

DB

Task 2

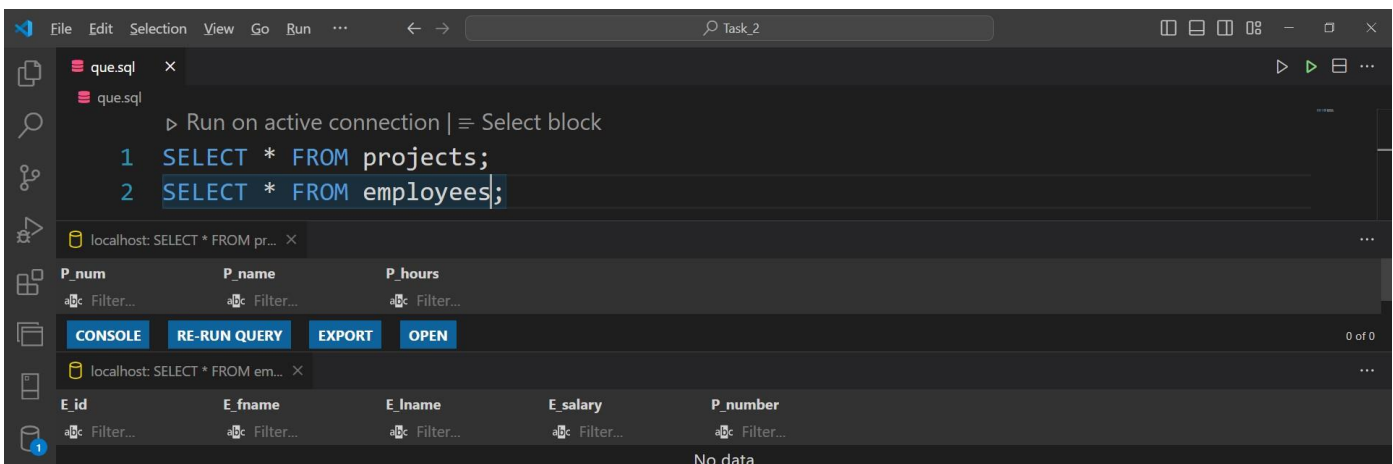
إسلام محمد عطية محمد

سكشن 1 (20220126)

Task 2

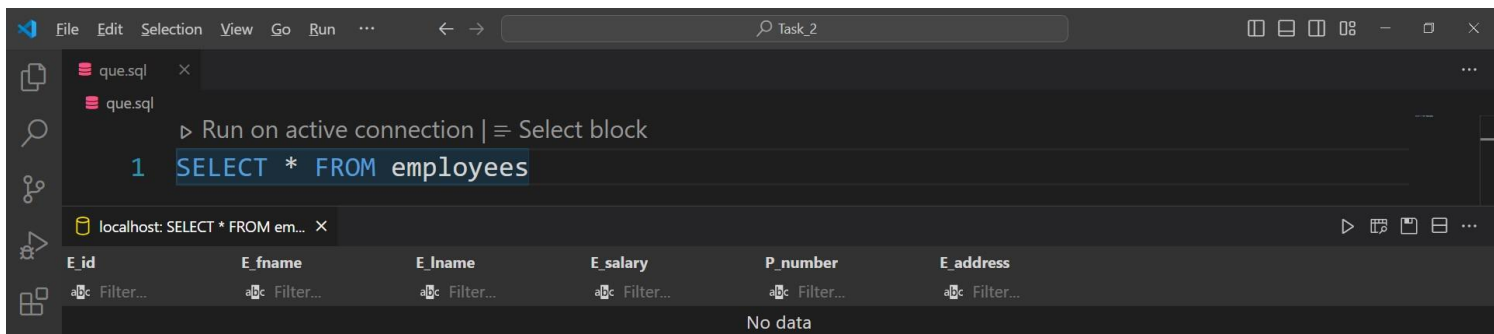
1. Create Employees and Projects tables with the following constraint:
(E_fname ,E_salary,P_name must be entered from user, E_salary must be greater than 2500, p_hours must be greater than 25).

```
1 CREATE TABLE Projects
2 (
3     P_num INT PRIMARY KEY ,
4     P_name VARCHAR(100) NOT NULL,
5     P_hours INT,
6     CHECK (P_hours > 25)
7 );
8 CREATE TABLE Employees
9 (
10    E_id INT PRIMARY KEY AUTO_INCREMENT,
11    E_fname VARCHAR(50) NOT NULL,
12    E_lname VARCHAR(50),
13    E_salary INT NOT NULL,
14    P_number int,
15    FOREIGN KEY (P_number) REFERENCES Projects(P_num),
16    CHECK (E_salary > 2500),
17
18 );
```



2. Add E_address column to employee table that must be only cairo, giza, and helwan,

```
1 ALTER TABLE employees
2 ADD E_address ENUM('cairo', 'giza', 'helwan')
```



The screenshot shows a SQL IDE interface. The top menu bar includes File, Edit, Selection, View, Go, Run, and a search icon. The main editor area shows a file named 'que.sql' with the following SQL query:

```
1 SELECT * FROM employees
```

Below the editor, there is a tab labeled 'localhost: SELECT * FROM em...'. The table view below this tab shows the following columns:

E_id	E_fname	E_lname	E_salary	P_number	E_address
Filter...	Filter...	Filter...	Filter...	Filter...	Filter...
No data					

3. Insert values into two tables.

```
1 INSERT INTO projects (P_num, P_name, P_hours)
2 VALUE
3     (10, 'IT', 35),
4     (20, 'C', 28),
5     (30, 'Network', 26),
6     (40, 'SQL Server', 30);
```

localhost: SELECT * FROM pr... X

P_num	P_name	P_hours
10	IT	35
20	C	28
30	Network	26
40	SQL Server	30

```
1 INSERT INTO employees(E_fname, E_lname, E_salary, P_Number)
2 VALUES
3     ('Ahmed', 'Mohamed', 2800, 20),
4     ('Nada', 'Ibrahim', 6000, 40),
5     ('Ali', '', 3000, 10),
6     ('Ahmed', 'Mohamed', 4200, 10);
```

Task_2

que.sql X

Run on active connection | Select block

```
1 SELECT * FROM employees;
```

localhost: SELECT * FROM em... X

E_id	E_fname	E_lname	E_salary	P_number	E_address
9	Ahmed	Mohamed	2800	20	NULL
10	Nada	Ibrahim	6000	40	NULL
11	Ali		3000	10	NULL
12	Ahmed	Mohamed	4200	10	NULL

4. Select Project name start with N.

The screenshot shows the SQL Server Enterprise Manager interface. The query editor displays the following SQL statement:

```
1 SELECT (P_name) FROM projects WHERE P_name LIKE 'N%'
```

The query is executed, and the results are shown in the Results pane. The results pane displays a table with one column, **P_name**, and one row with the value **Network**.

5. Select employee salary from 3000 to 6000.

The screenshot shows the SQL Server Enterprise Manager interface. The query editor displays the following SQL statement:

```
1 SELECT (E_salary) FROM employees WHERE E_salary BETWEEN 3000 AND 6000
```

The query is executed, and the results are shown in the Results pane. The results pane displays a table with one column, **E_salary**, and three rows with the values **6000**, **3000**, and **4200**.

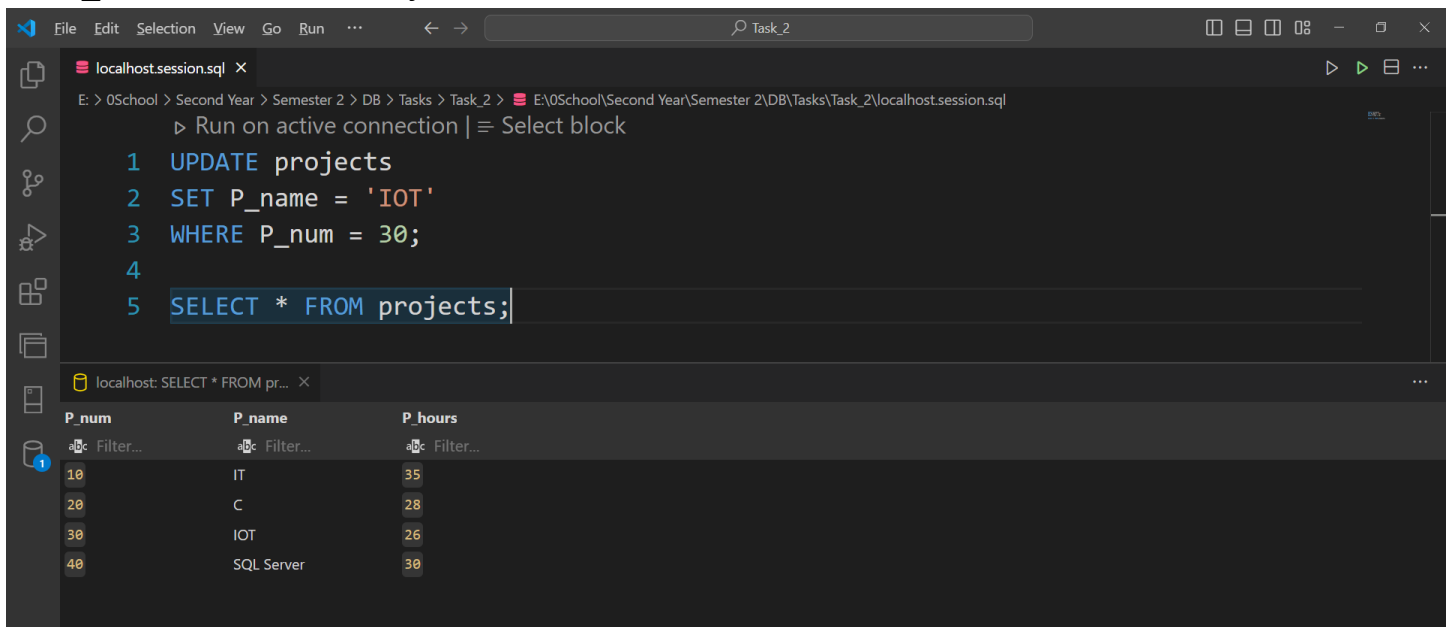
6. Select number of Employees without repeated name.

The screenshot shows the SQL Server Enterprise Manager interface. The query editor displays the following SQL statement:

```
1 SELECT COUNT(DISTINCT E_fname) FROM employees
```

The query is executed, and the results are shown in the Results pane. The results pane displays a table with one column, **COUNT(DISTINC...**, and one row with the value **3**.

7. Set P_name = IOT with Project number 30.



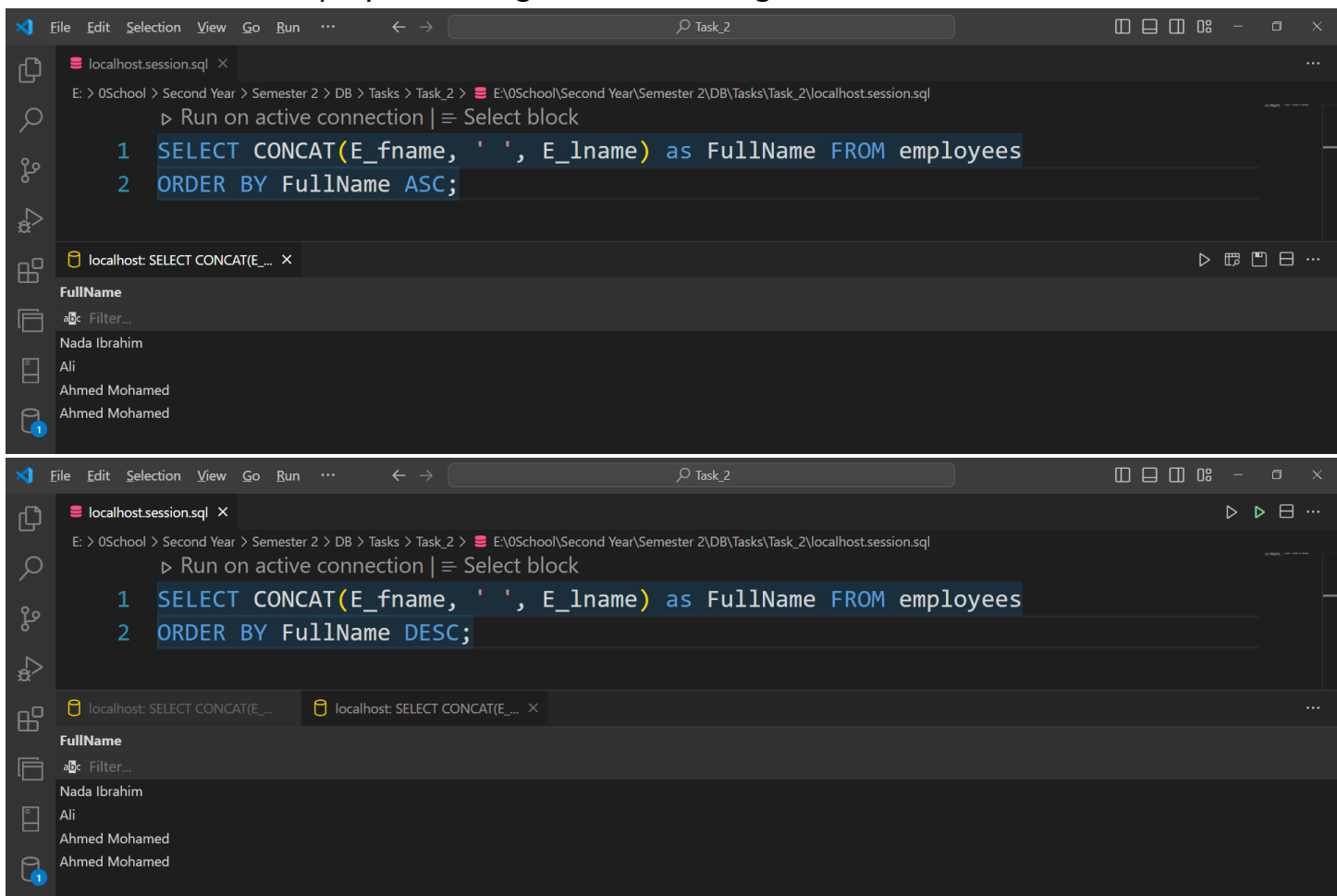
The screenshot shows the SQL Server Enterprise Manager interface. The query editor displays the following SQL code:

```
1 UPDATE projects
2 SET P_name = 'IOT'
3 WHERE P_num = 30;
4
5 SELECT * FROM projects;
```

The query results pane shows the following data:

P_num	P_name	P_hours
10	IT	35
20	C	28
30	IOT	26
40	SQL Server	30

8. Select full name of employees sorting then descending.



The first screenshot shows the SQL query editor with the following code:

```
1 SELECT CONCAT(E_fname, ' ', E_lname) as FullName FROM employees
2 ORDER BY FullName ASC;
```

The query results pane shows the following data:

FullName
Nada Ibrahim
Ali
Ahmed Mohamed
Ahmed Mohamed

The second screenshot shows the same SQL query editor with the following code:

```
1 SELECT CONCAT(E_fname, ' ', E_lname) as FullName FROM employees
2 ORDER BY FullName DESC;
```

The query results pane shows the following data:

FullName
Nada Ibrahim
Ali
Ahmed Mohamed
Ahmed Mohamed

9. Select P_name of maximum hours and E_name that works on it.

```
1 SELECT P_name, P_hours, E_fname FROM employees
2 JOIN projects
3 ON employees.P_number = projects.P_num AND projects.P_hours = (SELECT max(projects.p_hours) FROM projects)
```

localhost: SELECT P_name, P... X

P_name	P_hours	E_fname
IT	35	Ali
IT	35	Ahmed