**Equi join:**

Creates a join for equality or matching columns

A screenshot of a computer

Description automatically generated

Select columns

from table1 join table2

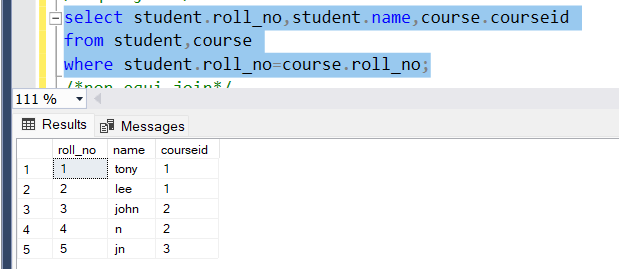
on table1.c=table2.c;

(or)

Select columns

From table1,tale2

Where table1.c=table2.c;



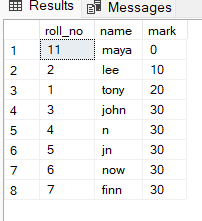
**Non equi join:**performs a join using comparison operator other than =

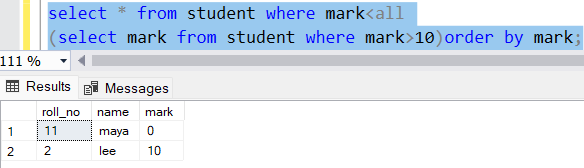
A screenshot of a computer

Description automatically generated

**All:**returns true if all of the subquery meet the conditions

Student table:





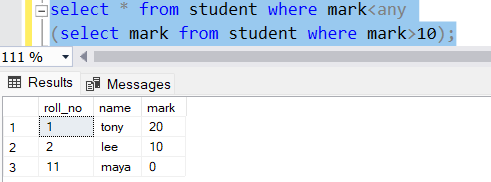
Here the subquery returns 20,30

So the outer query should return mark which is less than both 20 and 30

So the rows whose Marks are less than both 20 and 30 gets returned .

**Any:**

Return true if any of the subquery value meet the conditions



Subquery returns 20,30 so the mark should be less than 20 or 30.

Take 2 tables demo1,demo4:

A screenshot of a computer

Description automatically generated

**Union:**

combines data from two tables and eliminate duplicate rows

A screenshot of a computer

Description automatically generated

**union all**-combines data from 2 tables and includes the duplicate rows too

A screenshot of a computer

Description automatically generated

**intersect**-returns the common rows from 2 tables

A screenshot of a computer

Description automatically generated

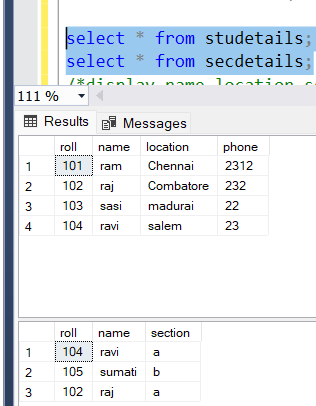
**except**-returns the left table rows except the common rows

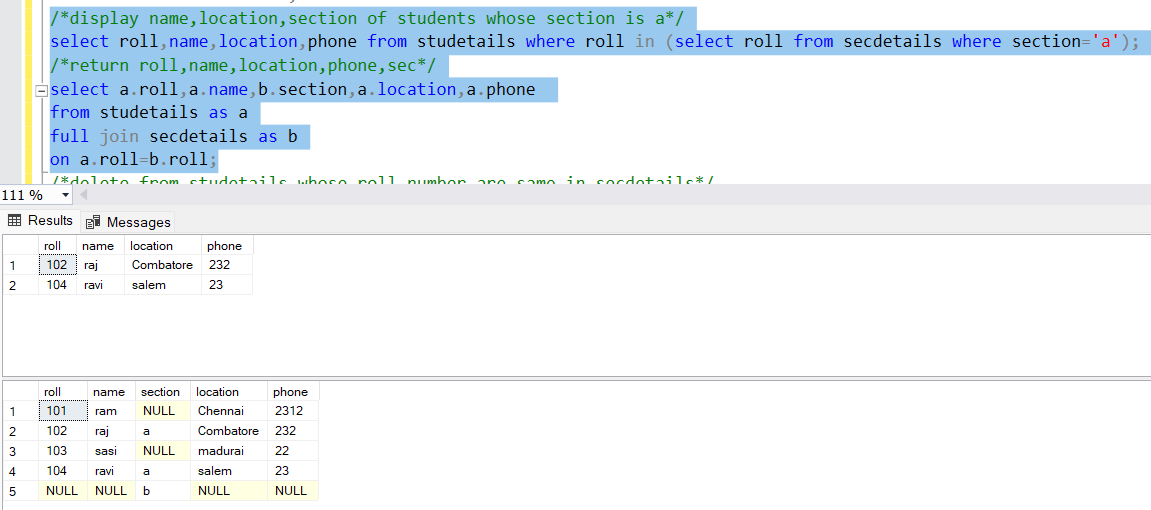
A screenshot of a computer

Description automatically generated

**Subquery:**

A query inside a query is called subquery



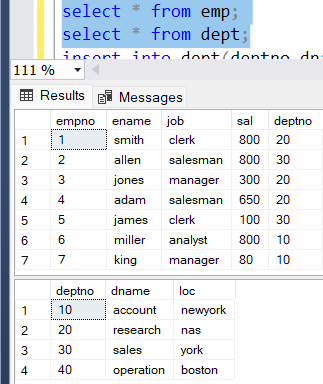


**Nested subquery:**

A subquery can be nested inside other subqueries ,nesting maybe used to any number of levels.

In nested subqueries we can execute inner queries independently.

below 2 tables are emp,dept



A screenshot of a computer

Description automatically generated

The most inner query returns deptno 30 and the inner query returns jobs(salesman,clerk)

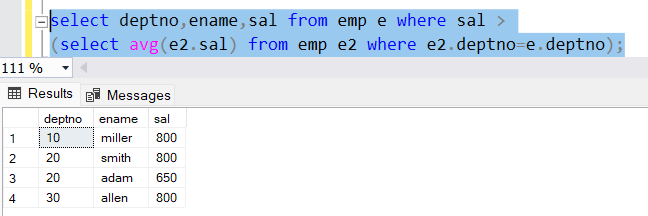
The outer query returns the above table.

**Corelated subquery:**

Is a subquery that uses values from the outer query requiring the inner query to execute once for each outer query.

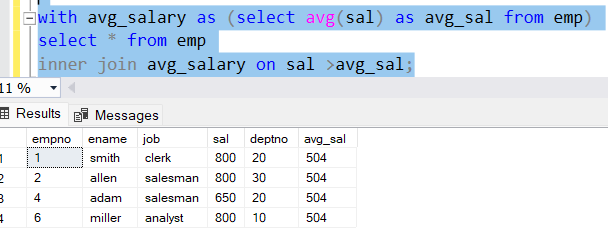
We cannot run inner query independently.

There is a co relation between the inner query and outer query.



**With clause:**

Instead of subquery we can use with clause cuz it performs good for complex queries



With:

The WITH clause provides a way to simplify complex queries by breaking them down into smaller, more manageable pieces. It can also be used to improve query performance by avoiding the need to repeat the same subquery multiple times within a larger query.