Transform : Intermediate Query

1. Tulis query untuk mendapatkan jumlah customer tiap bulan yang melakukan order pada tahun 1997.

```
SELECT MONTH(OrderDate) as bulan,
count(CustomerID) as Total
From Orders
where Orders.OrderDate between ('1997-01-01') and ('1997-12-31')
Group by MONTH(OrderDate)
Order by MONTH(OrderDate)
```

2. Tulis query untuk mendapatkan nama employee yang termasuk Sales Representative.

```
SELECT FirstName,LastName,Title
FROM Employees
where Title = 'Sales Representative'
```

3. Tulis query untuk mendapatkan top 5 nama produk yang quantitynya paling banyak diorder pada bulan Januari 1997.

```
select TOP(5)
    a.ProductName,
   b.Quantity
from Products as a
inner join OrderDetails as b on a.ProductID = b.ProductID
inner join Orders as c on b.OrderID=c.OrderID
where c.OrderDate between ('1997-01-01') and ('1997-01-31')
order by b.Quantity DESC
```

4. Tulis query untuk mendapatkan nama company yang melakukan order Chai pada bulan Juni 1997 **(no result)**

```
Select
c.ProductName,
d.CompanyName
FROM Orders as a
inner join OrderDetails as b on a.OrderID=b.OrderID
inner join Products as c on b.ProductID=c.ProductID
inner join Customers as d on a.CustomerID=d.CustomerID
Where c.ProductName='Chia' and a.OrderDate between ('1997-06-01') and ('1997-06-
30')
```

5. Tulis query untuk mendapatkan jumlah OrderID yang pernah melakukan pembelian (unit_price dikali quantity) <=100, 100<x<=250, 250<x<=500, dan >500.

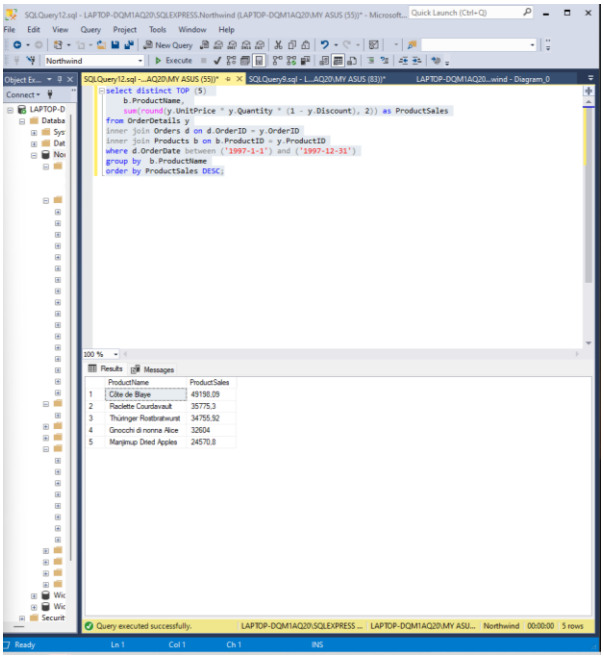
```
select distinct
x.OrderID,
round(x.UnitPrice * x.Quantity,1) as TotalPembelian

From OrderDetails as x
inner join Orders as y on x.OrderID=y.OrderID
where y.OrderDate between ('1997-01-01') and ('1997-12-31')
```

6. Tulis query untuk mendapatkan Company name pada tabel customer yang melakukan pembelian di atas 500 pada tahun 1997

```
select
a.CompanyName,
round(b.UnitPrice*b.Quantity,2) as Pembelian
From Customers as a
inner join Orders as c on a.CustomerID=c.CustomerID
inner join OrderDetails as b on c.OrderID=b.OrderID
where round(b.UnitPrice*b.Quantity,2) >500 and c.OrderDate between ('1997-01-01')
and ('1997-12-31')
```

7. Tulis query untuk mendapatkan nama produk yang merupakan Top 5 sales tertinggi tiap bulan di tahun 1997.

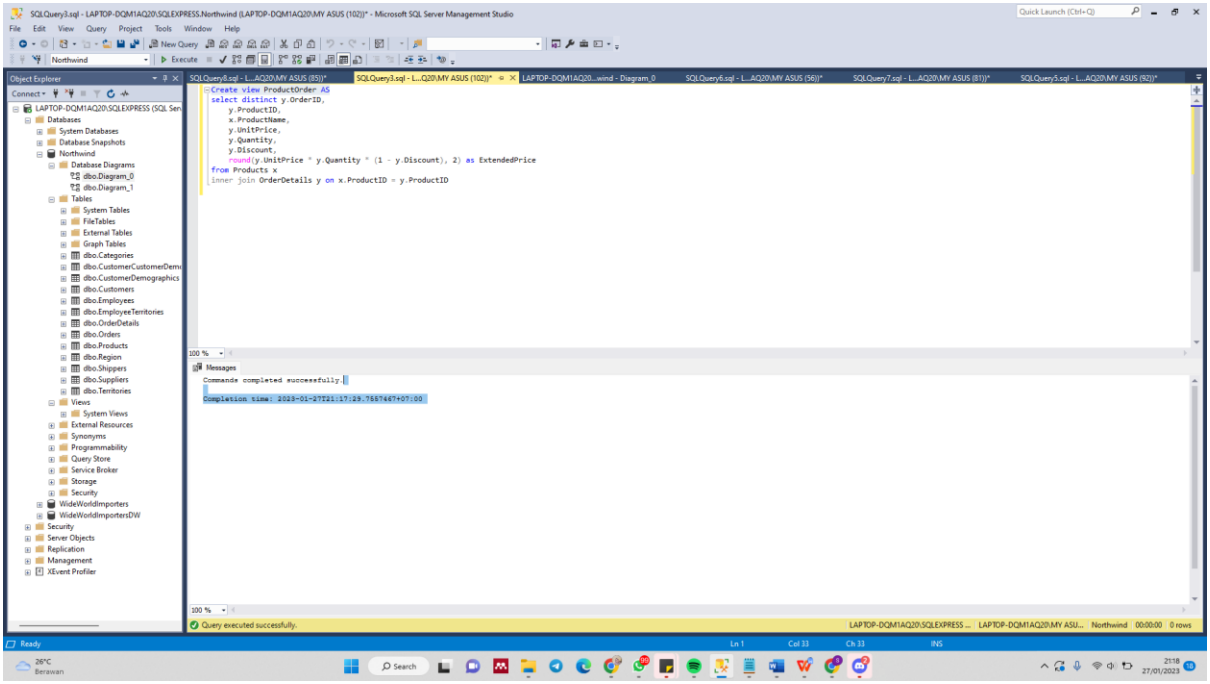
```
select distinct TOP (5)
    b.ProductName,
    sum(round(y.UnitPrice * y.Quantity * (1 - y.Discount), 2)) as ProductSales
from OrderDetails y
inner join Orders d on d.OrderID = y.OrderID
inner join Products b on b.ProductID = y.ProductID
where d.OrderDate between ('1997-1-1') and ('1997-12-31')
group by  b.ProductName
order by ProductSales DESC;
```



8. Buatlah view untuk melihat Order Details yang berisi OrderID, ProductID, ProductName, UnitPrice, Quantity, Discount, Harga setelah diskon

```
Create view ProductOrder AS
select distinct y.OrderID,
    y.ProductID,
    x.ProductName,
    y.UnitPrice,
    y.Quantity,
    y.Discount,
    round(y.UnitPrice * y.Quantity * (1 - y.Discount), 2) as ExtendedPrice
from Products x
```

```
inner join OrderDetails y on x.ProductID = y.ProductID
```



9. Buatlah procedure Invoice untuk memanggil CustomerID, CustomerName/company name, OrderID, OrderDate, RequiredDate, ShippedDate jika terdapat inputan CustomerID tertentu.

```
select  a.CustomerID,
    b.CompanyName,
    a.OrderID,
    a.OrderDate,
    a.RequiredDate,
    a.ShippedDate
from Orders a
inner join Customers b on a.CustomerID = b.CustomerID
```

Transform : Case Study
Product Analysis

- Untuk mendapatkan Nama product dengan jumlah penjualan produknya

```
select distinct
    b.ProductName,
    sum(round(y.UnitPrice * y.Quantity * (1 - y.Discount), 2)) as ProductSales
from OrderDetails y
inner join Orders d on d.OrderID = y.OrderID
inner join Products b on b.ProductID = y.ProductID
group by  b.ProductName
order by ProductSales DESC;
```

- Hubungan jumlah discount dengan jumlah order

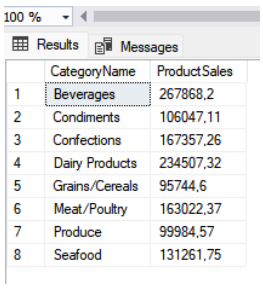
```
SELECT SUM(OD.Quantity) AS Total_Items_Sold, Count(OD.OrderID)AS Number_Of_Orders,
OD.Discount AS Discount_Percentage
FROM [OrderDetails] as OD
WHERE OD.Discount = 0.05 or OD.Discount =0.1 or OD.Discount =0.2 or OD.Discount =0.25
GROUP BY OD.Discount
ORDER BY OD.Discount DESC;
```

- Membandingkan jumlah barang yang terjual Ketika ada diskon dan tidak ada discount

```
-- For Items With Discounts:

SELECT SUM(OD.Quantity) AS Total_Items_Sold, Count(OD.OrderID)
FROM [OrderDetails] as OD
WHERE OD.Discount > 0;

-- For Items Without Discounts:

SELECT SUM(OD.Quantity) AS Total_Items_Sold, Count(OD.OrderID)
FROM [OrderDetails] as OD
WHERE OD.Discount = 0;
```

- Mengetahui Jumlah penjualan berdasarkan category name

```
select distinct
    a.CategoryName,
    sum(round(y.UnitPrice * y.Quantity * (1 - y.Discount), 2)) as ProductSales
from OrderDetails y
inner join Orders d on d.OrderID = y.OrderID
inner join Products b on b.ProductID = y.ProductID
inner join Categories a on a.CategoryID = b.CategoryID
group by a.CategoryName
order by a.CategoryName,ProductSales;
```



100 %

Results | Messages

	CategoryName	ProductSales
1	Beverages	267868,2
2	Condiments	106047,11
3	Confections	167357,26
4	Dairy Products	234507,32
5	Grains/Cereals	95744,6
6	Meat/Poultry	163022,37
7	Produce	99984,57
8	Seafood	131261,75

Customer Analysis

Supplier Analysis

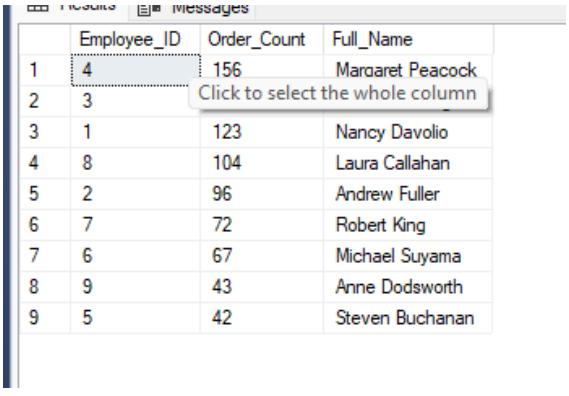
- menganalisis supplier berdasarkan kategori dan jumlah stock

```
select c.CategoryName as "Product Category",
       case when s.Country in
                 ('UK','Spain','Sweden','Germany','Norway',
                  'Denmark','Netherlands','Finland','Italy','France')
            then 'Europe'
            when s.Country in ('USA','Canada','Brazil')
            then 'America'
            else 'Asia-Pacific'
        end as "Supplier Continent",
        sum(p.UnitsInStock) as UnitsInStock
from Suppliers s
inner join Products p on p.SupplierID=s.SupplierID
inner join Categories c on c.CategoryID=p.CategoryID
group by c.CategoryName,
         case when s.Country in
                 ('UK','Spain','Sweden','Germany','Norway',
                  'Denmark','Netherlands','Finland','Italy','France')
              then 'Europe'
              when s.Country in ('USA','Canada','Brazil')
              then 'America'
              else 'Asia-Pacific'
         end;
```

Employee Analysis

- kita dapat menganalisis siapa dan title employee yang banyak berurusan dengan order

```
SELECT E.EmployeeID As Employee_ID,
COUNT(O.EmployeeID) AS Order_Count,
CONCAT(E.FirstName, ' ' ,E.LastName) As Full_Name,
E.Title
FROM Orders as O
inner join Employees as E on O.EmployeeID = E.EmployeeID
group by E.EmployeeID, E.FirstName, E.LastName,E.Title
order by Order_Count Desc
```



	Employee_ID	Order_Count	Full_Name
1	4	156	Margaret Peacock
2	3		
3	1	123	Nancy Davolio
4	8	104	Laura Callahan
5	2	96	Andrew Fuller
6	7	72	Robert King
7	6	67	Michael Suyama
8	9	43	Anne Dodsworth
9	5	42	Steven Buchanan