Name : Muhammad Naseem

Father name:Mirza Hussain

Section :4

Email :mnaseembaseeji@gmail.com.pk

Contect : 03554859971

Course : AI and Data Science

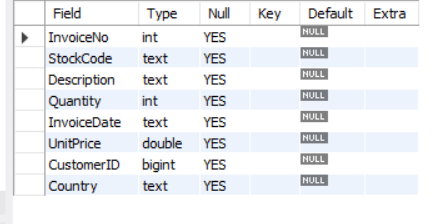
create schema retail\_shop;

use retail\_shop;

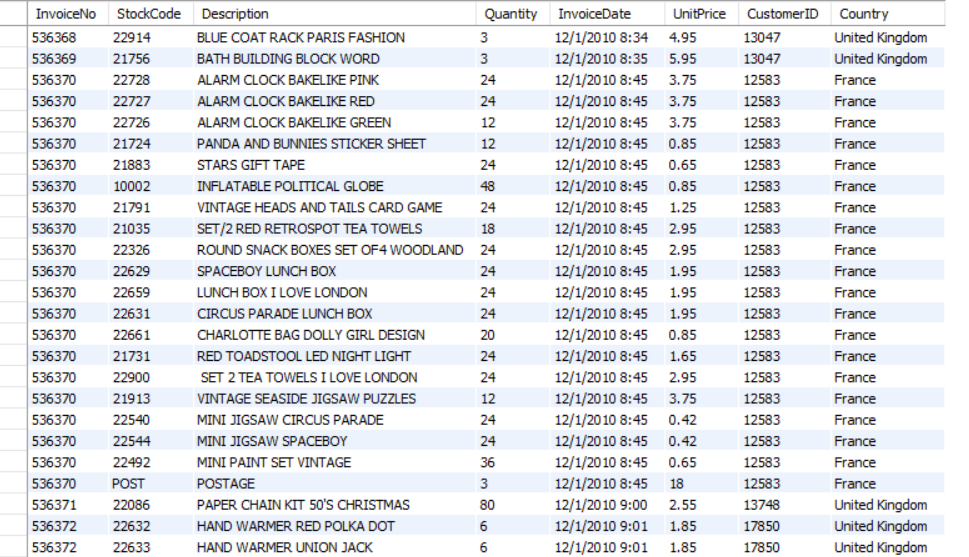
-- BEGINNER QUERIES

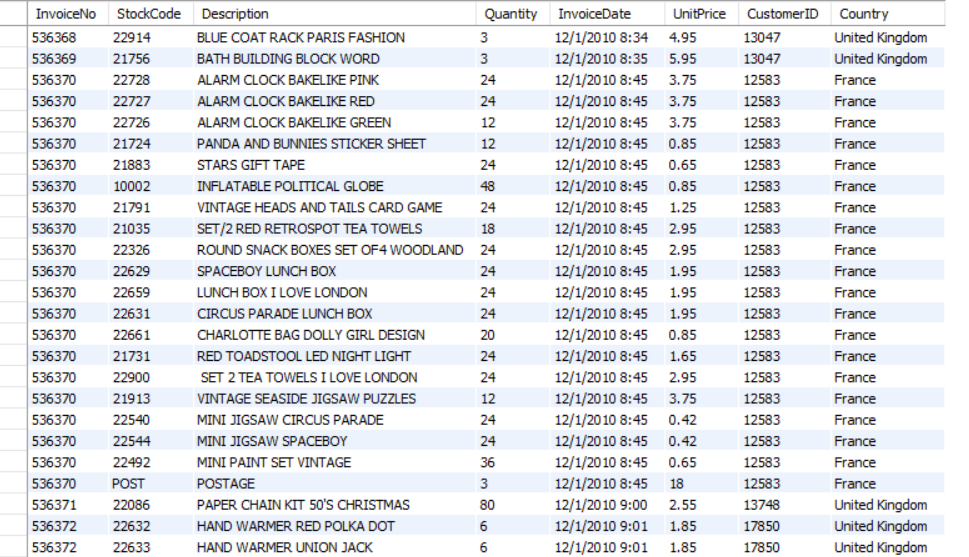
-- beginner\_query\_1. Definition of meta Data in mySQL Workbench

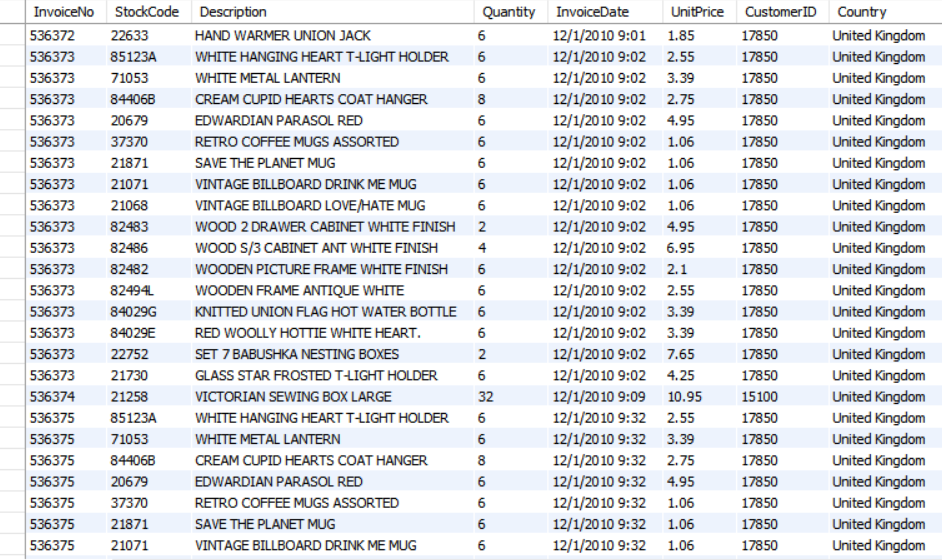
describe online\_retail;

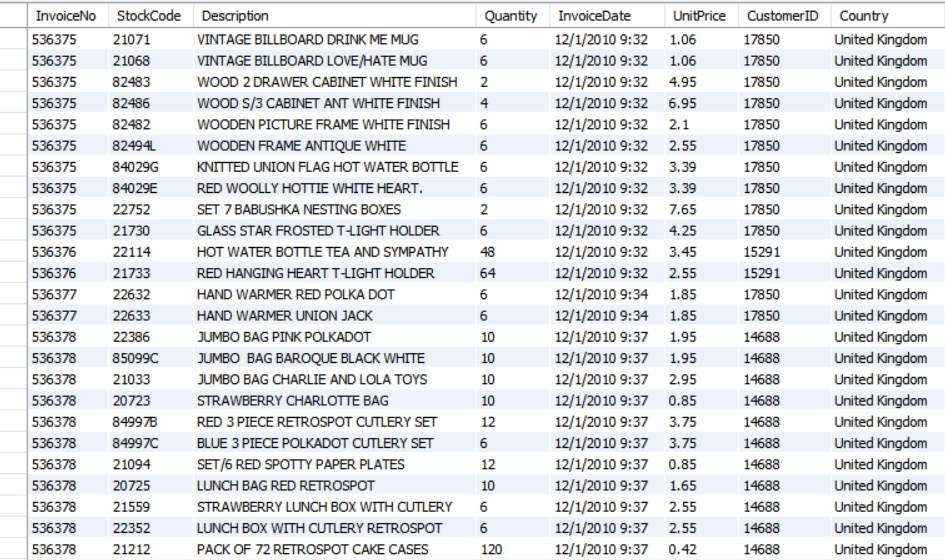


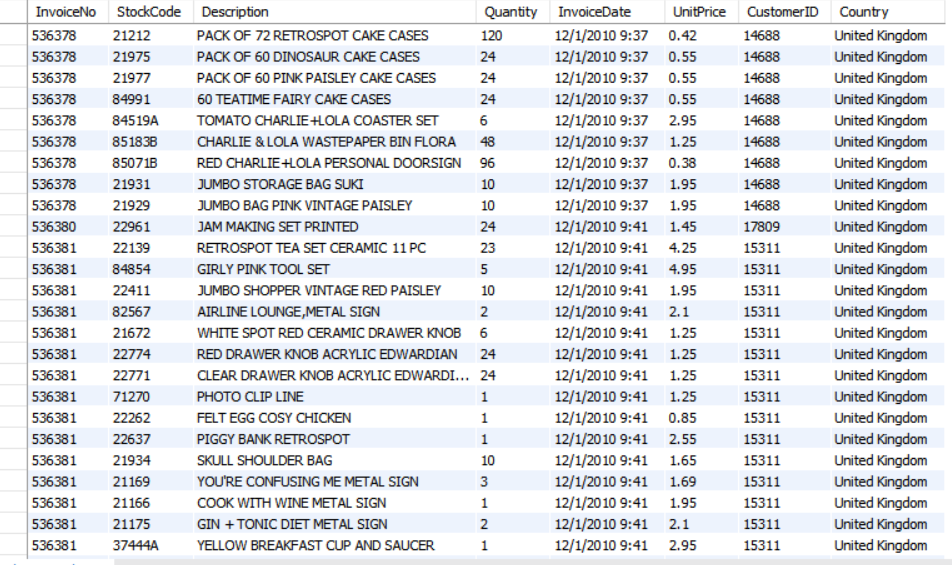
select \* from online\_retail;











.

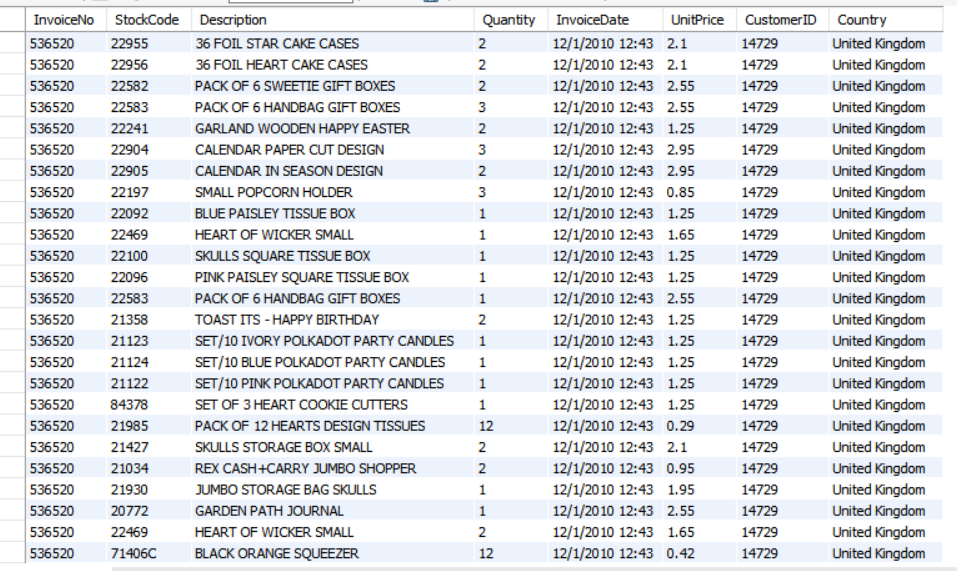
.

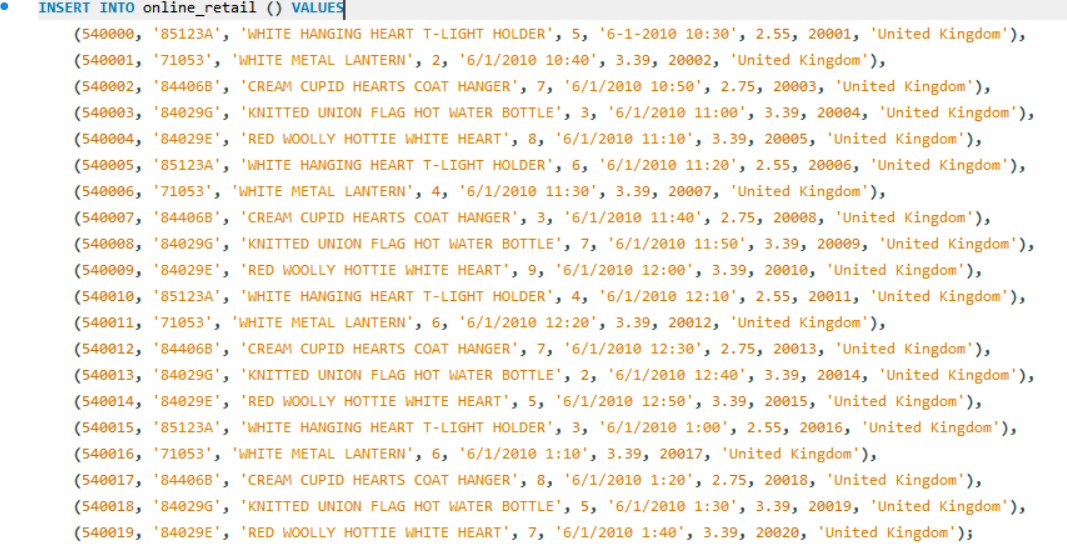
.

.

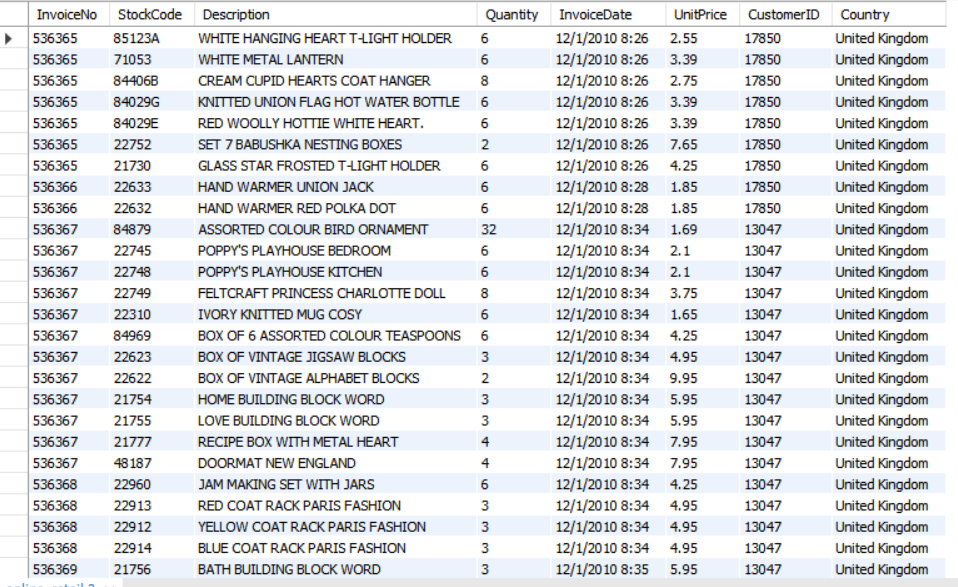
.

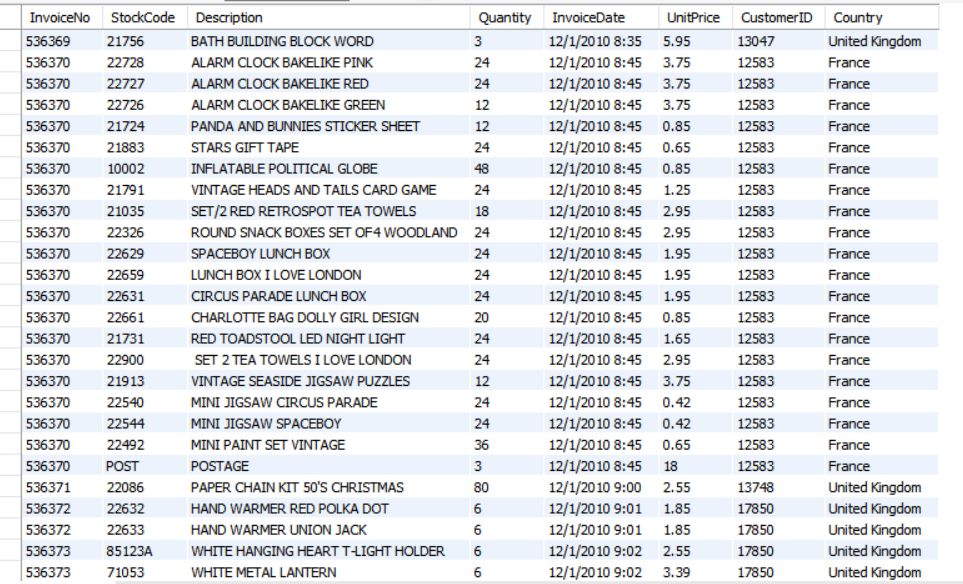
.





select \* from online\_retail;





.

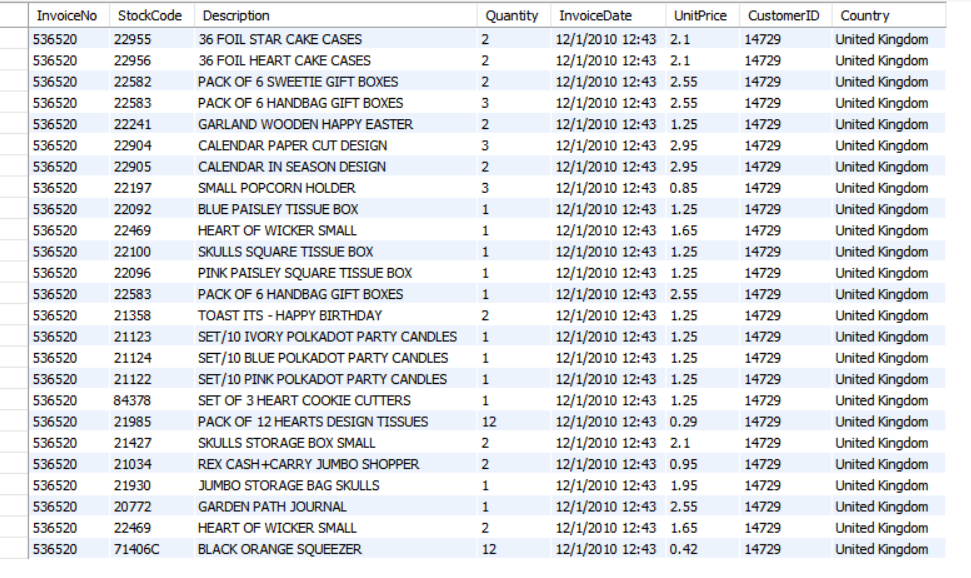
.

.

.

.

.



beginner\_query\_2. What is the distribution of order values across all customers in the dataset?

SELECT

Customer ID,

SUM(Quantity \* Unit Price) AS Total Order Value

FROM

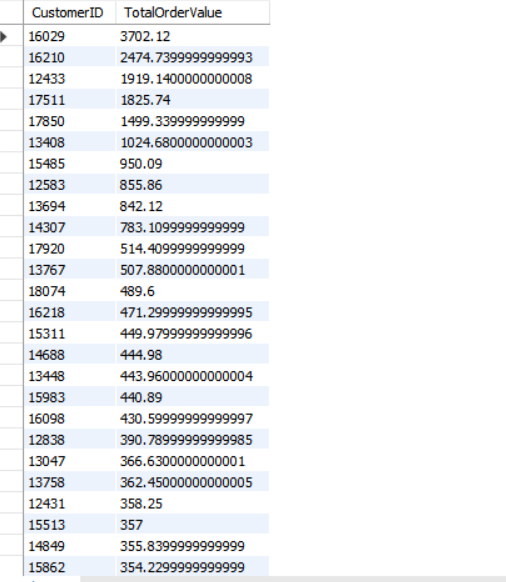
online\_retail

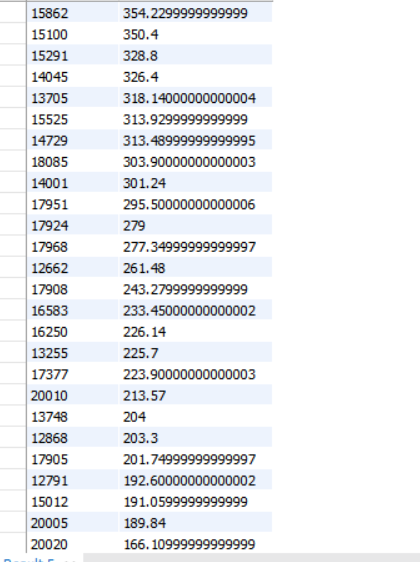
GROUP BY

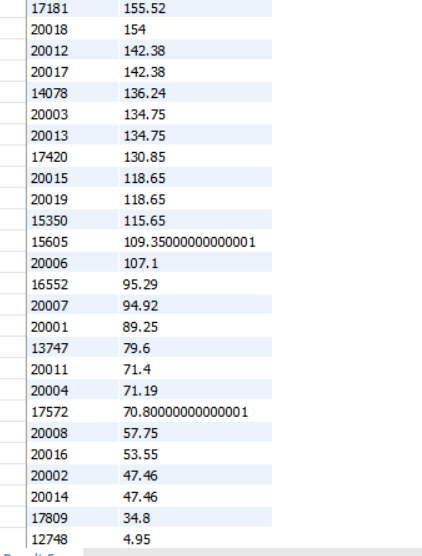
Customer ID

ORDER BY

Total Order Value DESC;







beginner\_query\_3. How many unique products has each customer purchased?

SELECT

Customer ID,

COUNT(DISTINCT Stock Code) AS Unique Products Purchased

FROM

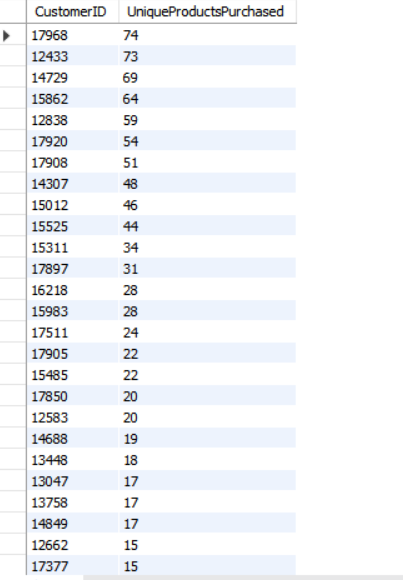
online\_retail

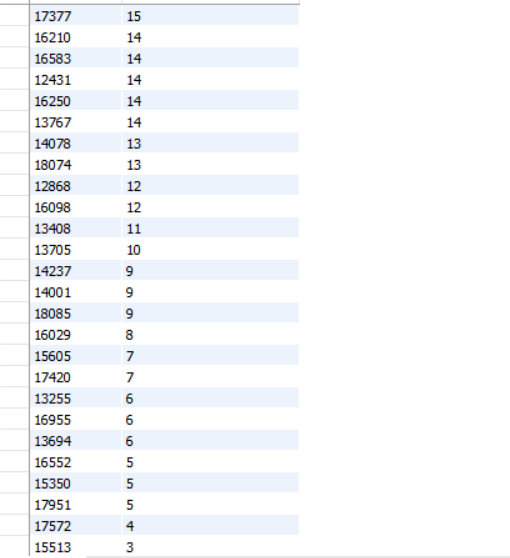
GROUP BY

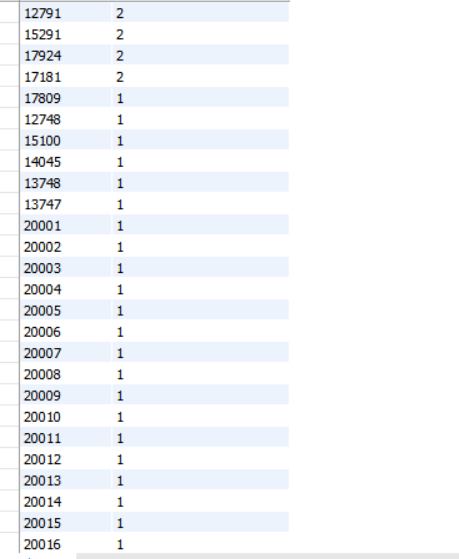
Customer ID

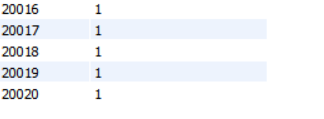
ORDER BY

Unique Products Purchased DESC;









-- beginner\_query\_4. Which customers have only made a single purchase from the company?

SELECT

Customer ID

FROM

online\_retail

GROUP BY

Customer ID

HAVING

COUNT(DISTINCT Invoice No) = null;

beginner\_query\_5. Which products are most commonly purchased together by customers in the dataset?

SELECT

p1.Stock Code AS Product1,

p2.Stock Code AS Product2,

COUNT(\*) AS Frequency

FROM

online\_retail p1

JOIN

online\_retail p2

ON

p1.Invoice No = p2.InvoiceNo

AND

p1.Stock Code < p2.StockCode

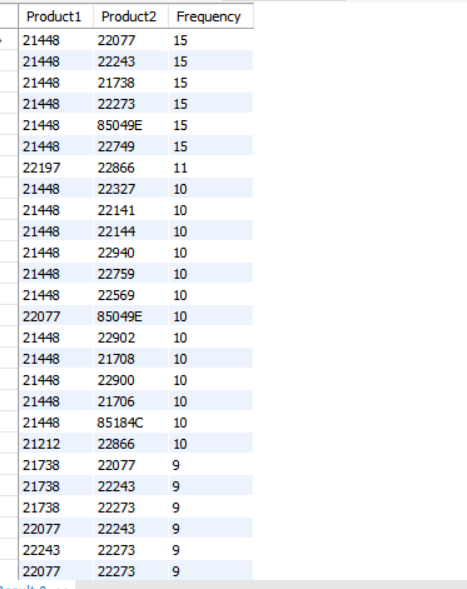
GROUP BY

p1.Stock Code,

p2.Stock Code

ORDER BY

Frequency DESC;



advanced\_query\_5. Time-based Analysis

-- Explore trends in customer behavior over time, such as monthly or quarterly sales patterns.

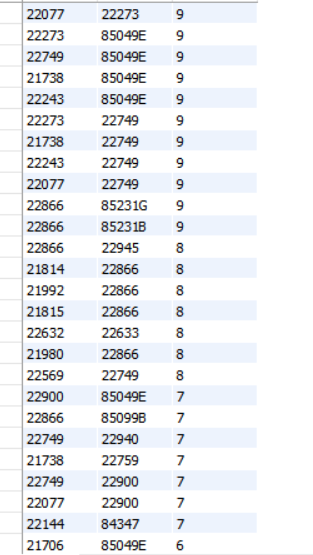
--

set SQL\_SAFE\_UPDATEs = 0;

UPDATE online\_retail

SET Invoice Date = STR\_TO\_DATE(Invoice Date, '%m/%d/%Y %H:%i')

WHERE Invoice Date IS NOT NULL;



.

.

.

.

.

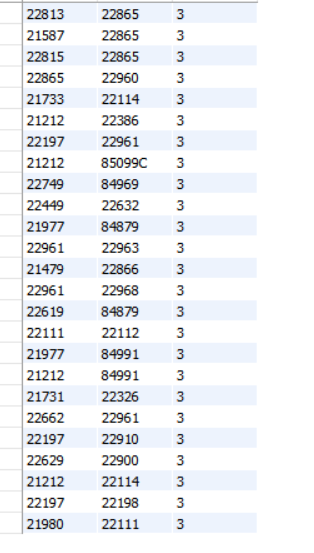
.

.

.

.

.



ADVANCED QUERIES

-- advanced\_query\_1. Customer Segmentation by Purchase Frequency

-- Group customers into segments based on their purchase frequency, such as high, medium, and low frequency

-- customers. This can help you identify your most loyal customers and those who need more attention

SELECT

Customer ID,

COUNT(DISTINCT Invoice No) AS Purchase Frequency,

CASE

WHEN COUNT(DISTINCT Invoice No) > 10 THEN 'High Frequency'

WHEN COUNT(DISTINCT Invoice No) BETWEEN 5 AND 10 THEN 'Medium Frequency'

ELSE 'Low Frequency'

END AS Customer Segment

FROM

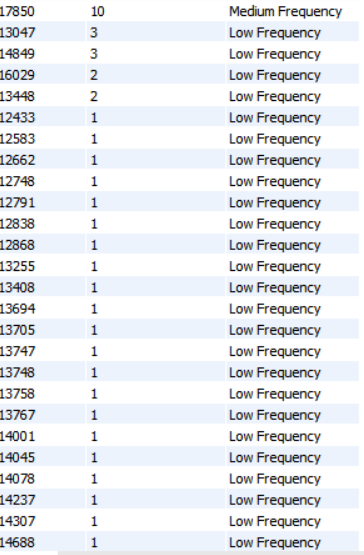
online\_retail

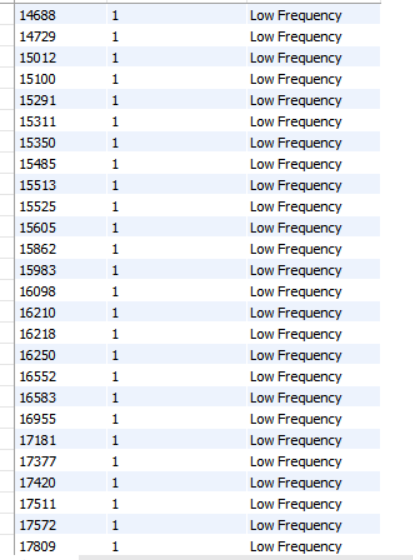
GROUP BY

Customer ID

ORDER BY

Purchase Frequency DESC;





.

.

.

.

.

.

.

.

.

.



advanced\_query\_2. Average Order Value by Country

-- Calculate the average order value for each country to identify where your most

-- valuable customers are located.

SELECT

Country,

AVG(Order Value) AS Average Order Value

FROM

(

SELECT

Invoice No,

Country,

SUM(Quantity \* Unit Price) AS Order Value

FROM

online\_retail

GROUP BY

Invoice No, Country

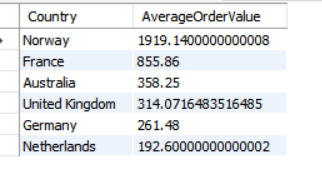
) AS order\_values

GROUP BY

Country

ORDER BY

Average Order Value DESC;



advanced\_query\_3. Customer Churn Analysis

-- Identify customers who haven't made a purchase in a specific period

-- (e.g., last 6 months) to assess churn.

SELECT

Customer ID,

MAX(Invoice Date) AS Last Purchase Date

FROM

online\_retail

GROUP BY

Customer ID

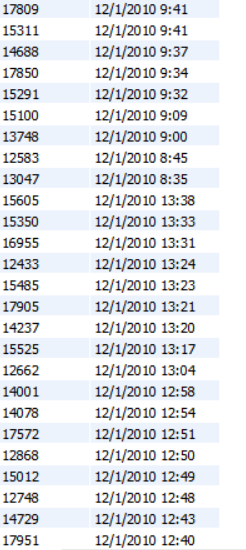
HAVING

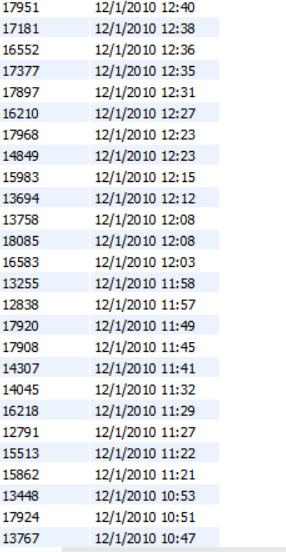
MAX(Invoice Date) < DATE\_SUB(CURDATE(), INTERVAL 6 MONTH)

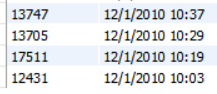
ORDER BY

Last Purchase Date DESC;









advanced\_query\_4. Product Affinity Analysis

-- Determine which products are often purchased together by calculating the

-- correlation between product purchases.

SELECT

p1.Stock Code AS Product1,

p2.Stock Code AS Product2,

COUNT(\*) AS Frequency

FROM

online\_retail p1

JOIN

online\_retail p2

ON

p1.Invoice No = p2.InvoiceNo

AND

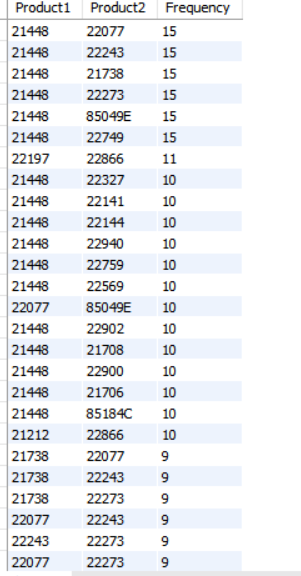
p1.Stock Code < p2.StockCode

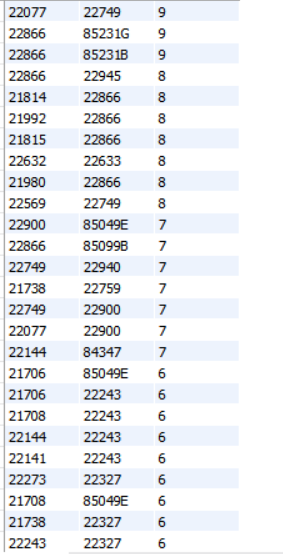
GROUP BY

p1.Stock Code, p2.StockCode

ORDER BY

Frequency DESC;





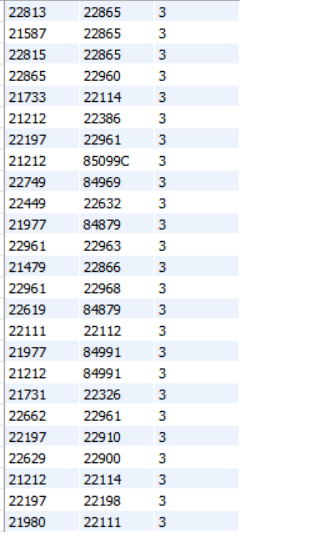
.

.

.

.

.



advanced\_query\_5. Time-based Analysis

-- Explore trends in customer behavior over time, such as monthly or quarterly sales patterns.

--

set SQL\_SAFE\_UPDATEs = 0;

UPDATE online\_ retail

SET Invoice Date = STR\_TO\_DATE(Invoice Date, '%m/%d/%Y %H:%i')

WHERE Invoice Date IS NOT NULL;

SELECT

DATE\_FORMAT(Invoice Date, '%Y-%m') AS Month,

SUM( Quantity \* Unit Price) AS Total Sales,

COUNT( DISTINCT Invoice No ) AS Number Of Orders

FROM

Online\_ Retail

GROUP BY

DATE\_FORMAT(Invoice Date, '%Y-%m')

ORDER BY

Month ASC; 

THE END PROJECT