**Co – related Queries :**

* They work on the principles of both **sub – queries & JOIN(s)**.
* Sub query are dependent on the output of the outer query then we call that as a co-related sub query.
* They are special type of sub – queries
* Here, both outer & inner queries are inter-dependent
* For each & every record of outer query, the entire inner query will be executed
* The outer query output will be fast to the sub query and this sub query execute.
* Then the output of the sub query pass to the outer query through outer query it displays the output for the user.

**Note:**

* Most of the time we use self join in co-related sub query.

**For ex, Display the employee who is earning the highest salary**

****

****

Thus, if an outer query column is being accessed inside the inner query, then that query is said to be co-related.

Let us see the logic i.e, how we get the 1st max salary :-

|  |  |  |
| --- | --- | --- |
| **Emp (A)** | | |
| **EmpNo** | **Ename** | **Sal** |
| 101 | Scott | 3000 |
| 102 | Blake | 4000 |
| 103 | King | 5000 |
| 104 | Smith | 2000 |
| 105 | Jones | 1000 |

|  |  |  |
| --- | --- | --- |
| **Emp (B)** | | |
| **EmpNo** | **Ename** | **Sal** |
| 101 | Scott | 3000 |
| 102 | Blake | 4000 |
| 103 | King | 5000 |
| 104 | Smith | 2000 |
| 105 | Jones | 1000 |

Since co-related queries are a combination of Joins and sub-queries.

It follows the concept of Joins and creates multiple copies of the same table.

Then it takes 1st record i.e, - Scott – sal is 3000. It starts comparing with the sal in the emp table,

3000 = 3000 - count starts from 0 – thus, 0 = 0

3000 < 4000 – thus, 0 ! = 1

3000 < 5000 – thus, 0 ! = 2

3000 > 2000 – thus , 0! = 4

3000 > 1000 – thus, 0 ! = 5 if the condition becomes false, then the count increments by 1. Here 3000 is less than 4000 & 5000, thus 0 ! = 2. Thus , Scott does not have the highest salary.

Similarly, it does for the next records,

Blake – salary of 4000 – but 4000 < 5000 – thus, 0 ! = 1. This is also false.

King – salary of 5000 – it is greater than everything – thus, 0 = 0. Thus, King has the highest salary.

But the query doesn’t stop here, it checks for Smith & Jones as well.

Similarly, if we want to find the 2nd maximum salary,

Then in the query, change ‘0’ to ‘1’ & here, the logic is – it compares until it gets 1 = 1.

For 3rd maximum salary – change 0 to 2 and so on – here, the logic is – it compares until it gets 2 = 2.

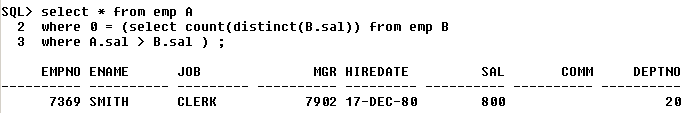
**For any highest, always put it as ‘0’ in the query.**

**If you want n(th) salary, pass (n-1).**

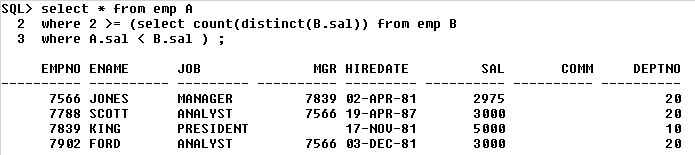
**In interview – this is a definite question. They will ask you what is co-related queries. And then they’ll ask you find, 1st or max or 3rd maximum salary – after you write the query – they will ask you to explain the logic as to how it gets the same – draw the table and explain it to them just as shown above.**

**Assignment**

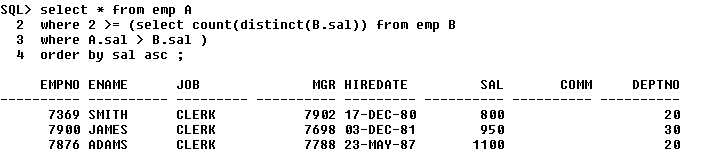
**1) Display the least salary from the employee table.**



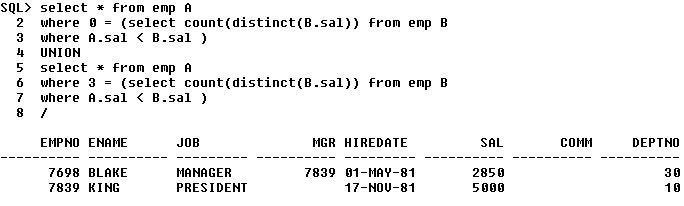
**2) Display top 3 person’s salaries from the employee table.**



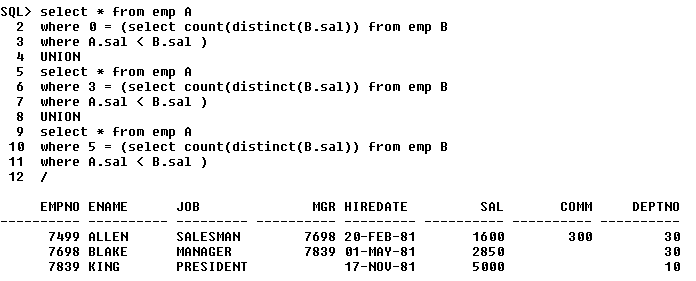
**3) Write a query to display bottom 3 salaries**

****

**4) Display 1st and 4th maximum salary**

****

**5) Display 1st, 4th & 6th highest salaries in a single query**

****