as "Good" or "Bad," can quickly identify which lines are performing above or below the average. This helps in assessing the relative success of each product line.

QUERY -->

```
-- Q.12) -- Identify the branch that exceeded the average number of products sold?
      select branch,avg(quantity) as avg_quantity from sales_data.amazon
      group by branch
      having avg_quantity > (select avg(quantity) from sales_data.amazon);
```

Output -->



Insight: Develop strategies to bring other branches up to the level of these high-performing ones, or replicate successful practices from top branches across others.

QUERY -->

```sql
-- Q.13) -- Which product line is most frequently associated with each gender?
  SELECT gender, product, MAX(total) AS max_sales
FROM (
    SELECT gender, product, COUNT(*) AS total
    FROM sales_data.amazon
    GROUP BY gender, product
) AS gender_product_sales
GROUP BY gender, product
ORDER BY gender, max_sales DESC;
```

Output -->

Gender	Product	Purchase_Frequency
Female	Fashion accessories	96
Female	Food and beverages	90
Female	Sports and travel	88
Female	Electronic accessories	84
Female	Home and lifestyle	79

Gender	Product	Purc
Male	Health and beauty	88
Male	Electronic accessories	86
Male	Food and beverages	84
Male	Fashion accessories	82
Male	Home and lifestyle	81

Insight: Understand the preferences of different genders for specific product lines. we can see fashion accessories category max sales.

QUERY -->

```
-- Q.14) -- Calculate the average rating for each product line.
          select product,avg(rating)as avg_rating from sales_data.amazon
          group by product;
```

Output -->

product	avg_rating
Health and beauty	7.003289473684212
Electronic accessories	6.92470588235294
Home and lifestyle	6.8375
Sports and travel	6.916265060240964
Food and beverages	7.113218390804598

Insight: High ratings might suggest that certain product lines could be expanded, while low ratings might indicate the need for product redesign, repositioning, or even discontinuation.

QUERY -->

```
-- Q.14) -- Count the sales occurrences for each time of day on every weekday?
        select dayname,count(total) as total_sales from sales_data.amazon
        group by dayname;
```

Output -->

dayname	total_sales
Saturday	164
Friday	139
Sunday	133
Monday	125
Thursday	138

Insight: Count the number of sales for each weekday and time of day of the week sales distribution across different times and days. Saturday is highest sales.

QUERY -->

```sql
-- Q.16) --Identify the customer type contributing the highest revenue.
select customer,sum(price * quantity) as revenue from sales_data.amazon
group by customer
order by revenue desc
limit 1;
```

Output -->

customer	revenue
Member	156403.2799999985

Insight: Identifying the member customer type that generates the most revenue is crucial for understanding where your business's key income comes from. It helps in segmenting customers and tailoring services or marketing strategies to maximize revenue.

QUERY -->

```sql
-- Q.17) -- Determine the city with the highest VAT percentage.
select city,round(sum(VAT),2) as vat_percentage from sales_data.amazon
group by city
order by vat_percentage desc
limit 1;
```

Output -->

Result Grid | Filter Rows:

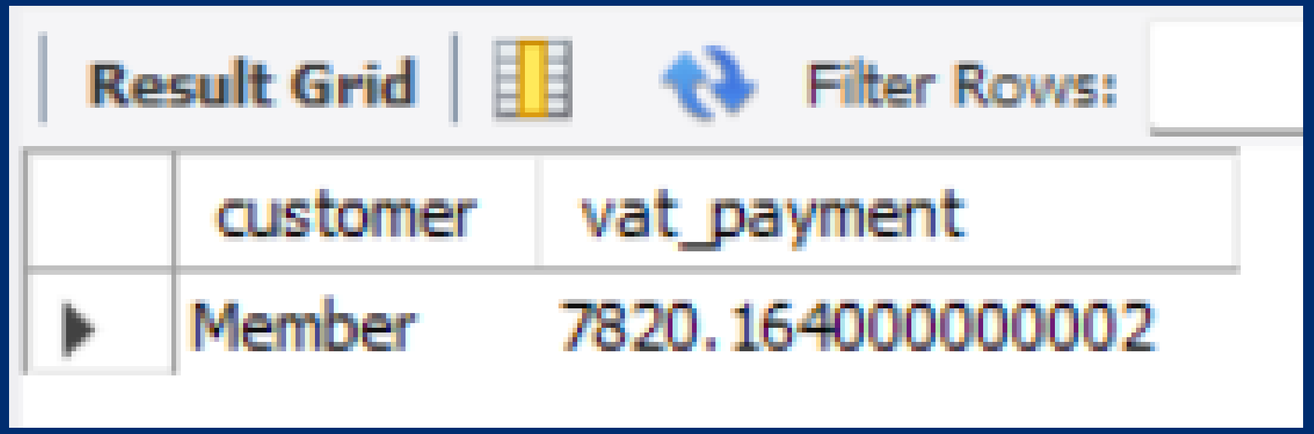
city	vat_percentage
Naypyitaw	5265.18

Insight: Knowing the naypyitaw city with the highest VAT percentage can inform pricing strategies. Businesses might need to adjust prices in cities with higher VAT to maintain profit margins.

QUERY -->

```sql
-- Q.18) -- Identify the customer type with the highest VAT payments.?
select customer,sum(vat) as vat_payment from sales_data.amazon
group by customer
order by vat_payment desc
limit 1;
```

Output -->



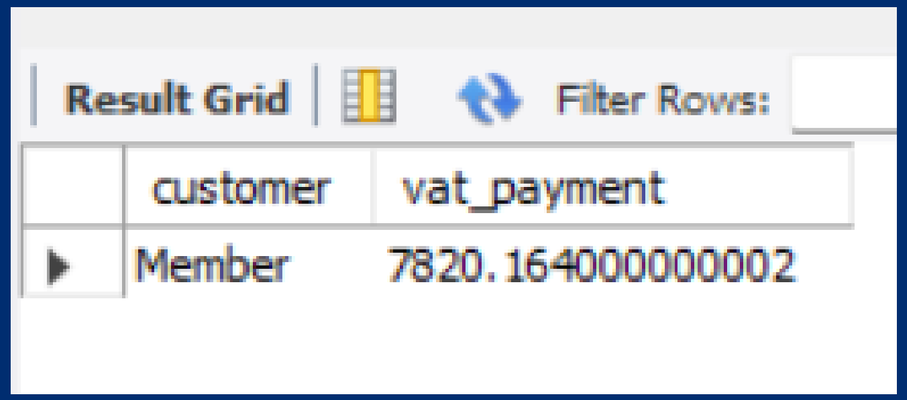
customer	vat_payment
Member	7820.164000000002

Insight: Identifying the member customer type with the highest VAT payments reveals which segment contributes the most to tax revenue. This can indicate which group is generating the most taxable sales, which often correlates with higher overall spending.

QUERY -->

```sql
-- Q.19) -- What is the count of distinct customer types in the dataset?
select count(distinct(customer)) as customer_count from sales_data.amazon;
```

Output -->



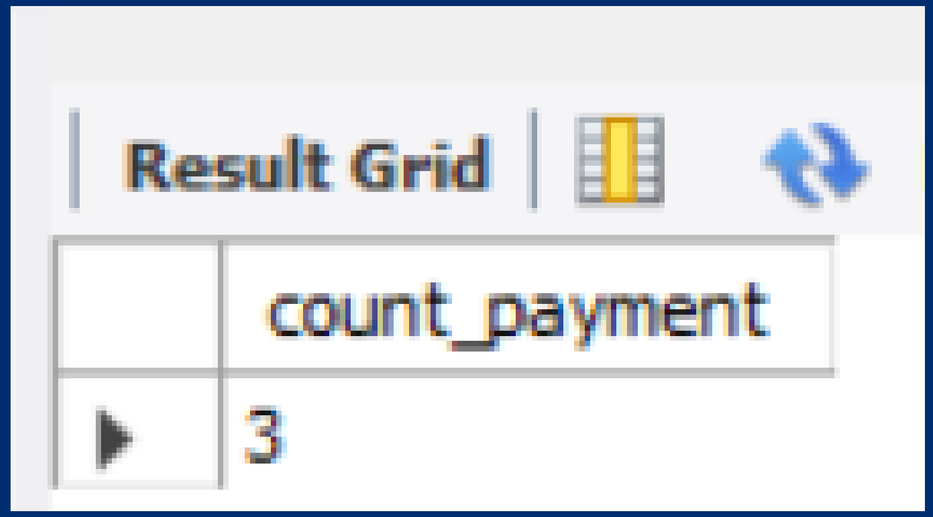
customer	vat_payment
Member	7820.164000000002

Insight: The count of distinct customer types member helps in understanding how the business segments its customers. If the dataset has multiple distinct customer types, it indicates a diversified customer base, which might require different marketing and sales strategies.

QUERY -->

```
-- Q.20) -- What is the count of distinct payment methods in the dataset?
   select count(distinct(payment)) as count_payment from sales_data.amazon;
```

Output -->

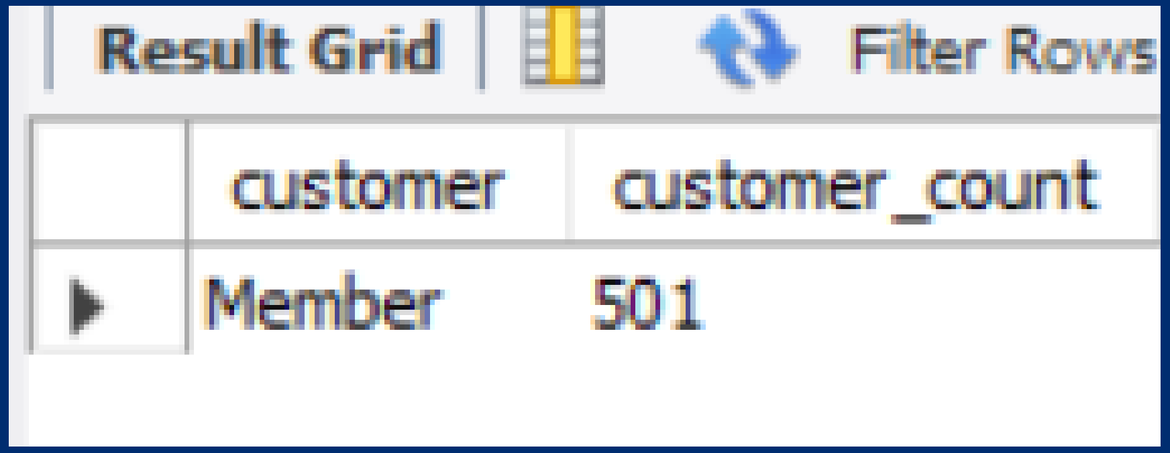


Insight: The count of distinct payment methods provides insight into how diverse the payment options are for customers. A higher number of distinct payment methods suggests that the business offers flexibility and convenience, which can improve customer satisfaction and potentially increase sales.

QUERY -->

```sql
-- Q.21) -- Which customer type occurs most frequently?
    select customer,count(customer) as customer_count from sales_data.amazon
    group by customer
    order by customer_count desc
    limit 1;
```

Output -->


Result Grid | Filter Rows

	customer	customer_count
▶	Member	501

Insight:

Understanding member customer type occurs most frequently helps in identifying the dominant segment of your customer base. This can guide your business in tailoring services, marketing, and customer engagement strategies to better meet the needs of this primary group.

QUERY -->

```sql
-- Q.22) -- Identify the customer type with the highest purchase frequency.
SELECT Customer,COUNT(quantity) as purchasefrequency FROM sales_data.amazon
GROUP BY Customer
ORDER BY purchasefrequency DESC;
```

Output -->

Customer	purchasefrequency
Member	501
Normal	499

Insight: High-frequency customers might provide insights into what's working well in offerings. If they're buying frequently, products or services might be meeting their needs effectively.

QUERY -->

```
-- Q.23) --Determine the predominant gender among customers.
    select gender, customer,count(customer) as customer_count from sales_data.amazon
    group by gender,customer
    order by customer_count;
```

Output -->

gender	customer	customer_count
Female	Normal	240
Male	Member	240
Male	Normal	259
Female	Member	261

Insight: Identifying the predominant gender can provide a snapshot of customer base. For instance, females make up 1% more than male of customer base, this suggests that products or services might appeal more to women.

QUERY -->

```
-- Q.24) -- Examine the distribution of genders within each branch.
      select branch,count(gender) as gender_distribution from sales_data.amazon
      group by branch
      order by gender_distribution desc;
```

Output -->

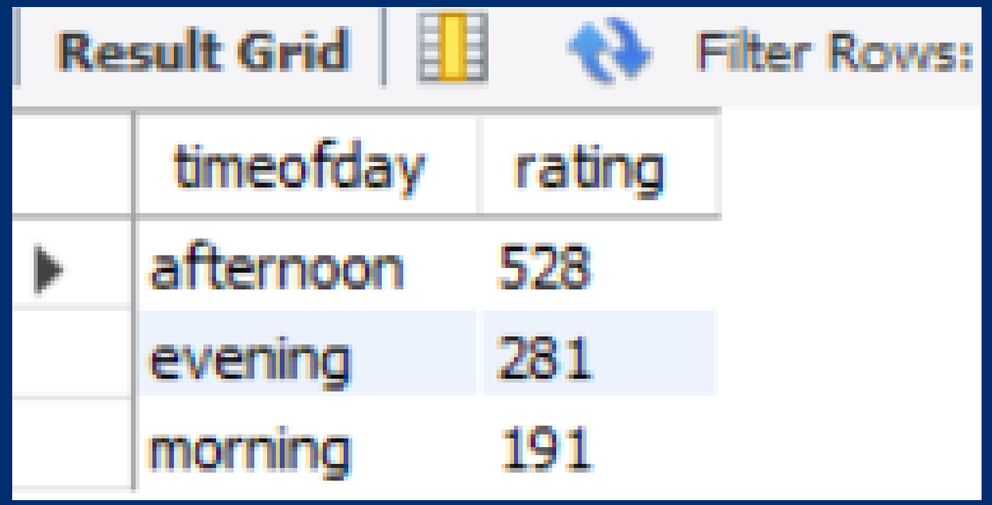
branch	gender_distribution
A	340
B	332
C	328

Insight: By analyzing the gender distribution in each branch, identify branches where one gender is particularly predominant. For example, Branch A might have a higher proportion of female customers, while Branch B might have more male customers. This could indicate regional or cultural preferences.

QUERY -->

```sql
-- Q.25) --Identify the time of day when customers provide the most ratings.
    select  timeofday,count(*) rating from sales_data.amazon
    group by timeofday
    order by rating desc;
```

Output -->



timeofday	rating
afternoon	528
evening	281
morning	191

Insight: Identifying the time of day afternoon most ratings are provided can help understand when customers are most engaged. For example, if most ratings occur in the afternoon between 12 PM and 6 PM, it suggests that customers are more likely to interact with platform after work hours.

QUERY -->

```sql
-- Q.26) -- Determine the time of day with the highest customer ratings for each branch.
WITH RatingsByHour AS ( SELECT  Branch,timeofday,COUNT(*) AS Rating FROM sales_data.amazon
GROUP BY Branch, timeOfDay)
SELECT Branch,timeOfDay,Rating FROM  RatingsByHour
WHERE (Branch, Rating) IN (SELECT Branch, MAX(Rating)  FROM RatingsByHour
GROUP BY  Branch)
ORDER BY Branch, timeOfDay;
```

Output -->

Branch	timeOfDay	Rating
A	afternoon	185
B	afternoon	162
C	afternoon	181

Insight: A branch has a consistent peak time for receiving ratings, it might indicate a good opportunity for localized marketing. For instance, Branch A might benefit from afternoon promotions if most of its ratings come in during the afternoon.

QUERY -->

```
-- Q.27) --Identify the day of the week with the highest average ratings.
      select dayname,avg(rating) as avg_rating from sales_data.amazon
      group by dayname
      order by avg_rating desc
      limit 1;
```

Output -->

	dayname	avg_rating
▶	Monday	7.153599999999999

Insight: Identifying dayname with the highest average ratings can provide insights into when customers are most satisfied with products or services. For example, if Monday have the highest average rating, it could indicate that customers have more positive experiences on weekday, possibly due to more leisure time or better service availability.

QUERY -->

```
-- Q.28) -- Determine the day of the week with the highest average ratings for each branch.
    WITH RatingsByDay AS (SELECT Branch,dayname,AVG(Rating) AS AvgRating FROM sales_data.amazon
    GROUP BY  Branch,dayname),
   RankedRatings AS ( SELECT Branch, dayname, AvgRating,
   RANK() OVER (PARTITION BY Branch ORDER BY AvgRating DESC) AS RatingRank FROM RatingsByDay)
   SELECT Branch,dayname, AvgRating FROM RankedRatings
   WHERE  RatingRank = 1
   ORDER BY Branch;
```

Output -->

Branch	dayname	AvgRating
A	Friday	7.311999999999985
B	Monday	7.335897435897434
C	Friday	7.278947368421051

Insight: Identify days of the week each branch tends to receive the highest ratings. This could suggest the best-performing day for customer satisfaction per branch.

CONCLUSION

- **Product Analysis**

The products lines food and beverages performing best and the product lines health and beauty that need to be improved.

- **Sales Analysis**

Highest sales in january month and low in february month
Highest city sales is Naypyitaw
Ewallet is the most frequent payment method
Female in Fashion and accessories - 96
Male in health and beauty - 88

- **Customer Analysis**

customer type member - 501 and Normal - 499

Recommendation

prioritize the areas with the most significant impact on revenue and customer satisfaction, such as focusing on top-selling products, improving customer retention, optimizing pricing, and enhancing operational efficiency. Regularly monitor the implementation of these strategies and adjust as needed based on can improve performance of data.

Thank You