MultiColumn SubQueries ====>
==========

* In certain situations, however, we can use subqueries that return two or more columns.

* Such subqueries are called MULTICOLUMN SUBQUERIES.

Example :-

* To understand the concept MULTICOLUMN SUBQUERY , consider the following table.

A_Name	A_Age	A_Country
Sunil	26	India
Mark	35	USA
Rajiv	22	India
Walter	31	Australia
Anwar	30	Pak
Suleman	32	Pak

WAQ to display name and age of eldest athelete of every country.
- SELECT A_name,A_age FROM Atheletes Where (A_age,A_country) in (Select max(A_age), A_country from Atheletes group by A_country);

* No of columns in the WHERE clause of OUTER QUERY and the SELECT clause of INNER QUERY must be same.

* It is compulsory to enclose all the columns mentioned in the WHERE clause of the OUTER QUERY in a pair of parenthesis.

* Order of columns in WHERE clause of OUTER QUERY and the SELECT clause of INNER QUERY must be same.

WAQ, to display the names of last hired employee along with his hiredate for each department.
- select ename, hiredate, deptno from emp where (deptno, hiredate) in (select deptno, max(hiredate) from emp group by deptno);

ENAME	HIREDATE	DEPTNO
JAMES	01-DEC-81	30
ADAMS	23-MAY-87	20
MILLER	23-JAN-82	10

SOME, ANY And ALL Operator ===>
==========

* These operators are used with relational operators and they compare the outer value to each of the value in the list generated by INNER QUERY .

* They have the power to make all relational operators behave like multi valued comparison operator.

* The ANY operator checks whether any value in the list makes the condition true.

* The ALL operator returns rows if the condition is true for all the values in the list.

* The SOME operator is identical to ANY, and the two can be used interchangeably.

WAQ to display the names of all the employees whose salary is greater than the salary of any SALESMAN ?
- select ename, sal from emp where sal > any (select sal from emp where job = 'SALESMAN');

```
ENAME             SAL
---------- ----------
KING             5000
FORD             3000
SCOTT            3000
JONES            2975
BLACK            2850
CLARK            2450
ALLEN            1600
TURNER           1500
MILLER           1300
```

OR

- select ename, sal from emp where sal > some (select sal from emp where job = 'SALESMAN');

```
ENAME             SAL
---------- ----------
KING             5000
FORD             3000
SCOTT            3000
JONES            2975
BLACK            2850
CLARK            2450
ALLEN            1600
TURNER           1500
MILLER           1300
```

WAQ to display the names of all the employees whose salary is greater than the salary of every SALESMAN ?
- select ename, sal from emp where sal > all (select sal from emp where job = 'SALESMAN');

```
ENAME             SAL
---------- ----------
CLARK            2450
BLACK            2850
JONES            2975
SCOTT            3000
FORD             3000
KING             5000
```

OR

- select ename, sal from emp where sal > (select max(sal) from emp where job = 'SALESMAN');

```
ENAME             SAL
---------- ----------
JONES            2975
BLACK            2850
CLARK            2450
SCOTT            3000
KING             5000
FORD             3000
```

The EXISTS Operator ===>
==========

* The EXISTS operator is used for CORRELATED SUBQUERIES and it tests whether the subquery returns at least one row.

* Because EXISTS tests only whether a row exists, the columns shown in the SELECT list of the subquery are irrelevant.

* Typically, we use a single-character text literal, such as 'X', or the keyword NULL or any number like 1.

WAQ to display names of those departments who have employees working in them.
- select dname from dept where exists (select deptno from emp where emp.deptno = dept.deptno);

```
DNAME
--------------
RESEARCH
SALES
ACCOUNTING
```

WAQ to display ename of those employees who are managers.
- select ename from emp e1 where exists (select 1 from emp e2 where e2.mgr = e1.empno);

```
ENAME
----------
FORD
BLACK
KING
JONES
SCOTT
CLARK
```

NOT EXISTS Operator ===>
==========

* The NOT EXISTS operator is the opposite of the EXISTS operator; it tests whether a matching row CANNOT BE FOUND.

* The operator is the most frequently used with CORRELATED SUBQUERY construct.

WAQ to display names of those departments who do not have employees working in them.
- select dname from dept where not exists (select 1 from emp where emp.deptno = dept.deptno);

```
DNAME
```

OPERATIONS