Design Laboratory (CS69202)
Spring Semester 2023

Topic: SQL

Date: January 24, 2024

Time: 2.30 PM - 4.00 PM

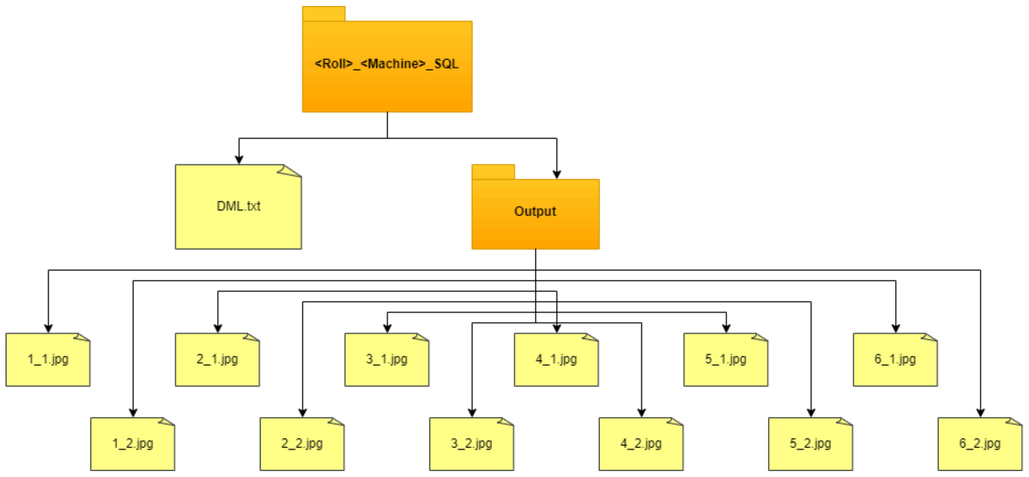
Instructions:

1. There will be no internet access during the exam. Make sure to download any required materials, SQL documentation, or reference guides beforehand.
2. No communication is allowed between students once the exam has begun. Any form of communication will be considered a violation of exam rules.
3. If you have any questions or need clarification on a particular question, please raise your hand, and the TA will assist you.
4. Manage your time wisely. Be aware of the allocated time for of the exam to ensure you have sufficient time to complete all questions.
5. We will be using the same database as the previous one. Please make sure the tables and data are in place.

Submission Format:

Submit the following files:

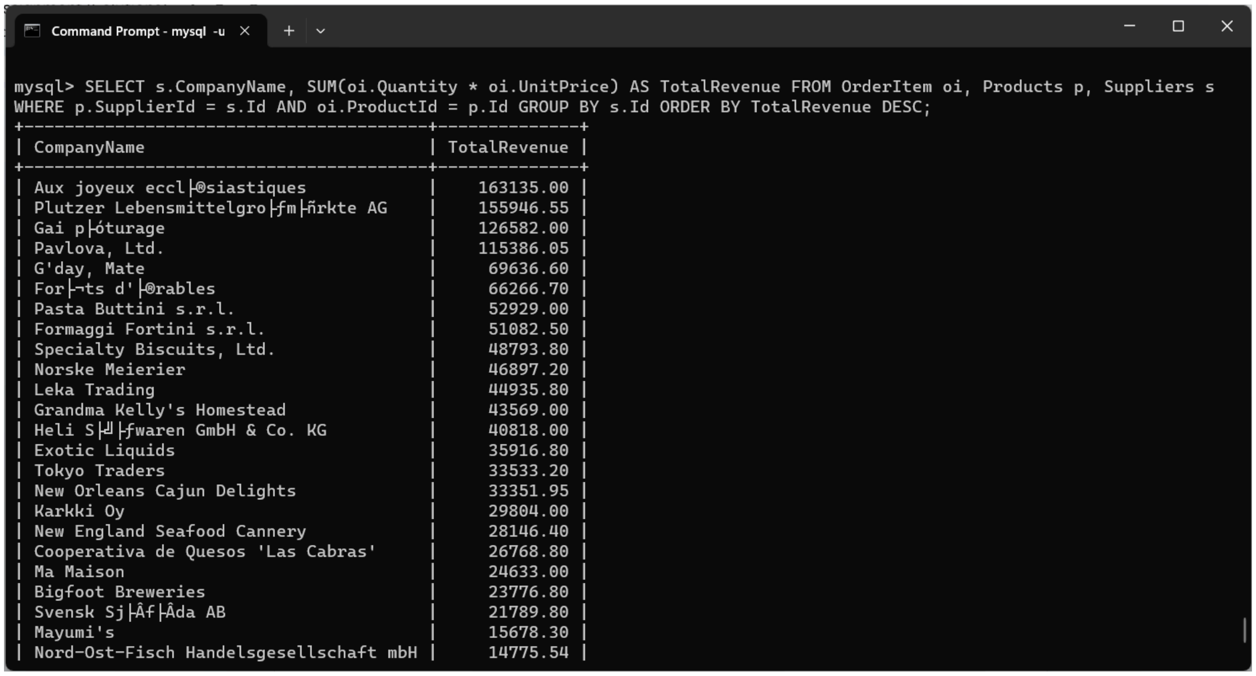
- The root folder name should be "<Roll>_<Machine Id>_SQL". For example, if your Roll is 22CS60R01 and you are giving the exam on machine number 20, then the root folder name should be 22CS60R01_20_SQL
- A text file containing SQL queries for all six questions (DML.txt)
- For each query
    - Screenshot of output head that shows query as well (<Ques No>_1.jpg)
    - Screenshot of output tail that shows number of rows (<Ques No>_2.jpg)
- Directory Structure:



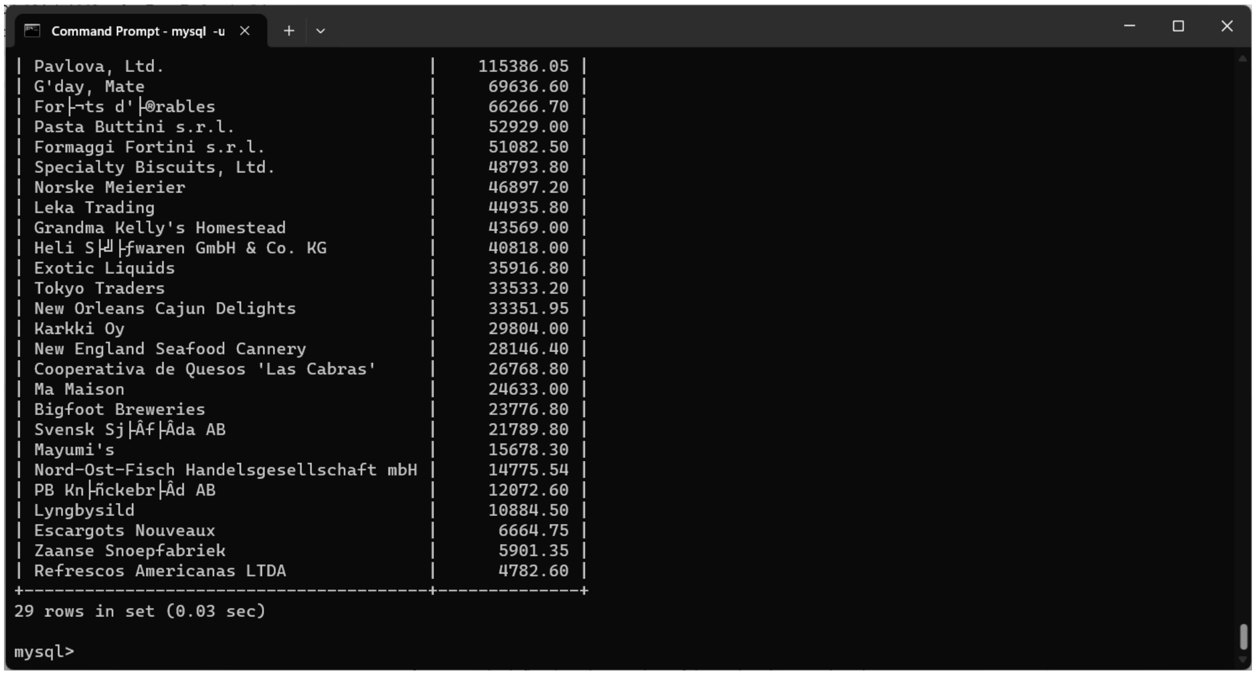
<u>Questions:</u>

1. **Calculate the total revenue for each supplier ordered by total revenue in descending order.**

SELECT s.CompanyName, SUM(oi.Quantity * oi.UnitPrice) AS TotalRevenue
FROM OrderItem oi, Products p, Suppliers s
WHERE p.SupplierId = s.Id AND oi.ProductId = p.Id
GROUP BY s.Id
ORDER BY TotalRevenue DESC;



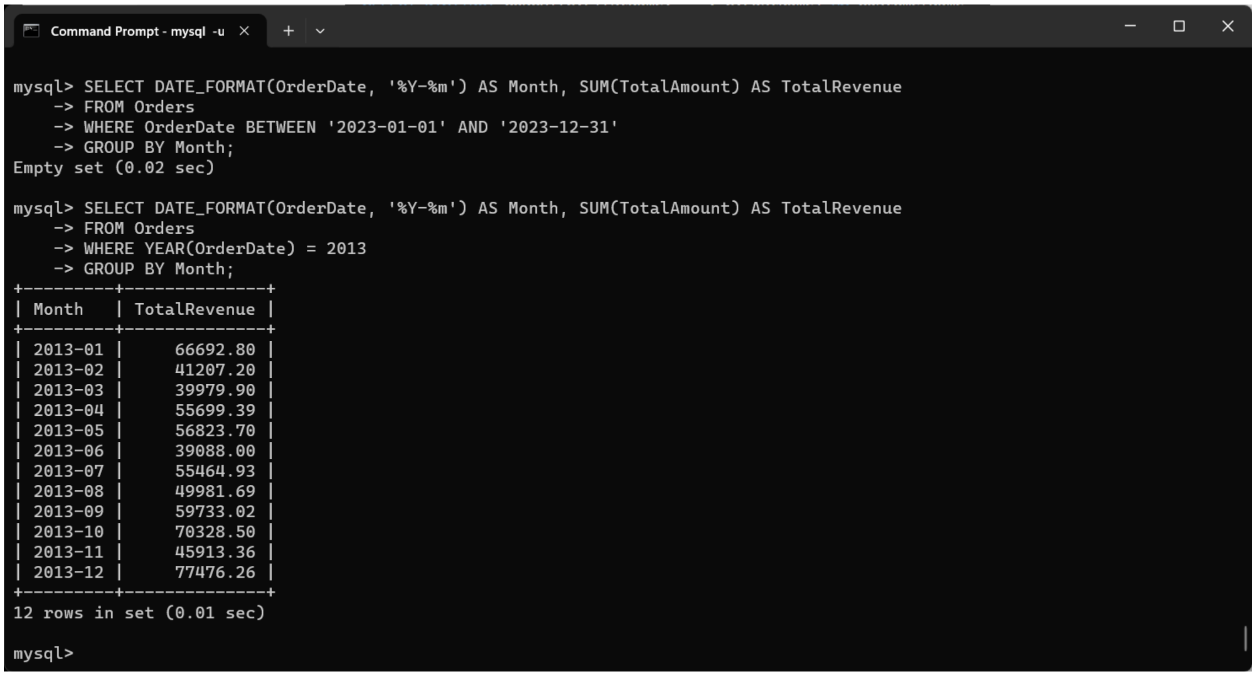
```
mysql> SELECT s.CompanyName, SUM(oi.Quantity * oi.UnitPrice) AS TotalRevenue FROM OrderItem oi, Products p, Suppliers s
WHERE p.SupplierId = s.Id AND oi.ProductId = p.Id GROUP BY s.Id ORDER BY TotalRevenue DESC;
+----------------------------------------+--------------+
| CompanyName                            | TotalRevenue |
+----------------------------------------+--------------+
| Aux joyeux eccl├®siastiques             |    163135.00 |
| Plutzer Lebensmittelgro├ƒm├ñrkte AG     |    155946.55 |
| Gai p├óturage                          |    126582.00 |
| Pavlova, Ltd.                          |    115386.05 |
| G'day, Mate                            |     69636.60 |
| For├¬ts d'├®rables                     |     66266.70 |
| Pasta Buttini s.r.l.                   |     52929.00 |
| Formaggi Fortini s.r.l.                |     51082.50 |
| Specialty Biscuits, Ltd.               |     48793.80 |
| Norske Meierier                        |     46897.20 |
| Leka Trading                           |     44935.80 |
| Grandma Kelly's Homestead              |     43569.00 |
| Heli S├┘├ƒwaren GmbH & Co. KG         |     40818.00 |
| Exotic Liquids                         |     35916.80 |
| Tokyo Traders                          |     33533.20 |
| New Orleans Cajun Delights             |     33351.95 |
| Karkki Oy                              |     29804.00 |
| New England Seafood Cannery            |     28146.40 |
| Cooperativa de Quesos 'Las Cabras'     |     26768.80 |
| Ma Maison                              |     24633.00 |
| Bigfoot Breweries                      |     23776.80 |
| Svensk Sj├Âf├Âda AB                   |     21789.80 |
| Mayumi's                               |     15678.30 |
| Nord-Ost-Fisch Handelsgesellschaft mbH |     14775.54 |
```



```
| Pavlova, Ltd.                          |    115386.05 |
| G'day, Mate                            |     69636.60 |
| For├¬ts d'├®rables                      |     66266.70 |
| Pasta Buttini s.r.l.                   |     52929.00 |
| Formaggi Fortini s.r.l.                |     51082.50 |
| Specialty Biscuits, Ltd.               |     48793.80 |
| Norske Meierier                        |     46897.20 |
| Leka Trading                           |     44935.80 |
| Grandma Kelly's Homestead              |     43569.00 |
| Heli S├╝├ƒwaren GmbH & Co. KG           |     40818.00 |
| Exotic Liquids                         |     35916.80 |
| Tokyo Traders                          |     33533.20 |
| New Orleans Cajun Delights             |     33351.95 |
| Karkki Oy                              |     29804.00 |
| New England Seafood Cannery            |     28146.40 |
| Cooperativa de Quesos 'Las Cabras'     |     26768.80 |
| Ma Maison                              |     24633.00 |
| Bigfoot Breweries                      |     23776.80 |
| Svensk Sj├Âf├Âda AB                     |     21789.80 |
| Mayumi's                               |     15678.30 |
| Nord-Ost-Fisch Handelsgesellschaft mbH |     14775.54 |
| PB Kn├ñckebr├Âd AB                      |     12072.60 |
| Lyngbysild                             |     10884.50 |
| Escargots Nouveaux                     |      6664.75 |
| Zaanse Snoepfabriek                    |      5901.35 |
| Refrescos Americanas LTDA              |      4782.60 |
+----------------------------------------+--------------+
29 rows in set (0.03 sec)

mysql>
```

2. **Calculate the total revenue for each month in 2013.**

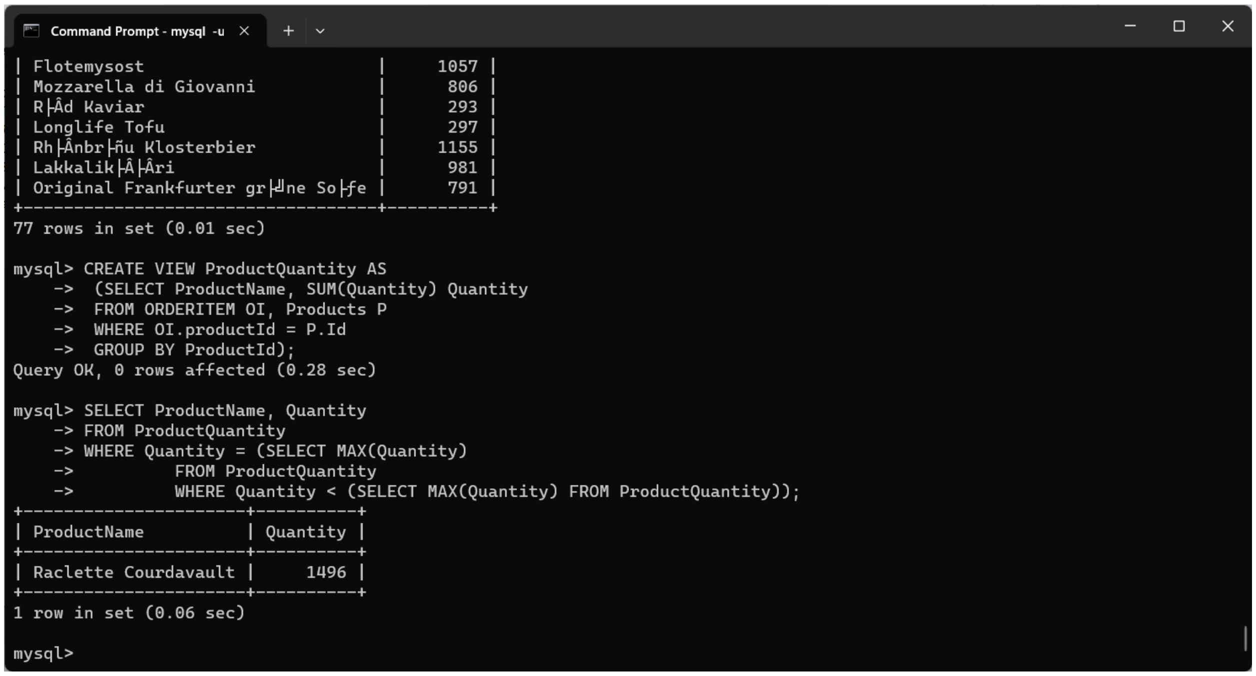
SELECT DATE_FORMAT(OrderDate, '%Y-%m') AS Month, SUM(TotalAmount) AS TotalRevenue
FROM Orders
WHERE YEAR(OrderDate) = 2013
GROUP BY Month;



```
mysql> SELECT DATE_FORMAT(OrderDate, '%Y-%m') AS Month, SUM(TotalAmount) AS TotalRevenue
    -> FROM Orders
    -> WHERE OrderDate BETWEEN '2023-01-01' AND '2023-12-31'
    -> GROUP BY Month;
Empty set (0.02 sec)

mysql> SELECT DATE_FORMAT(OrderDate, '%Y-%m') AS Month, SUM(TotalAmount) AS TotalRevenue
    -> FROM Orders
    -> WHERE YEAR(OrderDate) = 2013
    -> GROUP BY Month;
+---------+--------------+
| Month   | TotalRevenue |
+---------+--------------+
| 2013-01 |     66692.80 |
| 2013-02 |     41207.20 |
| 2013-03 |     39979.90 |
| 2013-04 |     55699.39 |
| 2013-05 |     56823.70 |
| 2013-06 |     39088.00 |
| 2013-07 |     55464.93 |
| 2013-08 |     49981.69 |
| 2013-09 |     59733.02 |
| 2013-10 |     70328.50 |
| 2013-11 |     45913.36 |
| 2013-12 |     77476.26 |
+---------+--------------+
12 rows in set (0.01 sec)

mysql>
```

3. **Find out second most popular Product in terms of Quantity ordered. List out Product Name and Product Id. Assume total quantity ordered for TOP 10 items is unique.**
   Note: ‘LIMIT’ cannot be used for the query, 0 marks will be awarded on violation.

```
CREATE VIEW ProductQuantity AS
        (SELECT ProductName, SUM(Quantity) Quantity
        FROM ORDERITEM OI, Products P
        WHERE OI.productId = P.Id
        GROUP BY ProductId);
SELECT ProductName, Quantity
FROM ProductQuantity
WHERE Quantity = (SELECT MAX(Quantity)
                FROM ProductQuantity
                WHERE Quantity < (SELECT MAX(Quantity) FROM ProductQuantity));
```

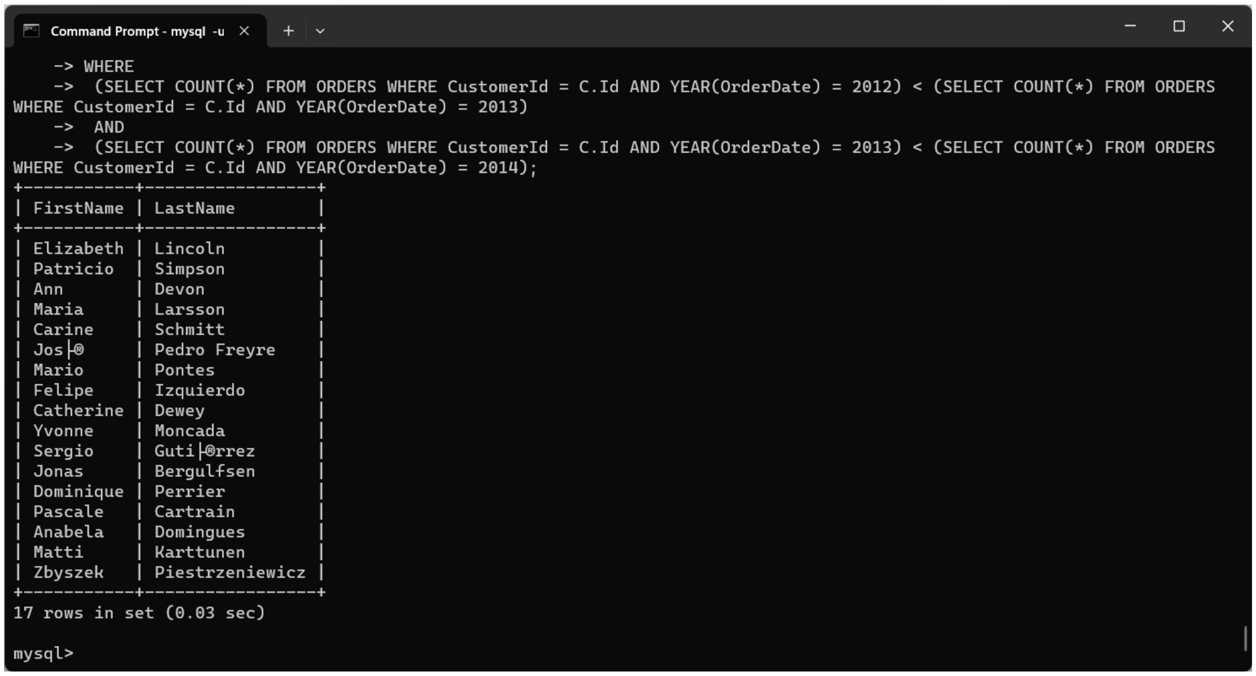


```
| Flotemysost                        |     1057 |
| Mozzarella di Giovanni             |      806 |
| R├Âd Kaviar                        |      293 |
| Longlife Tofu                      |      297 |
| Rh├Ânbr├ñu Klosterbier             |     1155 |
| Lakkalik├Â├Âri                     |      981 |
| Original Frankfurter gr├╝ne So├ƒe |      791 |
+------------------------------------+----------+
77 rows in set (0.01 sec)

mysql> CREATE VIEW ProductQuantity AS
    ->  (SELECT ProductName, SUM(Quantity) Quantity
    ->  FROM ORDERITEM OI, Products P
    ->  WHERE OI.productId = P.Id
    ->  GROUP BY ProductId);
Query OK, 0 rows affected (0.28 sec)

mysql> SELECT ProductName, Quantity
    -> FROM ProductQuantity
    -> WHERE Quantity = (SELECT MAX(Quantity)
    ->          FROM ProductQuantity
    ->          WHERE Quantity < (SELECT MAX(Quantity) FROM ProductQuantity));
+----------------------+----------+
| ProductName          | Quantity |
+----------------------+----------+
| Raclette Courdavault |     1496 |
+----------------------+----------+
1 row in set (0.06 sec)

mysql>
```

4. **Identify customers who have consistently increased their order frequency each year (2012, 2013, 2014) to identify our loyal customers. Order frequency is defined as the number of times they have ordered (independent of quantities).**

```
SELECT FirstName, LastName
FROM CUSTOMERS C
WHERE
```

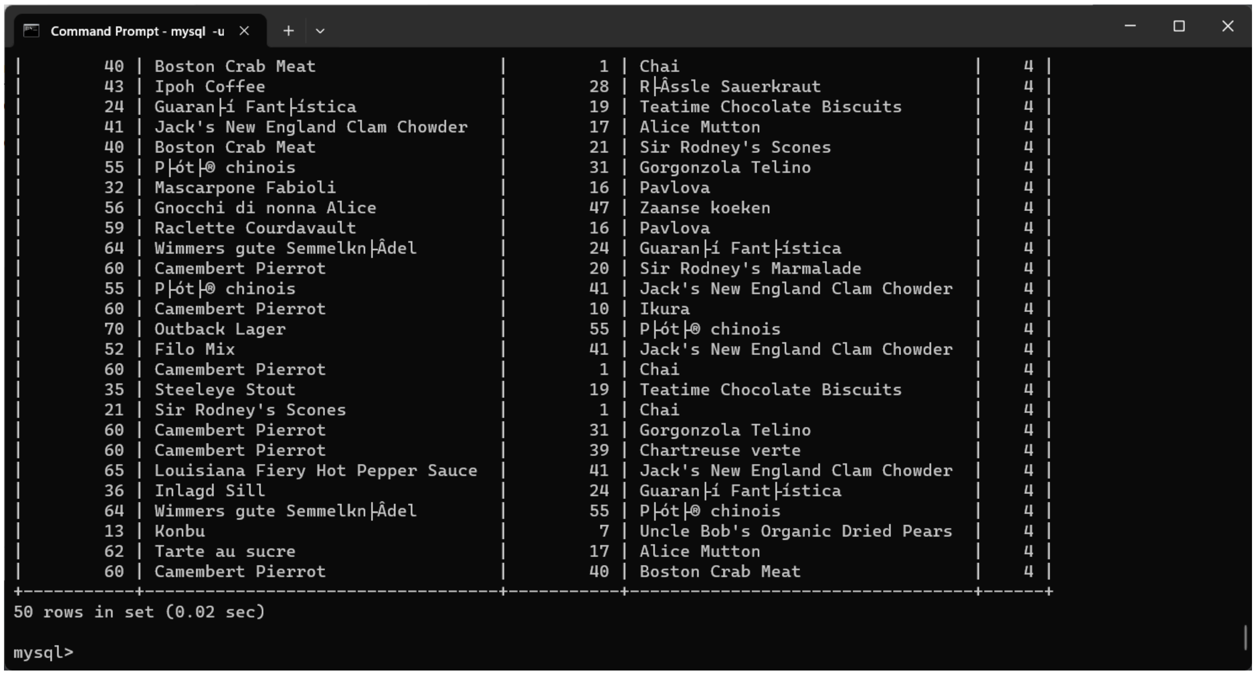
(SELECT COUNT(*) FROM ORDERS WHERE CustomerId = C.Id AND YEAR(OrderDate) = 2012) < (SELECT COUNT(*) FROM ORDERS WHERE CustomerId = C.Id AND YEAR(OrderDate) = 2013)
AND
(SELECT COUNT(*) FROM ORDERS WHERE CustomerId = C.Id AND YEAR(OrderDate) = 2013) < (SELECT COUNT(*) FROM ORDERS WHERE CustomerId = C.Id AND YEAR(OrderDate) = 2014);



```
    -> WHERE
    ->  (SELECT COUNT(*) FROM ORDERS WHERE CustomerId = C.Id AND YEAR(OrderDate) = 2012) < (SELECT COUNT(*) FROM ORDERS
WHERE CustomerId = C.Id AND YEAR(OrderDate) = 2013)
    ->  AND
    ->  (SELECT COUNT(*) FROM ORDERS WHERE CustomerId = C.Id AND YEAR(OrderDate) = 2013) < (SELECT COUNT(*) FROM ORDERS
WHERE CustomerId = C.Id AND YEAR(OrderDate) = 2014);
+-----------+-----------------+
| FirstName | LastName        |
+-----------+-----------------+
| Elizabeth | Lincoln         |
| Patricio  | Simpson         |
| Ann       | Devon           |
| Maria     | Larsson         |
| Carine    | Schmitt         |
| Jos├®      | Pedro Freyre    |
| Mario     | Pontes          |
| Felipe    | Izquierdo       |
| Catherine | Dewey           |
| Yvonne    | Moncada         |
| Sergio    | Guti├®rrez       |
| Jonas     | Bergulfsen      |
| Dominique | Perrier         |
| Pascale   | Cartrain        |
| Anabela   | Domingues       |
| Matti     | Karttunen       |
| Zbyszek   | Piestrzeniewicz |
+-----------+-----------------+
17 rows in set (0.03 sec)

mysql>
```

5. **Identify opportunities for cross-selling and up-selling based on product relationships. List out the top 50 product pairs that sold together sold together (One bucket). List out Product Names and their id along with the frequency ordered by Frequency**.

SELECT O1.ProductId, P1.ProductName, O2.ProductId, P2.ProductName, COUNT(O1.OrderId) Freq
FROM OrderItem O1, OrderItem O2, Products P1, Products P2
WHERE O1.OrderId = O2.OrderId AND O1.ProductId > O2.ProductId AND P1.Id = O1.ProductId AND P2.Id = O2.ProductId
GROUP BY O1.ProductId, O2.ProductId
ORDER BY Freq DESC LIMIT 5;
* Potential Mistake



```
|         40 | Boston Crab Meat                  |          1 | Chai                              |    4 |
|         43 | Ipoh Coffee                       |         28 | R├Âssle Sauerkraut                 |    4 |
|         24 | Guaran├í Fant├ística              |         19 | Teatime Chocolate Biscuits        |    4 |
|         41 | Jack's New England Clam Chowder   |         17 | Alice Mutton                      |    4 |
|         40 | Boston Crab Meat                  |         21 | Sir Rodney's Scones               |    4 |
|         55 | P├ót├® chinois                    |         31 | Gorgonzola Telino                 |    4 |
|         32 | Mascarpone Fabioli                |         16 | Pavlova                           |    4 |
|         56 | Gnocchi di nonna Alice            |         47 | Zaanse koeken                     |    4 |
|         59 | Raclette Courdavault              |         16 | Pavlova                           |    4 |
|         64 | Wimmers gute Semmelkn├Âdel        |         24 | Guaran├í Fant├ística              |    4 |
|         60 | Camembert Pierrot                 |         20 | Sir Rodney's Marmalade            |    4 |
|         55 | P├ót├® chinois                    |         41 | Jack's New England Clam Chowder   |    4 |
|         60 | Camembert Pierrot                 |         10 | Ikura                             |    4 |
|         70 | Outback Lager                     |         55 | P├ót├® chinois                    |    4 |
|         52 | Filo Mix                          |         41 | Jack's New England Clam Chowder   |    4 |
|         60 | Camembert Pierrot                 |          1 | Chai                              |    4 |
|         35 | Steeleye Stout                    |         19 | Teatime Chocolate Biscuits        |    4 |
|         21 | Sir Rodney's Scones               |          1 | Chai                              |    4 |
|         60 | Camembert Pierrot                 |         31 | Gorgonzola Telino                 |    4 |
|         60 | Camembert Pierrot                 |         39 | Chartreuse verte                  |    4 |
|         65 | Louisiana Fiery Hot Pepper Sauce  |         41 | Jack's New England Clam Chowder   |    4 |
|         36 | Inlagd Sill                       |         24 | Guaran├í Fant├ística              |    4 |
|         64 | Wimmers gute Semmelkn├Âdel        |         55 | P├ót├® chinois                    |    4 |
|         13 | Konbu                             |          7 | Uncle Bob's Organic Dried Pears   |    4 |
|         62 | Tarte au sucre                    |         17 | Alice Mutton                      |    4 |
|         60 | Camembert Pierrot                 |         40 | Boston Crab Meat                  |    4 |
+------------+-----------------------------------+------------+-----------------------------------+------+
50 rows in set (0.02 sec)

mysql>
```

6. **Detect potential fraud by identifying orders with unusually very high quantities. List the Product ID, Product Name, Customer Name, and Quantity Ordered. (Assuming orders with quantities that are greater than Fourth Standard Deviations are potential fraud for each product.) We understand that some products may be ordered in high quantities and others in low quantities. So, fraud needs to be calculated for each product's stats.**

   For each product P1, list the order of O1; let P1's mean order quantity is μ_{P_1}, and the standard deviation is σ_{P_1}:
   List out an order if:

   [Quantity Ordered In O1 of P1] > $\mu_{P_1} + 4 \times \sigma_{P_1}$

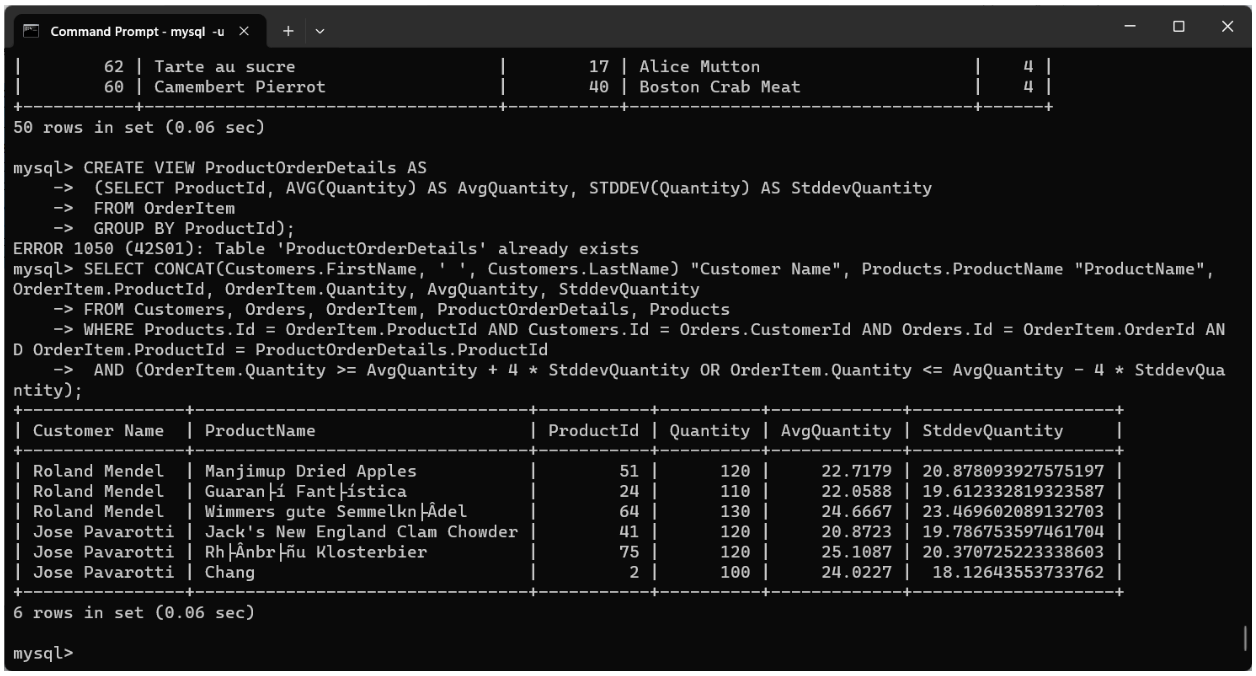
   For example:

Id	Product	Quantity Ordered	Customer Id
1	Chai Pac	100	-
2	Chai Pac	110	-
3	Ikura	1	-
4	Ikura	1	-
……………………………………..			
17	Ikura	1	-

18	Ikura	100	

Here, the order with ID 18 is the only outlier. Whereas orders with ID 1 and 2 are not outliers. This is just an example and not related to the actual answer.

```
CREATE VIEW ProductOrderDetails AS
	(SELECT ProductId, AVG(Quantity) AS AvgQuantity, STDDEV(Quantity) AS StddevQuantity
	FROM OrderItem
	GROUP BY ProductId);
SELECT CONCAT(Customers.FirstName, ' ', Customers.LastName) "Customer Name", Products.ProductName "ProductName", OrderItem.ProductId, OrderItem.Quantity, AvgQuantity, StddevQuantity
FROM Customers, Orders, OrderItem, ProductOrderDetails, Products
WHERE Products.Id = OrderItem.ProductId AND Customers.Id = Orders.CustomerId AND Orders.Id = OrderItem.OrderId AND OrderItem.ProductId = ProductOrderDetails.ProductId
	AND OrderItem.Quantity >= AvgQuantity + 4 * StddevQuantity;
```



```
|       62 | Tarte au sucre                     |        17 | Alice Mutton                      |    4 |
|       60 | Camembert Pierrot                  |        40 | Boston Crab Meat                  |    4 |
+----------+------------------------------------+-----------+-----------------------------------+------+
50 rows in set (0.06 sec)

mysql> CREATE VIEW ProductOrderDetails AS
    ->  (SELECT ProductId, AVG(Quantity) AS AvgQuantity, STDDEV(Quantity) AS StddevQuantity
    ->  FROM OrderItem
    ->  GROUP BY ProductId);
ERROR 1050 (42S01): Table 'ProductOrderDetails' already exists
mysql> SELECT CONCAT(Customers.FirstName, ' ', Customers.LastName) "Customer Name", Products.ProductName "ProductName",
OrderItem.ProductId, OrderItem.Quantity, AvgQuantity, StddevQuantity
    -> FROM Customers, Orders, OrderItem, ProductOrderDetails, Products
    -> WHERE Products.Id = OrderItem.ProductId AND Customers.Id = Orders.CustomerId AND Orders.Id = OrderItem.OrderId AN
D OrderItem.ProductId = ProductOrderDetails.ProductId
    ->  AND (OrderItem.Quantity >= AvgQuantity + 4 * StddevQuantity OR OrderItem.Quantity <= AvgQuantity - 4 * StddevQua
ntity);
+----------------+----------------------------------+-----------+----------+-------------+--------------------+
| Customer Name  | ProductName                      | ProductId | Quantity | AvgQuantity | StddevQuantity     |
+----------------+----------------------------------+-----------+----------+-------------+--------------------+
| Roland Mendel  | Manjimup Dried Apples            |        51 |      120 |     22.7179 | 20.878093927575197 |
| Roland Mendel  | Guaran├í Fant├ística               |        24 |      110 |     22.0588 | 19.612332819323587 |
| Roland Mendel  | Wimmers gute Semmelkn├Âdel         |        64 |      130 |     24.6667 | 23.469602089132703 |
| Jose Pavarotti | Jack's New England Clam Chowder  |        41 |      120 |     20.8723 | 19.786753597461704 |
| Jose Pavarotti | Rh├Ânbr├ñu Klosterbier             |        75 |      120 |     25.1087 | 20.370725223338603 |
| Jose Pavarotti | Chang                            |         2 |      100 |     24.0227 |  18.12643553733762 |
+----------------+----------------------------------+-----------+----------+-------------+--------------------+
6 rows in set (0.06 sec)

mysql>
```