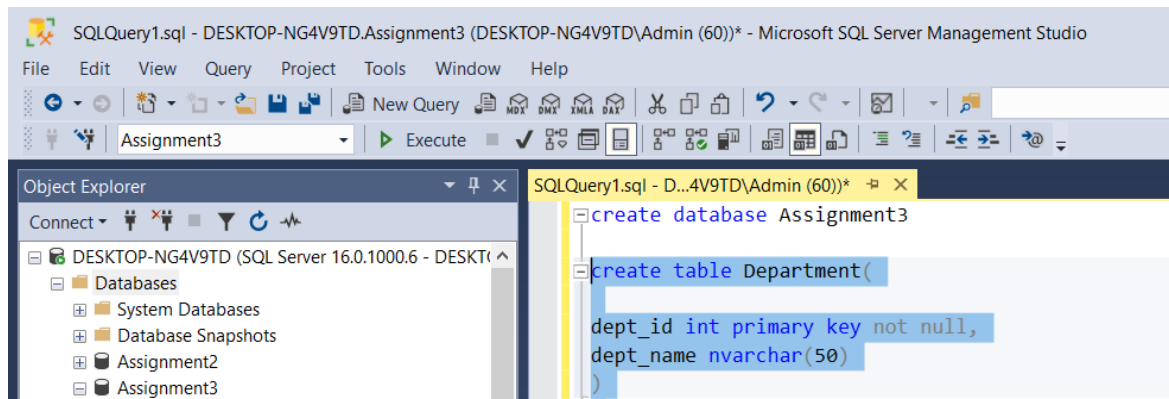


Assignment 3 : Retrieve data using Group By clause

Sample table1:Department

-dept_id

-dept_name



Sample table2: Employee

-emp_id

-dept_id

-mgr_id

-emp_name

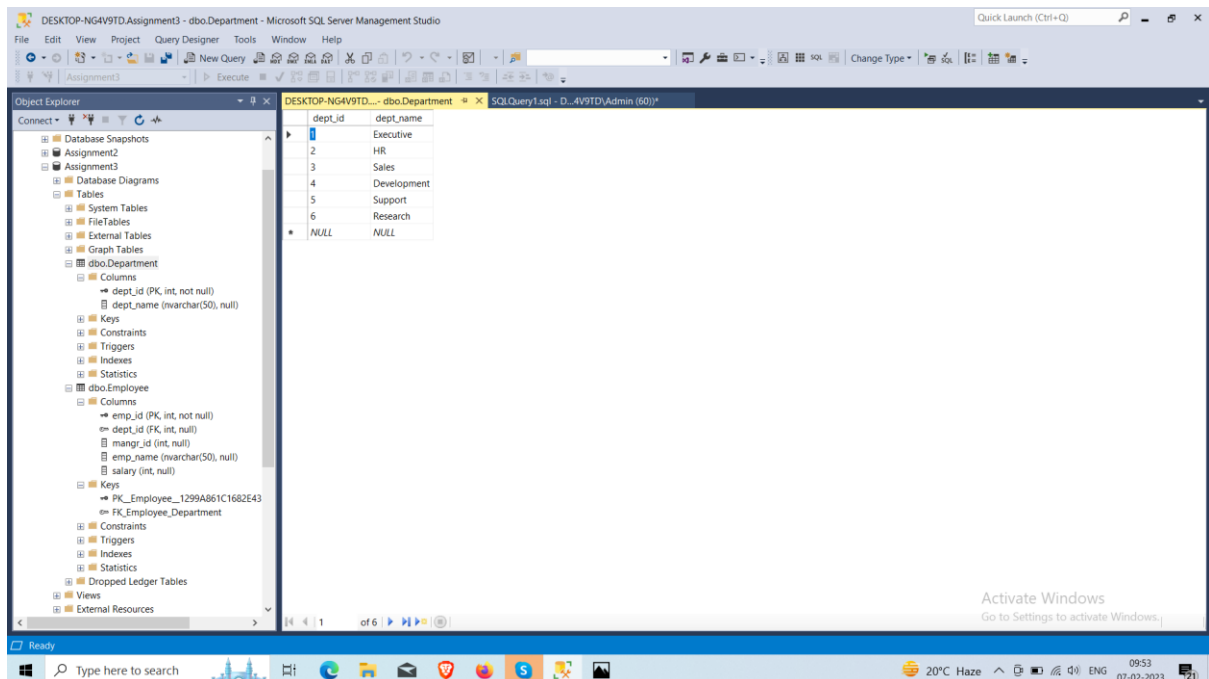
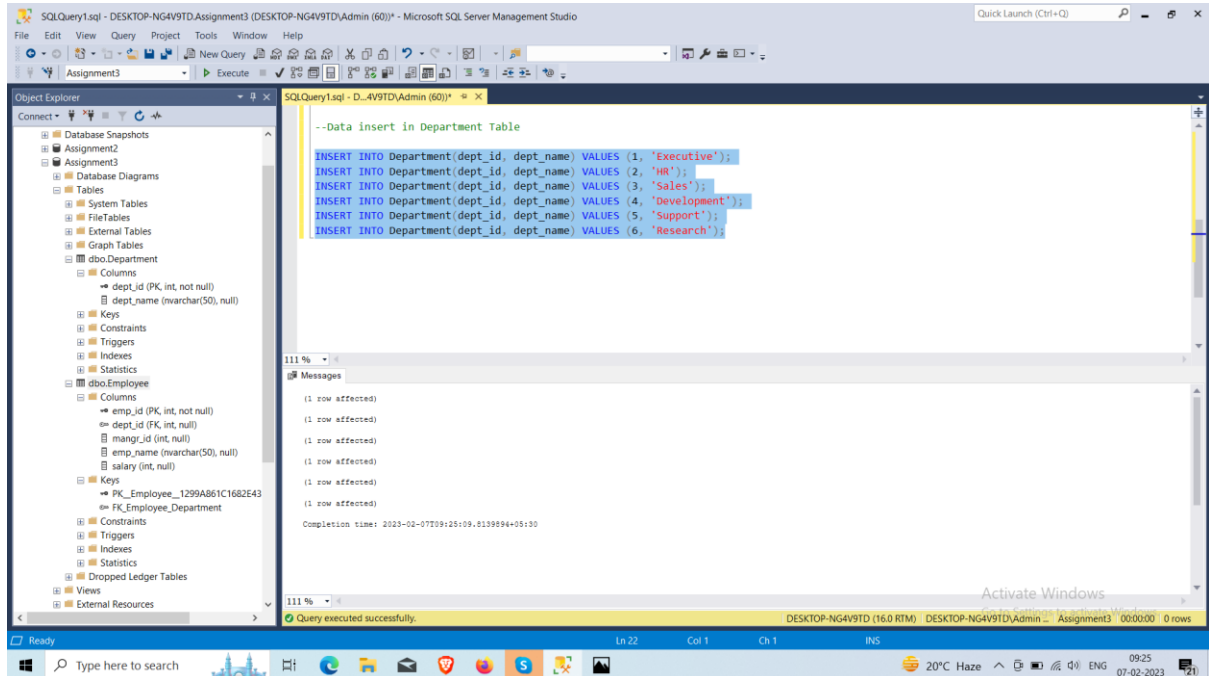
-salary

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'Tables' folder is expanded, showing the 'dbo.Department' table. The 'Columns' folder for 'dbo.Department' is expanded, showing two columns: 'dept_id' (PK, int, not null) and 'dept_name' (nvarchar(50), null). On the right, the SQL script for creating the 'Employee' table is shown. The script defines the following columns: 'emp_id' (int, primary key, not null), 'dept_id' (int), 'mgr_id' (int), 'emp_name' (nvarchar(50)), and 'salary' (int).

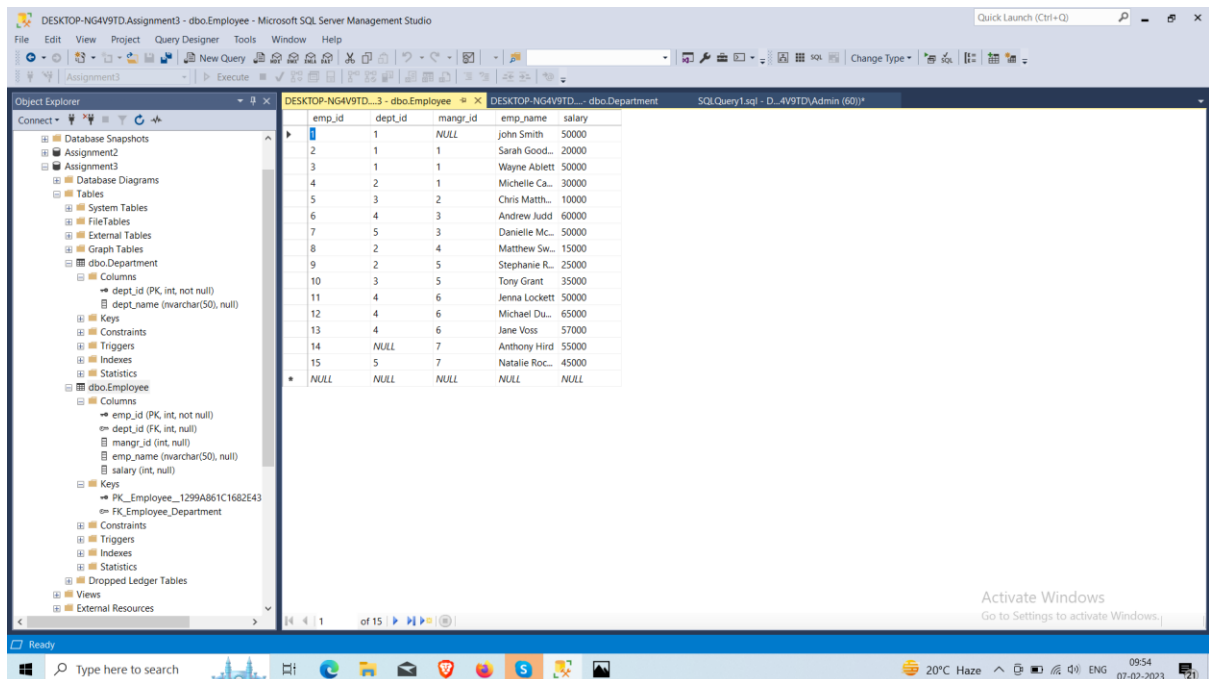
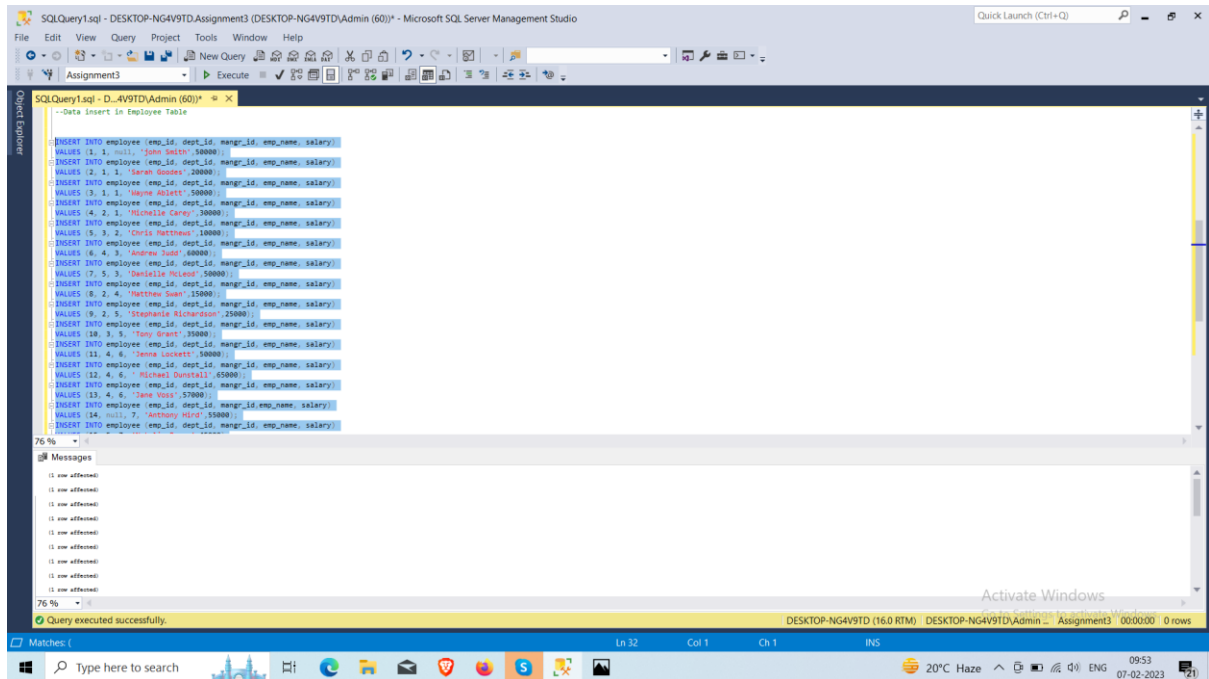
```
create table Employee(  
    emp_id int primary key not null,  
    dept_id int,  
    mgr_id int,  
    emp_name nvarchar(50),  
    salary int  
)
```

Insert data in table

- Table Department



• Table Employoy

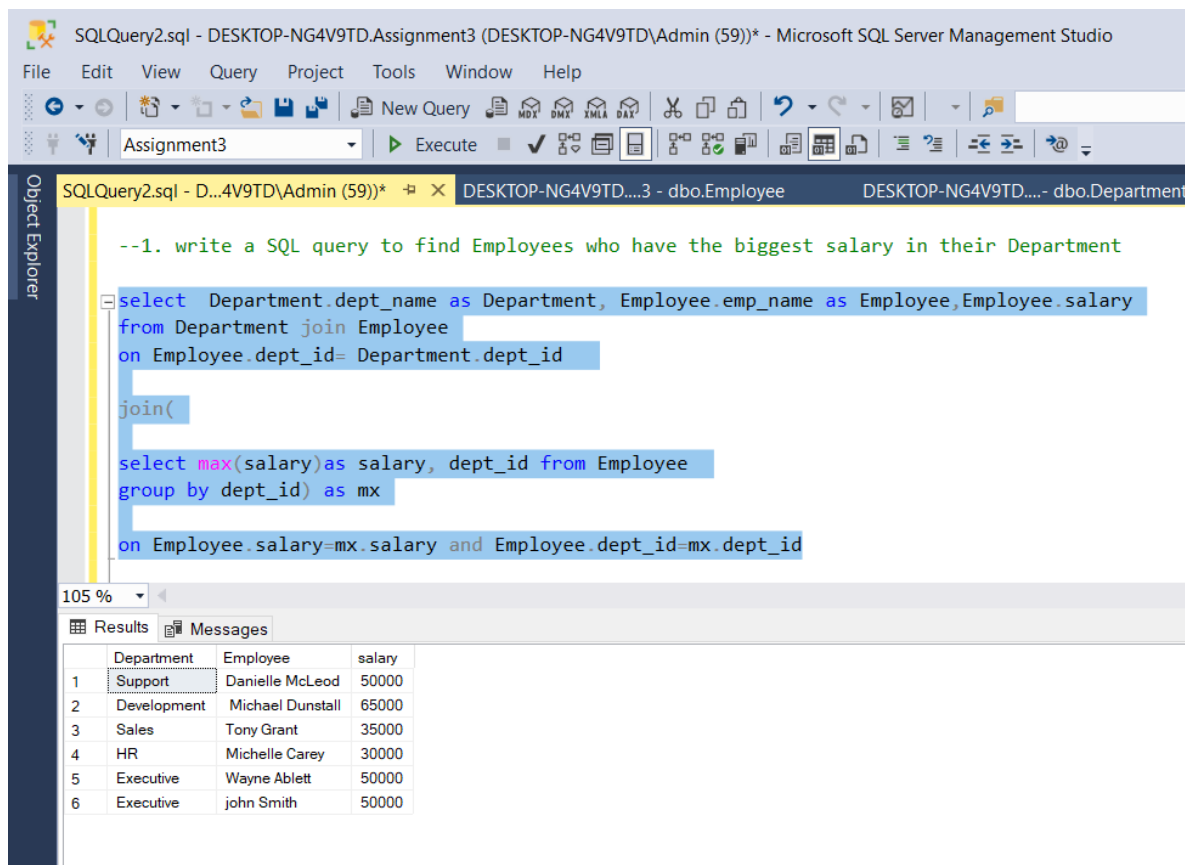


1. write a SQL query to find Employees who have the biggest salary in their Department

```
select Department.dept_name as Department, Employee.emp_name as  
Employee, Employee.salary  
from Department join Employee  
on Employee.dept_id= Department.dept_id
```

join(

```
select max(salary) as salary, dept_id from Employee  
group by dept_id) as mx  
on Employee.salary=mx.salary and Employee.dept_id=mx.dept_id
```



The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the file is 'SQLQuery2.sql' in 'DESKTOP-NG4V9TD.Assignment3'. The menu bar includes File, Edit, View, Query, Project, Tools, Window, and Help. The toolbar contains various icons for file operations, query execution, and formatting. The 'Object Explorer' on the left shows the database structure. The main query editor displays the following SQL code:

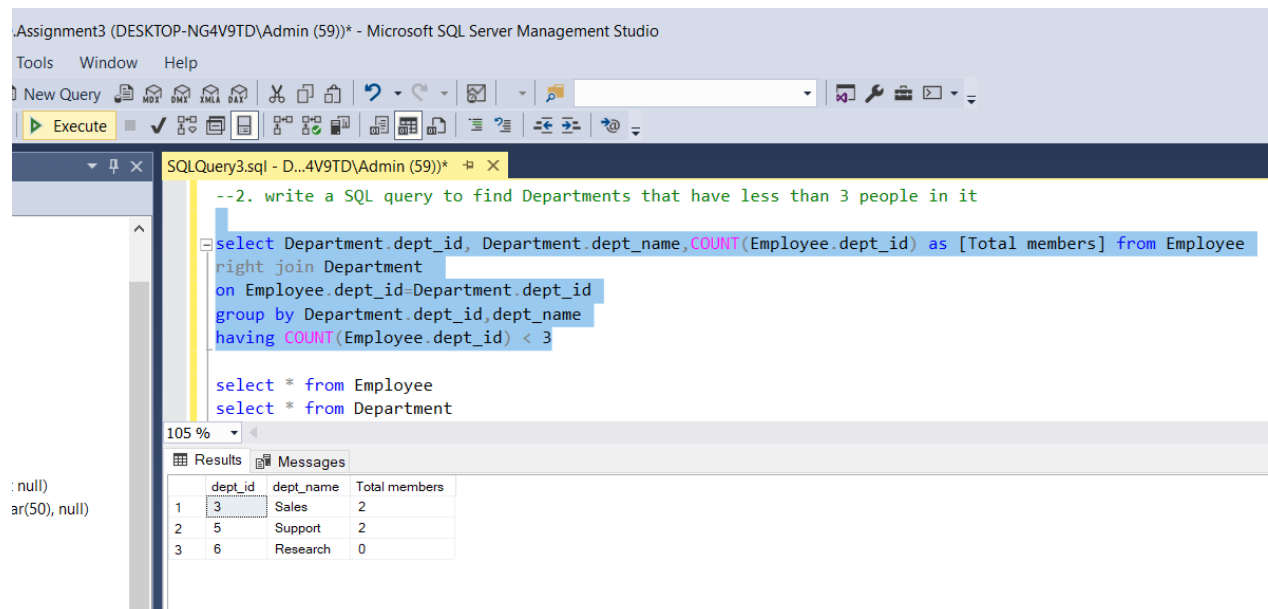
```
--1. write a SQL query to find Employees who have the biggest salary in their Department  
  
select Department.dept_name as Department, Employee.emp_name as Employee, Employee.salary  
from Department join Employee  
on Employee.dept_id= Department.dept_id  
  
join(  
  
select max(salary) as salary, dept_id from Employee  
group by dept_id) as mx  
  
on Employee.salary=mx.salary and Employee.dept_id=mx.dept_id
```

The 'Results' pane at the bottom shows the output of the query as a table with 6 rows and 3 columns: Department, Employee, and salary.

	Department	Employee	salary
1	Support	Danielle McLeod	50000
2	Development	Michael Dunstall	65000
3	Sales	Tony Grant	35000
4	HR	Michelle Carey	30000
5	Executive	Wayne Ablett	50000
6	Executive	john Smith	50000

2. write a SQL query to find Departments that have less than 3 people in it

```
select Department.dept_id, Department.dept_name, COUNT(Employee.dept_id) as [Total members]
from Employee
right join Department
on Employee.dept_id=Department.dept_id
group by Department.dept_id, dept_name
having COUNT(Employee.dept_id) < 3
```



The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the file is 'Assignment3 (DESKTOP-NG4V9TD\Admin (59))* - Microsoft SQL Server Management Studio'. The menu bar includes 'Tools', 'Window', and 'Help'. The toolbar contains various icons for file operations, execution, and formatting. The 'SQLQuery3.sql' file is open, displaying the following SQL query:

```
--2. write a SQL query to find Departments that have less than 3 people in it
select Department.dept_id, Department.dept_name, COUNT(Employee.dept_id) as [Total members] from Employee
right join Department
on Employee.dept_id=Department.dept_id
group by Department.dept_id, dept_name
having COUNT(Employee.dept_id) < 3

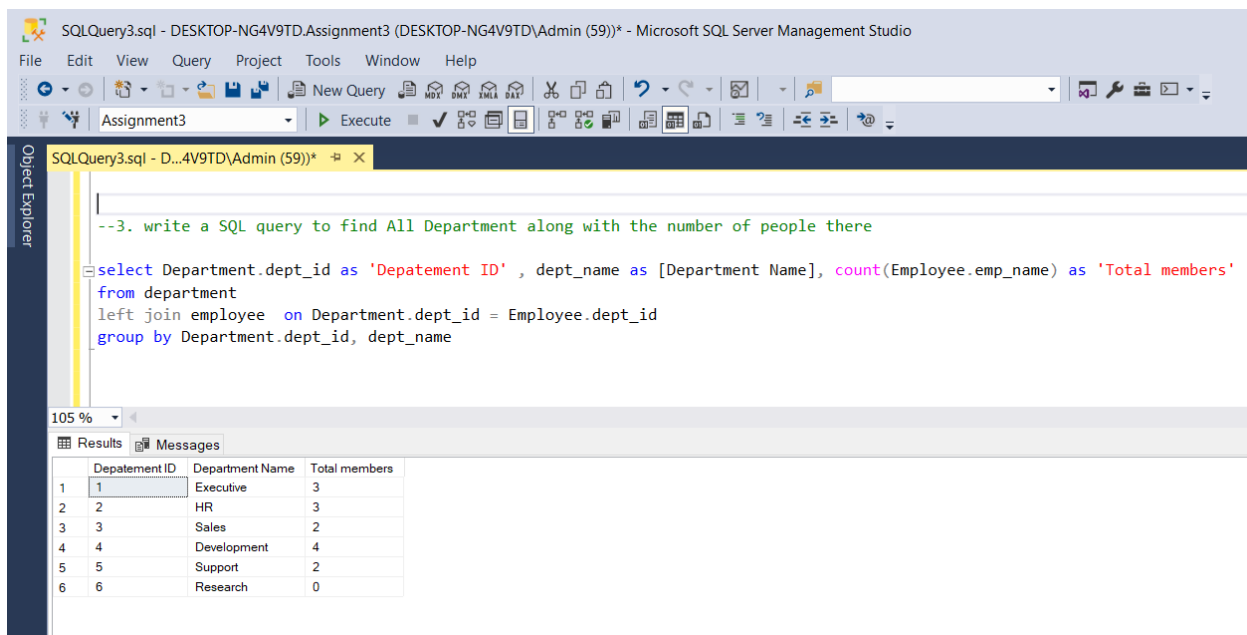
select * from Employee
select * from Department
```

The 'Results' tab is active, showing the output of the query. The results are displayed in a table with three columns: 'dept_id', 'dept_name', and 'Total members'. The table contains three rows of data:

dept_id	dept_name	Total members
1	Sales	2
2	Support	2
3	Research	0

3. write a SQL query to find All Department along with the number of people there

```
select Department.dept_id as 'Depatement ID' , dept_name as [Department Name],  
count(Employee.emp_name) as 'Total members'  
from department  
left join employee on Department.dept_id = Employee.dept_id  
group by Department.dept_id, dept_name
```



The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the file is 'SQLQuery3.sql' located in 'DESKTOP-NG4V9TD.Assignment3 (DESKTOP-NG4V9TD\Admin (59))'. The menu bar includes File, Edit, View, Query, Project, Tools, Window, and Help. The toolbar contains various icons for file operations, query execution, and formatting. The 'Object Explorer' on the left shows the 'Assignment3' database. The main query editor displays the following SQL query:

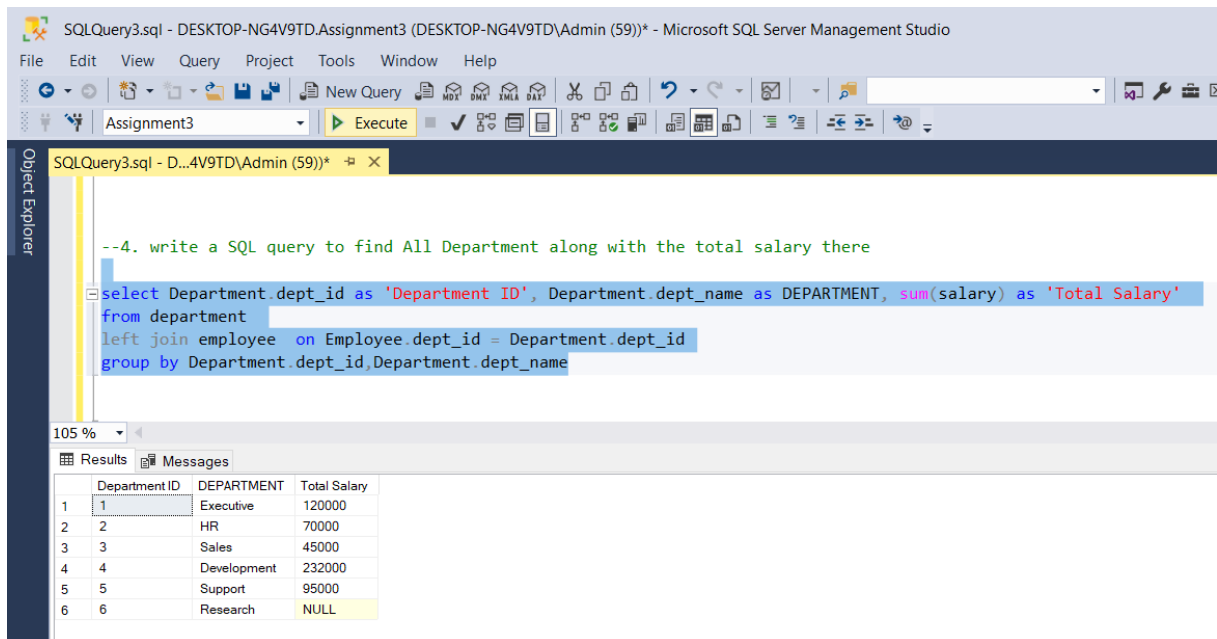
```
--3. write a SQL query to find All Department along with the number of people there  
  
select Department.dept_id as 'Depatement ID' , dept_name as [Department Name], count(Employee.emp_name) as 'Total members'  
from department  
left join employee on Department.dept_id = Employee.dept_id  
group by Department.dept_id, dept_name
```

Below the query editor, the 'Results' tab is active, showing a table with 3 columns: 'Depatement ID', 'Department Name', and 'Total members'. The table contains 6 rows of data:

	Depatement ID	Department Name	Total members
1	1	Executive	3
2	2	HR	3
3	3	Sales	2
4	4	Development	4
5	5	Support	2
6	6	Research	0

4. write a SQL query to find All Department along with the total salary there

```
select Department.dept_id as 'Department ID', Department.dept_name as DEPARTMENT,  
sum(salary) as 'Total Salary'  
from department  
left join employee on Employee.dept_id = Department.dept_id  
group by Department.dept_id, Department.dept_name
```



The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor displays the following SQL query:

```
--4. write a SQL query to find All Department along with the total salary there  
  
select Department.dept_id as 'Department ID', Department.dept_name as DEPARTMENT, sum(salary) as 'Total Salary'  
from department  
left join employee on Employee.dept_id = Department.dept_id  
group by Department.dept_id, Department.dept_name
```

The query results are displayed in the Results pane below the query editor. The results are as follows:

	Department ID	DEPARTMENT	Total Salary
1	1	Executive	120000
2	2	HR	70000
3	3	Sales	45000
4	4	Development	232000
5	5	Support	95000
6	6	Research	NULL