

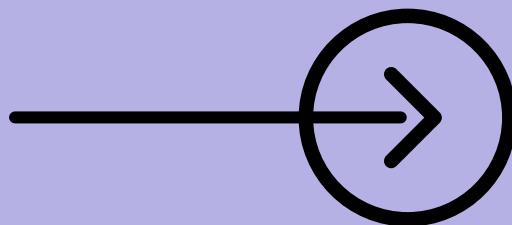
Part - 2

# Advanced SQL Interview

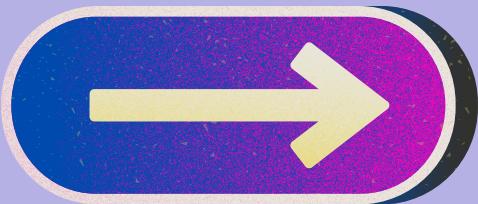
Q & A



*Krishan kumar*  
@Krishan kumar



# **How do you delete duplicates in a SQL table?**



# Delete duplicates in a SQL:

→ To remove duplicates in a SQL table,

**the ROW\_NUMBER() function can be combined with a CTE to identify and delete duplicate records efficiently.**



I want example to understand this concept

# Example Scenario:

→ Suppose you have a table named Employees with columns EmployeeID, FirstName, LastName, and Email.

You want to delete rows with duplicate emails, keeping only the first occurrence.

```
WITH CTE AS (
    SELECT
        EmployeeID,
        ROW_NUMBER() OVER (PARTITION BY Email ORDER BY EmployeeID) AS RowN
    FROM
        Employees
)
DELETE FROM Employees
WHERE EmployeeID IN (
    SELECT EmployeeID FROM CTE WHERE RowNum > 1
);
```

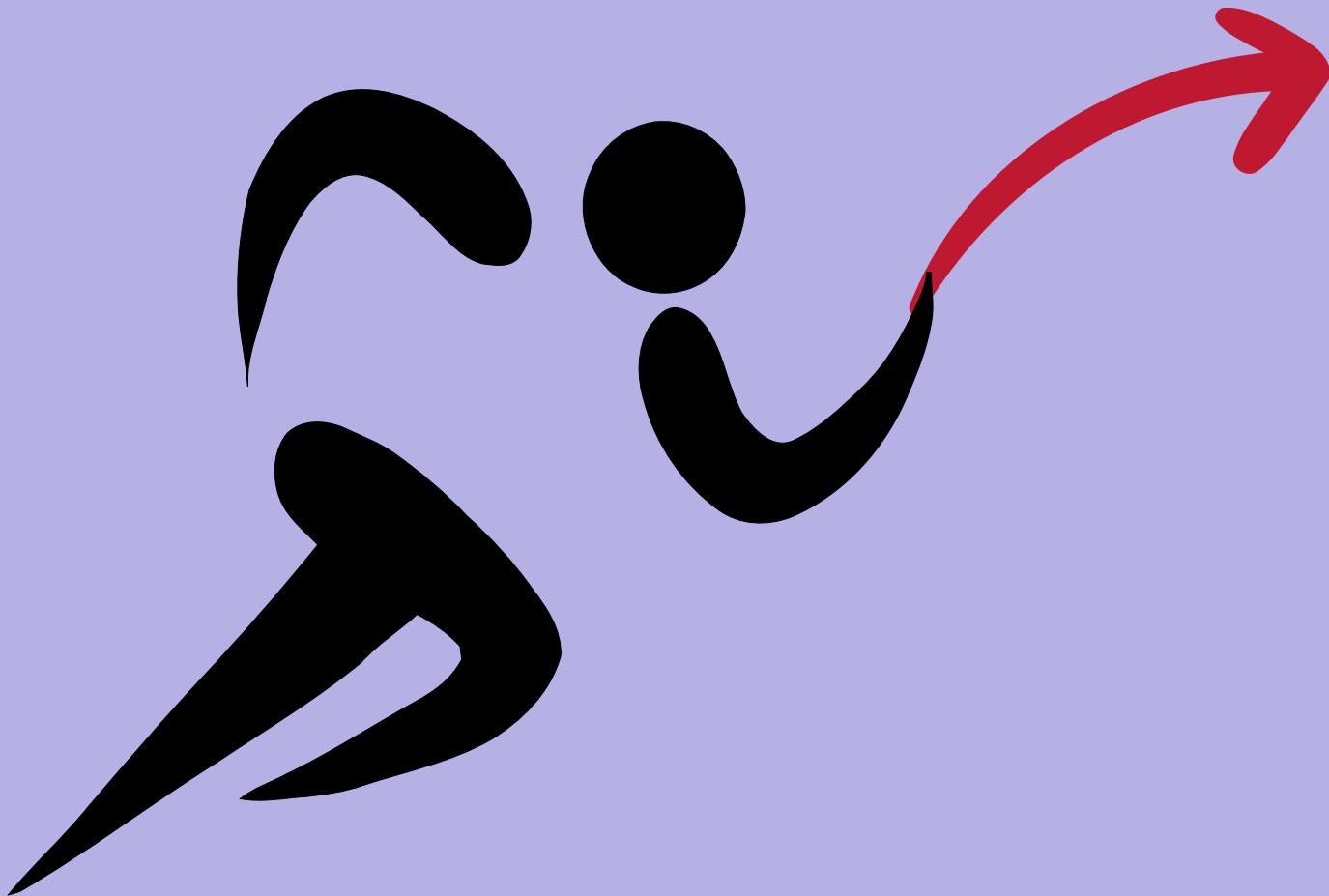


I noted some additional tips for you

# Key Tips:

## → **Tips:**

- **Backup:** Always back up your data before performing delete operations to prevent data loss.
- **Testing:** Test the delete logic first with a **SELECT** statement to confirm you're identifying the correct rows.



**Swipe right for the jackpot of the day!** 😊





wanna see some  
Counter Questions



# **1. How would you handle duplicates if the table has a composite key?**

- **Use the composite key columns in the PARTITION BY clause.**

**For instance, if the duplicate is based on both Email and PhoneNumber,**

**write PARTITION BY Email, PhoneNumber.**

**Next Question**



## **2. What if you want to retain the most recent duplicate based on a timestamp?**

- Adjust the **ORDER BY** in **ROW\_NUMBER()** to use the timestamp in descending order (**ORDER BY Timestamp DESC**),  
ensuring the latest record is assigned  
**RowNum = 1.**



*Next Question*

### **3. Can you delete duplicates without using a CTE?**

→ Yes, a subquery can be used.

**But a CTE generally provides a cleaner and more readable approach.**



*Next Question*

## **4. What happens if the duplicate criterion changes later?**

- **Modify the PARTITION BY clause in ROW\_NUMBER() to reflect the new criteria,**

**Such as additional columns in the partition.**



## 5. How do you ensure performance with large tables?

- Ensure indexes are created on the columns in **PARTITION BY** and **ORDER BY** clauses.

To improve query performance on large tables.



you completed one interview question  
with me,

can you do me a favour



## Find This Useful



**Time to hit that like button  
and give it some love! 😍**

Visit my LinkedIn for such amazing Content 😊

[in krishan kumar](#)