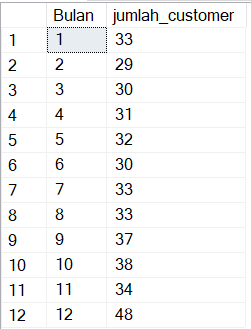
**Intermediate Queries**

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

| SELECT MONTH(OrderDate) Bulan,  COUNT(CustomerID) jumlah\_customer  FROM Orders  WHERE YEAR(OrderDate) = 1997  GROUP BY MONTH(OrderDate); |
| --- |



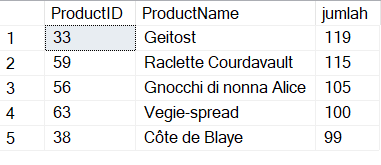
1. Tulis query untuk mendapatkan nama employee yang termasuk Sales Representative.

| SELECT TOP(1) FirstName+' '+LastName nama\_employee, SUM(Quantity\*od.UnitPrice) total\_sales  FROM Orders  LEFT JOIN Employees AS em ON em.EmployeeID = Orders.EmployeeID  LEFT JOIN [Order Details] AS od ON Orders.OrderID = od.OrderID  GROUP BY FirstName, LastName  ORDER BY total\_sales DESC; |
| --- |



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

| SELECT TOP(5) od.ProductID, pd.ProductName, SUM(od.Quantity) jumlah  FROM Orders  LEFT JOIN [Order Details] AS od ON Orders.OrderID=od.OrderID  LEFT JOIN Products AS pd ON od.ProductID=pd.ProductID  WHERE MONTH(OrderDate) = 1  AND YEAR(OrderDate) = 1997  GROUP BY od.ProductID, pd.ProductName  ORDER BY jumlah DESC; |
| --- |



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

| SELECT cs.CustomerID, cs.CompanyName, pd.ProductName  FROM Orders  LEFT JOIN Customers AS cs ON Orders.CustomerID=cs.CustomerID  LEFT JOIN [Order Details] AS od ON Orders.OrderID=od.OrderID  LEFT JOIN Products AS pd ON od.ProductID=pd.ProductID  WHERE pd.ProductName = 'Chai'; |
| --- |



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

| WITH pembelian\_orderid AS(  SELECT Orders.OrderID, SUM(od.UnitPrice\*od.Quantity) pembelian  FROM Orders  LEFT JOIN [Order Details] AS od ON Orders.OrderID=od.OrderID  GROUP BY Orders.OrderID  )  SELECT COUNT(CASE WHEN pembelian <= 100 THEN OrderID END) '>500',  COUNT(CASE WHEN pembelian > 100 AND pembelian <= 250 THEN OrderID END) '100<x<=200',  COUNT(CASE WHEN pembelian > 250 AND pembelian <= 500 THEN OrderID END) '250<x<=500',  COUNT(CASE WHEN pembelian > 500 THEN OrderID END) '>500'  FROM pembelian\_orderid; |
| --- |



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

| WITH pembelian\_customer AS(  SELECT Orders.OrderID, cs.CompanyName, SUM(od.UnitPrice\*od.Quantity) pembelian  FROM Orders  LEFT JOIN Customers AS cs ON Orders.CustomerID=cs.CustomerID  LEFT JOIN [Order Details] AS od ON Orders.OrderID=od.OrderID  WHERE YEAR(OrderDate) = 1997  GROUP BY Orders.OrderID, cs.CompanyName  HAVING SUM(od.UnitPrice\*od.Quantity) > 500  )  SELECT DISTINCT(CompanyName)  FROM pembelian\_customer; |
| --- |



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

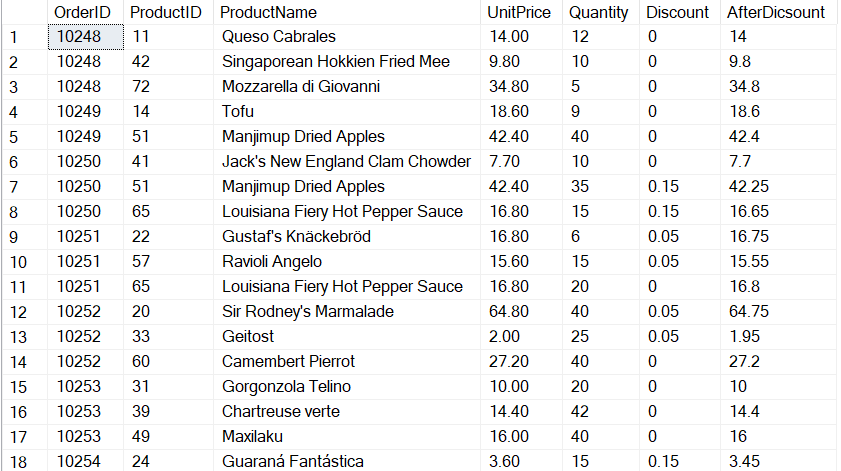
| WITH sales\_prod AS (SELECT pd.ProductName,  MONTH(OrderDate) bulan,  SUM(od.UnitPrice\*od.Quantity) sales,  ROW\_NUMBER() OVER (PARTITION BY MONTH(OrderDate) ORDER BY SUM(od.UnitPrice\*od.Quantity) DESC) AS 'unit\_rank'  FROM Orders  LEFT JOIN [Order Details] AS od ON Orders.OrderID=od.OrderID  LEFT JOIN Products AS pd ON od.ProductID=pd.ProductID  WHERE YEAR(OrderDate) = 1997  GROUP BY pd.ProductName, MONTH(OrderDate)  )  SELECT bulan, ProductName  FROM sales\_prod  WHERE unit\_rank <= 5; |
| --- |



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

| CREATE VIEW InfoProdukView AS  SELECT Orders.OrderID,  od.ProductID,  pd.ProductName,  od.UnitPrice, od.Quantity,  od.Discount,  (od.UnitPrice-od.Discount) AfterDicsount  FROM Orders  LEFT JOIN [Order Details] AS od ON Orders.OrderID=od.OrderID  LEFT JOIN Products AS pd ON od.ProductID=pd.ProductID; |
| --- |

| SELECT \*  FROM InfoProdukView; |
| --- |



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

| CREATE PROCEDURE MakeInvoiceByCust  @CustomerID nvarchar(20)  AS  SELECT  @CustomerID CustomerID,  cs.CompanyName,  OrderID,  OrderDate,  RequiredDate,  ShippedDate  FROM Orders  LEFT JOIN Customers AS cs ON Orders.CustomerID=cs.CustomerID  WHERE cs.CustomerID=@CustomerID; |
| --- |

| EXEC MakeInvoiceByCust @CustomerID='ANTON'; |
| --- |

