SQL data insight such as top-selling products, high-revenue stores, customer behavior, market growth, Inventory mangement and revenue trends.



1. Total sales and revenue for each handbag model

SELECT

h.Model,

SUM(s.QuantitySold) AS TotalSales,

SUM(s.TotalRevenue) AS TotalRevenue

FROM Sales s

JOIN Handbags h ON s.BagID = h.BagID

GROUP BY h.Model

2. Stores generating the highest revenue

SELECT

st.StoreName,

SUM(s.TotalRevenue) AS TotalRevenue

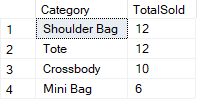
FROM Sales s

JOIN Stores st ON s.StoreID = st.StoreID

GROUP BY st.StoreName

ORDER BY TotalRevenue DESC

3. Most popular handbag categories

SELECT

h.Category,

COUNT(\*) AS TotalSold

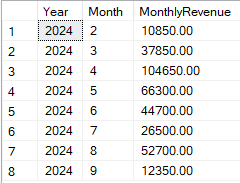
FROM Sales s

JOIN Handbags h ON s.BagID = h.BagID

GROUP BY h.Category

ORDER BY TotalSold DESC

4. Market Growth 🡪 Revenue growth over the past year



SELECT

YEAR(s.SaleDate) AS Year,

MONTH(s.SaleDate) AS Month,

SUM(s.TotalRevenue) AS MonthlyRevenue

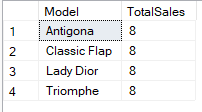
FROM Sales s

WHERE s.SaleDate >= DATEADD(YEAR, -1, GETDATE())

GROUP BY YEAR(s.SaleDate), MONTH(s.SaleDate)

ORDER BY Year, Month

5. Inventory Management 🡪 Underperforming handbags (low sales)

SELECT

h.Model,

SUM(s.QuantitySold) AS TotalSales

FROM Sales s

JOIN Handbags h ON s.BagID = h.BagID

GROUP BY h.Model

HAVING SUM(s.QuantitySold) < 10

1. Customer Insights 🡪 Top 10 customers by total purchase value



SELECT Top 10

c.Name,

c.Email,

SUM(s.TotalRevenue) AS TotalPurchaseValue,

RANK() OVER (ORDER BY SUM(s.TotalRevenue) DESC) AS Rank

FROM Sales s

JOIN Customers c ON s.StoreID = c.CustomerID

GROUP BY c.Name, c.Email

ORDER BY Rank



2. Average purchase value per customer

SELECT

c.Name,

AVG(s.TotalRevenue) AS AvgPurchaseValue

FROM Sales s

JOIN Customers c ON s.StoreID = c.CustomerID

GROUP BY c.Name

3. Operational Optimization 🡪 Sales performance of leather bags vs canvas bags

SELECT

 h.Material,

SUM(s.QuantitySold) AS TotalSales,

SUM(s.TotalRevenue) AS TotalRevenue

FROM Sales s

JOIN Handbags h ON s.BagID = h.BagID

WHERE h.Material IN ('Leather', 'Canvas')

GROUP BY h.Material

4. Stores needing restock for high-selling handbag models

SELECT

h.Model,

st.StoreName,

SUM(s.QuantitySold) AS TotalSold

FROM Sales s

JOIN Handbags h ON s.BagID = h.BagID

JOIN Stores st ON s.StoreID = st.StoreID

GROUP BY h.Model, st.StoreName

HAVING SUM(s.QuantitySold) > 10