DAMG7370 - Designing Advanced Data Architectures for Business Intelligence

Name: Krisha Lakhani
NUID: 002334794

1. Total Sales by Country

```
SELECT
   a.CountryRegion AS Country,                                      -- Name of the country
   COUNT(DISTINCT soh.CustomerID) AS CustomerCount,    -- How many unique customers
   COUNT(DISTINCT soh.SalesOrderID) AS OrderCount,      -- How many orders were placed
   SUM(soh.TotalDue) AS TotalSales                               -- Total revenue generated
FROM SalesLT.SalesOrderHeader soh
JOIN SalesLT.Address a
   ON soh.BillToAddressID = a.AddressID            -- Use billing address to track sales location
GROUP BY a.CountryRegion
ORDER BY TotalSales DESC;                            -- Show countries with highest revenue first
```

---

2. Total Sales by City & Country

```
SELECT
   a.CountryRegion AS Country,                                                    -- Country name
   a.City AS City,                                                  -- City name within the country
   COUNT(DISTINCT soh.CustomerID) AS CustomerCount,    -- Unique customers in that city
   SUM(soh.TotalDue) AS TotalSales                                -- Total revenue from the city
FROM SalesLT.SalesOrderHeader soh
JOIN SalesLT.Address a
   ON soh.BillToAddressID = a.AddressID                          -- Link order to billing address
GROUP BY a.CountryRegion, a.City                         -- Group results by country then city
ORDER BY a.CountryRegion, TotalSales DESC;         -- Sort by country, then top-earning cities
```

---

3. Total Sales by Customer (Person)

```
SELECT
   c.CustomerID,                                                   -- Unique ID for each customer
   c.FirstName + ' ' + ISNULL(c.MiddleName + ' ', ") + c.LastName AS CustomerName,
                                                               -- Full name (middle name optional)
   c.EmailAddress,                                                                -- Customer email
   COUNT(soh.SalesOrderID) AS OrderCount,       -- Number of orders placed by the customer
   SUM(soh.TotalDue) AS TotalSales                           -- Total spending by this customer
FROM SalesLT.Customer c
JOIN SalesLT.SalesOrderHeader soh
   ON c.CustomerID = soh.CustomerID                            -- Link customer to their orders
WHERE c.CompanyName IS NULL
```

```
GROUP BY c.CustomerID, c.FirstName, c.MiddleName, c.LastName, c.EmailAddress
ORDER BY TotalSales DESC;                              -- List top-spending customers first
```

---

4. Total Sales by Customer (Company)

```
SELECT
   c.CompanyName,                                                  -- Name of the company
   COUNT(DISTINCT c.CustomerID) AS CustomerCount,  -- Number of customers associated
with this company
   COUNT(soh.SalesOrderID) AS OrderCount,       -- Total number of orders from this company
   SUM(soh.TotalDue) AS TotalSales                        -- Total revenue from this company
FROM SalesLT.Customer c
JOIN SalesLT.SalesOrderHeader soh
   ON c.CustomerID = soh.CustomerID                          -- Link company to their orders
WHERE c.CompanyName IS NOT NULL
GROUP BY c.CompanyName                                                   -- Group by company
ORDER BY TotalSales DESC;                                       -- Rank companies by revenue
```

---

5. Total Sales by Product Name

```
SELECT
   p.ProductID,                                                     -- Unique product identifier
   p.Name AS ProductName,                                                               -- Product
name
   p.ProductNumber,                                                               -- Product code
   COUNT(DISTINCT soh.SalesOrderID) AS OrderCount,  -- How many orders included this
product
   SUM(sod.OrderQty) AS TotalQuantitySold,                                  -- Total units sold
   SUM(sod.LineTotal) AS TotalSales                              -- Revenue from this product
FROM SalesLT.Product p
JOIN SalesLT.SalesOrderDetail sod
   ON p.ProductID = sod.ProductID                              -- Link product to order details
JOIN SalesLT.SalesOrderHeader soh
   ON sod.SalesOrderID = soh.SalesOrderID                  -- Link orders to their header info
GROUP BY p.ProductID, p.Name, p.ProductNumber
ORDER BY TotalSales DESC;                         -- Show top revenue-generating products first
```

DAMG7370 - Designing Advanced Data Architectures for Business Intelligence

Name: Krisha Lakhani
NUID: 002334794

6. Total Sales by Product by Company

```
SELECT
   c.CompanyName,                                    -- Company purchasing the product
   p.Name AS ProductName,                                      -- Product purchased
   SUM(sod.OrderQty) AS QuantitySold,             -- Total units purchased by this company
   SUM(sod.LineTotal) AS TotalSales      -- Revenue generated by this company-product pair
FROM SalesLT.Customer c
JOIN SalesLT.SalesOrderHeader soh
   ON c.CustomerID = soh.CustomerID                       -- Link company to their orders
JOIN SalesLT.SalesOrderDetail sod
   ON soh.SalesOrderID = sod.SalesOrderID                      -- Link orders to products
JOIN SalesLT.Product p
   ON sod.ProductID = p.ProductID                         -- Link products to order details
WHERE c.CompanyName IS NOT NULL
GROUP BY c.CompanyName, p.Name                     -- Aggregate by company and product
ORDER BY c.CompanyName, TotalSales DESC;            -- List each company’s top products
```

---

7. Total Sales by Product Category

```
SELECT
   pc.ProductCategoryID,                                                 -- Category ID
   pc.Name AS CategoryName,                                              -- Category name
   COUNT(DISTINCT p.ProductID) AS ProductCount,     -- Number of products in this category
   SUM(sod.OrderQty) AS TotalQuantitySold,                -- Total units sold in this category
   SUM(sod.LineTotal) AS TotalSales                      -- Total revenue from this category
FROM SalesLT.ProductCategory pc
JOIN SalesLT.Product p
   ON pc.ProductCategoryID = p.ProductCategoryID                -- Link category to products
JOIN SalesLT.SalesOrderDetail sod
   ON p.ProductID = sod.ProductID                           -- Link products to order details
GROUP BY pc.ProductCategoryID, pc.Name
ORDER BY TotalSales DESC;                                       -- Rank categories by revenue
```