**1.Demographic Distribution: Customers by Gender and Continent**

SELECT Gender, Continent,

COUNT(\*) AS customer\_count

FROM customers

GROUP BY Gender,Continent

ORDER BY customer\_count DESC;

**2.Customer Distribution by Gender**

SELECT Gender,

COUNT(\*) AS customer\_count

FROM customers

GROUP BY Gender

ORDER BY customer\_count DESC;

3.**Customer Distribution by Location (Country and State):**

SELECT

country,

state, gender,

COUNT(\*) AS customer\_count

FROM customers

GROUP BY country, state, gender

ORDER BY customer\_count DESC;

**4.Customer Distribution by Location (City):**

SELECT

country, city,

COUNT(CustomerKey) AS customer\_count

FROM customers

GROUP BY country, city

ORDER BY customer\_count DESC

LIMIT 10;

**5.Customer Distribution by Gender, Country and Age Range**

SELECT Gender, country,

CASE

WHEN YEAR(CURDATE()) - YEAR(Birthday) BETWEEN 0 AND 25 THEN '0-25'

WHEN YEAR(CURDATE()) - YEAR(Birthday) BETWEEN 25 AND 50 THEN '25-50'

WHEN YEAR(CURDATE()) - YEAR(Birthday) BETWEEN 50 AND 75 THEN '50-75'

ELSE '75+'

END AS AgeGroup,

COUNT(\*) AS Customer\_Count

FROM customers

GROUP BY Gender, country, AgeGroup

ORDER BY country ASC , Customer\_Count DESC ;

**6.Segmentation: Top 3 Product Categories Preferred by Male Customer**

SELECT Subcategory,Gender, sum(Quantity) AS total\_purchased from products

JOIN sales ON products.ProductKey=sales.ProductKey

JOIN customers ON customers.CustomerKey=sales.CustomerKey

GROUP BY Subcategory,Gender

HAVING Gender="male"

ORDER BY total\_purchased desc

LIMIT 3;

**7.Overall Sales Performance Over Time:**

SELECT

DATE\_FORMAT(`OrderDate`, '%Y') AS year,

SUM(Quantity \* CAST(REPLACE(UnitPriceUSD, '$', '') AS DECIMAL(10,2))) AS total\_sales,

AVG(Quantity \* CAST(REPLACE(UnitPriceUSD, '$', '') AS DECIMAL(10,2))) AS avg\_sales

FROM sales join products ON sales.ProductKey=products.ProductKey

GROUP BY year

ORDER BY year DESC;

**8. Top Performing Products by Quantity Sold:**

SELECT ProductName,

SUM(Quantity) AS total\_quantity\_sold,

SUM(Quantity \* CAST(REPLACE(UnitPriceUSD, '$', '') AS DECIMAL(10,2))) AS total\_revenue

FROM products JOIN sales ON products.ProductKey = sales.ProductKey

GROUP BY ProductName

ORDER BY total\_quantity\_sold DESC;

**9.Geographical Analysis by Sales:**

SELECT City,

SUM(Quantity \* CAST(REPLACE(UnitPriceUSD,'$','') AS DECIMAL (10,2))) AS total\_sales

FROM products JOIN sales ON products.ProductKey = sales.ProductKey

JOIN customers ON customers.CustomerKey=sales.CustomerKey

GROUP BY City

ORDER BY total\_sales DESC;

**10. Sales by Currency:**

SELECT CurrencyCode,

SUM(Quantity \* CAST(REPLACE(UnitPriceUSD,'$','') AS DECIMAL(10,2)))AS total\_sales,

AVG(Exchange) AS average\_exchange\_rate

FROM products JOIN sales

ON products.ProductKey = sales.ProductKey

JOIN exchange\_rates

ON exchange\_rates.Date = sales.OrderDate

GROUP BY CurrencyCode

ORDER BY total\_sales DESC;

**11. Profitability Analysis by Product,Product Popularity:**

SELECT ProductName,

SUM(Quantity) AS total\_num\_of\_products,

(SUM(CAST(REPLACE(UnitPriceUSD,'$','') AS DECIMAL(10,2))) - SUM(CAST(REPLACE(UnitCostUSD,'$','') AS DECIMAL(10,2)))) AS profit\_margin

FROM products JOIN sales ON products.ProductKey = sales.ProductKey

GROUP BY ProductName

HAVING

profit\_margin > 0

ORDER BY

profit\_margin DESC;

**12.Category Analysis:**

SELECT Category,Subcategory,

SUM(Quantity) AS total\_num\_of\_products,

(SUM(CAST(REPLACE(UnitPriceUSD,'$','') AS DECIMAL(10,2))) - SUM(CAST(REPLACE(UnitCostUSD,'$','') AS DECIMAL(10,2)))) AS profit\_margin

FROM products JOIN sales ON products.ProductKey = sales.ProductKey

GROUP BY Category,Subcategory

HAVING

profit\_margin > 0

ORDER BY

profit\_margin DESC;

**13.Analyze sales by store location and Store Performance :**

SELECT State,Gender,SquareMeters AS Store,

COUNT(DISTINCT OrderNumber) AS total\_orders,

SUM(Quantity \* (CAST(REPLACE(UnitPriceUSD,'$','') AS DECIMAL(10,2)))) AS total\_sales

FROM stores JOIN sales ON stores.StoreKey = sales.StoreKey

JOIN customers ON customers.CustomerKey = sales.CustomerKey

JOIN products ON products.ProductKey = sales.ProductKey

GROUP BY Store,State,Gender

ORDER BY total\_orders DESC,total\_sales DESC;

**14.Product Analysis Top Five Products :**

SELECT ProductName,

SUM(Quantity) AS total\_quantity\_sold,

SUM(Quantity \* CAST(REPLACE(UnitPriceUSD, '$', '') AS DECIMAL(10,2))) AS total\_revenue

FROM products JOIN sales ON products.ProductKey = sales.ProductKey

GROUP BY ProductName

ORDER BY total\_quantity\_sold DESC

LIMIT 5: