

Intermediate Query

Luthfi Muhammad Irsyad

DSLS Data Consultant Bootcamp

Query 1

```
--Tulis query untuk mendapatkan jumlah customer tiap bulan  
--yang melakukan order pada tahun 1997.  
SELECT MONTH(OrderDate) AS mth,  
       COUNT(CustomerID) AS total_cust,  
       COUNT(DISTINCT(CustomerID)) AS total_unq_cust  
FROM Orders ord  
WHERE YEAR(OrderDate) = 1997  
GROUP BY MONTH(OrderDate);
```

90 %

Results Messages

	mth	total_cust	total_unq_cust
1	1	33	27
2	2	29	21
3	3	30	24
4	4	31	27
5	5	32	26
6	6	30	25
7	7	33	25
8	8	33	30
9	9	37	27
10	10	38	28
11	11	34	27
12	12	48	37

Query 2

```
--Tulis query untuk mendapatkan nama  
--employee yang termasuk Sales Representative  
SELECT LastName, FirstName, Title  
FROM Employees  
WHERE Title = 'Sales Representative';
```

90 %

Results Messages

	LastName	FirstName	Title
1	Davolio	Nancy	Sales Representative
2	Leverling	Janet	Sales Representative
3	Peacock	Margaret	Sales Representative
4	Suyama	Michael	Sales Representative
5	King	Robert	Sales Representative
6	Dodsworth	Anne	Sales Representative

Query 3

```
--Tulis query untuk mendapatkan top 5 nama produk yang  
--quantitynya paling banyak diorder pada bulan Januari 1997.  
SELECT TOP 5 ProductName, SUM(Quantity) AS total_prod  
FROM [Order Details] as orde  
LEFT JOIN Products as prod  
    ON prod.ProductID = orde.ProductID  
WHERE orde.OrderID IN (SELECT OrderID  
                        FROM Orders  
                        WHERE YEAR(OrderDate) = 1997  
                        AND MONTH(OrderDate) = 1)  
GROUP BY ProductName  
ORDER BY total_prod DESC;
```

10 %

Results Messages

	ProductName	total_prod
1	Geitost	119
2	Raclette Courdavault	115
3	Gnocchi di nonna Alice	105
4	Vegie-spread	100
5	Côte de Blaye	99

Query 4

--Tulis query untuk mendapatkan nama company yang melakukan
--order Chai pada bulan Juni 1997.

```
SELECT CompanyName
FROM [Order Details] as orde
LEFT JOIN Orders as ord
    ON orde.OrderID = ord.OrderID
LEFT JOIN Customers as cus
    ON ord.CustomerID = cus.CustomerID
LEFT JOIN Products as prod
    ON prod.ProductID = orde.ProductID
WHERE orde.OrderID IN (SELECT OrderID
                        FROM Orders
                        WHERE YEAR(OrderDate) = 1997
                           AND MONTH(OrderDate) = 6)
    AND ProductName = 'Chai'
GROUP BY CompanyName;
```

90 %

Results Messages

	CompanyName
1	Tortuga Restaurante

Query 5

```
--Tulis query untuk mendapatkan jumlah OrderID yang pernah melakukan pembelian
--(unit_price dikali quantity) <=100, 100<x<=250, 250<x<=500, dan >500.
WITH A AS (
    SELECT OrderID, SUM(UnitPrice * Quantity) AS pembelian,
           CASE
               WHEN SUM(UnitPrice * Quantity) <= 100 THEN '<=100'
               WHEN SUM(UnitPrice * Quantity) > 100
                AND SUM(UnitPrice * Quantity) <= 250 THEN '100<x<=250'
               WHEN SUM(UnitPrice * Quantity) > 250
                AND SUM(UnitPrice * Quantity) <= 500 THEN '250<x<=500'
               WHEN SUM(UnitPrice * Quantity) > 500 THEN '>500'
           ELSE NULL END AS cat_pembelian
    FROM [Order Details]
    GROUP BY OrderID
)
SELECT cat_pembelian, COUNT(OrderID) AS total_order
FROM A
GROUP BY cat_pembelian;
```

90 %

Results Messages

	cat_pembelian	total_order
1	250<x<=500	118
2	100<x<=250	64
3	>500	611
4	<=100	37

Query 6

yang melakukan pembelian di atas 500 pada tahun 1997.

```

WITH A AS (
    SELECT cus.CompanyName,
           orde.OrderID,
           SUM(UnitPrice * Quantity) AS pembelian
    FROM [Order Details] AS orde
    LEFT JOIN Orders as ord
        ON orde.OrderID = ord.OrderID
    LEFT JOIN Customers as cus
        ON ord.CustomerID = cus.CustomerID
    WHERE orde.OrderID IN (SELECT OrderID
                          FROM Orders
                          WHERE YEAR(OrderDate) = 1997)
    GROUP BY cus.CompanyName, orde.OrderID
    HAVING SUM(UnitPrice * Quantity) > 500
)
SELECT CompanyName
FROM A
GROUP BY CompanyName;

```

90 %

Results Messages

	CompanyName
1	Alfreds Futterkiste
2	Antonio Moreno Taquería
3	Around the Horn
4	Berglunds snabbköp
5	Blondesddsl père et fils
6	Bólido Comidas preparadas
7	Bon app'
8	Bottom-Dollar Markets
9	B's Beverages

Query 7

```
--Tulis query untuk mendapatkan nama produk yang merupakan Top 5 sales
--tertinggi tiap bulan di tahun 1997.
WITH A AS (
    SELECT ProductName,
           SUM(Quantity) AS total_prod,
           MONTH(OrderDate) AS mth,
           YEAR(OrderDate) AS yr,
           RANK() OVER (PARTITION BY MONTH(OrderDate)
                        ORDER BY SUM(Quantity) DESC) AS Rank_
    FROM [Order Details] as orde
    LEFT JOIN Orders as ord
        ON orde.OrderID = ord.OrderID
    LEFT JOIN Products as prod
        ON prod.ProductID = orde.ProductID
    WHERE orde.OrderID IN (SELECT OrderID
                          FROM Orders
                          WHERE YEAR(OrderDate) = 1997)
    GROUP BY ProductName, MONTH(OrderDate), YEAR(OrderDate)
)
SELECT ProductName, Rank_, mth, yr
FROM A
WHERE Rank_ <= 5
ORDER BY mth, Rank_;
```

	ProductName	Rank_	mth	yr
1	Geitost	1	1	1997
2	Raclette Courdavault	2	1	1997
3	Gnocchi di nonna Alice	3	1	1997
4	Vegie-spread	4	1	1997
5	Côte de Blaye	5	1	1997
6	Pâté chinois	1	2	1997
7	Sirop d'érable	2	2	1997
8	Tourtière	3	2	1997
9	Gula Malacca	4	2	1997
10	Pavlova	5	2	1997
11	Raclette Courdavault	1	3	1997
12	Guaraná Fantástica	2	3	1997
13	Outback Lager	3	3	1997
14	Boston Crab Meat	4	3	1997
15	Camembert Pierrot	5	3	1997
16	Chang	5	3	1997
17	Louisiana Hot Spiced...	5	3	1997
18	Gnocchi di nonna Alice	1	4	1997
19	Schoggi Schokolade	2	4	1997
20	Jack's New England ...	3	4	1997
21	Camembert Pierrot	4	4	1997
22	Rhönbräu Klosterbier	5	4	1997
23	Gorgonzola Telino	1	5	1997
24	Inlagd Sill	2	5	1997

Query 8

```
--Buatlah view untuk melihat Order Details yang berisi OrderID,  
--ProductID, ProductName, UnitPrice, Quantity, Discount, Harga setelah diskon.  
CREATE VIEW [View Order Details]  
AS  
SELECT OrderID, orde.ProductID, ProductName,  
       orde.UnitPrice, Quantity, Discount,  
       orde.UnitPrice - ( orde.UnitPrice * Discount ) AS [Harga setelah diskon]  
FROM [Order Details] AS orde  
LEFT JOIN Products as prod  
      ON prod.ProductID = orde.ProductID;
```

0 %

Messages

Commands completed successfully.

Completion time: 2023-01-29T22:37:12.4084335+07:00

Query 9

```
--Buatlah procedure Invoice untuk memanggil CustomerID, CustomerName/company name,  
--OrderID, OrderDate, RequiredDate, ShippedDate jika terdapat inputan CustomerID tertentu.  
CREATE PROCEDURE proc_Orders (  
    @CustomerID AS varchar(5)  
    ) AS  
  
BEGIN  
    IF @CustomerID IN (SELECT ord.CustomerID  
                      FROM Orders AS ord  
                      GROUP BY CustomerID)  
    BEGIN  
        SELECT ord.CustomerID, cus.CompanyName, OrderID, OrderDate, RequiredDate, ShippedDate  
        FROM Orders AS ord  
        LEFT JOIN Customers as cus  
            ON ord.CustomerID = cus.CustomerID  
        WHERE ord.CustomerID = @CustomerID;  
    END  
  
    ELSE  
    BEGIN  
        RETURN PRINT 'CustomerID not Exist in Table Orders';  
    END  
END;
```

%

Messages

Commands completed successfully.

Completion time: 2023-01-29T22:38:16.4808214+07:00