MAX & MIN :

EID	ENAME	SAL	DEPTNO
1	ALLEN	1000	20
2	BLAKE	2000	10
3	CLARK	3000	30
4	MILLER	1500	10
5	ADAMS	2500	20

1. WAQTD maximum salary of an employee .

```
SELECT MAX( SAL )
FROM EMP ;
```

2. WAQTD name of the employee getting maximum salary .

```
SELECT ENAME , MAX( SAL )
FROM EMP ;
```

```
SELECT ENAME
FROM EMP
WHERE SAL = MAX( SAL ) ;
```

```
SELECT ENAME
FROM EMP
WHERE SAL = ( SELECT MAX( SAL )
FROM EMP ) ;
```

3. WAQTD name and salary earned by the employee getting Minimum salary .

```
SELECT ENAME , SAL
FROM EMP
WHERE SAL = ( SELECT MIN( SAL )
FROM EMP ) ;
```

TYPES OF SUB - QUERY :

1. **SINGLE ROW SUB QUERY**
2. **MULTI ROW SUB QUERY**

Example :

Emp

EID	ENAME	SAL	DEPTNO
1	ALLEN	1000	20
2	BLAKE	2000	10
3	CLARK	3000	30
4	MILLER	1500	10
5	SMITH	2500	10

DEPT

DEPTNO	DNAME	LOC
10	D1	L1
20	D2	L2
30	D3	L3

1. WAQTD dname of ALLEN .

```
SELECT DNAME
FROM DEPT
WHERE DEPTNO    =   ( SELECT DEPTNO
                      FROM EMP
                      WHERE ENAME = 'ALLEN' ) ;
```

20

2. WAQTD dnames of allen and smith .

```
SELECT DNAME
FROM DEPT
WHERE DEPTNO    =   ( SELECT DEPTNO
                      FROM EMP
                      WHERE ENAME IN
                      ( 'ALLEN' ,'SMITH' ) ) ;
```

20
10

DEPTNO
20
10
30
10
10

Here , since the sub query returns 2 records we cannot use '=' Op . We've to use IN Op .

1. SINGLE ROW SUB QUERY :

- If the sub query returns exactly 1 record / value we call it as Single Row Sub Query .
- *If it returns only 1 value then we can use the normal operators Or the Special Operators to compare the values .*

2. MULTI ROW SUB QUERY :

- If the sub query returns more than1 record / value we call it as Multi Row Sub Query .
- *If it returns more than 1 value then we **cannot use the normal operators** We have to use only Special Operators to compare the values .*

Note : *It is difficult to identify whether a query Belongs Single or Multi row So , it is always recommended to use Special Operators to Compare The values .*

1. WAQTD ename and salary of the employees earning *more than* Employees of dept 10 .

EID	ENAME	SAL	DEPTNO
1	ALLEN	1000	20
2	BLAKE	2000	10
3	CLARK	3000	30
4	MILLER	1500	10
5	SMITH	2500	10

2000
1500
2500

```
SELECT ENAME , SAL
FROM EMP
WHERE SAL  >   ( SELECT SAL
                 FROM EMP
                 WHERE DEPTNO = 10 ) ;
```

Here we cannot use > symbol to compare Multiple values .

We cant use IN or. NOT IN as well because It is used for = and != symbols .

*Therefore we have to use **Sub Query Operators** For Comparing Relational Operators such as (> , < , >= , <=) .*

Sub Query Operators :

1. ALL :

"It is special Op used along with a relational Op (> , < , > = , <=) to compare the values present at the RHS ".

- *ALL Op returns true if all the values at the RHS have satisfied the condition .*

Example :

CLARK ,3000

2000
1500
2500

```
SELECT ENAME , SAL
FROM EMP
WHERE SAL  > ALL  ( SELECT SAL
                    FROM EMP
                    WHERE DEPTNO = 10 ) ;
```

SAL
1000
2000
3000
1500
2500

1000 > ALL (2000 , 1500 , 2500)

1000 > 2000	False
1000 > 1500	False
1000 > 2500	False

2000 > ALL (2000 , 1500 , 2500)

2000 > 2000	False
2000 > 1500	True
2000 > 2500	False

3000 > ALL (2000 , 1500 , 2500)

3000 > 2000	True
3000 > 1500	True
3000 > 2500	True

1500 > ALL (2000 , 1500 , 2500)

1500 > 2000	False
1500 > 1500	False
1500 > 2500	False

2500 > ALL (2000 , 1500 , 2500)

2500 > 2000	True
2500 > 1500	True
2500 > 2500	False

2. **ANY :**

"It is special Op used along with a relational Op (> , < , > = , <=)

2. **ANY :**

"It is special Op used along with a relational Op (> , < , > = , <=) to compare the values present at the RHS ".

- *ANY Op returns true if one of the values at the RHS have satisfied the condition .*

Example :

2000
1500
2500

```
SELECT ENAME , SAL
FROM EMP
WHERE SAL  > ANY  ( SELECT SAL
                    FROM EMP
                    WHERE DEPTNO = 10 ) ;
```

SAL
1000
2000
3000
1500
2500

1000 > ANY (2000 , 1500 , 2500)

1000 > 2000	False
1000 > 1500	False
1000 > 2500	False

2000 > ANY (2000 , 1500 , 2500)

2000 > 2000	False
2000 > 1500	True
2000 > 2500	False

3000 > ANY (2000 , 1500 , 2500)

3000 > 2000	True
3000 > 1500	True
3000 > 2500	True

1500 > ANY (2000 , 1500 , 2500)

1500 > 2000	False
1500 > 1500	False

1500 > ANY (2000 , 1500 , 2500)

1500 > 2000	False
1500 > 1500	False
1500 > 2500	False

2500 > ANY (2000 , 1500 , 2500)

2500 > 2000	True
2500 > 1500	True
2500 > 2500	False

1. WAQTD name of the employee if the employee earns less than The employees working as salesman .

```
SELECT ENAME
FROM EMP
WHERE SAL < ALL ( SELECT SAL
```



```
FROM EMP
WHERE JOB='SALESMAN' ) ;
```

2. WAQTD name of the employee if the employee earns less than At least a salesman .

```
SELECT ENAME
FROM EMP
WHERE SAL < ANY ( SELECT SAL
FROM EMP
WHERE JOB ='SALESMAN' ) ;
```

3. WAQTD names of the employees earning more than ADAMS .

```
SELECT ENAME
FROM EMP
WHERE SAL > ALL  ( SELECT SAL
FROM EMP
WHERE ENAME ='ADAMS' ) ;
```

NESTED SUB QUERY :

" A sub query written inside a sub query is known as Nested Subquery "

SAL
1000

- WE CAN NEST ABOUT **255** SUB QUERIES

1000
2000
4000
3000
5000

1. WAQTD maximum salary given to an employee .

   SELECT MAX(SAL) **5000**
   FROM EMP ;

2. WAQTD second maximum salary given to an employee .

   4000
   SELECT MAX(SAL)
   FROM MP **5000**
   WHERE SAL < (SELECT MAX(SAL)
   FROM EMP) ;

SAL
1000
2000
4000
3000
5000

3. WAQTD 3rd maximum salary .

   SELECT MAX(SAL) 3000
   FROM EMP
   WHERE SAL < (SELECT MAX(SAL) 4000
   FROM EMP

3. WAQTD 3rd maximum salary .

```
SELECT MAX( SAL ) 3000
FROM EMP
WHERE SAL < ( SELECT MAX( SAL ) 4000
FROM EMP
WHERE SAL < ( SELECT MAX( SAL ) 5000
FROM EMP ) )
```

4. WAQTD 4th maximum salary .

```
SELECT MAX( SAL ) 2000
FROM EMP
WHERE SAL < ( SELECT MAX( SAL ) 3000
FROM EMP
WHERE SAL < ( SELECT MAX( SAL ) 4000
FROM EMP
WHERE SAL < ( SELECT MAX( SAL ) 5000
FROM EMP ) ) )
```

5. WAQTD 3 minimum salary .

```
SELECT MIN(SAL )
```



```
FROM EMP
WHERE SAL > ( SELECT MIN(SAL )
              FROM EMP
              WHERE SAL > ( SELECT MIN ( SAL )
                            FROM EMP ) ) ;
```

6. WAQTD Dept name of the employee getting 2nd Minimum salary .

```
SELECT DNAME
FROM DEPT
WHERE DEPTNO = ( SELECT DEPTNO
FROM EMP
WHERE SAL = (SELECT MIN( SAL )
FROM EMP
WHERE SAL > ( SELECT MIN( SAL )
FROM EMP ) ) );
```

EMPLOYEE AND MANAGER RELATION :

EID	ENAME	MGR
1	ALLEN	3
2	SMITH	1
3	JAMES	2
4	KING	3

CASE 1 :

- WAQTD name of Allen's manager .

JAMES

```
SELECT ENAME
FROM EMP                    3
WHERE EID = ( SELECT MGR
              FROM EMP
              WHERE ENAME ='ALLEN' )
```

1
2
3
4

- WAQTD name of SMITH's manager .

```
SELECT ENAME
FROM EMP
WHERE EID = ( SELECT MGR
              FROM EMP
              WHERE ENAME ='SMITH' ) ;
```

- WAQTD name of SMITH's manager's manager .

EID	ENAME	MGR
1	ALLEN	3
2	SMITH	1
3	JAMES	2
4	KING	3

```
SELECT ENAME
FROM EMP
WHERE EID = ( SELECT MGR
FROM EMP
WHERE EID = ( SELECT MGR
FROM EMP
```

```
WHERE ENAME ='SMITH' ) ) ;
```

- WAQTD dname of King's Manager .

```
SELECT DNAME
FROM DEPT
WHERE DEPTNO = ( SELECT DEPTNO
FROM EMP
WHERE EID = ( SELECT MGR
FROM EMP
WHERE ENAME ='KING' ) ) ;
```

- WAQTD Location of Adams's manager's manager .

```
SELECT LOC
FROM DEPT
WHERE DEPTNO = ( SELECT DEPTNO
FROM EMP
WHERE EID = ( SELECT MGR
FROM EMP
WHERE EID = ( SELECT MGR
FROM EMP
WHERE ENAME ='ADAMS' ) ) ) ;
```

CASE -2

- WAQTD Names of the employees reporting to KING.

```
SELECT ENAME
FROM EMP
WHERE MGR = ( SELECT EID
FROM EMP
WHERE ENAME ='KING' ) ;
```

- WAQTD Name and salary given to the employees reporting To James .

```
SELECT ENAME , SAL
FROM EMP
WHERE MGR = ( SELECT EID
FROM EMP
WHERE ENAME ='JAMES' ) ;
```

To find **Manager**	Select **MGR** in Sub Q
To find **Employees**	Select **EID** in Sub Q