**-- Total Sales by Category**

SELECT c.Category, SUM(od.Quantity \* od.amount) AS TotalSales

FROM Order\_Details od

JOIN Category c ON c.Category\_ID = od.Category\_ID

GROUP BY c.Category

ORDER BY TotalSales DESC;

**-- Monthly Sales Trends**

SELECT

DATE\_FORMAT(o.Order\_Date, '%Y-%m') AS Month,

SUM(od.Quantity \* od.amount) AS Sales\_Amt

FROM Order\_Details od

JOIN orders o

ON od.order\_id = o.order\_id

GROUP BY DATE\_FORMAT(o.Order\_Date, '%Y-%m');

**-- Find the top 5 users who have spent the most amount of money.**

SELECT u.name, SUM(od.amount) AS total\_spent

FROM Orders o

JOIN Order\_Details od ON o.order\_id = od.order\_id

JOIN Users u ON o.user\_id = u.user\_id

GROUP BY u.name

ORDER BY total\_spent DESC

LIMIT 5;

**-- Calculate the monthly revenue for the year 2019.**

SELECT DATE\_FORMAT(o.order\_date, '%Y-%m') AS month,

SUM(od.amount) AS monthly\_revenue

FROM Orders o

JOIN Order\_Details od ON o.order\_id = od.order\_id

WHERE YEAR(o.order\_date) = 2019

GROUP BY month

ORDER BY month;

**-- Identify the category with the highest average profit per order.**

SELECT c.category, ROUND(AVG(od.profit),2) AS avg\_profit

FROM Order\_Details od

JOIN Category c ON od.category\_id = c.category\_id

GROUP BY c.category

ORDER BY avg\_profit DESC

LIMIT 1;

SELECT DATE\_FORMAT(order\_date, '%Y-%m') AS month, COUNT(\*)

FROM orders

WHERE YEAR(order\_date) = 2019

GROUP BY month

ORDER BY month ASC;

**-- Find the user who has placed orders in the most number of different categories.**

SELECT u.name, COUNT(DISTINCT c.category\_id) AS category\_count

FROM Orders o

JOIN Order\_Details od ON o.order\_id = od.order\_id

JOIN Category c ON od.category\_id = c.category\_id

JOIN Users u ON o.user\_id = u.user\_id

GROUP BY u.name

ORDER BY category\_count DESC

LIMIT 1;

**-- Calculate the total profit for each user and categorize them into 'High', 'Medium', and 'Low' profit groups.**

SELECT u.name, SUM(od.profit) AS total\_profit,

CASE

WHEN SUM(od.profit) > 1000 THEN 'High'

WHEN SUM(od.profit) BETWEEN 500 AND 1000 THEN 'Medium'

WHEN SUM(od.profit) < 0 THEN 'Negative'

ELSE 'Low'

END AS profit\_group

FROM Orders o

JOIN Order\_Details od ON o.order\_id = od.order\_id

JOIN Users u ON o.user\_id = u.user\_id

GROUP BY u.name;

**-- Find the top 3 cities with the highest average order amount.**

SELECT u.city, ROUND(AVG(od.amount),2) AS avg\_order\_amount

FROM Orders o

JOIN Order\_Details od ON o.order\_id = od.order\_id

JOIN Users u ON o.user\_id = u.user\_id

GROUP BY u.city

ORDER BY avg\_order\_amount DESC

LIMIT 3;