`GROUP BY` Clause in SQL

The `GROUP BY` clause is used in SQL to **group rows** that have the **same values** in specified columns. It is typically used with **aggregate functions** (e.g., `COUNT()`, `SUM()`, `AVG()`, `MAX()`, `MIN()`) to perform operations on each group.

Syntax:

sql Copy code

```sql
SELECT column_name, aggregate_function(column_name)
FROM table_name
GROUP BY column_name;
```

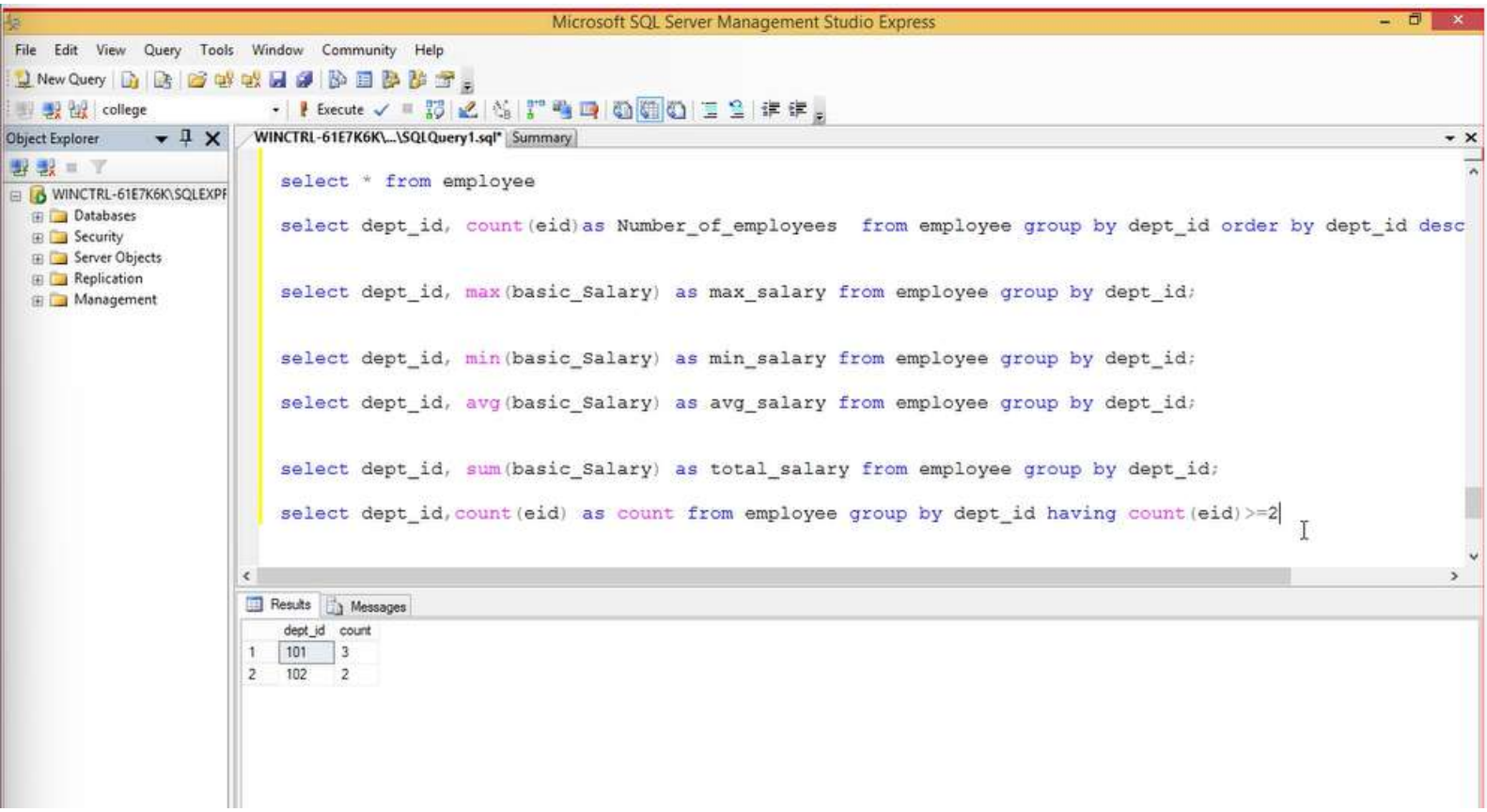
SQL GroupBy Clause Having

To filter groups based on condition, you use the HAVING clause.

- *SQL GROUPBY clause Having*

```
SELECT customer_id, YEAR (order_date), COUNT (order_id) order_count
FROM sales.orders
GROUP BY customer_id, YEAR (order_date)
HAVING COUNT (order_id) >= 2
ORDER BY customer_id;
```

customer_id	order_year	order_count
1	2018	2
2	2018	2
3	2018	3
4	2017	2
5	2016	2
6	2018	2
7	2018	2
9	2018	2
10	2018	2


Microsoft SQL Server Management Studio Express
File Edit View Query Tools Window Community Help
New Query
college
Execute
Object Explorer
WINCTRL-61E7K6K\SQLEXPF
Databases
Security
Server Objects
Replication
Management
WINCTRL-61E7K6K\...\SQLQuery1.sql*
Summary
select * from employee
select dept_id, count(eid)as Number_of_employees from employee group by dept_id order by dept_id desc
select dept_id, max(basic_Salary) as max_salary from employee group by dept_id;
select dept_id, min(basic_Salary) as min_salary from employee group by dept_id;
select dept_id, avg(basic_Salary) as avg_salary from employee group by dept_id;
select dept_id, sum(basic_Salary) as total_salary from employee group by dept_id;
select dept_id,count(eid) as count from employee group by dept_id having count(eid)>=2
Results
Messages
dept_id count
1 101 3
2 102 2

HAVING vs. WHERE

- The WHERE clause applies the condition to individual rows before the rows are summarized into groups by the GROUP BY clause.
- The HAVING clause applies the condition to the groups after the rows are grouped into groups.
- Therefore, it is important to note that the HAVING clause is applied after whereas the WHERE clause is applied before the GROUP BY clause.

Introduction to Subquery

- A subquery is a query nested within another query such as SELECT, INSERT, UPDATE or DELETE. Also, a subquery can be nested within another subquery.
- A subquery is called an inner query while the query that contains the subquery is called an outer query.
- A subquery can be used anywhere that expression is used and must be closed in parentheses.

- You can use a subquery in many places such as:
  - With the IN or NOT IN operator
  - With comparison operators
  - With the EXISTS or NOT EXISTS operator
  - With the ANY or ALL operator
  - In the FROM clause
  - In the SELECT clause

Outer Query

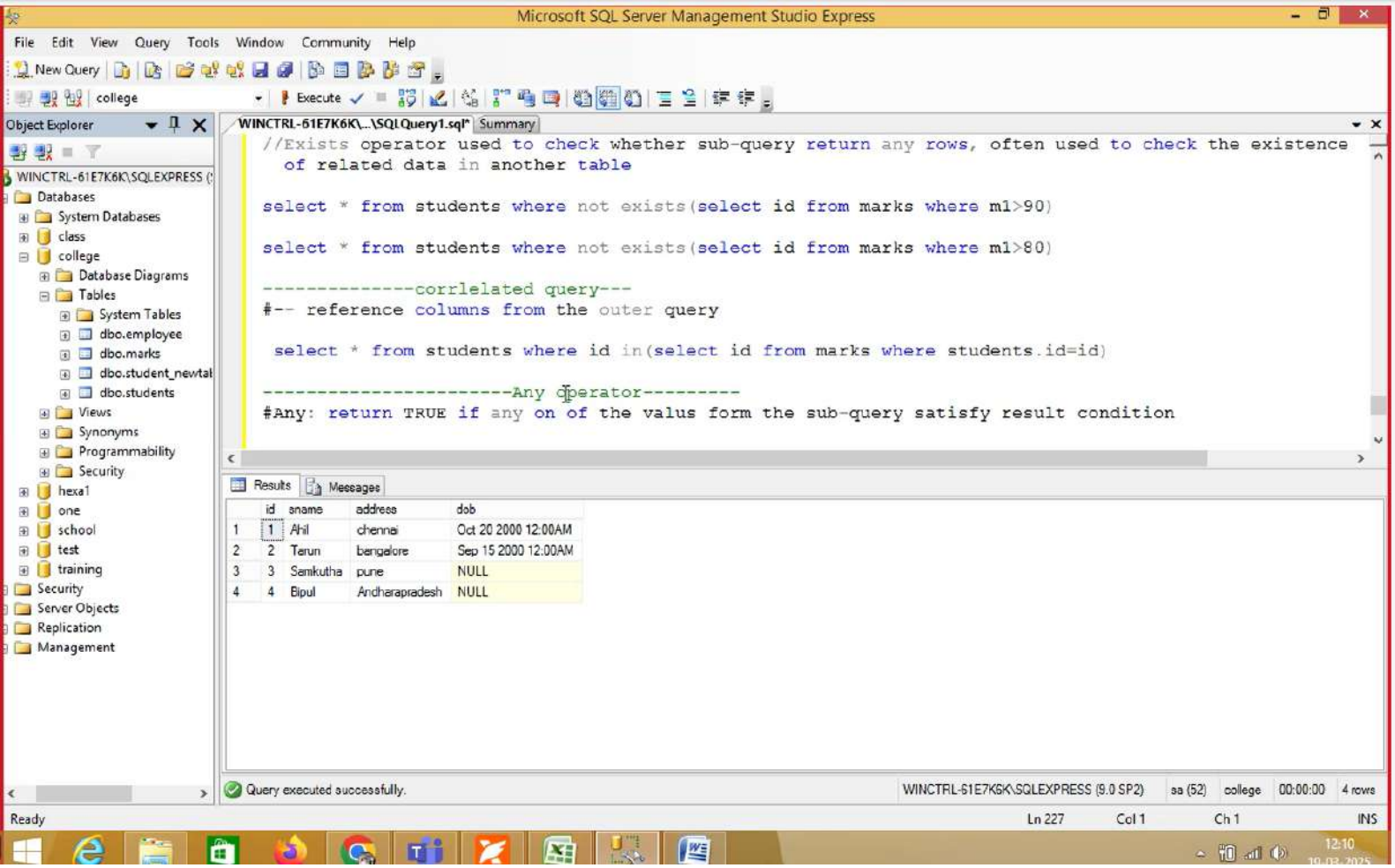
Subquery or Inner Query

```
SELECT lastname, firstname
FROM employees
WHERE officeCode IN (SELECT officeCode
 FROM offices
 WHERE country = 'USA')
```

MySQL Subquery

Guidelines

- Subqueries must be enclosed within parentheses.
- A subquery can have only one column in the SELECT clause, unless multiple columns are in the main query for the subquery to compare its
- An ORDER BY command cannot be used in a subquery, although the main query can use an ORDER BY. The GROUP BY command can be used to perform the same function as the ORDER BY in a subquery.
- Subqueries that return more than one row can only be used with multiple value operators such as the IN operator.
- The SELECT list cannot include any references to values that evaluate to a BLOB, ARRAY, CLOB, or NCLOB.
- A subquery cannot be immediately enclosed in a set function.
- The BETWEEN operator cannot be used with a subquery. However, the BETWEEN operator can be used within the


Microsoft SQL Server Management Studio Express
File Edit View Query Tools Window Community Help
New Query
college
Execute
Object Explorer
WINCTRL-61E7K6K\SQLEXPRESS (
Databases
System Databases
class
college
Database Diagrams
Tables
System Tables
dbo.employee
dbo.marks
dbo.student_newtal
dbo.students
Views
Synonyms
Programmability
Security
hexa1
one
school
test
training
Security
Server Objects
Replication
Management
WINCTRL-61E7K6K\...\SQLQuery1.sql*
Summary
//Exists operator used to check whether sub-query return any rows, often used to check the existence of related data in another table
select * from students where not exists(select id from marks where m1>90)
select * from students where not exists(select id from marks where m1>80)
--------------corrlelated query---
#-- reference columns from the outer query
select * from students where id in(select id from marks where students.id=id)
----------------------Any operator---------
#Any: return TRUE if any on of the valus form the sub-query satisfy result condition
Results
Messages
id sname address dob
1 1 Ahil chennai Oct 20 2000 12:00AM
2 2 Tarun bangalore Sep 15 2000 12:00AM
3 3 Samkutha pune NULL
4 4 Bipul Andharapradesh NULL
Query executed successfully.
WINCTRL-61E7K6K\SQLEXPRESS (9.0 SP2)
sa (52)
college
00:00:00
4 rows
Ready
Ln 227
Col 1
Ch 1
INS
12:10

SQL Subquery with FROM clause

find all employees whose salaries are greater than or equal to the highest salary of every department.

➢ *SQL subquery with the ANY operator*

```
SELECT
AVG(order_count) average_order_count_by_staff
FROM
(
  SELECT
  staff_id, COUNT(order_id) order_count
  FROM sales.orders
  GROUP BY staff_id
) t;
```

average_order_count_by_staff
269

- The query that you place in the FROM clause must have a table alias. In this example, we used the t as the table alias for the subquery. To come up with the final result, SQL Server carries the following steps:

Objective

- ORDER BY
- AGGREFATE FUNCTION
- GROUP BY
- INTRODUCTION TO JOINS
- JOIN QUERY
- STRING FUNCTION
- DATE FUNCTION
- MATHEMATICAL FUNCTION

- A string function accepts a string value as an input and returns a string value regardless of the data type (string or numeric).

Category	Function	Description
Position	CHARINDEX	Find position of one or more characters in another value
	LEN	Return the number of characters
	PATINDEX	CHARINDEX on super vitamins...
Transformation	LEFT	Return beginning portion of value
	LOWER	Return value as all lower case characters
	LTRIM	Remove any beginning spaces
	QUOTENAME	Make the value legal for SQL code generation
	REPLACE	Replace one set of characters with another
	REPLICATE	Repeat characters
	REVERSE	Flip the value end to end
	RIGHT	Return the last portion of the value
	RTRIM	Remove any trailing spaces
	SPACE	Create a value of repeated spaces
	STR	Convert a number to a text value.
	STUFF	Insert characters inside another value
	SUBSTRING	Return a portion of a value, such as the middle.
	UPPER	Return value as all UPPER CASE characters
Character set	ASCII	Return the ASCII code for a character
	CHAR	Return the Character for the corresponding ASCII code
	NCHAR	Like CHAR but for UNICODE.
	UNICODE	Like ASCII but for UNICODE.
Soundex	DIFFERENCE	An interesting way to compare differences in strings.

Mathematical Functions

Category	Function	Brief Description
Scientific and Trig Functions	ACOS	Arc Cosine (inverse cosine)
	ASIN	Arc Sine (inverse sine)
	ATAN	Arc Tangent (inverse tangent)
	ATN2	Arc Tangent (inverse tangent)
	COS	Cosine
	COT	Cotangent
	DEGREES	Degrees from Radians
	EXP	Exponent
	LOG	Natural logarithm (ln)
	LOG10	Log base 10
	PI	Apple pie? I think not!
	POWER	Power function
	RADIANS	Radians from Degrees
	SIN	Sine
	SQRT	Square Root
	SQUARE	Square
	TAN	Tangent
Rounding Functions	CEILING	Ceiling
	FLOOR	Floor
	ROUND	Round Number

93.20%

15:17
19-03-2025