**Name – Davender Singh**

**Course – DBW624V1A**

**Professor – Les King**

**1. Sales Volumes Analysis by fiscal quarter**

a. By Store

SELECT

S.StoreName,

D.FiscalQuarter,

SUM(RF.SalesVolume) AS TotalSalesVolume

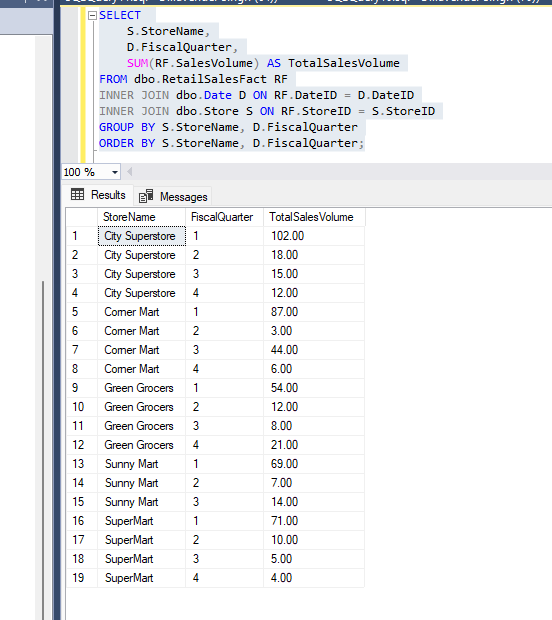
FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

INNER JOIN dbo.Store S ON RF.StoreID = S.StoreID

GROUP BY S.StoreName, D.FiscalQuarter

ORDER BY S.StoreName, D.FiscalQuarter;



b. By Product

SELECT

P.ProductName,

D.FiscalQuarter,

SUM(RF.SalesVolume) AS TotalSalesVolume

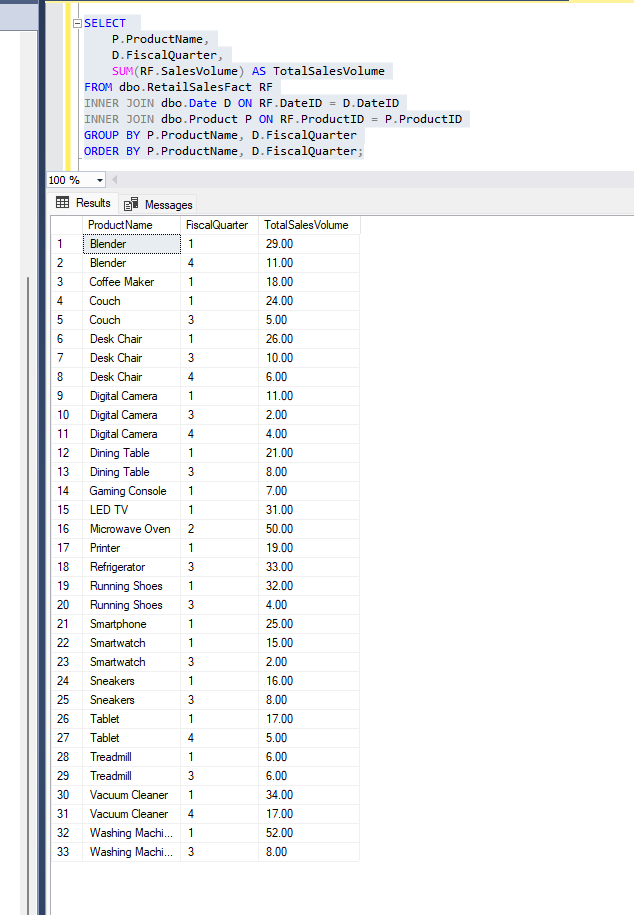
FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductName, D.FiscalQuarter

ORDER BY P.ProductName, D.FiscalQuarter;



c. By Product Group (Age Group)

SELECT

P.ProductGroup,

D.FiscalQuarter,

SUM(RF.SalesVolume) AS TotalSalesVolume

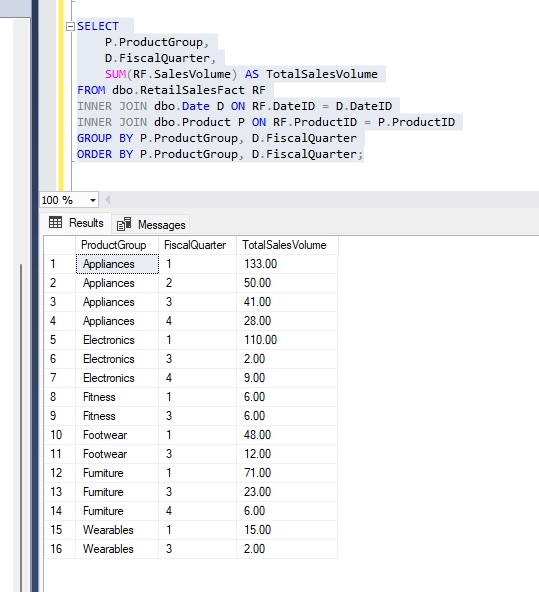
FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductGroup, D.FiscalQuarter

ORDER BY P.ProductGroup, D.FiscalQuarter;



**2. Sales Revenue Analysis by fiscal quarter**

a. By Store

SELECT

S.StoreName,

D.FiscalQuarter,

SUM(RF.SalesRevenue) AS TotalSalesRevenue

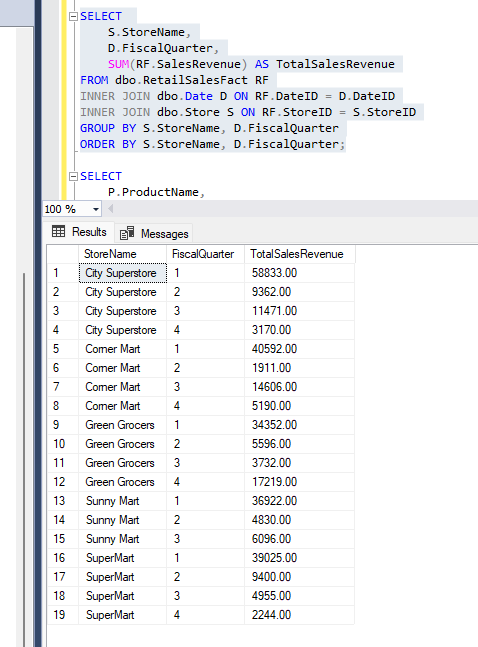
FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

INNER JOIN dbo.Store S ON RF.StoreID = S.StoreID

GROUP BY S.StoreName, D.FiscalQuarter

ORDER BY S.StoreName, D.FiscalQuarter;



b. By Product

SELECT

P.ProductName,

D.FiscalQuarter,

SUM(RF.SalesRevenue) AS TotalSalesRevenue

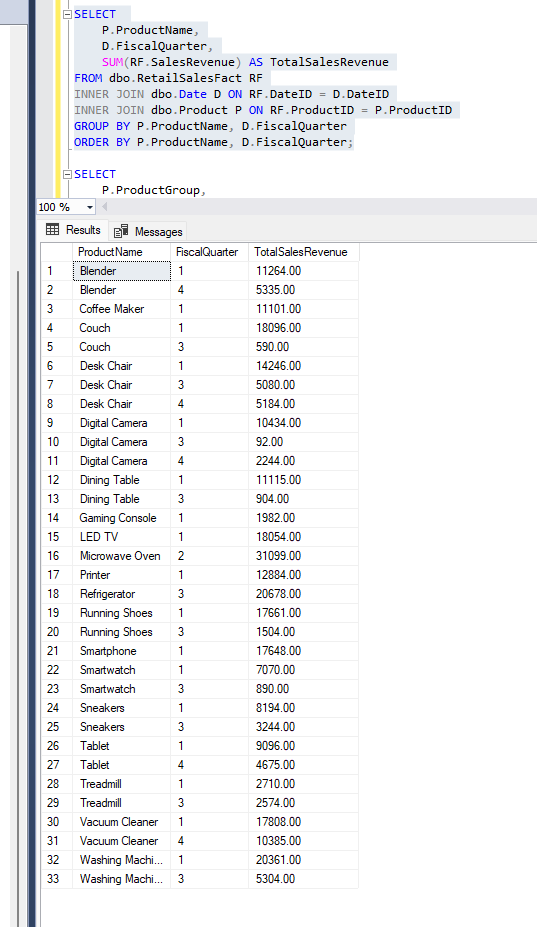
FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductName, D.FiscalQuarter

ORDER BY P.ProductName, D.FiscalQuarter;



c. By Product Group (Age Group)

SELECT

P.ProductGroup,

D.FiscalQuarter,

SUM(RF.SalesRevenue) AS TotalSalesRevenue

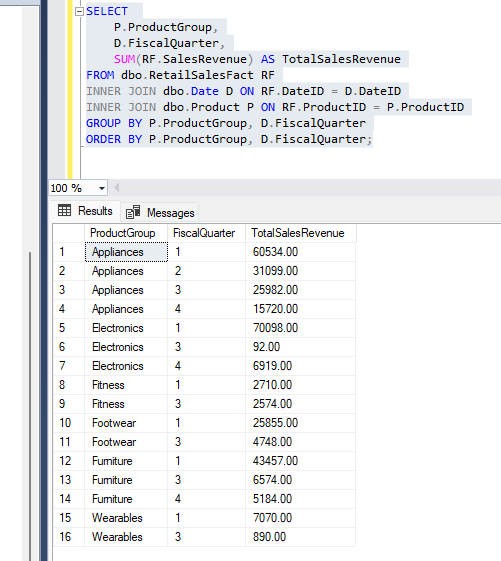
FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductGroup, D.FiscalQuarter

ORDER BY P.ProductGroup, D.FiscalQuarter;



**3. Sales Profit Analysis by fiscal quarter**

a. By Store

SELECT

S.StoreName,

D.FiscalQuarter,

SUM(RF.SalesProfit) AS TotalSalesProfit

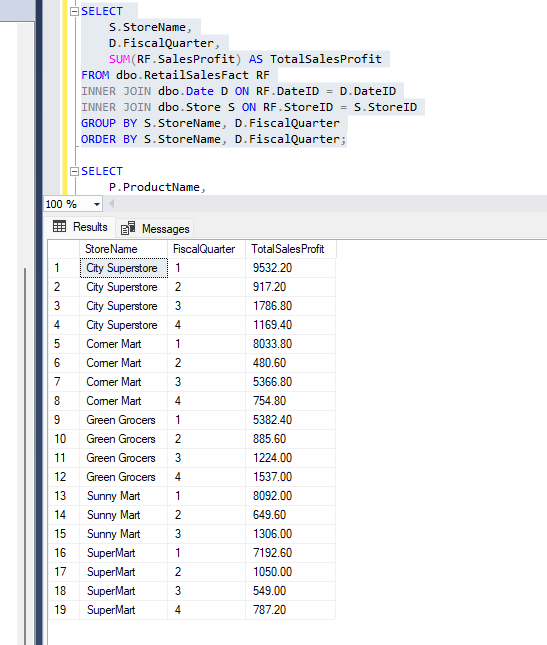
FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

INNER JOIN dbo.Store S ON RF.StoreID = S.StoreID

GROUP BY S.StoreName, D.FiscalQuarter

ORDER BY S.StoreName, D.FiscalQuarter;



b. By Product

SELECT

P.ProductName,

D.FiscalQuarter,

SUM(RF.SalesProfit) AS TotalSalesProfit

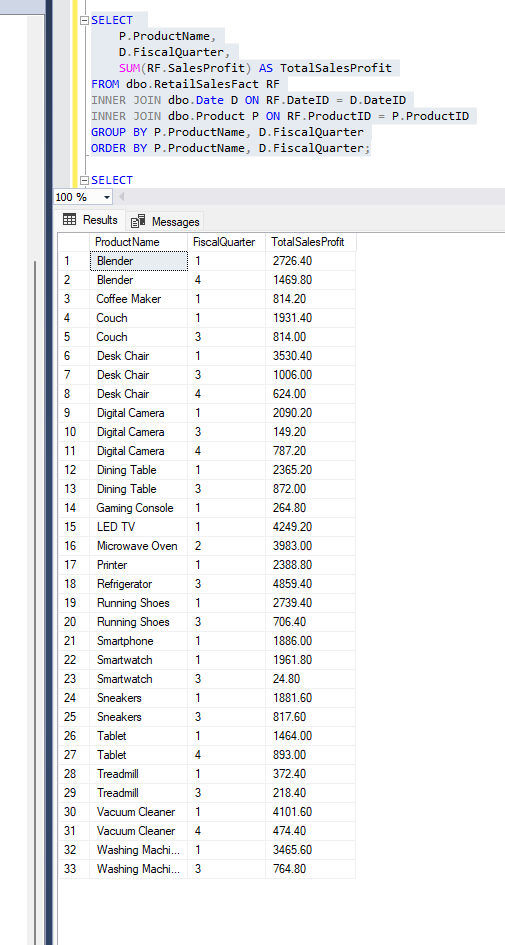
FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductName, D.FiscalQuarter

ORDER BY P.ProductName, D.FiscalQuarter;



c. By Product Group (Age Group)

SELECT

P.ProductGroup,

D.FiscalQuarter,

SUM(RF.SalesProfit) AS TotalSalesProfit

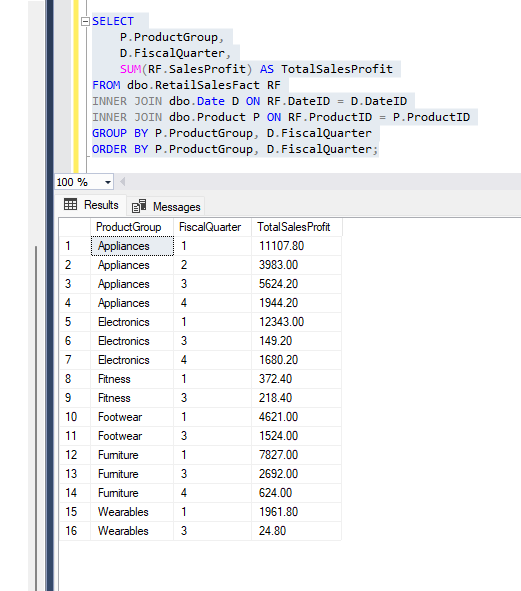
FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductGroup, D.FiscalQuarter

ORDER BY P.ProductGroup, D.FiscalQuarter;



**4. Product Line Analysis by fiscal quarter, measured by revenue and profit**

a. Which products have been the most / least successful

-- Most successful products by revenue

SELECT TOP 10

P.ProductName,

SUM(RF.SalesRevenue) AS TotalRevenue

FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductName

ORDER BY TotalRevenue DESC;

-- Least successful products by revenue

SELECT TOP 10

P.ProductName,

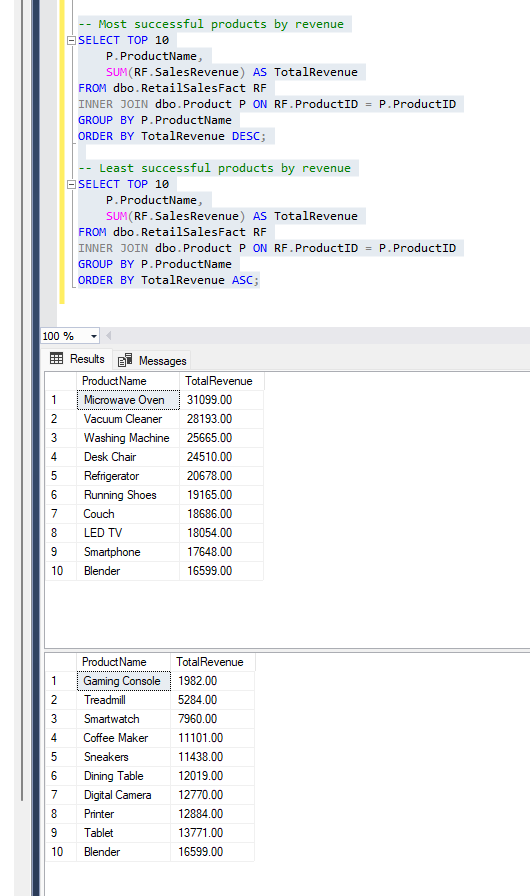
SUM(RF.SalesRevenue) AS TotalRevenue

FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductName

ORDER BY TotalRevenue ASC;



b. Which product groups have been the most / least successful

-- Most successful product groups by revenue

SELECT TOP 10

P.ProductGroup,

SUM(RF.SalesRevenue) AS TotalRevenue

FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductGroup

ORDER BY TotalRevenue DESC;

-- Least successful product groups by revenue

SELECT TOP 10

P.ProductGroup,

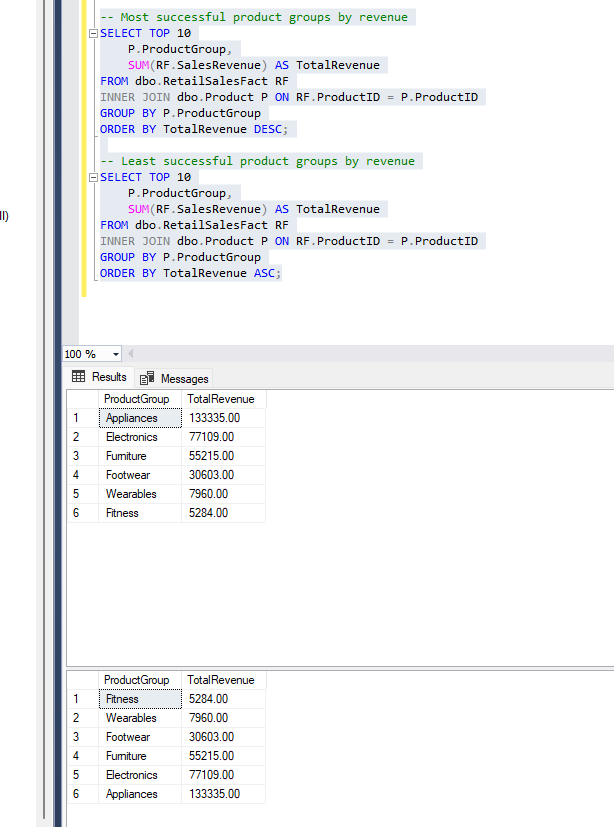
SUM(RF.SalesRevenue) AS TotalRevenue

FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

GROUP BY P.ProductGroup

ORDER BY TotalRevenue ASC;



c. What is the product trends (growth or declining)?

WITH QuarterlyRevenueProfit AS (

SELECT

P.ProductID,

P.ProductName,

D.FiscalQuarter,

SUM(RF.SalesRevenue) AS TotalRevenue,

SUM(RF.SalesProfit) AS TotalProfit

FROM dbo.RetailSalesFact RF

INNER JOIN dbo.Product P ON RF.ProductID = P.ProductID

INNER JOIN dbo.Date D ON RF.DateID = D.DateID

WHERE D.Year = 2023 -- Filtering for the year 2023

GROUP BY P.ProductID, P.ProductName, D.FiscalQuarter

),

QuarterlyGrowth AS (

SELECT

ProductID,

ProductName,

FiscalQuarter,

TotalRevenue,

TotalProfit,

LAG(TotalRevenue) OVER (PARTITION BY ProductID ORDER BY FiscalQuarter) AS PreviousQuarterRevenue,

LAG(TotalProfit) OVER (PARTITION BY ProductID ORDER BY FiscalQuarter) AS PreviousQuarterProfit

FROM QuarterlyRevenueProfit

)

SELECT

ProductID,

ProductName,

FiscalQuarter,

TotalRevenue,

TotalProfit,

PreviousQuarterRevenue,

PreviousQuarterProfit,

CASE

WHEN PreviousQuarterRevenue IS NOT NULL AND TotalRevenue > PreviousQuarterRevenue THEN 'Growth'

WHEN PreviousQuarterRevenue IS NOT NULL AND TotalRevenue < PreviousQuarterRevenue THEN 'Decline'

ELSE 'Stable' -- Assuming the first quarter or no change as stable

END AS RevenueTrend,

CASE

WHEN PreviousQuarterProfit IS NOT NULL AND TotalProfit > PreviousQuarterProfit THEN 'Growth'

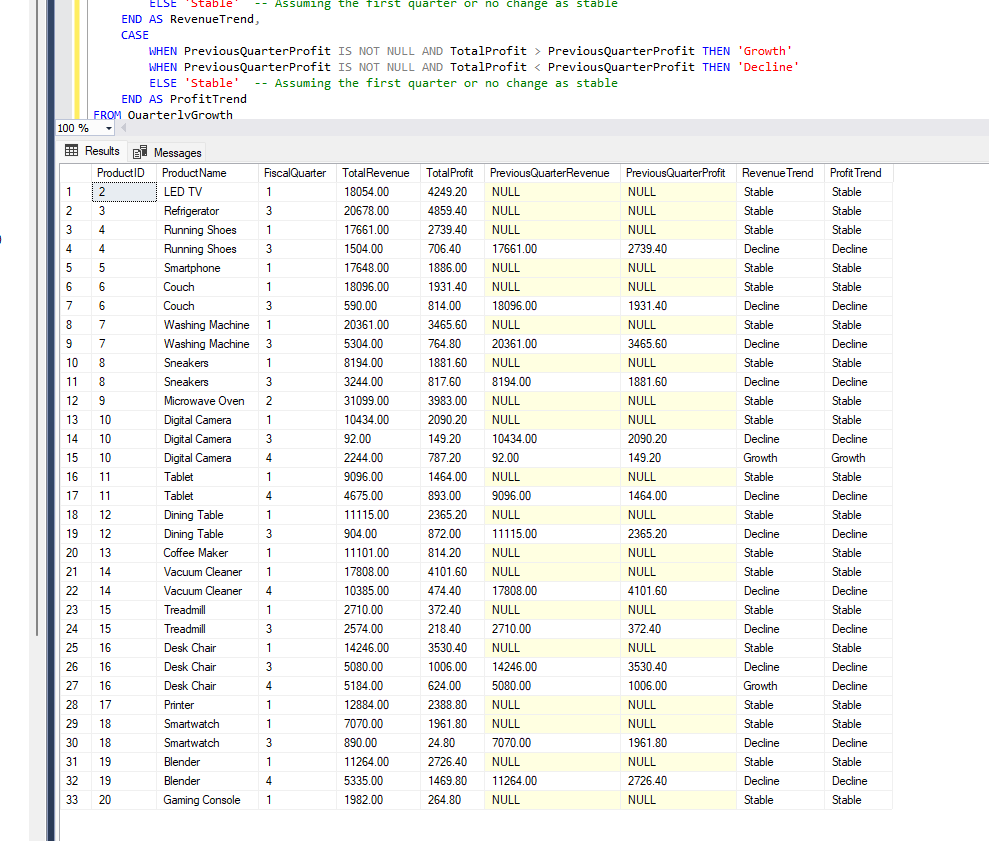
WHEN PreviousQuarterProfit IS NOT NULL AND TotalProfit < PreviousQuarterProfit THEN 'Decline'

ELSE 'Stable' -- Assuming the first quarter or no change as stable

END AS ProfitTrend

FROM QuarterlyGrowth

ORDER BY ProductID, FiscalQuarter;



**5. Store Analysis by fiscal quarter, measured by revenue and profit**

a. Which stores are the most / least successful

-- Most and Least Successful Stores by Revenue and Profit

WITH StoreRevenueProfit AS (

SELECT

RSF.StoreID,

D.FiscalQuarter,

SUM(RSF.SalesRevenue) AS TotalRevenue,

SUM(RSF.SalesProfit) AS TotalProfit

FROM

dbo.RetailSalesFact RSF

JOIN dbo.Date D ON RSF.DateID = D.DateID

GROUP BY

RSF.StoreID, D.FiscalQuarter

)

SELECT

StoreID,

FiscalQuarter,

TotalRevenue,

TotalProfit,

RANK() OVER (PARTITION BY FiscalQuarter ORDER BY TotalRevenue DESC) AS RevenueRank,

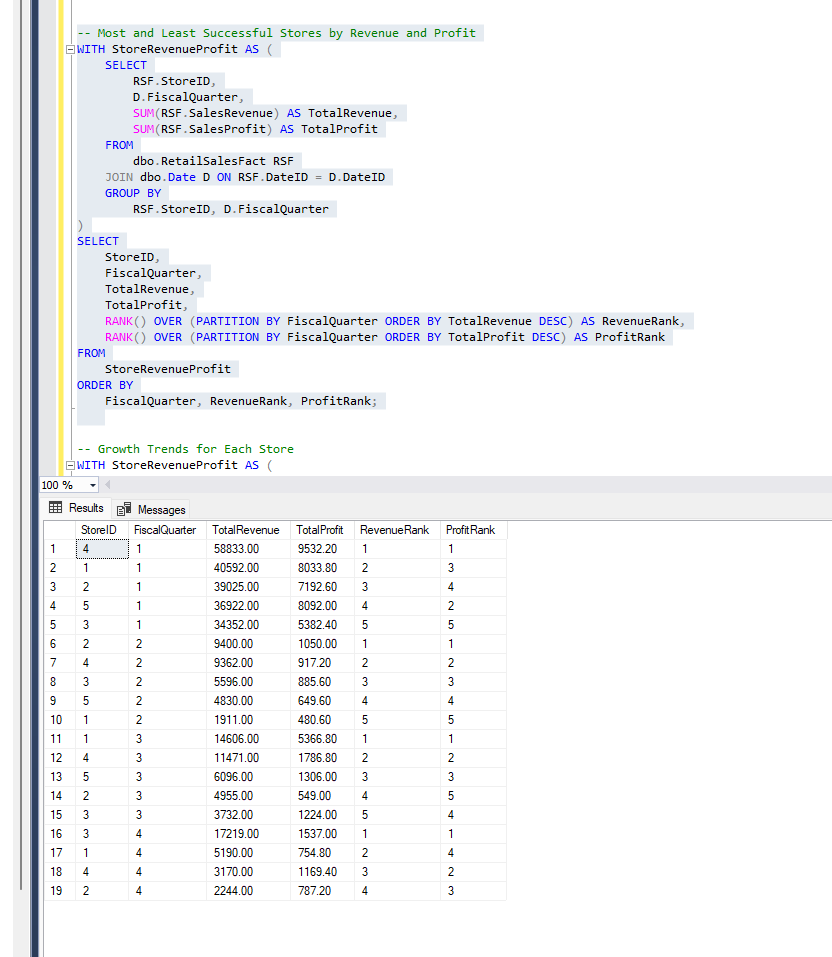
RANK() OVER (PARTITION BY FiscalQuarter ORDER BY TotalProfit DESC) AS ProfitRank

FROM

StoreRevenueProfit

ORDER BY

FiscalQuarter, RevenueRank, ProfitRank;



b. What is the growth trends for each store (growth or declining)?

-- Growth Trends for Each Store

WITH StoreRevenueProfit AS (

SELECT

RSF.StoreID,

D.FiscalQuarter,

SUM(RSF.SalesRevenue) AS TotalRevenue,

SUM(RSF.SalesProfit) AS TotalProfit

FROM

dbo.RetailSalesFact RSF

JOIN dbo.Date D ON RSF.DateID = D.DateID

GROUP BY

RSF.StoreID, D.FiscalQuarter

),

RevenueProfitPreviousQuarter AS (

SELECT

StoreID,

FiscalQuarter,

TotalRevenue,

TotalProfit,

LAG(TotalRevenue) OVER (PARTITION BY StoreID ORDER BY FiscalQuarter) AS PreviousQuarterRevenue,

LAG(TotalProfit) OVER (PARTITION BY StoreID ORDER BY FiscalQuarter) AS PreviousQuarterProfit

FROM

StoreRevenueProfit

)

SELECT

StoreID,

FiscalQuarter,

TotalRevenue,

TotalProfit,

CASE

WHEN PreviousQuarterRevenue IS NOT NULL AND TotalRevenue > PreviousQuarterRevenue THEN 'Growth'

WHEN PreviousQuarterRevenue IS NOT NULL AND TotalRevenue < PreviousQuarterRevenue THEN 'Decline'

ELSE 'Stable'

END AS RevenueTrend,

CASE

WHEN PreviousQuarterProfit IS NOT NULL AND TotalProfit > PreviousQuarterProfit THEN 'Growth'

WHEN PreviousQuarterProfit IS NOT NULL AND TotalProfit < PreviousQuarterProfit THEN 'Decline'

ELSE 'Stable'

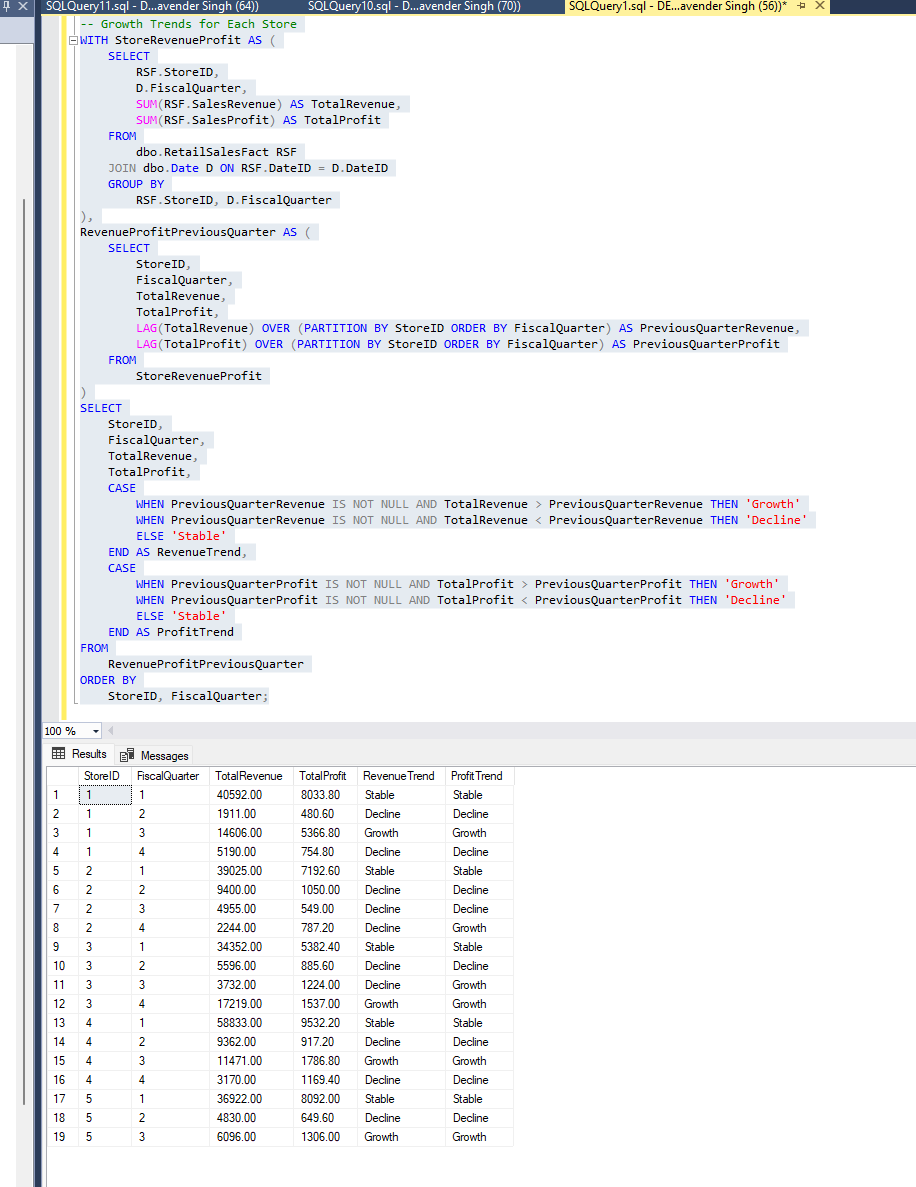
END AS ProfitTrend

FROM

RevenueProfitPreviousQuarter

ORDER BY

StoreID, FiscalQuarter;



**6. Additional Analysis**

a. Which names have been most successful by volume

SELECT

BN.Name,

SUM(RSF.SalesVolume) AS TotalVolume

FROM

dbo.RetailSalesFact RSF

JOIN

dbo.Customer C ON RSF.CustomerID = C.CustomerID

JOIN

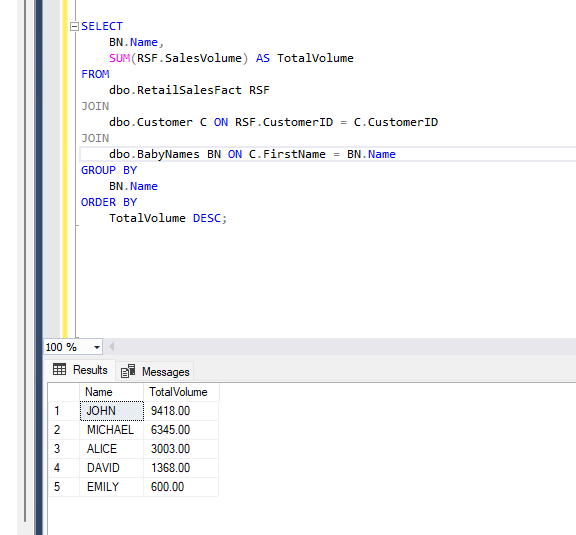
dbo.BabyNames BN ON C.FirstName = BN.Name

GROUP BY

BN.Name

ORDER BY

TotalVolume DESC;



b. Which gender has been most successful by volume

SELECT

BN.Gender,

SUM(RSF.SalesVolume) AS TotalVolume

FROM

dbo.RetailSalesFact RSF

JOIN

dbo.Customer C ON RSF.CustomerID = C.CustomerID

JOIN

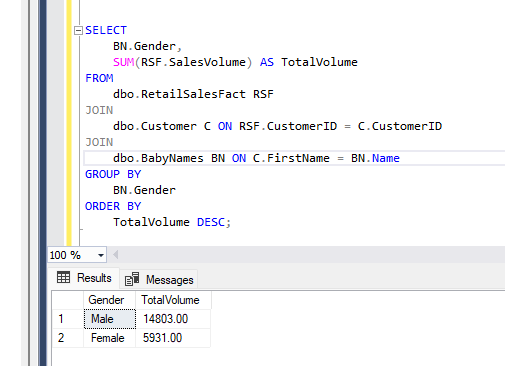
dbo.BabyNames BN ON C.FirstName = BN.Name

GROUP BY

BN.Gender

ORDER BY

TotalVolume DESC;



c. Who was the top sales person for the quarter?

WITH SalespersonVolume AS (

SELECT

E.EmployeeID,

E.EmployeeName,

D.FiscalQuarter,

SUM(RSF.SalesVolume) AS TotalVolume,

RANK() OVER (PARTITION BY D.FiscalQuarter ORDER BY SUM(RSF.SalesVolume) DESC) AS Rank

FROM

dbo.RetailSalesFact RSF

JOIN

dbo.Employee E ON RSF.EmployeeID = E.EmployeeID

JOIN

dbo.Date D ON RSF.DateID = D.DateID

GROUP BY

E.EmployeeID, E.EmployeeName, D.FiscalQuarter

)

SELECT

EmployeeID,

EmployeeName,

FiscalQuarter,

TotalVolume

FROM

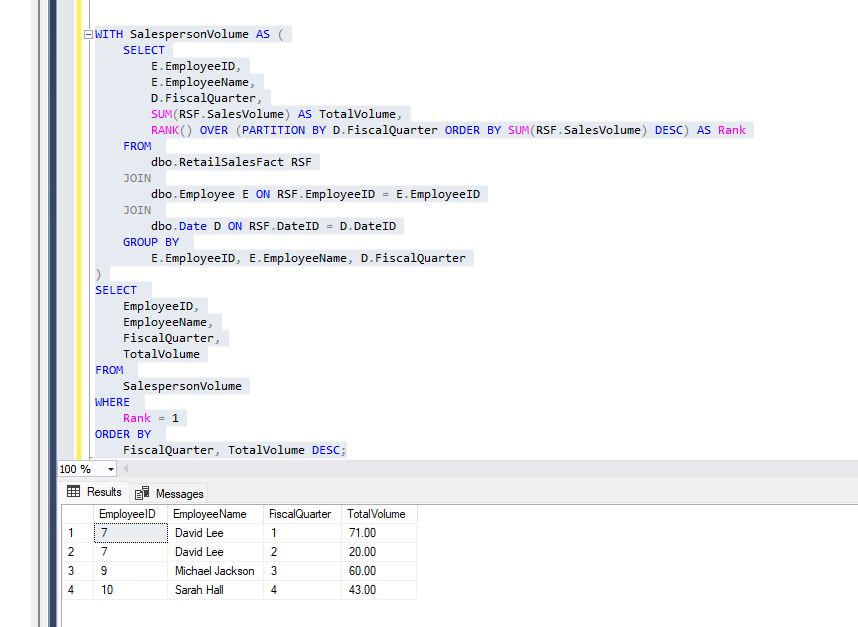
SalespersonVolume

WHERE

Rank = 1

ORDER BY

FiscalQuarter, TotalVolume DESC;



d. What percentage of sales are cash versus credit card?

SELECT

PaymentMethod,

SUM(SalesVolume) AS TotalVolume,

100.0 \* SUM(SalesVolume) / (SELECT SUM(SalesVolume) FROM dbo.RetailSalesFact) AS PercentageOfTotalVolume

FROM

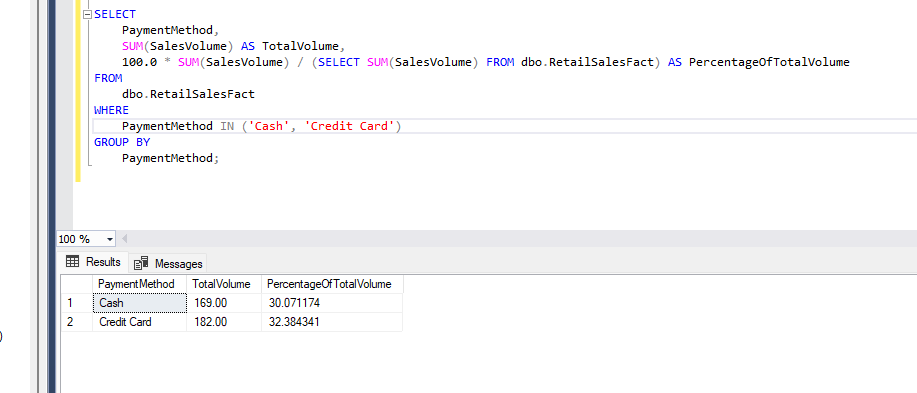
dbo.RetailSalesFact

WHERE

PaymentMethod IN ('Cash', 'Credit Card')

GROUP BY

PaymentMethod;



e. What percentage of sales were using a marketing campaign?

SELECT

MarketingCampaignFlag,

SUM(SalesVolume) AS TotalVolume,

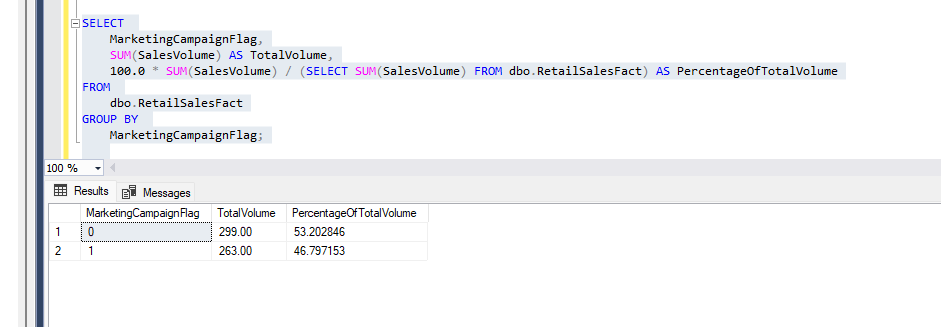
100.0 \* SUM(SalesVolume) / (SELECT SUM(SalesVolume) FROM dbo.RetailSalesFact) AS PercentageOfTotalVolume

FROM

dbo.RetailSalesFact

GROUP BY

MarketingCampaignFlag;



**7. Analytics Against Reference Tables**

a. Which ten provinces should we open stores in, based on population?

SELECT TOP 10

Province,

SUM(Population) AS TotalPopulation

FROM

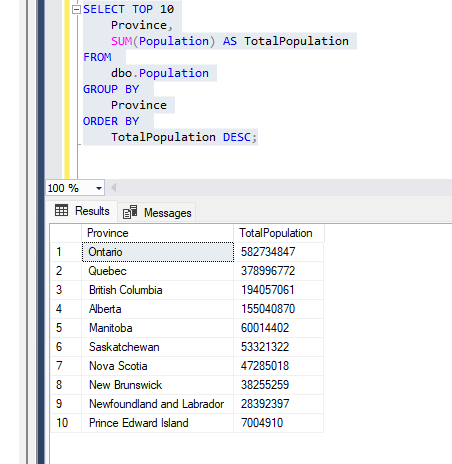
dbo.Population

GROUP BY

Province

ORDER BY

TotalPopulation DESC;



b. Which names should we expect will be the most popular for our personalized products?

SELECT TOP 10

Name,

SUM(Frequency) AS TotalFrequency

FROM

dbo.BabyNames

GROUP BY

Name

ORDER BY

TotalFrequency DESC;

