SQL Practice Solutions

1. SELECT * FROM Customers;
2. SELECT * FROM Products WHERE price > 50;
3. SELECT * FROM Orders WHERE MONTH(order_date) = 6 AND YEAR(order_date) = 2025;
4. SELECT name, country FROM Customers;
5. SELECT DISTINCT category FROM Products;
6. SELECT country, COUNT(*) FROM Customers GROUP BY country;
7. SELECT AVG(price) FROM Products;
8. SELECT * FROM Products ORDER BY price DESC LIMIT 1;
9. SELECT * FROM Customers WHERE age > 30;
10. SELECT * FROM Orders WHERE quantity > 2;
11. SELECT COUNT(*) FROM Orders;
12. SELECT product_id, SUM(quantity) FROM Orders GROUP BY product_id;
13. SELECT o.product_id, SUM(o.quantity * p.price) AS total_revenue FROM Orders o JOIN Products p ON o.product_id = p.product_id GROUP BY o.product_id;
14. SELECT DISTINCT c.* FROM Customers c JOIN Orders o ON c.customer_id = o.customer_id JOIN Products p ON o.product_id = p.product_id WHERE p.name = 'Yoga Mat';
15. SELECT * FROM Products WHERE category = 'Fitness';
16. SELECT * FROM Customers WHERE country = 'India';
17. SELECT * FROM Customers ORDER BY age ASC LIMIT 1;
18. SELECT * FROM Products WHERE product_id NOT IN (SELECT DISTINCT product_id FROM Orders);
19. SELECT customer_id FROM Orders GROUP BY customer_id HAVING COUNT(DISTINCT product_id) > 1;
20. SELECT o.customer_id, SUM(o.quantity * p.price) AS total_spent FROM Orders o JOIN Products p ON o.product_id = p.product_id GROUP BY o.customer_id ORDER BY total_spent DESC LIMIT 1;
21. SELECT * FROM Customers WHERE customer_id NOT IN (SELECT DISTINCT customer_id FROM Orders);
22. SELECT country, AVG(age) FROM Customers GROUP BY country;
23. SELECT DISTINCT p.* FROM Products p JOIN Orders o ON p.product_id = o.product_id JOIN Customers c ON o.customer_id = c.customer_id WHERE c.country = 'USA';
24. SELECT DISTINCT c.* FROM Customers c JOIN Orders o ON c.customer_id = o.customer_id JOIN Products p ON o.product_id = p.product_id WHERE p.category = 'Home Kitchen';
25. SELECT COUNT(DISTINCT product_id) FROM Orders;
26. SELECT DISTINCT c.* FROM Customers c JOIN Orders o1 ON c.customer_id = o1.customer_id JOIN Orders o2 ON c.customer_id = o2.customer_id WHERE DATEDIFF(o2.order_date, o1.order_date) = 1;
27. SELECT o.customer_id, SUM(o.quantity * p.price) AS total_spent FROM Orders o JOIN Products p ON o.product_id = p.product_id GROUP BY o.customer_id;
28. SELECT * FROM Products ORDER BY price DESC LIMIT 3;
29. SELECT * FROM Products WHERE price BETWEEN 20 AND 100;
30. SELECT customer_id, COUNT(*) AS total_orders FROM Orders GROUP BY customer_id;
31. SELECT o.order_id, c.name AS customer_name, p.name AS product_name, o.quantity FROM Orders o JOIN Customers c ON o.customer_id = c.customer_id JOIN Products p ON o.product_id = p.product_id;

32. SELECT customer_id FROM Orders o JOIN Products p ON o.product_id = p.product_id GROUP BY customer_id HAVING SUM(p.category != 'Electronics') = 0;
33. SELECT DISTINCT p.* FROM Products p JOIN Orders o ON p.product_id = o.product_id JOIN Customers c ON o.customer_id = c.customer_id WHERE c.age < 30;
34. SELECT SUM(o.quantity * p.price) AS total_revenue FROM Orders o JOIN Products p ON o.product_id = p.product_id WHERE MONTH(o.order_date) = 6 AND YEAR(o.order_date) = 2025;
35. SELECT o.customer_id FROM Orders o JOIN Products p ON o.product_id = p.product_id GROUP BY o.customer_id HAVING COUNT(DISTINCT p.category) >= 2;
36. SELECT o.*, p.category FROM Orders o JOIN Products p ON o.product_id = p.product_id;
37. SELECT product_id, SUM(quantity) AS total_quantity FROM Orders GROUP BY product_id ORDER BY total_quantity DESC LIMIT 1;
38. SELECT * FROM Orders WHERE quantity > 3;
39. SELECT customer_id, COUNT(*) AS total_orders FROM Orders GROUP BY customer_id ORDER BY total_orders DESC LIMIT 1;
40. SELECT customer_id, AVG(quantity) AS avg_quantity FROM Orders GROUP BY customer_id ORDER BY avg_quantity DESC;
41. SELECT c.* FROM Customers c WHERE c.customer_id NOT IN (SELECT DISTINCT o.customer_id FROM Orders o JOIN Products p ON o.product_id = p.product_id WHERE p.price > 50);
42. SELECT category, MAX(price) FROM Products GROUP BY category;
43. SELECT customer_id, MAX(order_date) AS last_order_date FROM Orders GROUP BY customer_id;
44. SELECT DISTINCT c.* FROM Customers c JOIN Orders o ON c.customer_id = o.customer_id JOIN Products p ON o.product_id = p.product_id WHERE p.name = 'Running Shoes';
45. SELECT p.product_id, p.name, SUM(o.quantity) AS total_quantity_sold FROM Products p JOIN Orders o ON p.product_id = o.product_id GROUP BY p.product_id, p.name;
46. SELECT o.*, c.age FROM Orders o JOIN Customers c ON o.customer_id = c.customer_id;
47. SELECT o.customer_id FROM Orders o JOIN Products p ON o.product_id = p.product_id GROUP BY o.customer_id HAVING COUNT(DISTINCT p.category) >= 3;
48. SELECT p.product_id, p.name, IFNULL(SUM(o.quantity),0) AS total_quantity FROM Products p LEFT JOIN Orders o ON p.product_id = o.product_id GROUP BY p.product_id, p.name ORDER BY total_quantity ASC LIMIT 1;
49. SELECT o.* FROM Orders o WHERE o.order_date = (SELECT MIN(order_date) FROM Orders o2 WHERE o2.customer_id = o.customer_id);
50. SELECT c.*, IF(o.customer_id IS NULL, 'No', 'Yes') AS has_ordered FROM Customers c LEFT JOIN Orders o ON c.customer_id = o.customer_id GROUP BY c.customer_id;