RDBMS Task #3

1. Find the Total Number of Orders for Each Customer.

Ans:SELECT c.customer_id,

   CONCAT(c.first_name, ' ', c.last_name) AS customer_name,

   COUNT(o.order_id) AS total_orders

FROM

   Customers c

LEFT JOIN

   Orders o ON c.customer_id = o.customer_id

GROUP BY

   c.customer_id, c.first_name, c.last_name;

2. Find the Total Sales Amount for Each Product (Revenue per Product).

Ans:SELECT p.product_id, p.product_name,

   SUM(od.quantity * od.unit_price) AS total_revenue

FROM

   Products p

JOIN

   OrderDetails od ON p.product_id = od.product_id

GROUP BY

   p.product_id, p.product_name;

3. Find the Most Expensive Product Sold.

Ans:SELECT p.product_id, p.product_name,   MAX(od.unit_price) AS max_price

FROM Products JOIN   OrderDetails od ON p.product_id = od.product_id;

4. Get the List of Customers Who Have Placed Orders in the Last 30 Days.

Ans:SELECT   DISTINCT c.customer_id,   CONCAT(c.first_name, ' ', c.last_name) AS customer_name, c.email

FROM Customers c

JOIN Orders o ON c.customer_id = o.customer_id

WHERE   o.order_date >= CURDATE() - INTERVAL 30 DAY;

5. Calculate the Total Amount Paid by Each Customer.

Ans:SELECT c.customer_id, CONCAT(c.first_name, ' ', c.last_name) AS customer_name,
 SUM(p.payment_amount) AS total_paid

FROM Customers JOIN

    Orders o ON c.customer_id = o.customer_id

JOIN

    Payments p ON o.order_id = p.order_id

GROUP BY

    c.customer_id, c.first_name, c.last_name;

6. Get the Number of Products Sold by Category.

Ans:SELECT p.category,

SUM(od.quantity) AS total_products_sold

FROM Products JOIN

OrderDetails od ON p.product_id = od.product_id

GROUP BY   p.category;

7. List All Orders That Are Pending (i.e., Orders that haven't been shipped yet).

Ans:SELECT o.order_id, CONCAT(c.first_name, ' ', c.last_name) AS customer_name, o.order_date,
 o.total_amount FROM   Orders o JOIN   Customers c ON o.customer_id = c.customer_id

WHWHERE o.order_status = 'Pending';

8. Find the Average Order Value (Total Order Amount/Number of Orders).

Ans:SELECT AVG(o.total_amount) AS average_order_value FROM Orders o;

9. List the Top 5 Customers Who Have Spent the Most Money.

Ans:SELECT c.customer_id, CONCAT(c.first_name, ' ', c.last_name) AS customer_name,
SUM(p.payment_amount) AS total_spent FROM   Customers c JOIN Orders o ON c.customer_id =
o.customer_id JOIN   Payments p ON o.order_id = p.order_id GROUP BY   c.customer_id,
c.first_name, c.last_name ORDER BY   total_spent DESC

LIMIT 5;

10. Find the Products That Have Never Been Sold.

Ans:SELECT p.product_id,   p.product_name,   p.category

FROM   Products p LEFT JOIN OrderDetails od ON p.product_id = od.product_id

WHERE   od.order_id IS NULL;