**Find 2nd highest sal paid employee details**

SQL>Select max(sal) from emp where sal<(Select max(sal) from emp);

SQL>Select max(sal) from emp where sal not in(Select max(sal) from emp);

**Examples**

**1)Find all the customers who have both a loan and an account at the bank**

SQL> Select distinct customer\_name from borrower where customer\_name in( Select customer\_name from depositor);

**2)Find all customers who do have a loan at the bank but do not have an account at the bank**

SQL>Select distinct customer\_name from borrower where customer\_name not in (Select customer\_name from depositor);

**3)Find the names of customers who have a loan at the bank, and whose names are neither smith nor jones**

SQL>Select distinct customer\_name from borrower where customer\_name not in (‘SMITH’,’JONES’);

**Set operations**

1. **Customer names who is having only account in the bank but not loan**

SQL> Select customer\_name from depositor

MINUS

Select customer\_name from borrower;

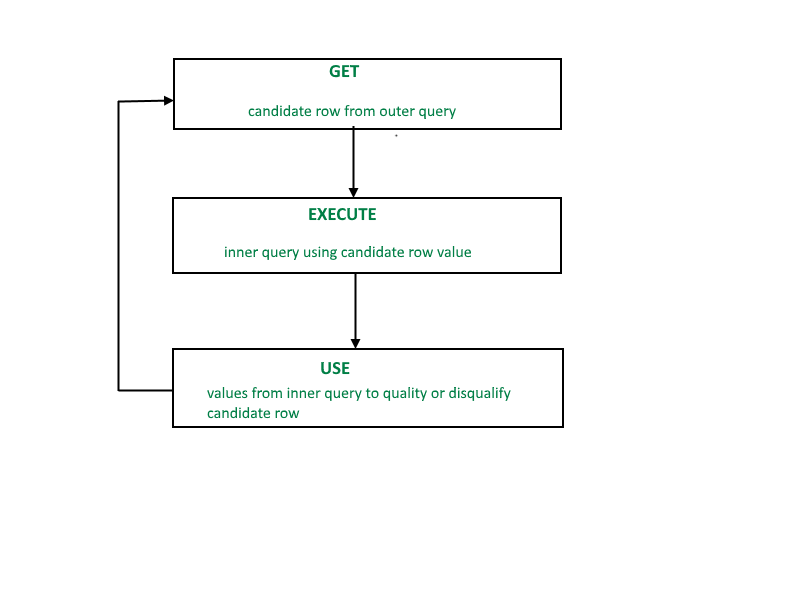
1. **Customer names who has both loan and account in the bank**

SQL>Select customer\_name from borrower

UNION

Select customer\_name from depositor;

**Corelated sub quries**

Correlated subqueries are used for row-by-row processing. Each subquery is executed once for every row of the outer query.  


A correlated subquery is evaluated once for each row processed by the parent statement. The parent statement can be a **SELECT**, **UPDATE**, or **DELETE** statement.

**1)Display Top 3 highest salaries in emp table**

SQL> Select \* from (Select \* from emp order by sal desc) where rownum<=3 order by sal

desc;

**2)How to create rownumbers**

SQL>Select row\_number() over(order by ename),empno,ename from emp;

SQL>Select row\_number() over(order by ename) as rono,empno,ename,sal from emp;

**3)4th highest sal**

SQL>Select \* from(Select row\_number() over(order by sal desc) as rono,empno,sal from emp) emp where rono=4;

**Query to create a database table from another table**

SQL>Create table new as Select \* from emp;

**Example queries**

**Q1**) **SQL> select \* from (select sal, rank () over (order by sal desc) as R from emp);**

**O/P:**

**SAL R**

**---------- ----------**

50000 1

30000 2

30000 2

29750 4

28500 5

26000 6

25000 7

24500 8

23000 9

22500 10

22500 10

21000 12

19500 13

18000 14

14 rows selected.

**Q2) SQL> select \* from (select sal, dense\_rank () over (order by sal desc) as R from emp);**

**O/P:**

SAL R

---------- ----------

50000 1

30000 2

30000 2

29750 3

28500 4

26000 5

25000 6

24500 7

23000 8

22500 9

22500 9

21000 10

19500 11

18000 12

14 rows selected.

**Q3) SQL> select \* from (select sal, rank () over (order by sal desc) as R from emp) where R=4;**

**O/P:**

**SAL R**

**---------- ----------**

29750 4

**Q4) SQL> select \* from (select sal, dense\_rank () over (order by sal desc) as R from emp) where R=4;**

**O/P:**

**SAL R**

---------- ----------

28500 4

**Q5) SQL> Select \* from (select job, deptno, rank () over (partition by deptno order by sal desc) as R from emp);**

**o/p:**

**JOB DEPTNO R**

**--------- ---------- ----------**

PRESIDENT 10 1

MANAGER 10 2

CLERK 10 3

ANALYST 20 1

ANALYST 20 1

MANAGER 20 3

CLERK 20 4

CLERK 20 5

MANAGER 30 1

SALESMAN 30 2

SALESMAN 30 3

SALESMAN 30 4

SALESMAN 30 4

CLERK 30 6

14 rows selected.

**Q6) SQL> Select \* from (select job, deptno, dense\_rank () over (partition by deptno order by sal desc) as R from emp);**

**O/P:**

**JOB DEPTNO R**

**--------- ---------- ----------**

PRESIDENT 10 1

MANAGER 10 2

CLERK 10 3

ANALYST 20 1

ANALYST 20 1

MANAGER 20 2

CLERK 20 3

CLERK 20 4

MANAGER 30 1

SALESMAN 30 2

SALESMAN 30 3

SALESMAN 30 4

SALESMAN 30 4

CLERK 30 5

14 rows selected.

**Q7) SQL> select \* from (select sal, Row\_number () over (order by sal desc) as R from emp) where R=4;**

**O/P:**

**SAL R**

**---------- ----------**

29750 4

Row\_number () is same as Rank () function