SQL- Advanced

Consider the following Salary table for employees.

EMP

EXAMPLE 1] RUNNING TOTAL.

ENAME	SALARY	EMP_NOs.
SMITH	800	1
ALLEN	1600	2
WARD	1250	3
JONES	2975	4
MARTIN	1250	5
BLAKE	2850	6
CLARK	2450	7
SCOTT	3000	8
KING	5000	9
TURNER	1500	10
ADAMS	1100	11
JAMES	950	12
FORD	3000	—13
MILLER	1300	14

§ Problem statement

Generate a Running total of
. Employee Salaries .

A] Traditional approach .

→ For every Row of an outer
query → Execute a sub
Query which
gives the total
till that Row
of the Salary column

Let us try to
Understand ←
this.

```
SELECT e.name, e.Sal,
(SELECT Sum(d.Sal) from Emp.
where d.empno ≤ e. empno)
as running total from Emp.
ORDER BY 3
```

Consider 1st Row of the table.

[emp e]

[SMITH   800   1] → Now Run the inner query.

$$emp.d \leq emp.e$$
nos   nos

means

$$emp.d \leq 1$$ which means
-nos

[SMITH   800   1]

Now sum (sal) Means Sum (800)

Thus the output Row Returned is.

Ename	Sal	Sum(Sal)
SMITH	800	800

Running total.

Consider 2nd row of the table.

[emp e]   —

[ALLEN   1600   2] → Now run the inner query

$$emp\_nos.d \leq emp\_nos.e$$
means

$$emp nos.d \leq 2$$

Thus Rows Returned will be.

NAME	SAL	Emp-nos
SMITH	800	1
ALLEN	1600	2

$$SUM (SAL) = \binom{800}{+1600} = 2400$$ Thus NET Result is a row

[ALLEN   1600   2400]

B] Approach using WINDOWS FUNCTIONS.

The Correlated subquery is done with
and we just have .

SELECT e.name, e.sal,

SUM (sal) OVER (order by sal, emp-nos)
as running - total
From emp
order by 2.

Column of interest

KEY WORD

Function to be applied

Sequence of Rows.
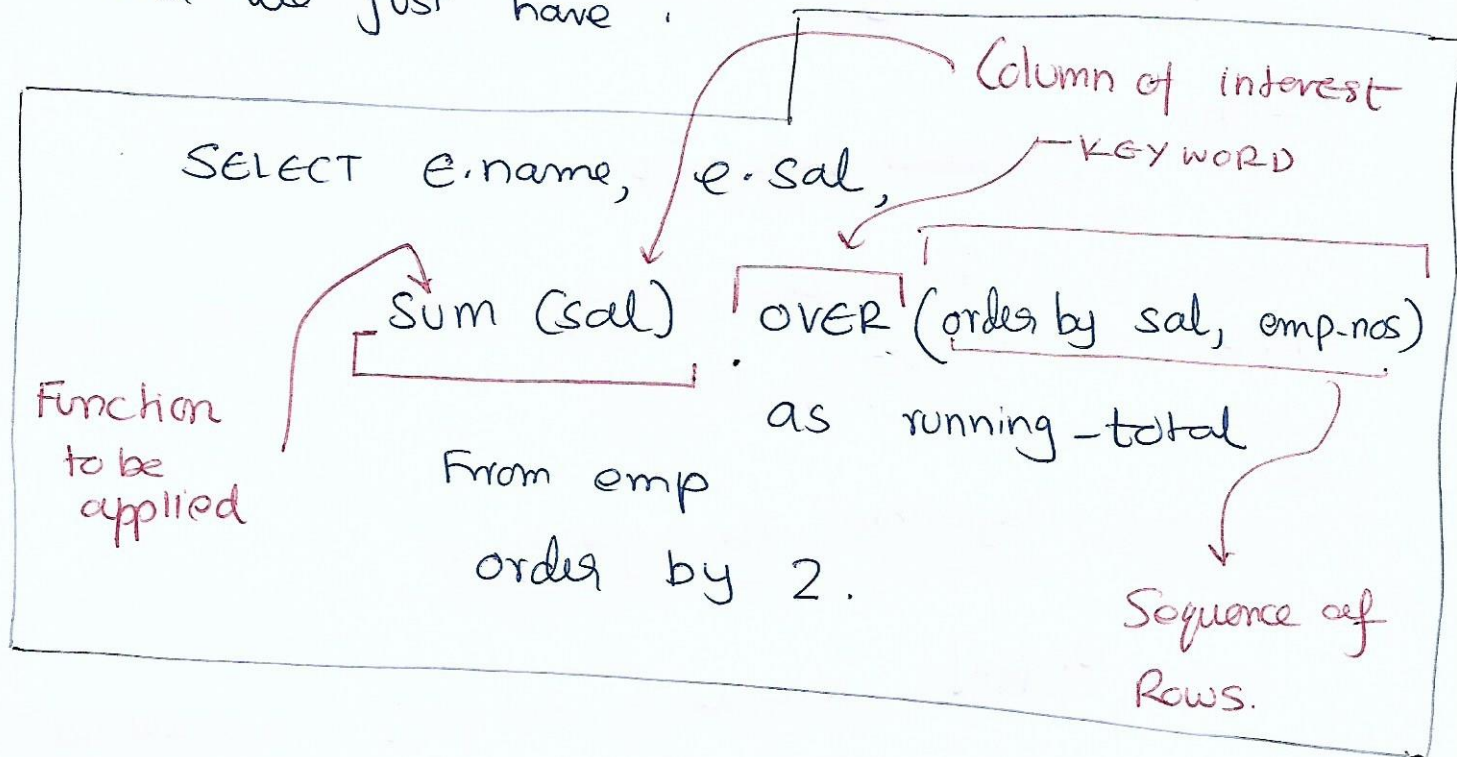
EXAMPLE 2 →

COMPUTING MODE

CONSIDER THE FOLLOWING DATA FOR emp table
WITH Dept- NOS = 20.

SALARY

SALARY
800
1100
2975
3000
3000

§ Problem statement

Find the Salary which occurs maximum
Nos of times.

Algorithm For getting it

Step 1] Generate a subquery which will get
the below Result Set in
Descending
order of
'CNT'
i.e. Count

SALARY	Cnt
3000	2
800	1
1100	1
2975	1

Step 2] Select the Salary with the highest Coun
(that is the 1st Row of this table
generated with the Subquery.

EXAMPLE 2 → Contd.   ⑤

Approach 1]   Using Subqueries.

```
SELECT  SAL
   FROM   emp
   WHERE  DEPT_NO = 20
       GROUP BY   SAL
       HAVING   Count(*) >= all
```

inner query fetches
Count of
All Salaries

Outer query selects
that Salary For which
the nos of occurrence
is the maximum

```
(Select   Count(*)
   From  emp
   where Dept-no = 20
   group by SAL)
```

From the set of occurrences of the Sub Query

Approach 2] USING WINDOWS FUNCTIONS.

```
SELECT  SAL
   FROM
   (SELECT SAL,  DENSE_RANK() OVER (ORDER BY CNT DESC) AS RI
       FROM (SELECT SAL, COUNT(*) AS CNT
               FROM EMP
               WHERE DEPT_NO = 20
               GROUP BY SAL) X
                                      ) Y
   WHERE RNK = 1
```

Ranks

highest Count
1st Rank.

→ INNER QUERY

] Mid Level Query

↳ outer query

EXAMPLE -3] Finding Percentage of a Total.

Problem Statement.

Find the sum of salaries of DeptNo = 10, ~~espo~~
as a % of the total salaries of all Depts.

Algorithm:

Step1 [→ We Need a Sum (Sal) over entire table
     → and Sum (Sal) For Dept = 10

Step2 [We Need to pull the sums in one LINE
    To get the percentage of Total, and the
Sum of Salary For that Dept in one Row]

Approach 1] USING SUBQUERIES.

→ Sums Salaries From Dept 10   Selects only Salaries From Dept 10

```
SELECT  ( Sum ( Case when DeptNo = 10  then SAL END)
              / Sum (SAL)) * 100   AS PCT
       FROM EMP
```
↳ Sums all Salaries.

Approach 2 : USING WINDOWS FUNCTIONS

→ MORE Elegant Ratio of two Sums

```
SELECT  DISTINCT (d10/total) * 100 AS PCT
```

INNER QUERY →
```
  FROM (SELECT  DEPT_NO,
                SUM (SAL) OVER () TOTAL,          ← all Rows
                SUM (SAL) OVER (PARTITION BY DEPTNO) d10
         FROM EMP                    ↳ Dept wise Sal totals
       ) X
     WHERE dept_NO = 10 ] Filter For Dept = 10
```

Example - 4] Paginating through a Result set ⑦

Problem statement

Return the ⊗ Salaries 6 to 10 in ASC ORDER of Salaries.

Algorithm

→ order by salaries ascending
→ Then Return Rows 6 to 10.

Solution

```
SELECT    SAL
    FROM (
Sorting         SELECT    row_number() OVER (ORDER BY SAL) AS
and     ←               SAL.
Assign          FROM EMP
row_number    ) X
        WHERE    RN between    6 AND 10.
```

`Nth' Maximum Record.

Problem Statement : Find the `n'th maximum Salary From Emp Table.

Algorithm ①

Using WINDOWS Function.

→ Select Distinct salaries,
→ Order by Salary Descending,
→ Rank Salaries, or get a Row number
→ Select Rank = `N' or Row number = N

Algorithm 2

For WINDOWS Function.

Use the Logic that `N'th maximum Salary will have (N-1) Distinct Salaries which are greater than itself.

G
Thus Count the Nos of Salaries
> the desired salary, that
Count = N-1

Solution using Sub Query:

```
SELECT NAME, SAL FROM EMP A
  WHERE (N-1) = (SELECT COUNT (DISTINCT SALARY)
                  FROM EMP. B
                  . WHERE B.SALARY > A.SALARY)
```

Solution using WINDOWS Func

```
SELECT SAL, EMPNAME
  FROM
      ( SELECT EMPNAME, SAL,
          row_number () OVER (ORDER BY SAL DESC) AS RN
        FROM EMP) X
      WHERE RN = N
```