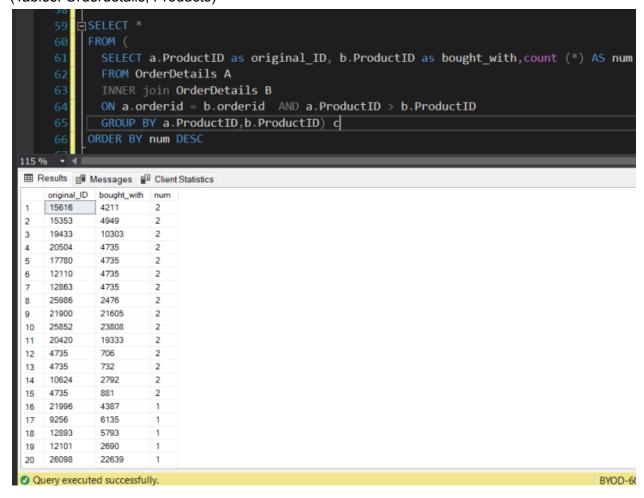
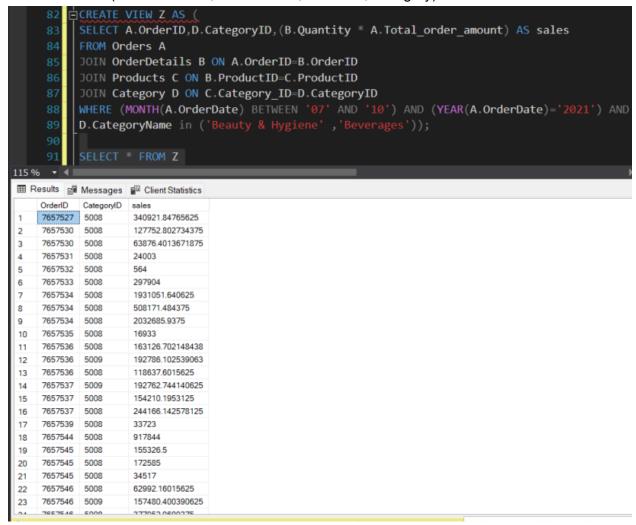
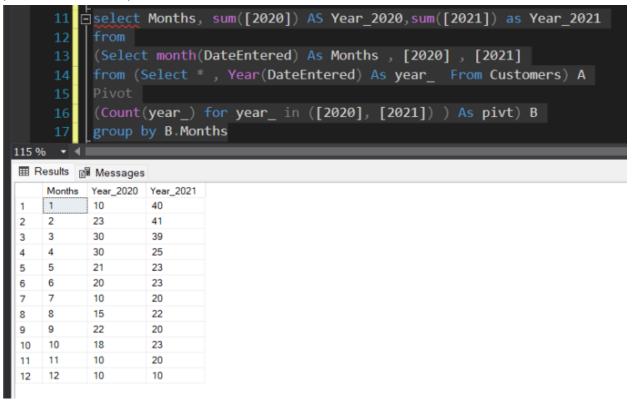
Q1. We are trying to find paired products that are often purchased together by the same user, such as chips and soft drinks, milk and curd etc.. Find the top paired products names. (Tables: Orderdetails, Products)



Q2. We want to understand the impact of running a campaign during July'21-Oct'21 what was the total sales generated for the categories "Beauty & Hygiene" and "Bevarages" by entire customer base. (Tables: Orders, Orderdetails, Products, Category)

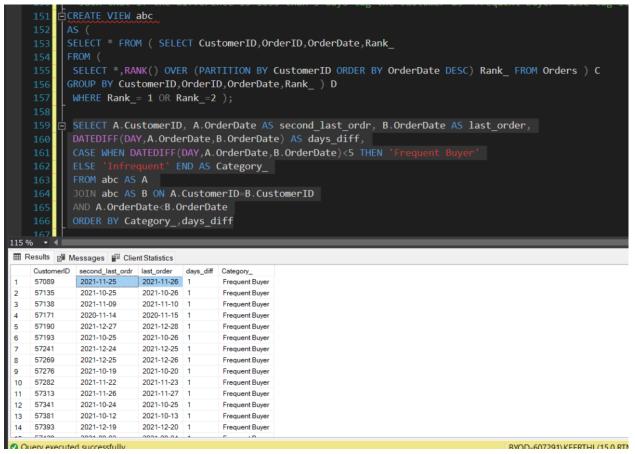


Q3. Create a YOY analysis for the count of customers enrolled with the company each month. (Tables: Customers)



Q4. Find out the difference between the last two order dates for each of the customers and categorize the customers in two categories such that if the difference is less than 5 days tag the customer as "Frequent Buyer" else tag it as "Infrequent".

(Tables: Orders)

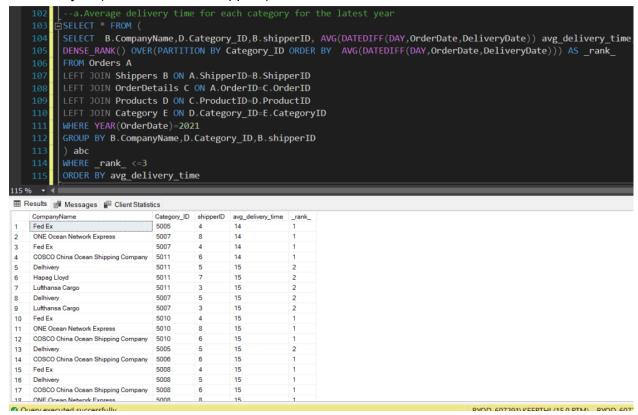


Q5. Create a Procedure to filter the orders from Orders table where total purchase amount in a each order id < @x and purchase of year is @Y where @x and @y are the inputs provided by the user.

(Tables: Orders)

```
CREATE PROCEDURE sp proc name(@x FLOAT,@Y DATE)
    275
             AS
    276
             BEGIN
    277
    278
                    SELECT *
    279
                    FROM
                        Orders
    281
                   WHERE
                        total order amount<@x AND
    282
    283
                        orderdate = @Y
    284
             END;
             EXEC sp proc name 522230, '2022-09-22';
    285
    287
Messages Client Statistics
  Commands completed successfully.
  Completion time: 2022-03-11T23:27:03.5481015+05:30
```

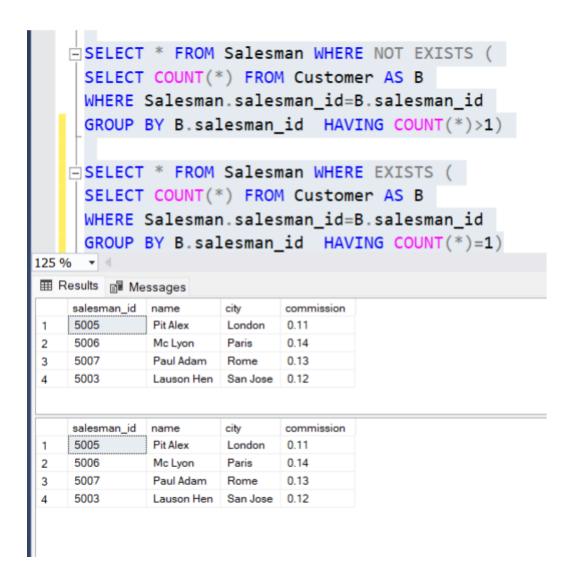
Q6. Find the top 3 Shipper companies in terms of average delivery time for each category for the latest year(Tables: Orders, Shippers)



Q7. Write a SQL query to find the salespeople who deal with a single customer. Return salesman_id, name, city and commission. (Tables: Salesman, Customer)

Answer-

Approach 1:

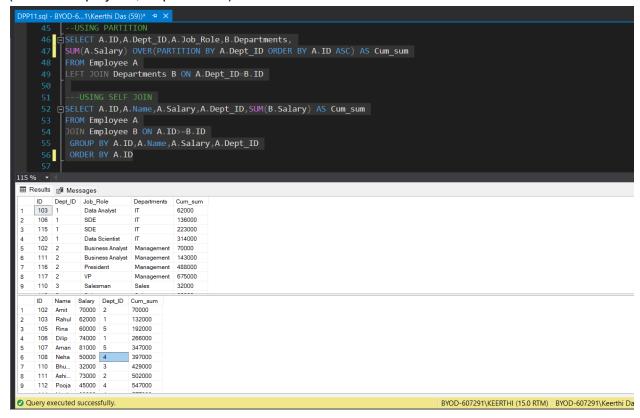


Approach 2:

(Tables: Salesman, Customer)

```
LQuery∠.sqr - BY...T\Keertini Das (ככ))^ → 🔨
 ⇒SELECT *
   FROM salesman
   WHERE salesman_id IN (
       SELECT DISTINCT salesman id
       FROM customer a
       WHERE NOT EXISTS (
           SELECT * FROM customer b
           WHERE a.salesman_id=b.salesman_id
           AND a.cust name <> b.cust name));
Results Messages
  salesman_id name
                             commission
                     city
  5005
            Pit Alex
                     London
                             0.11
  5006
                             0.14
            Mc Lyon
                     Paris
  5007
            Paul Adam
                     Rome
                             0.13
  5003
            Lauson Hen San Jose 0.12
```

Q8. Find the cumulative sum using Partition by and self join. (Tables: Employees, Departments)



Q9. Find all the owners whose pet's name does not begin from letters 'D' to 'S' (Tables: Owners, Pets)

