Nth Highest Salary in Oracle

Nth Highest Salary

Q. “How can I select the Nth highest salary of the EMP table?”

This is a question that every Oracle newbie stumbles over. Ask it on a forum and you’re pointed to the archives. That gets you nowhere as when you search the archives, all you find is a host of other messages also asking you to search the archives.

Here comes the answer to the problem of finding the Nth highest salary. You can extend this logic to find the Nth highest row of any table.

First things first: The question is ambiguous!

Let’s say this is your data:

Name Salary

KING 5000

FORD 3000

SCOTT 3000

JONES 2975

BLAKE 2850

CLARK 2850

ALLEN 1600

Q. Who is second – FORD or SCOTT or both?

What will you say about JONES’s salary – is it the 3rd highest salary, or the 4th highest?

If you are looking for the set of people earning the Nth highest salary, with no gaps in case of ties, then JONES should be ranked 3rd, after KING [5000, 1st], followed by FORD and SCOTT [both 3000, 2nd].

If you are looking for exact ranks with gaps if there are ties, then JONES is the 4th highest paid employee, as there are 3 people earning more than him – KING, FORD and SCOTT. In this system of ranking, FORD and SCOTT are 2nd jointly and no employee is 3rd.

This is how your ranks will look, in the 2 cases:

Scenario 1: No gaps in case of ties

Name	Salary	Rank
KING	5000	1
FORD	3000	2
SCOTT	3000	2
JONES	2975	3
BLAKE	2850	4
CLARK	2850	4
ALLEN	1600	5

Scenario 2: Gaps in case of ties

Name	Salary	Rank
KING	5000	1
FORD	3000	2
SCOTT	3000	2
JONES	2975	4
BLAKE	2850	5
CLARK	2850	5
ALLEN	1600	7

Once you have your question sorted out –

(a) Set of people earning the Nth highest salary, with continuous ranks if there are ties, OR

(b) Set of people earning the Nth highest salary, with skipped rank numbers if there are ties

Then you can proceed to writing the queries.

Scenario 1: DENSE_RANK () for Nth highest row, no gaps in case of ties

The analytic function dense_rank() will rank the rows with no gaps in ranking sequence if there are ties.

The ranks are calculated as:

```
SQL>    select ename,sal,dense_rank() over (order by sal desc) ranking from emp;

ENAME      SAL      RANKING
---------- ---------- ----------
KING       5000     1
FORD       3000     2
SCOTT      3000     2
JONES      2975     3
CLARK      2850     4
BLAKE      2850     4
ALLEN      1600     5
```

Wrap a filter around and pick out the Nth highest salary, say the 4th highest salary.

```
    select * from
   (select ename,sal,dense_rank() over (order by sal desc) ranking
    from emp )
   where ranking = 4; -- Replace 4 with any value of N

ENAME                SAL      RANKING
---------- ---------- ----------
BLAKE                2850             4
CLARK                2850             4
```

The 4th position has a tie between BLAKE and CLARK.

Scenario 2: RANK () for Nth highest row, gaps in case of ties

The analytic function rank() will rank the rows with gaps in ranking sequence if there are ties.

The ranks are calculated as:

```
SQL> select ename,sal,rank() over (order by sal desc) ranking from emp;

ENAME      SAL       RANKING
---------- ---------- ----------
KING       5000              1
FORD       3000              2
SCOTT      3000              2
JONES      2975              4
CLARK      2850              5
BLAKE      2850              5
ALLEN      1600              7
TURNER     1500              8
```

Wrap a filter around and pick out the Nth highest salary, say the 4th highest salary.

```
 select * from
( select ename,sal,rank() over (order by sal desc) ranking from  emp )
where ranking = 4 -- Replace 4 with any value of N
```

```
ENAME                SAL      RANKING
---------- ---------- ----------
JONES                2975         4
```

A different answer from the previous query, as there is no rank 3 because of the tied 2nd place.

Closing Notes

The requirement to “find Nth highest row” is incomplete, until the following questions are also answered:

Can the result match more than one value? If not, on what basis should the one record be chosen if there is a tie?

How should the subsequent records be ranked in case of ties – contiguously or with gaps?

Depending on the answer for (2), DENSE_RANK (for contiguous) or RANK (for gaps) can be used. Depending on the answer for (1), extra filter criteria can be applied to the SQL.

There are other approaches for calculating the Nth highest row, too. The next is a non-analytic approach, which works the same way as the RANK query (gaps for ties).

```
SQL> select ename, sal from emp a
where 3 = ( select count(*) -- Replace 3 with any value of (N - 1)
        from emp b
        where b.sal > a.sal);

ENAME      SAL
---------- ----------
JONES      2975
```