National University of Computer and Emerging Sciences



Lab Manual 5

"Nested Queries"

Database Systems

Spring 2022



Contents

C	ontents	1
O	bjective	2
Pr	Pre-requisites	
1.	Join Operation	3
	Inner Join:	3
	Left/Right/Full Outer Join	3
	Cross Join	4
	Joining More than two tables	5
2.	Aggregation-Grouping	6
	Grouping:	8
	Having Clause	9
3.	Nested Queries	9
	A subquery (inner query) is a SQL select query nested inside a another select query (outer query)	9
	A subquery can be nested inside:	9
	There are two types of subqueries	10
	Scalar Vs Non-scalar	10
	Non-Correlated Query:	11
	Non-Correlated Subqueries in SELECT clause	11
	Non-Correlated Subqueries in From Clause	11
	Non-Correlated Subqueries in Where Clause	12
	Correlated queries	12
	Correlated Subquery in Select Clause	12
	Correlated Subquery in Where Clause	13
	Correlated Subquery in Having Clause	13



Objective

• The purpose of this manual is to get stared with data retrieval queries, starting from Simple Join operation, covering Order by clause and Aggregate functions, Group by.

Pre-requisites

- Lab 2 manual, on how to get started with MS-SQL server
- How Select from Where clause work
- How Joining and all its type work
- How Order by clause works
- Aggregate functions, Group by

Task Distribution

Total Time	170 Minutes
Joining	15 Minutes
Group by	15 Minutes
Exercise	100 Minutes
Evaluation	Last 30 Minutes

1.



1. Nested Queries

A subquery (inner query) is a SQL select query nested inside a another select query (outer query)

A subquery may occur in:

- SELECT clause of outer query
- FROM clause of outer query
- WHERE clause of outer query (most commonly used)

A subquery can be nested inside:

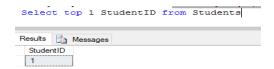
- SELECT statement
- INSERT statement
- UPDATE statement
- DELETE statement
- Another subquery.

There are two types of subqueries

- Correlated subqueries: where we use some attribute of outer query in inner query, result of inner query will then change according to the attribute of outer query.
- Non-correlated Subqueries: where no attribute of outer query is used in inner query, in this case inner query always return same value

Scalar Vs Non-scalar

A select query can return a scalar value or a table. Scalar value means one column and one row Example: result of the following query is scalar



A select query can also return non-scalar value, with more than one column and/or more than one row Example:

Select StudentID from Students

Will give non-scalar result.

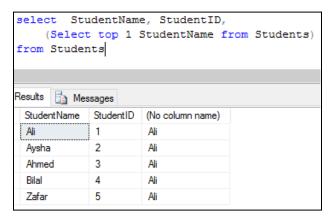
If you are writing a sub query in Select Clause, the inner query should be Scalar If you are writing a subquey in From Clause, inner query can be scalar or non-Scalar If you are writing a subquery in Where Clause, inner query can be scalar or non-Scalar depending on conditon.



Non-Correlated Query:

Non-Correlated Subqueries in SELECT clause

TRY IT: Non-correlated nested query in Select is not very useful



Non-Correlated Subqueries in From Clause

```
SELECT <List of columns of T ( result of inner query)>
FROM (select ColumnName from <TableName>) as T WHERE <condition>
**inner query can be scalar of non-scalar
***always give alias to inner query in from clause
```

TRY THIS

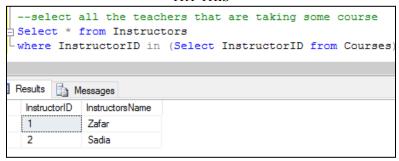
```
select
 from
  ( select StudentName, CourseID, GPA From
     Students S inner join Registration R on R.StudentID=S.StudentID
 -) as T
Results 🔓 Messages
   StudentName CourseID GPA
              3
                      3
                      2
   Αli
              4
   Ali
              5
                      3
              1
                      2.5
   Aysha
              2
                      0
   Aysha
                      3
   Aysha
```



Non-Correlated Subqueries in Where Clause

```
SELECT <List of columns of T >
FROM TableName as T
WHERE <condition> (select ColumnName from <TableName>)
```

TRY THIS

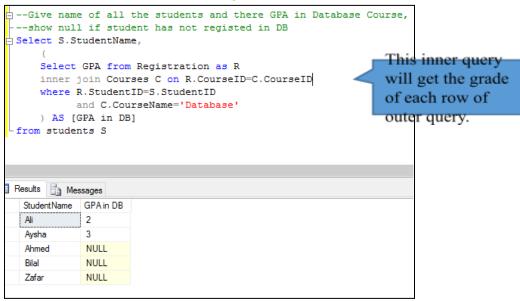


Correlated queries

When inner query is correlated with outer query, then the inner query is executed for each row of outer query.

Correlated Subquery in Select Clause

TRY THIS





Correlated Subquery in Where Clause

TRY THIS

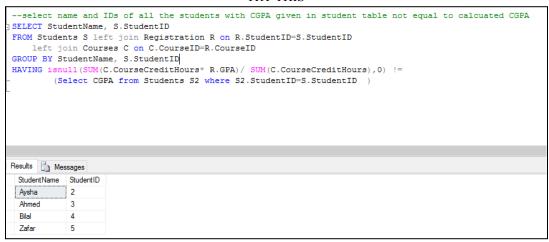
```
--Select Names of all the students with Grade Higher GPA 2 in any course
 Select *
 from Students S
 where exists
          (Select * from
          Registration R
          where R.StudentID=S.StudentID
          and GPA>2)
Results 🔓 Messages
   StudentID StudentName StudentBatch CGPA
           Ali
                      2013
                                 3.3
   1
                      2013
                                 4
   2
           Aysha
```

** WHAT DOES THE EXIST CLAUSE DO?

Correlated Subquery in Having Clause

You can also use subquery in having clause (correlated on non-correlated)

TRY THIS



Modify the query given above to, Shown name, IDs, Calcuated CGPA and CGPA given in Student table of all the students with CGPA given in student table lesser to calcuated CGPA

^{**}Refer to the slides on Neste Queries for more details