**Basic / Fresher Level**

1. Write a query to select all columns from the employees table.

SELECT \* FROM employees

1. How do you find all customers from the state 'Nevada'?

SELECT customer FROM customers WHERE state = ‘Nevada’

1. Write a query to count total sales in the sales table.

SELECT COUNT(sales) FROM sales

1. How to get all products where price is greater than 100?

SELECT \* FROM products WHERE price > 100

1. Select employee names and their salaries from the employees table.

SELECT name, salary FROM employees

1. Write a query to insert a new record into the products table.

INSERT INTO products(product\_id, name, price, min\_qty)

VALUES(43, ‘dinning table’ 14000, 1)

1. How do you update the status of an order to 'Completed'?

UPDATE orders SET status = 'Completed'

WHERE order\_id = 654775;

1. Write a query to delete a customer with customer\_id = 2.

DELETE FROM customer WHERE customer\_id = 2

**Intermediate Level**

1. Write a query to get total sales amount grouped by salesperson.

SELECT salesperson, SUM(sales\_amount) AS total\_sales FROM sales

GROUP BY salesperson

1. Join the sales and products table and show product name with sales quantity.

SELECT p.name, s.quantity FROM sales AS s

INNER JOIN products AS p ON s.product\_id = p.product\_id

1. Get the top 5 products by total sales.

SELECT p.name, s.quantity FROM sales AS s

INNER JOIN products AS p ON s.product\_id = p.product\_id

ORDERBY total\_sales DESC

LIMIT 5

1. How do you get all orders placed after '2023-01-01'?

SELECT \* FROM orders WHERE order\_date > ‘2023-01-01’

1. Write a query to find employees who manage other employees.

SELECT m.manager\_name FROM employees AS e

JOIN managers As m ON e.employee\_id = m.manager\_id

1. Retrieve customer details with their latest order date.

SELECT \* FROM customers

ORDER BY MAX(order\_date) DESC

1. Write a query to calculate total revenue grouped by product category.

SELECT product\_category, SUM(quantity\*price) AS total\_revenue FROM sales

GROUP BY product\_category

1. How to find products that are discontinued and have stock quantity > 0?

SELECT \* FROM products

WHERE is\_discontinued = 1 AND stock\_quantity > 0

**Advanced Level**

1. Write a query to show customers along with total amount they have spent.

SELECT cust\_id, cust\_name, SUM(amount\_spent) AS total\_spent FROM customers

GROUP BY customer\_name

ORDER BY total\_spent DESC

1. Using a join, display all orders with corresponding salesperson details.

SELECT \* FROM orders AS o

JOIN salespersons AS s ON o.salesperson\_id = s. salesperson\_id

1. Find sales with a discount greater than 0.15 and sort by total\_sale descending.

SELECT \* FROM sales

WHERE discount > 0.15

ORDER BY total\_sale DESC

1. Write a query to find the average salary of employees per department.

SELECT department\_name, AVG(salary) AS average\_salary FROM employees

GROUP BY department\_name

ORDER BY average\_salary DESC

1. Write a query to find the region that generated the highest total sales.

SELECT region, SUM(total\_amount) AS total\_region\_sale FROM sales

GROUP BY region

ORDER BY total\_region\_sale DESC

LIMIT 1

1. Find all customers who have placed orders but have no sales recorded in the sales table.

SELECT c.customer\_id, c.customer\_name, s.order\_id, s.order\_date

FROM customers AS c

JOIN orders AS o ON c.customer\_id = o.customer\_id

LEFT JOIN sales AS s On o.order\_id = s.order\_id

WHERE s.sales\_id IS NULL

1. Write a subquery to find products that have never been sold.

SELECT \* FROM products

WHERE product\_id NOT IN (

SELECT DISTINCT product\_id FROM sales

)

1. How would you optimize a slow query joining customers and sales on customer\_id?

Answer- I will follow below Methods

* Select Important columns only
* Filter the data first and then join
* Create Indexes on join columns

1. Write a window function query to rank salespersons based on total sales amount.

⁂