Top SQL Interview Questions for Data Analyst Roles

#### ✔️ SELECT, JOINs, GROUP BY

#### ✔️ Aggregations (SUM, AVG, COUNT)

#### ✔️ Subqueries, CTEs

#### ✔️ Window Functions (ROW\_NUMBER, RANK, etc.)

#### ✔️ Filtering (WHERE, HAVING)

#### ✔️ Data Cleaning using SQL

## ------------------------------------------------------------------------------------------------------------

## 1. Find the total revenue by month

SELECT   
 DATE\_FORMAT(order\_date, '%Y-%m') AS month,  
 SUM(order\_amount) AS total\_revenue  
FROM orders  
GROUP BY DATE\_FORMAT(order\_date, '%Y-%m');

## 2. Show the top 3 products by total sales

SELECT   
 product\_id,   
 SUM(sales\_amount) AS total\_sales  
FROM sales  
GROUP BY product\_id  
ORDER BY total\_sales DESC  
LIMIT 3;

## 3. Get the first purchase date for each customer

SELECT   
 customer\_id,   
 MIN(purchase\_date) AS first\_purchase  
FROM purchases  
GROUP BY customer\_id;

## 4. Calculate the percentage of returning customers

WITH purchase\_counts AS (  
 SELECT customer\_id, COUNT(\*) AS order\_count  
 FROM orders  
 GROUP BY customer\_id  
)  
SELECT   
 ROUND(100 \* SUM(CASE WHEN order\_count > 1 THEN 1 ELSE 0 END) / COUNT(\*), 2) AS returning\_customer\_percentage  
FROM purchase\_counts;

## 5. Find customers who haven’t purchased in the last 30 days

SELECT customer\_id  
FROM customers  
WHERE customer\_id NOT IN (  
 SELECT DISTINCT customer\_id   
 FROM orders   
 WHERE order\_date >= CURDATE() - INTERVAL 30 DAY  
);

## 6. What is the average order value (AOV)?

SELECT   
 ROUND(SUM(order\_amount) / COUNT(DISTINCT order\_id), 2) AS AOV  
FROM orders;

## 7. Rank products by total sales within each category

SELECT   
 category,  
 product\_id,  
 SUM(sales) AS total\_sales,  
 RANK() OVER (PARTITION BY category ORDER BY SUM(sales) DESC) AS sales\_rank  
FROM product\_sales  
GROUP BY category, product\_id;

## 8. Find month-over-month revenue growth

WITH monthly\_revenue AS (  
 SELECT   
 DATE\_FORMAT(order\_date, '%Y-%m') AS month,  
 SUM(order\_total) AS revenue  
 FROM orders  
 GROUP BY month  
)  
SELECT   
 month,  
 revenue,  
 LAG(revenue) OVER (ORDER BY month) AS prev\_month\_revenue,  
 ROUND(((revenue - LAG(revenue) OVER (ORDER BY month)) / LAG(revenue) OVER (ORDER BY month)) \* 100, 2) AS growth\_percentage  
FROM monthly\_revenue;

## 9. Detect duplicate customer emails

SELECT email, COUNT(\*)   
FROM customers  
GROUP BY email  
HAVING COUNT(\*) > 1;

## 10. List the top 5 customers by lifetime value (LTV)

SELECT   
 customer\_id,  
 SUM(order\_amount) AS lifetime\_value  
FROM orders  
GROUP BY customer\_id  
ORDER BY lifetime\_value DESC  
LIMIT 5;