Part - 1

# Advanced SQL Interview Q & A !!

Krishan kumar

@Krishan kumar

# How do you find duplicates in a given table?

# Finding Duplicates:

- To identify duplicates in a SQL table, you can use the GROUP BY and HAVING clauses.

  This method groups the data based on columns you want to check for duplicates and filters groups with more than one occurrence.

# Example Scenario:

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

  **To find duplicate records based on the Email column, the query would look like this:**

```
SELECT Email, COUNT(*)
FROM Employees
GROUP BY Email
HAVING COUNT(*) > 1;
```

- **Output:**

  **The result will list duplicate email addresses along with the count of how often each appears in the table.**

**Getting Detailed Duplicate Records**

# Detailed Duplicate Records:

➡ **To view full details for each duplicate row, you can use a CTE (Common Table Expression) or a subquery:**

```sql
WITH DuplicateEmails AS (
    SELECT Email
    FROM Employees
    GROUP BY Email
    HAVING COUNT(*) > 1)
SELECT e.*
FROM Employees e
JOIN DuplicateEmails d ON e.Email = d.Email;
```

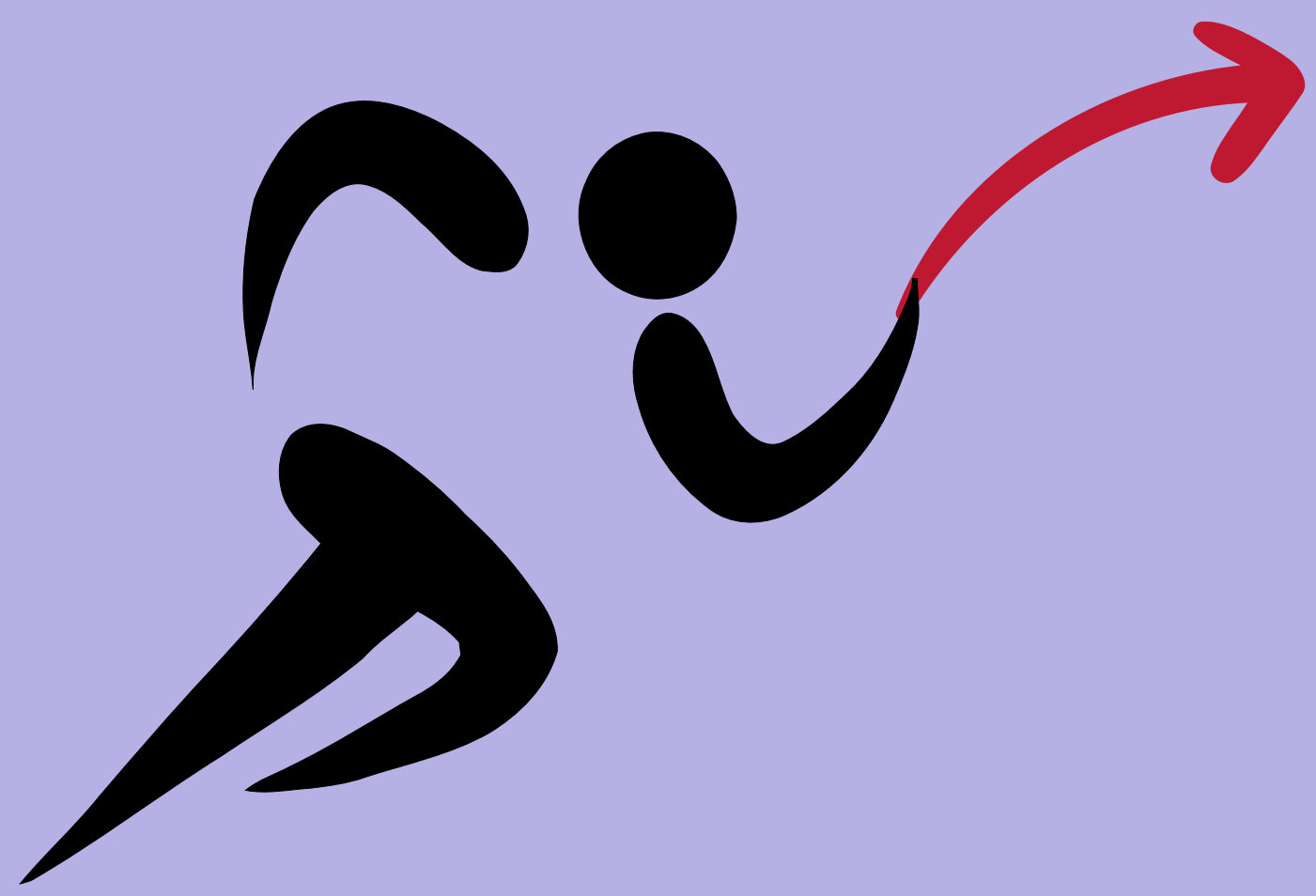
I create some additional Notes

# Additional Notes:

## Notes:

- **Flexible Columns:** To find duplicates based on multiple columns, replace Email with those columns in both the SELECT and GROUP BY clauses.

- **Performance:** For large tables, ensure your columns used for filtering are indexed to speed up the duplicate detection process.



Swipe right for the jackpot of the day 😉

wanna see some

Counter Questions

# 1. Can you find duplicates based on more than one column?

- **Yes, just add the additional columns to both SELECT and GROUP BY.**

**For example, GROUP BY FirstName, LastName.**

# 2. What is the difference between GROUP BY and PARTITION BY when finding duplicates?

➡ **GROUP BY groups the entire result set and is paired with HAVING for filtering duplicates.**

**PARTITION BY is used within window functions and can't directly find duplicates.**

Next Question

# 3. How can you handle duplicate entries in SQL after finding them?

- **You can delete duplicates using a DELETE with a ROW_NUMBER() function in a CTE to isolate duplicates.**

# 4. Are there other methods to find duplicates without GROUP BY?

- **Yes, using the ROW_NUMBER() window function with PARTITION BY and filtering rows where ROW_NUMBER > 1.**

# 5. Does this approach work with NULL values?

➡ **Yes, GROUP BY treats NULLs as equal,**

**so all NULLs in a column will be grouped together.**

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.

# 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.**

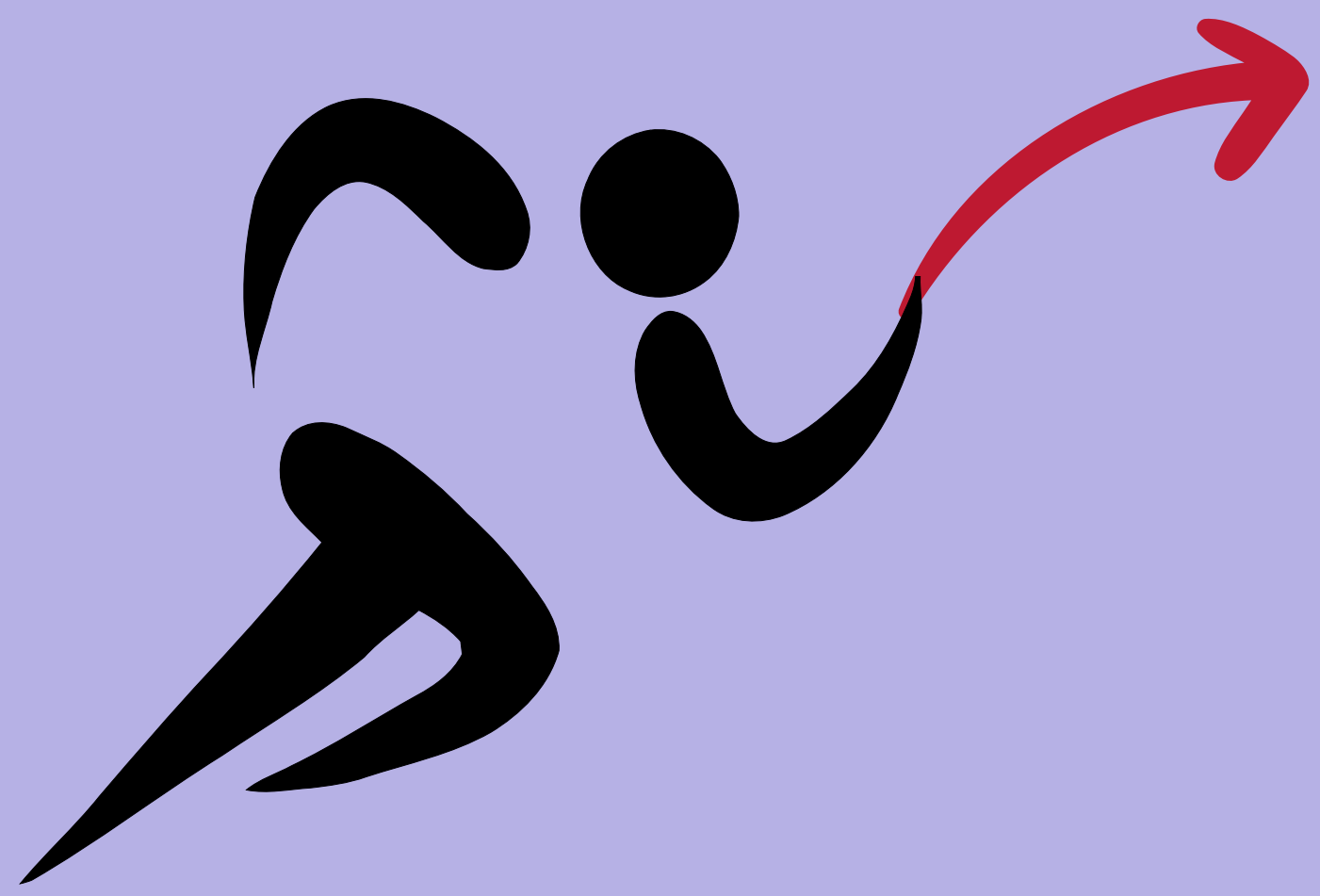
```sql
WITH CTE AS (
    SELECT
        EmployeeID,
        ROW_NUMBER() OVER (PARTITION BY Email ORDER BY EmployeeID) AS
RowNum FROM
        Employees
)
DELETE FROM Employees
WHERE EmployeeID IN (
    SELECT EmployeeID FROM CTE WHERE RowNum > 1
);
```

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.**

# 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.**

# 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.**

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.

# 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.**

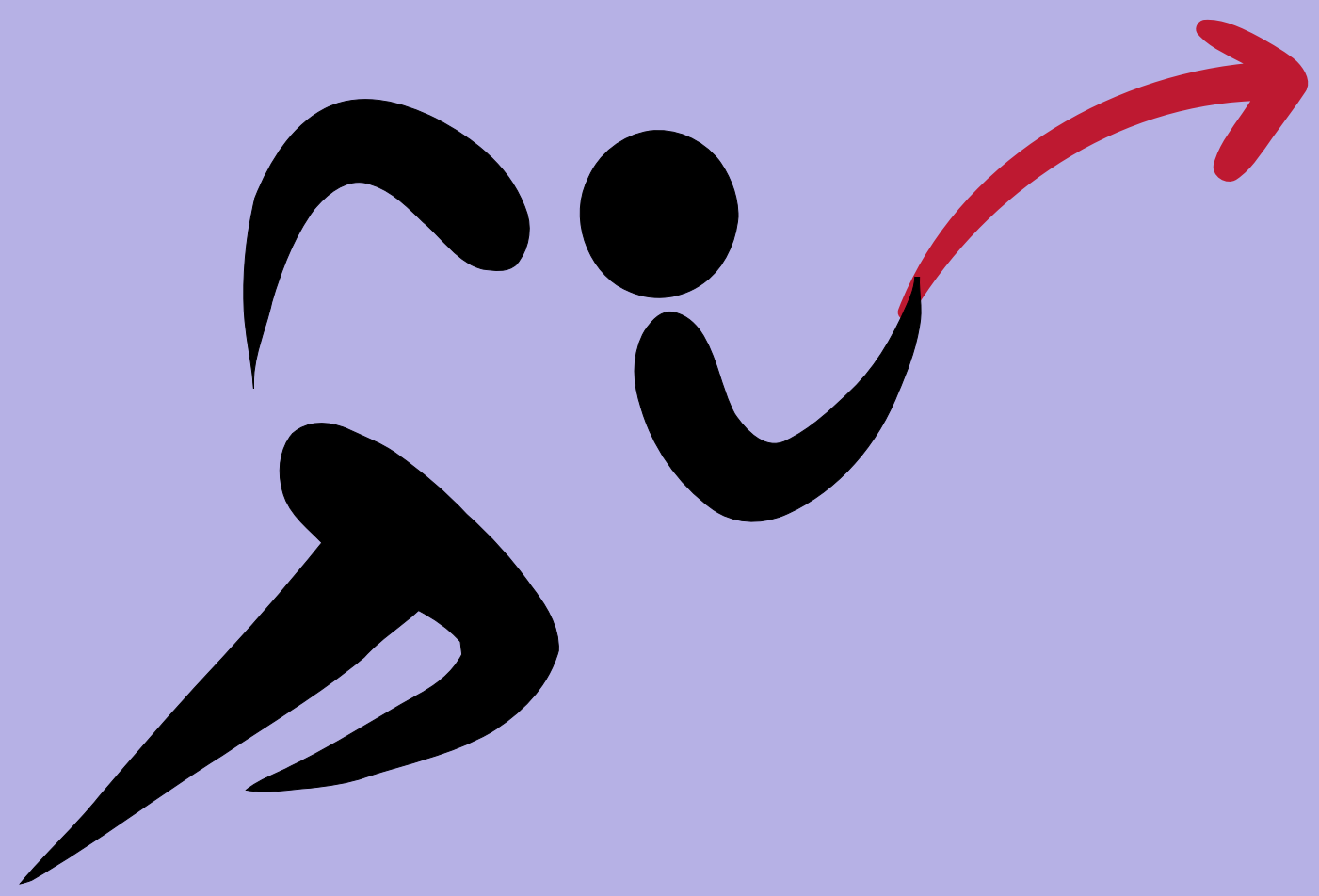
```sql
SELECT
    COALESCE(A.ID, B.ID) AS ID,
    A.Column1 AS Column1_A,
    B.Column1 AS Column1_B
FROM
    TableA A

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.**

# 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.**

Part - 4

# Advanced SQL Interview Q & A !!

*Krishan kumar*

@Krishan kumar

# How can you select unique records from a table?

# Select unique records from a table

➡ **To select unique records from a table in SQL, you can use the DISTINCT keyword in your SELECT statement.**

What are the exactly Steps to Select Unique Records

# Steps to Select Unique Records:

- **Use DISTINCT Keyword:** Include the DISTINCT keyword immediately after the SELECT keyword in your query.

# Example Scenario:

- **Suppose you have a table named Employees with columns EmployeeID, FirstName, LastName, and DepartmentID.**

**To select unique combinations of FirstName and LastName,**

**you would write:**

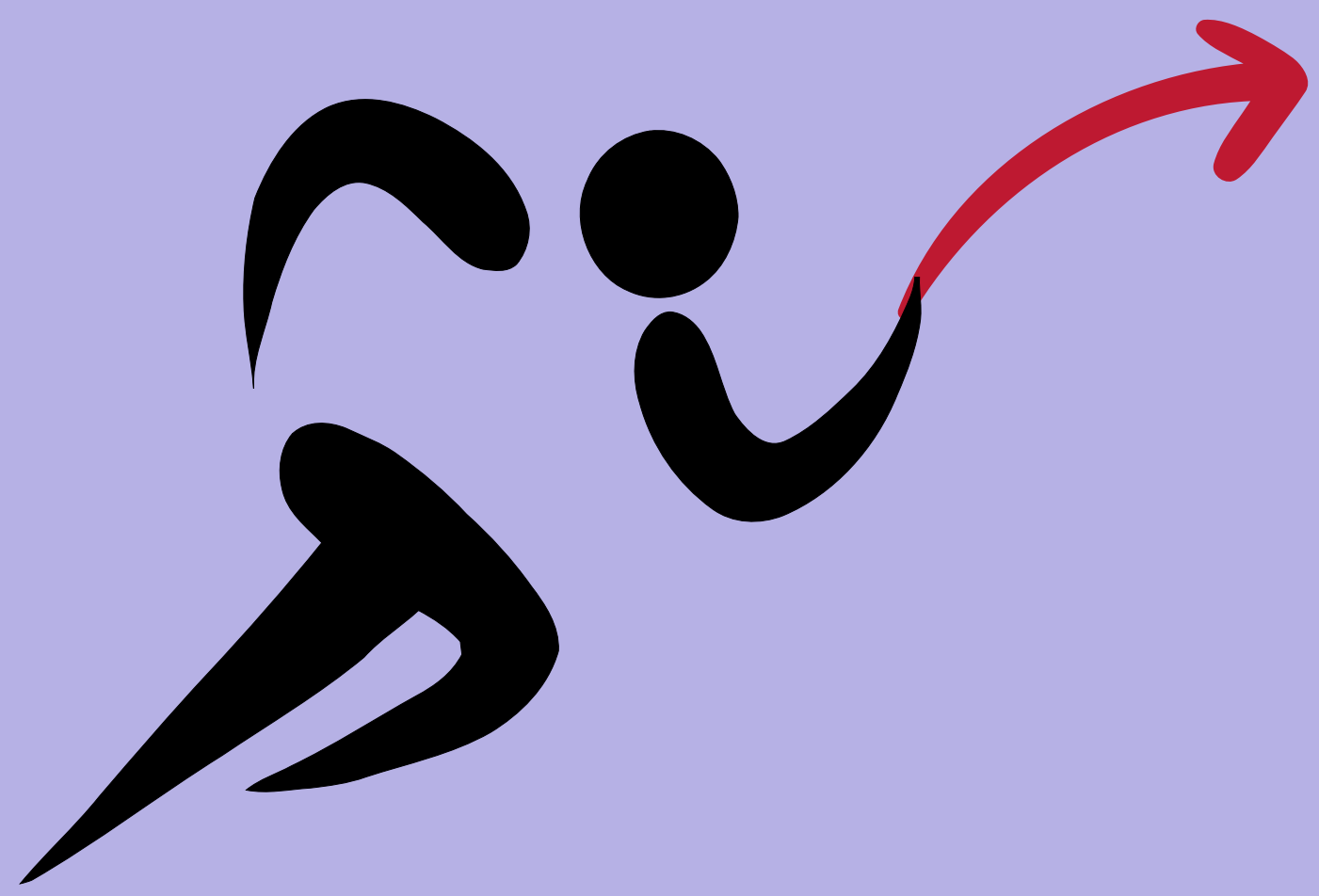
```
SELECT SUBSTRING(column_name, start_position, length)
FROM table_name;
```

I noted some key Points: for you

# Key Tips:

- **Column Selection:** You can specify multiple columns after the SELECT DISTINCT keywords to retrieve unique combinations of those columns.

- **Ordering:** The DISTINCT keyword applies to the entire row, so the query returns unique combinations of all selected columns.



Swipe for the jackpot of the day 😉

wanna see some

Counter Questions

# 1. How does the DISTINCT keyword differ from using GROUP BY?

- **DISTINCT is used to retrieve unique combinations of values across selected columns in the result set.**

  **In contrast, GROUP BY groups identical values and is typically used with aggregate functions like COUNT, SUM, etc., to summarize grouped data.**

# 2. Can you explain how the DISTINCT keyword affects performance?

➡ **The DISTINCT keyword can slow down queries, especially on large datasets,**

**as it requires sorting and comparing rows to remove duplicates.**

**Using it with well-indexed columns helps reduce performance issues.**

Next Question

# 3. How would you modify the query to select unique records based on a specific column?

- **Simply specify the desired column(s) after the DISTINCT keyword in the SELECT statement.**

**For example, SELECT DISTINCT DepartmentID FROM Employees; returns unique department IDs.**

Next Question

# 4. Can DISTINCT be combined with aggregate functions?

- **Yes, you can use DISTINCT within aggregate functions, like**

  **COUNT(DISTINCT ColumnName), to count unique values in a specific column.**

# 5. Are there any alternatives to using DISTINCT?

- **An alternative to DISTINCT is using ROW_NUMBER() with a CTE or subquery,**

  **allowing you to remove duplicates with more control over which duplicates to keep in the result set.**

Part - 5

# Advanced SQL Interview Q & A !!

*krishan kumar*

@Krishan kumar

How can you convert a text into date format ?

Consider the given text as 31-01-202

# Convert a text into date format:

- To convert a text into a date format in SQL, you can use the CAST or CONVERT functions, specifying the appropriate format for the input text.

Here's how you can convert the given text '31-01-2023' into a date format:

Can you explain both functions with an example

# Example:

## Using CAST Function:

```
SELECT CAST('31-01-2023' AS DATE) AS ConvertedDate;
```

## Explanation:

**The CAST function converts an expression of one data type to another.**

**In this case, it converts the text '31-01-2023' into a date format.**

How we got this using CONVERT Function

# Example:

## Using CONVERT Function:

```
SELECT CONVERT(DATE, '31-01-2023', 105) AS
ConvertedDate;
```

## Explanation:

**The CONVERT function is similar to CAST but allows you to specify a style parameter to indicate the format of the input text.**

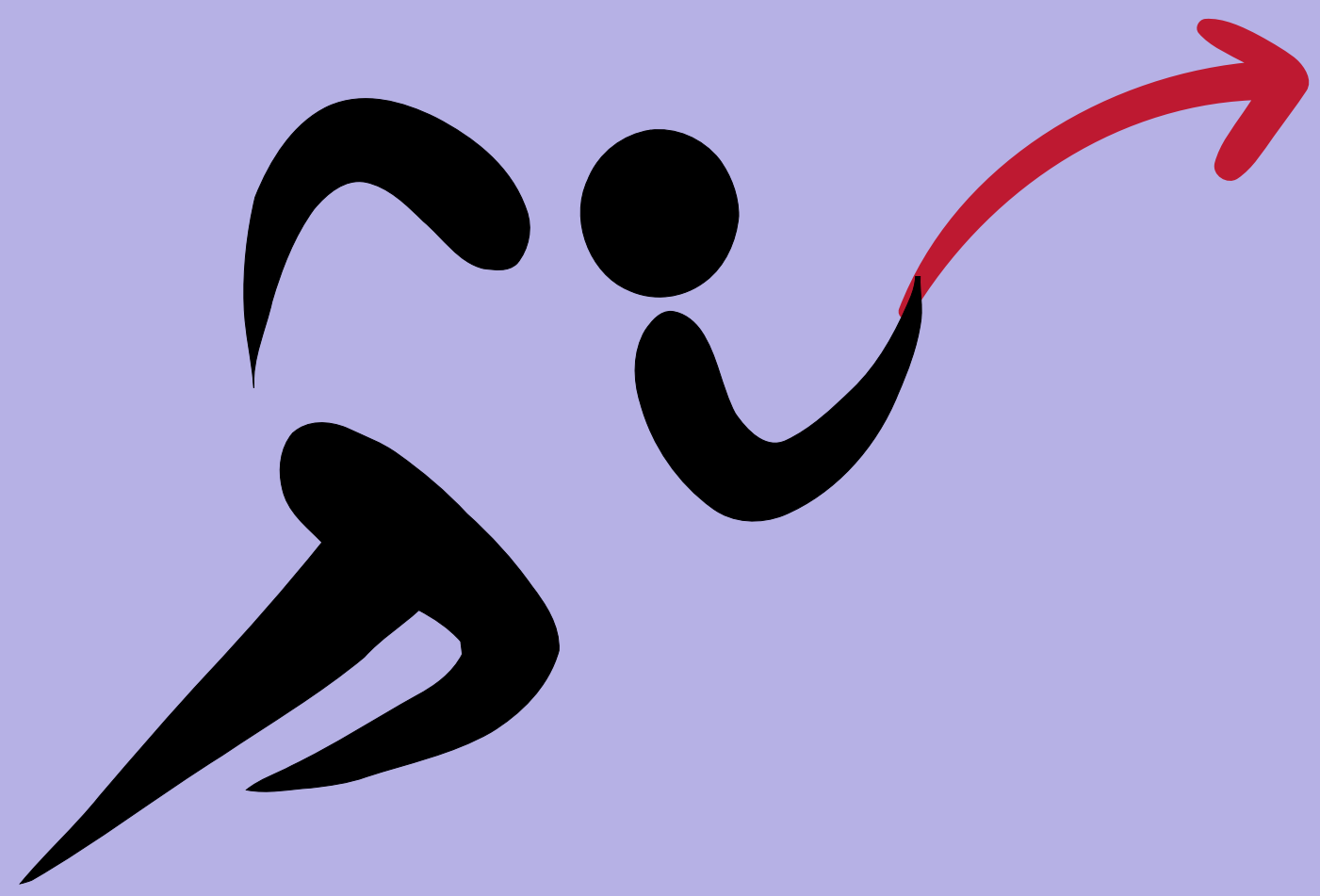
**Here, style 105 corresponds to the Italian date format 'dd-mm-yyyy'.**

I noted some key Points: for you

# Key Tips:

- **Input Format:** Ensure that the input text matches the specified format for date conversion.

- **Date Format Codes:** Different databases may use different style codes for date formats. Refer to the documentation for your specific database system for the correct style code.



Swipe for the jackpot of the day 😉

wanna see some

Counter Questions

# 1. What is the difference between CAST and CONVERT functions?

- **CAST is ANSI standard and widely used for converting data types without format specifications.**

  **CONVERT allows additional format options via style codes, but its usage can vary slightly across database systems.**

## 2. Can I use these functions for other date formats?

➡ **Yes, you can specify different formats in CONVERT by changing the style code.**

**For instance, style 101 is for the U.S. format mm-dd-yyyy.**

Next Question

# 3. How does the database handle invalid date formats?

- **If the text doesn’t match the specified date format, the query may result in an error or NULL.**

**It's essential to validate formats before conversion.**

## 4. Can I format dates without changing their data type?

➡ **Yes, you can use FORMAT() or database-specific date functions to display dates in various formats without converting the data type.**

# 5. What if I need to convert dates in bulk?

- You can apply CAST or CONVERT in the SELECT statement for each row,

  Or create a view to handle all date conversions automatically.