

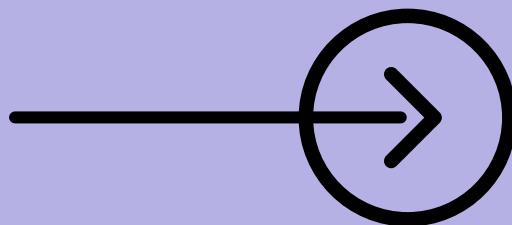
Part - 3

# Advanced SQL Interview

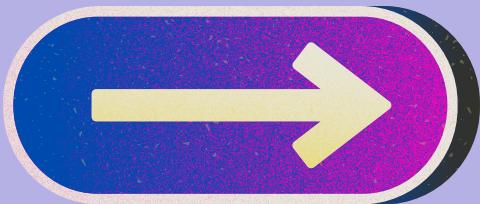
Q & A



*Krishan kumar*  
@Krishan kumar



**How can you get rows that  
are not common in both  
tables?**



# Getting rows that are not common

- To retrieve rows that exist in one table but not the other, you can use a **FULL OUTER JOIN** in SQL along with filtering.

This approach helps in finding the rows that are unique to each table based on a specific key column.



What are the exactly steps to Retrieve Non-Common Rows

# Steps to Retrieve Non-Common Rows:

- • **Use FULL OUTER JOIN:** Perform a **FULL OUTER JOIN** on both tables to include all rows from both tables, with nulls in place of missing matches.
- **Filter Non-Common Rows:** Apply a **WHERE clause** to filter rows where one table's key column is null, indicating that the row exists only in one of the tables.



I want example to understand this concept

# Example Scenario:

→ Let's say you have two tables,

**TableA** and **TableB**, and you want to find rows that are not common in both tables based on a key column, ID.

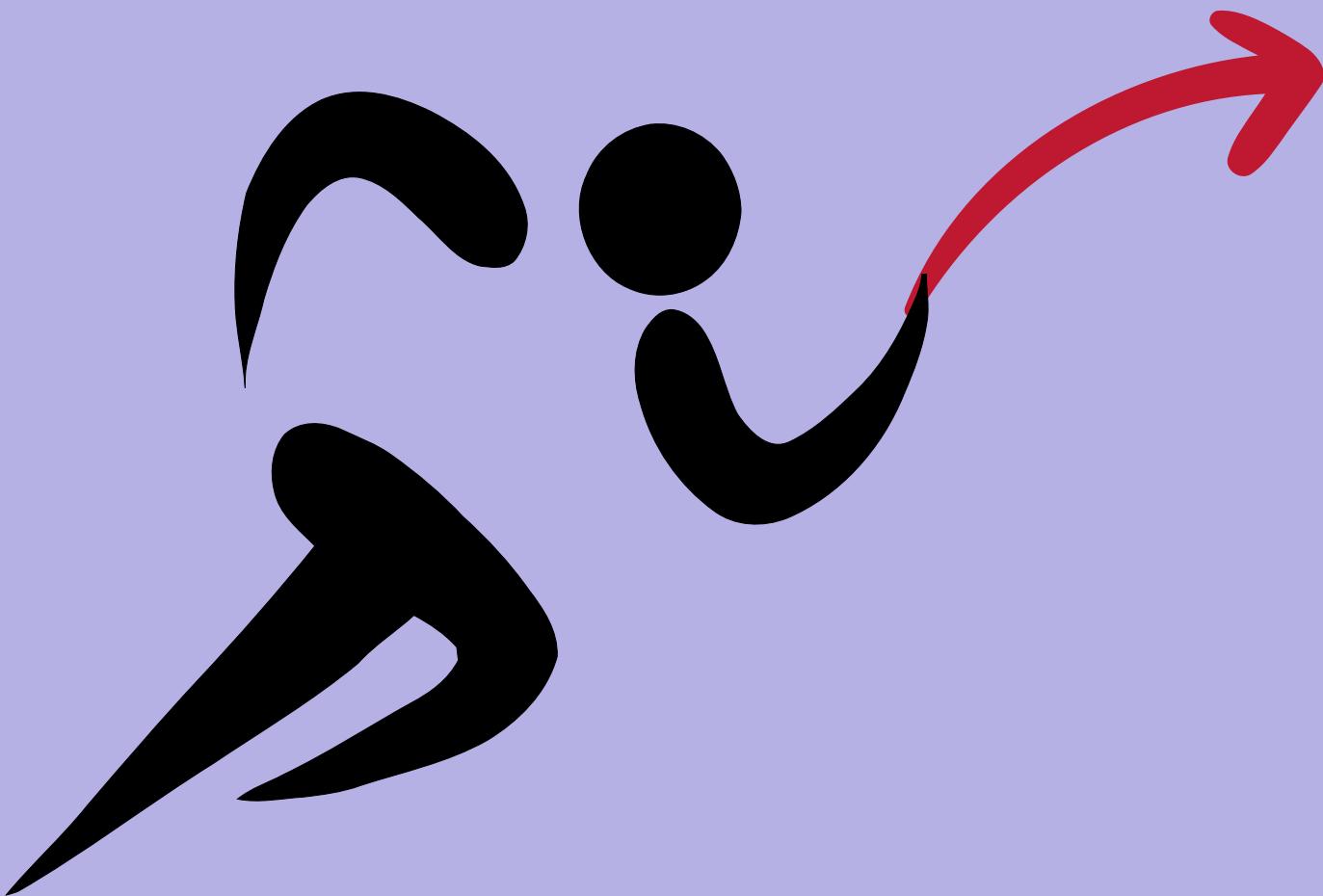
```
● ● ●  
SELECT  
    COALESCE(A.ID, B.ID) AS ID,  
    A.Column1 AS Column1_A,  
    B.Column1 AS Column1_B  
FROM  
    TableA A  
  
FULL OUTER JOIN  
    TableB B ON A.ID = B.ID  
WHERE  
    A.ID IS NULL OR B.ID IS NULL;
```



I noted some key Points: for you

## Key Tips:

- **COALESCE Function:** COALESCE() returns the first non-null value in the provided expressions, ensuring the ID column displays the non-null ID from either TableA or TableB.
- **Understanding NULLs:** Rows with null values in A.ID or B.ID signify the row exists in one table but lacks a match in the other, which identifies it as a non-common row.



*Swipe for the jackpot of the day* 😊





wanna see some  
Counter Questions



# **1. What if you only want rows that exist in TableA but not in TableB?**

- Use a **LEFT JOIN** instead of **FULL OUTER JOIN** and add **WHERE B.ID IS NULL**.

**This retrieves only rows that exist in TableA without any matches in TableB.**



## **2. How can you identify rows present only in TableB but not in TableA?**

→ Use a **RIGHT JOIN** and include **WHERE A.ID IS NULL** in the query.

**This will return rows that exist only in TableB without any corresponding rows in TableA.**



*Next Question*

### **3. Can this method handle multiple key columns?**

→ Yes, specify each key column in the **ON** clause and adjust the **WHERE** clause to check for nulls on all keys.

**This approach works as long as each column used for matching is consistent in both tables.**



*Next Question*

## **4. How do you get common rows between TableA and TableB instead?**

- Replace the **WHERE** clause with **WHERE A.ID IS NOT NULL AND B.ID IS NOT NULL.**

**This will filter out only the rows that are present in both tables, showing common rows.**



## **5. Is there an alternative to FULL OUTER JOIN for finding non-common rows?**

- Yes, you can use UNION and EXCEPT to identify rows unique to each table.

**While effective, this method may be more complex than FULL OUTER JOIN but works well for simple comparisons.**



**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](#)