

Gemini SQL Assistant (Charan_sols company DB)

Ask questions in English. Get SQL queries and results instantly!

Enter your question whenever:

show the third highest employee salary

Try These Examples



Run

Generated SQL Query:

```
LECT employee_salary FROM Charan_sols ORDER BY employee_salary DESC LIMIT 1 OFFSET
```

Query Result:

	employee_salary
0	75000

Gemini Explains the SQL



Okay, let's break down this SQL query step-by-step:

1. `SELECT employee_salary`

- What it does:** This part tells the database *what* you want to retrieve.
- In simple terms:** "Give me the values from the column named `employee_salary` ." Imagine you have a table with many columns, and this specifies that you only want to see the contents of one particular column: "employee_salary".

2. `FROM Charan_sols`

- What it does:** This specifies the table where the `employee_salary` data is located.

- **In simple terms:** "Look for the `employee_salary` column in the table called `Charan_sols`." `Charan_sols` is likely the name of a table in your database that stores information about employees.

3. `ORDER BY employee_salary DESC`

- **What it does:** This sorts the results (the `employee_salary` values) in a specific order.
- **In simple terms:** "Arrange the salaries from highest to lowest."
 - `ORDER BY employee_salary` : This part means "sort by the `employee_salary` column."
 - `DESC` : This is short for "descending," meaning arrange in reverse order (highest to lowest). If you had `ASC` (ascending), it would sort from lowest to highest.

4. `LIMIT 1`

- **What it does:** This restricts the number of results returned.
- **In simple terms:** "Give me only one result (the first result after the sorting and any offset)."

5. `OFFSET 2`

- **What it does:** This skips a specified number of rows before starting to return results.
- **In simple terms:** "Skip the first two salaries (the very highest and the second highest), and *then* start giving me results."

Putting it all together:

The query essentially asks the database to:

1. Go to the `Charan_sols` table.
2. Grab the values from the `employee_salary` column.
3. Sort those salaries from highest to lowest.
4. Skip the first two highest salaries.
5. Return the salary that comes *after* those two (meaning, the *third* highest salary).

In other words, this query finds the third highest employee salary in the `Charan_sols` table.