How to Find Nth Highest Salary from a Table? →

Last Updated : 11 Apr, 2025

Finding the second-highest salary in SQL is a common requirement in data analysis and SQL interviews. This query is important for applications where tracking **employee ranks**, calculating **percentile based bonuses**, or analysing **hierarchical data** is necessary. Understanding how to retrieve the second-highest salary can be essential for data insights and optimizing queries.

Understanding the Second-Highest Salary Query

The **second-highest salary** query is used to find the employee who **ranks second** among **salaries** in a dataset. It's a common task for business analysts, HR departments, and database administrators. This query is a subset of **ranking** and **filtering operations** and can be extended to find the **Nth highest salary**.

Real World Use Cases:

- **Ranking Employees:** Help HR determine employee rankings based on salary.
- **Data Analysis:** Analyse salary distribution trends.
- **Reward Calculations:** Assign percentile-based rewards based on salary ranks.

Methods to Find the Second-Highest Salary in SQL

Given an **employee table** with the following columns: `name` and `salary`, the goal is to find the employee(s) with the **second-highest salary**. Below is an example of such a dataset:

Employee Table

Name	Salary
Aman	100000
Shubham	1000000
Naveen	40000
Nishant	500000

Skip to content

1. Using Subqueries

A **subquery** can be used to exclude the **maximum salary** and find the **second-highest salary**. Below is a simple query to find the **employee** whose salary is the **highest** and exclude it to get the second-highest salary.

Query:

```
SELECT name, salary
FROM employee
WHERE salary = (
    SELECT MAX(salary)
    FROM employee
    WHERE salary < (SELECT MAX(salary) FROM employee)
);
```

Output

Name	Salary
Nishant	500,000

Note: If you're using **MySQL**, be aware that we may encounter **ERROR 1140** when running this query. This happens because the `name` column is unaggregated while using `MAX(salary)`

2. Using LIMIT (MySQL -Specific)

In MSQL, the LIMIT clause is often used to **retrieve** the **desired row** based on rank. We can nest the previous query to find the **second-highest salary**.

Query:

```
select *from employee
group by salary
order by  salary desc limit 1,1;
```

Output

Skip to content

Name	Salary
Nishant	500000

Note: The **LIMIT 1,1** clause is used to fetch one row starting from the second row in a descending order.

3. Using Common Table Expressions (CTEs) with Ranking Functions

In **SQL Server** and other databases that support **ranking functions**, such as **DENSE_RANK()**, we can make the process of finding the second-highest salary much simpler and more flexible.

Query:

```
WITH RankedSalaries AS (
    SELECT name, salary,
           DENSE_RANK() OVER (ORDER BY salary DESC) AS Rank
    FROM employee
)
SELECT name, salary
FROM RankedSalaries
WHERE Rank = 2;
```

Output:

Name	Salary
Nishant	500,000

Note : Why Use Ranking Functions? Using **DENSE_RANK()** ensures that the query accounts for ties in salaries. If two employees have the same highest salary, the next row will be ranked as the second-highest salary.

4. Finding the Nth-Highest Salary

To generalize the query for the **Nth highest salary**, instead of nesting for the **second**, **third**, etc largest salary, we can find the **nth salary** using a general query like in MySQL:

MySQL Query:

Skip to content

```
SELECT salary
FROM employee
ORDER BY salary desc
limit n-1,1
```

Generic Query:

```
SELECT name, salary
FROM employee A
WHERE n-1 = (SELECT count(1)
             FROM employee B
             WHERE B.salary>A.salary)
```

5. Handling Duplicates in Salary

What if multiple employees have the same salary and you still need to find the second-highest salary? In this case, you can use a **DISTINCT** clause to ensure you only work with unique salary values.

Query for the 4th highest salary

```
SELECT * FROM employee
WHERE salary= (SELECT DISTINCT(salary)
FROM employee ORDER BY salary DESC LIMIT 3,1);
```

General Query for Nth Highest Salary

```
SELECT * FROM employee
WHERE salary= (SELECT DISTINCT(salary)
FROM employee ORDER BY salary DESC LIMIT n-1,1);
```

Explanation: This query finds the **4th highest salary** by selecting distinct salary values and using `LIMIT` to skip the first three unique salaries

6. Troubleshooting Common Errors (ERROR 1140)

While querying the database to fetch an employee with maximum salary, we might get **ERROR 1140**, specifically when **GROUP BY** is not used correctly with aggregated queries.

Error Message:

Skip to content

```
ERROR 1140 (42000): In aggregated query without GROUP BY, expression #1 of SELECT list contains
nonaggregated column...
```

We can clearly understand from the error message that our query is an **aggregation query** from '**MAX(salary)**' and at the same time, it uses an unaggregated column '**name**', which creates ambiguity for MySQL. This error will be captured by those versions of MySQL which contain the value '**only_full_group_by**' in the 'sql_mode' variable. We can check this variable in **MySQL** using the following command.

```
SHOW VARIABLES LIKE "sql_mode";
```

Solution for ERROR 1140:

To avoid confusion, we must avoid using the **aggregated column** and the **unaggregated column** in the same query to eliminate this error. The following command would help in this.

```
// EMPLOYEE WITH HIGHEST SALARY
SELECT name, salary FROM employee ORDER BY salary DESC LIMIT 1;

// EMPLOYEE WITH SECOND HIGHEST SALARY
SELECT name, salary FROM employee WHERE salary < (SELECT MAX(salary) FROM employee) ORDER BY salary
DESC LIMIT 1;

// EMPLOYEE WITH Nth HIGHEST SALARY
SELECT name, salary FROM employee ORDER BY salary DESC LIMIT (N-1), 1;
```

Conclusion

This guide explained multiple ways to find the **second-highest salary** in SQL using subqueries, **CTEs**, and **LIMIT functions**. These techniques are essential for solving **real-world** database **problems** and acing **SQL interview questions**. Whether ranking employees or analyzing **salary trends**, mastering these queries enhances your **data analysis skills** and prepares you for handling **complex datasets**.

Comment More info Advertise with us

Next Article

How to Find Nth Highest Salary from a Table?

Similar Reads

Skip to content

PL/SQL Query to List the Second Highest Salary By Department

Fetching the highest salary or nth lowest salary can be considered a common task when we are dealing with some real case scenarios. We often face such scenarios when there is sales performance analysis, merit-based promotions, salary benchmarking, etc. These are a few use cases to fetch nth...

5 min read

SQL Query to Find the Highest Salary of Each Department

Structured Query Language (SQL) is a standard database language widely used for managing, maintaining, and retrieving data from relational databases like MySQL, Oracle, and Microsoft SQL Server. In this article, we will demonstrate how to use SQL to find the highest salary in each department. This i

2 min read

List the Second Highest Salary By Department

Listing the second-highest salary by department is a common task that explains the use of advanced PostgreSQL techniques. It offers several functions and methods to achieve this. In this article, we will explore three approaches that help us to List the Second Highest Salary By Department with the h

5 min read

How to find Nth highest salary from a table?

Finding the Nth highest salary in a database table is a common challenge and interview question for SQL enthusiasts and professionals. This tutorial will guide you step-by-step through multiple methods of solving this problem in SQL, from using ranking functions to subqueries and the LIMIT...

6 min read

How to Limit Query Results in SQL?

SQL provides powerful features like the LIMIT clause to control the number of rows returned, making it an essential tool for optimizing queries and retrieving precise results. Whether weâ€™re managing large datasets or working on performance tuning, understanding how to limit query results is...

4 min read

PL/SQL Query to List the First 50% Rows in a Result Set

Listing the First 50% Rows in a Result Set can be considered as a normal day-to-day problem. When dealing with data analysis-related tasks, especially with large data sets, fetching all the data at once might create a problem. Getting the first 50% of rows can help in performing the initial analysis

5 min read

SQL Query to Find Monthly Salary of Employee If Annual Salary is Given

SQL stands for Structured Query Language, which used in the database to retrieve data, update and modify data in relational databases like MySql, Oracle, etc. And a query is a question or request for data from the database, that is if we ask someone any question then the question is the query. Simil

3 min read

SQL Query to Get the Latest Record from the Table

Fetching the latest record from a table is a frequent and essential task when managing relational databases. Whether you want to retrieve all columns or a specific subset of them, SQL provides a variety of techniques to accomplish this efficiently. In this article, we will explain how to retrieve th

5 min read

Skip to content

PL/SQL Query to List the Last 25% Rows in a Result Set

We sometimes need to fetch some specific rows from our result set. There are many reasons to do so. When we are working on real-time analytics, focusing on recent data is then our need. This helps us in quicker decision-making as it provides us the quicker insight into the latest data. PL/SQL is a p

5 min read

How to Use SQL DISTINCT and TOP in Same Query?

Structured Query Language (SQL) is a computer language used to interact with relational databases. It allows us to organize, manage, and retrieve data efficiently. In this article, we will explain how to use the DISTINCT keyword and the TOP clause together in a query, explaining their purpose, usage

4 min read

GeeksforGeeks
Sanchhaya Education Private Limited

Corporate & Communications Address:
A-143, 7th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar
Pradesh (201305)

Registered Address:
K 061, Tower K, Gulshan Vivante
Apartment, Sector 137, Noida, Gautam
Buddh Nagar, Uttar Pradesh, 201305

GET IT ON Google Play
Download on the App Store

Advertise with us

Company
- About Us
- Legal
- Privacy Policy
- In Media
- Contact Us
- Advertise with us
- GFG Corporate Solution
- Placement Training Program

Languages
- Python
- Java
- C++
- PHP
- GoLang
- SQL
- R Language
- Android Tutorial
- Tutorials Archive

DSA
- Data Structures
- Algorithms
- DSA for Beginners
- Basic DSA Problems
- DSA Roadmap
- Top 100 DSA Interview Problems
- DSA Roadmap by Sandeep Jain
- All Cheat Sheets

Data Science & ML
- Data Science With Python
- Data Science For Beginner
- Machine Learning
- ML Maths
- Data Visualisation
- Pandas
- NumPy
- NLP
- Deep Learning

Web Technologies
- HTML
- CSS
- JavaScript
- TypeScript
- ReactJS
- NextJS
- Bootstrap
- Web Design

Python Tutorial
- Python Programming Examples
- Python Projects
- Python Tkinter
- Python Web Scraping
- OpenCV Tutorial
- Python Interview Question
- Django

Computer Science
- Operating Systems
- Computer Network
- Database Management System
- Software Engineering
- Digital Logic Design
- Engineering Maths
- Software Development
- Software Testing

DevOps
- Git
- Linux
- AWS
- Docker
- Kubernetes
- Azure
- GCP
- DevOps Roadmap

System Design
- High Level Design
- Low Level Design
- UML Diagrams
- Interview Guide
- Design Patterns
- OOAD
- System Design Bootcamp
- Interview Questions

Inteview Preparation
- Competitive Programming
- Top DS or Algo for CP
- Company-Wise Recruitment Process
- Company-Wise Preparation
- Aptitude Preparation
- Puzzles

School Subjects
- Mathematics
- Physics
- Chemistry
- Biology
- Social Science
- English Grammar
- Commerce
- World GK

GeeksforGeeks Videos
- DSA
- Python
- Java
- C++
- Web Development
- Data Science
- CS Subjects