Amazon Sales Analysis

SQL Capstone Project

Introduction

To gain important knowledge for enhancing sales performance, this capstone project will examine sales data from Amazon branches in Mandalay, Yangon, and Naypyitaw. The research looks for patterns that affect revenue and customer behavior by analyzing 1,000 transactions from a variety of product lines and customer demographics.

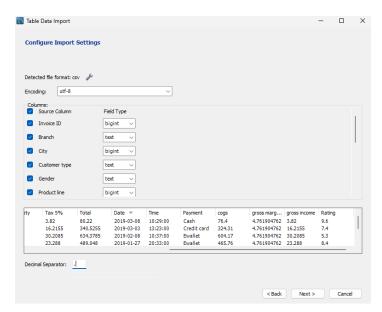
Objective

To improve business strategy by gaining understanding of the elements influencing sales in various branches.

Business Challenge

Amazon finds it difficult to identify which product categories and consumer groups bring them the most money. By offering practical insights to improve overall profitability and sales methods, the initiative will solve these problems.

Data Loading



Data Wrangling

```
-- Data Wrangling
SELECT * from amazon where `Invoice ID` is NULL
or Branch is null
or City is null
or `Customer type` is null
 or Gender is null
 or `Product line` is null
 or `Unit price` is null
 or `Quantity` is null
 or `Tax 5%` is null
 or Total is null
 or Date is null
 or Time is null
or Payment is null
or cogs is null
or `gross margin percentage` is null
or `gross income` is null
or Rating is null;
```

Feature Engineering

```
ALTER TABLE amazon
ADD COLUMN timeofday VARCHAR(10),
ADD COLUMN dayname VARCHAR(10),
ADD COLUMN monthname 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;
UPDATE amazon
SET dayname = DAYNAME('date');
UPDATE amazon
SET dayname = DATE_FORMAT(`date`, '%a');
UPDATE amazon
SET monthname = MONTHNAME('date');
UPDATE amazon
SET monthname = DATE_FORMAT(`date`, '%b');
SELECT time, timeofday, date, dayname, monthname
FROM amazon
LIMIT 10;
```

	time	timeofday	date	dayname	monthname
•	13:08:00	Afternoon	2019-01-05	Sat	Jan
	10:29:00	Morning	2019-03-08	Fri	Mar
	13:23:00	Afternoon	2019-03-03	Sun	Mar
	20:33:00	Evening	2019-01-27	Sun	Jan
	10:37:00	Morning	2019-02-08	Fri	Feb
	18:30:00	Evening	2019-03-25	Mon	Mar
	14:36:00	Afternoon	2019-02-25	Mon	Feb
	11:38:00	Morning	2019-02-24	Sun	Feb

Research Questions

1. What is the count of distinct cities in the dataset?

Code:

```
select count(distinct City) from amazon;
```

Output:



Insights: There are three Distinct cities: Yangon, Naypyitaw, Mandalay

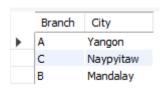
2. For each branch, what is the corresponding city?

Code:

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

```
select Branch, City from amazon
Group by branch, city;
```

Output:



Insights: $A \rightarrow Yangon$, $B \rightarrow Naypyitaw$, $C \rightarrow Mandalay$

3. What is the count of distinct product lines in the dataset?

Code:

```
-- 4. Which payment method occurs most frequently?

SELECT Payment, COUNT(*) AS frequency

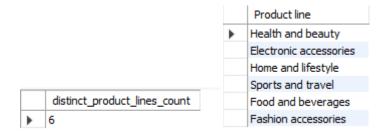
FROM amazon

GROUP BY Payment

ORDER BY frequency DESC

limit 1;
```

Output:

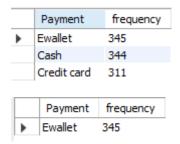


Insights: There are six product_lines:

- Health and beauty
- Electronic acessories
- Home and lifestyle
- Sports and travel
- Food and beverages
- Fashion accessories
- 4. Which payment method occurs most frequently?

```
SELECT Payment, COUNT(*) AS frequency
FROM amazon
GROUP BY Payment
ORDER BY frequency DESC
limit 1;
```

Output:



Insights: The most used Payment methos id E-wallet

5. Which product line has the highest sales?

Code:

```
-- 5. Which product line has the highest sales?

SELECT 'Product line', SUM(total) AS total_sales

FROM amazon

GROUP BY 'Product line'

ORDER BY total_sales DESC

LIMIT 1;
```

Output:

				Product line	total_sales
			•	Food and beverages	56144.844000000005
				Sports and travel	55122.826499999996
				Electronic accessories	54337.531500000005
	5 1 11			Fashion accessories	54305.895
	Product line	total_sales		Home and lifestyle	53861.91300000001
•	Food and beverages	56144.844000000005		Health and beauty	49193.739000000016

Insights: Food and Beverages is the Product Line with Highest Sales

6. How much revenue is generated each month?

Code:

```
-- 6. How much revenue is generated each month?

SELECT monthname AS month, SUM(`Unit price` * quantity) AS monthly_revenue
FROM amazon
GROUP BY month
ORDER BY month;
Output:
```

	month	monthly_revenue
•	Feb	92589.88
	Jan	110754.16
	Mar	104243.33999999997

Insights: Revenue is Calculated Using Unit price * quantity and Most Revenue is generates during January followed by March and February

7. In which month did the cost of goods sold reach its peak?

Code:

```
-- 7. In which month did the cost of goods sold reach its peak?

SELECT monthname AS month, SUM(cogs) AS total_cogs

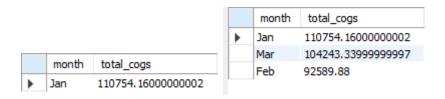
FROM amazon

GROUP BY month

ORDER BY total_cogs DESC

LIMIT 1;
```

Output:



Insights: The cost of goods are it's peak during January

8. Which product line generated the highest revenue?

Code:

```
-- 8. Which product line generated the highest revenue?

SELECT `Product line`, SUM(`unit price` * quantity) AS total_revenue
FROM amazon
GROUP BY `Product line`
ORDER BY total_revenue DESC
LIMIT 1;
```

Output:

	Product line	total_revenue
•	Food and beverages	53471.28000000006
	Product line	total_revenue
•	Food and beverages	53471.28000000006
	Sports and travel	52497.93000000002
	Electronic accessories	51750.029999999984
	Fashion accessories	51719.89999999997
	Home and lifestyle	51297.05999999998
	Health and beauty	46851, 17999999998

Insights: Food and Beverages generates the highest revenue

9. In which city was the highest revenue recorded?

Code:

```
-- 9. In which city was the highest revenue recorded?

SELECT City, SUM(`unit price` * quantity) AS total_revenue
FROM amazon
GROUP BY City
ORDER BY total_revenue DESC
LIMIT 1;
Output:
```

Insights: Naypyitaw is the City with highest revenue

City total_revenue

Naypyitaw 105303.53

10. Which product line incurred the highest Value Added Tax?

Code:

```
-- 10. Which product line incurred the highest Value Added Tax?

SELECT `Product line`, SUM(`Tax 5%`) AS total_vat

FROM amazon

GROUP BY `Product line`

ORDER BY total_vat DESC

LIMIT 1;

Output:

Productline total_vat

Food and beverages 2673.5639999999994
```

Insights: Food and beverages have the Highest Value Added Tax

11. For each product line, add a column indicating "Good" if its sales are above average, otherwise "Bad."

```
■ 

○ WITH ProductSales AS (
       SELECT
           `Product line`,
            SUM(`unit price` * quantity) AS total sales
        FROM amazon
        GROUP BY 'Product line'
   ),
 AverageSales AS (
        SELECT AVG(total_sales) AS avg_sales
        FROM ProductSales
    SELECT
        ps. Product line,
        ps.total_sales,
        a.avg_sales,
            WHEN ps.total_sales > a.avg_sales THEN 'Good'
            ELSE 'Bad'
        END AS sales_performance
    FROM ProductSales ps, AverageSales a;
```

Output:

	Product line		total_sales		avg_sales		sales_performanc
•	Health and beauty		46851.1799999999	8	51264.563333333	333	Bad
	Electronic accessor	ies	51750.0299999999	84	51264.563333333	333	Good
	Home and lifestyle		51297.0599999999	8	51264.563333333	333	Good
	Sports and travel		52497.9300000000	2	51264.563333333	333	Good
	Food and beverage	es	53471.2800000000	6	51264.563333333	333	Good
Spo	rts and travel	524	97.93000000002	5126	54.56333333333	Good	d
FSp	oorts and travel	534	71.28000000006	5126	54.56333333333	Good	d
Fas	hion accessories	517	19.8999999997	5126	54.56333333333	Good	d

Insights: Health and Beauty has sales below average and performing quite low.

12. Identify the branch that exceeded the average number of products sold.

Code:

```
-- 12. Identify the branch that exceeded the average number of products sold.
```

```
WITH BranchSales AS (
          SELECT branch, SUM(quantity) AS total_products_sold
          FROM amazon
          GROUP BY branch
- )
SELECT
          branch,
          total_products_sold,
          (SELECT AVG(total_products_sold) FROM BranchSales) AS avg_products_sold
FROM BranchSales
HAVING total_products_sold > avg_products_sold;
```

Output:

	branch	total_products_sold		branch	total_products_sold	avg_products_sold
>	Α	1859	•	Α	1859	1836.6667

Insights: A \rightarrow Yangon exceeded the average number of products sold.

13. Which product line is most frequently associated with each gender?

Code:

Insights: Females frequently use Fashion accessories whereas Males use Health and beauty

14. Calculate the average rating for each product line.

Female Fashion accessories

Male Health and beauty

```
-- 14. Calculate the average rating for each product line.

SELECT 'Product line', AVG(rating) AS average_rating

FROM amazon

GROUP BY 'Product line';

Output:
```

	Product line	average_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
	Fashion accessories	7.029213483146067

Insights: Food and beverages have the most average rating rating followed by Fashion and accessories and Health and beauty

15. Count the sales occurrences for each time of day on every weekday.

Code:

```
-- 15. Count the sales occurrences for each time of day on every weekday.

SELECT

dayname AS weekday,

timeofday,

COUNT(*) AS sales_count

FROM amazon

GROUP BY weekday, timeofday

ORDER BY weekday, FIELD(timeofday, 'Morning', 'Afternoon', 'Evening');
```

Output:

	weekday	timeofday	sales_count
•	Mon	Morning	21
	Sun	Morning	22
	Wed	Morning	22
	Sat	Morning	28
	Fri	Morning	29
	Mon	Evening	29
	Thu	Evening	29
	Thu	Morning	33
	Fri	Evening	36
	Tue	Morning	36
	Wed	Evening	40
	Sun	Evenina	41

WEEKDAY	TIMEOFDAY	SALES_COUNT
Fri	Morning	29
Fri	Afternoon	74
Fri	Evening	36
Mon	Morning	21
Mon	Afternoon	75
Mon	Evening	29
Sat	Morning	28
Sat	Afternoon	81

Sat	Evening	55	
Sun	Morning	22	
Sun	Afternoon	70	
Sun	Evening	41	
Thu	Morning	33	
Thu	Afternoon	76	
Thu	Evening	29	
Tue	Morning	36	
Tue	Afternoon	71	
Tue	Evening	51	
Wed	Morning	22	
Wed	Afternoon	81	
Wed	Evening	40	

Insights: Most sales occur during Afternoons and mostly on Wednesday and Saturday

16. Identify the customer type contributing the highest revenue.

Code:

```
-- 16. Identify the customer type contributing the highest revenue.

SELECT `customer type`, SUM(`unit price` * quantity) AS total_revenue

FROM amazon

GROUP BY `customer type`

ORDER BY total_revenue DESC

LIMIT 1;
```

Output:

	customer type	total_revenue
•	Member	164223.44400000002

Insights: Most of the customers affiliated as Member contribute to the most revenue

17. Determine the city with the highest VAT percentage.

Code:

```
-- 17. Determine the city with the highest VAT percentage.

SELECT City, MAX(`Tax 5%`) AS highest_vat_percentage

FROM amazon

GROUP BY City

ORDER BY highest_vat_percentage DESC

LIMIT 1;

Output:

City highest_vat_percentage
```

Insights: Naypyitaw has the highest VAT percentage

Naypyitaw 49.65

18. Identify the customer type with the highest VAT payments.

Code:

```
-- 18. Identify the customer type with the highest VAT payments.

SELECT `customer type`, SUM(`Tax 5%`) AS total_vat

FROM amazon

GROUP BY `customer type`

ORDER BY total_vat DESC

LIMIT 1;

Output:
```



Insights: : Most of the customers affiliated as Member contribute to the most VAT

19. What is the count of distinct customer types in the dataset?

```
-- 19. What is the count of distinct customer types in the dataset?

SELECT COUNT(DISTINCT `customer type`) AS distinct_customer_types_count
FROM amazon;
```

Output:

```
distinct_customer_types_count

2
```

Insights: There are two customer types in the dataset: Member, Normal

20. What is the count of distinct payment methods in the dataset?

Code:

```
-- 20. What is the count of distinct payment methods in the dataset?

SELECT COUNT(DISTINCT Payment) AS distinct_payment_methods

FROM amazon;

Output:

distinct_payment_methods

3
```

Insights: Three Payment methods: E-wallet, Cash and Credit Card

21. Which customer type occurs most frequently?

Code:

```
-- 21. Which customer type occurs most frequently?

SELECT `customer type`, COUNT(*) AS occurrence_count
FROM amazon
GROUP BY `customer type`
ORDER BY occurrence_count DESC
LIMIT 1;

Output:
```



Insights: Member Customer type occurs most frequently

22. Identify the customer type with the highest purchase frequency.

Code:

```
-- 22. Identify the customer type with the highest purchase frequency.

SELECT `customer type`, COUNT(*) AS purchase_frequency

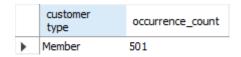
FROM amazon

GROUP BY `customer type`

ORDER BY purchase_frequency DESC

LIMIT 1;
```

Output:



Insights: Member Customer type purchases more often.

23. Determine the predominant gender among customers.

Code:

```
-- 23. Determine the predominant gender among customers.

SELECT gender, COUNT(*) AS gender_count

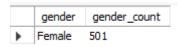
FROM amazon

GROUP BY gender

ORDER BY gender_count DESC

LIMIT 1;
```

Output:



Insights: Female make more Purchases compared to male

24. Examine the distribution of genders within each branch.

Code:

```
-- 24. Examine the distribution of genders within each branch.

SELECT branch, gender, COUNT(*) AS gender_count

FROM amazon

GROUP BY branch, gender

ORDER BY branch, gender_count DESC;
```

Output:

	branch	gender	gender_count
•	Α	Male	179
	Α	Female	161
	В	Male	170
	В	Female	162
	С	Female	178
	С	Male	150

Insights: Highest number of males in Branch A and Highest number of females in Branch B

25. Identify the time of day when customers provide the most ratings.

Code:

```
-- 25. Identify the time of day when customers provide the most ratings.

SELECT

timeofday,

COUNT(rating) AS rating_count

FROM amazon

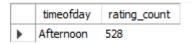
WHERE rating IS NOT NULL

GROUP BY timeofday

ORDER BY rating_count DESC

LIMIT 1;
```

Output:



Insights: During Afternoons customers provide most Ratings

26. Determine the time of day with the highest customer ratings for each branch.

Code:

```
SELECT branch, timeofday, average_rating AS highest_average_rating
FROM (
     SELECT
         branch,
         timeofday,
         AVG(rating) AS average_rating,
         RANK() OVER (PARTITION BY branch ORDER BY AVG(rating) DESC) AS rnk
     FROM amazon
     WHERE rating IS NOT NULL
     GROUP BY branch, timeofday
) AS TimeRatings
 WHERE rnk = 1
 ORDER BY branch;
Output:
    branch timeofday highest_average_rating
           Afternoon
                    7.0567567567567595
```

Insights: Brach C is the one that produces the most ratings.

6.891525423728813

Afternoon 7.0955801104972345

7.153599999999999

27. Identify the day of the week with the highest average ratings.

Code:

Mon

В

С

Morning

```
-- 27. Identify the day of the week with the highest average ratings.

SELECT dayname AS weekday, AVG(rating) AS average_rating

FROM amazon

WHERE rating IS NOT NULL

GROUP BY weekday

ORDER BY average_rating DESC

LIMIT 1;

Output:

weekday average_rating
```

Insights: Monday is the day with the highest average ratings

28. Determine the day of the week with the highest average ratings for each branch.

Code:

C

Friday

```
-- 28. Determine the day of the week with the highest average ratings for each branch.
 SELECT
     d.branch,
     d.weekday AS day_of_week,
     d.average_rating AS highest_average_rating
FROM (
     SELECT
        branch,
        DAYNAME(date) AS weekday,
         AVG(rating) AS average_rating
     FROM amazon
     WHERE rating IS NOT NULL
     GROUP BY branch, weekday
 ) AS d
 INNER JOIN (
     SELECT
         branch,
         MAX(average_rating) AS max_average_rating
     FROM (
         SELECT
             branch,
             DAYNAME(date) AS weekday,
             AVG(rating) AS average_rating
         FROM amazon
         WHERE rating IS NOT NULL
         GROUP BY branch, weekday
     ) AS sub
     GROUP BY branch
 ON d.branch = m.branch AND d.average_rating = m.max_average_rating
 ORDER BY d.branch;
Output:
branch day_of_week highest_average_rating
       Friday
                   7.3119999999999985
В
       Monday 7.335897435897434
```

Insights: Branch B is the one that make highest ratings during Monday.

7.278947368421051