

CIS 430

Esha Attiq

Lab Assignment 3

Objective: Querying a Relational Database COMPANY Database

*note: this is Lab 3, I did it inside my previous Lab 2 since it's using the same data

1. Added myself inside EMPLOYEE, DEPENDENT, WORKS_ON table then used a single SELECT to get my entries. -- insert myself into EMPLOYEE

```
INSERT INTO EMPLOYEE VALUES ('Esha', '', 'Attiq', '77777777', '6-Mar-02', '1010 Cleveland, CSU, OH', 'F', 40000, 987654321,6);
```

```
-- insert my own DEPENDENT, a Cat named Gato, with same ESSN value
```

```
INSERT INTO DEPENDENT VALUES ('77777777', 'Gato', 'F', '22-Mar-99', 'Cat');
```

```
-- Insert my data into WORKS_ON with same ESSN value
```

```
INSERT INTO WORKS_ON VALUES ('77777777', 3, 33.5);
```

```
INSERT INTO WORKS_ON VALUES ('77777777', 4, 33.5);
```

```
-- one SELECT to get my entries i just entered
```

```
SELECT e.*, d.*, w.*
```

```
FROM EMPLOYEE E, DEPENDENT D, WORKS_ON W
```

```
WHERE E.FNAME = 'Esha' AND E.LNAME = 'Attiq' AND E.SSN=D.ESSN AND E.SSN=W.ESSN;
```

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

```
-- insert myself into EMPLOYEE
INSERT INTO EMPLOYEE VALUES ('Esha', '', 'Attiq', '77777777', '6-Mar-02', '1010 Cleveland, CSU, OH', 'F', 40000, 987654321,6);

-- insert my own DEPENDENT, a Cat named Gato, with same ESSN value
INSERT INTO DEPENDENT VALUES ('77777777', 'Gato', 'F', '22-Mar-99', 'Cat');

-- Insert my data into WORKS_ON with same ESSN value
INSERT INTO WORKS_ON VALUES ('77777777', 3, 33.5);
INSERT INTO WORKS_ON VALUES ('77777777', 4, 33.5);

-- one SELECT to get my entries i just entered
SELECT e.*, d.*, w.*
FROM EMPLOYEE E, DEPENDENT D, WORKS_ON W
WHERE E.FNAME = 'Esha' AND E.LNAME = 'Attiq' AND E.SSN=D.ESSN AND E.SSN=W.ESSN;
```

The Results pane shows the output of the query, displaying columns FNAME, MINIT, LNAME, SSN, BDATE, ADDRESS, SEX, SALARY, SUPERSSN, DNO, ESSN, DEPENDENT_NAME, SEX, BDATE, RELATIONSHIP, ESSN, PNO, and HOURS. The results are as follows:

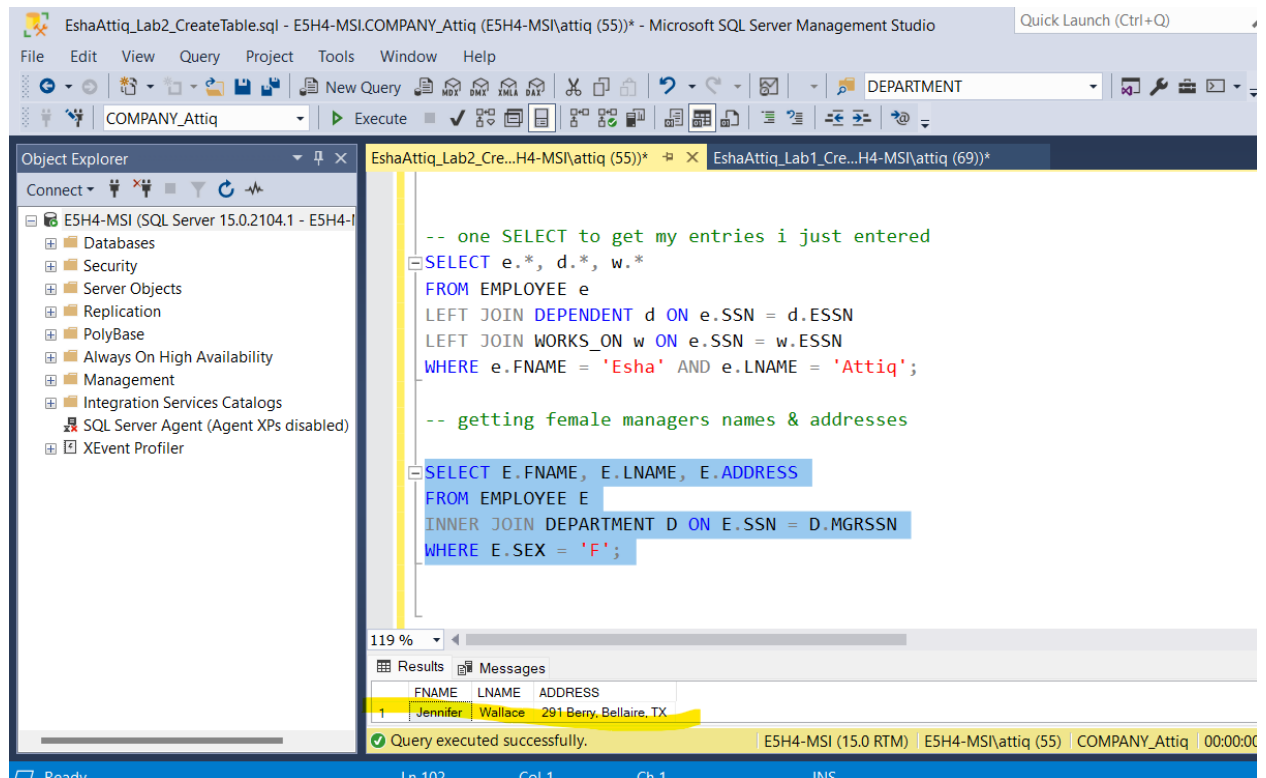
	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO	ESSN	DEPENDENT_NAME	SEX	BDATE	RELATIONSHIP	ESSN	PNO	HOURS
1	Esha	A	Attiq	77777777	2002-03-06	1010 Cleveland, CSU, OH	F	40000.00	987654321	6	77777777	Gato	F	1999-03-22	Cat	77777777	3	33.50
2	Esha	A	Attiq	77777777	2002-03-06	1010 Cleveland, CSU, OH	F	40000.00	987654321	6	77777777	Gato	F	1999-03-22	Cat	77777777	4	33.50

The status bar at the bottom indicates "Query executed successfully." and "E5H4-MSI (15.0 RTM) | E5H4-MSI\attiq (59) | COMPANY_Attig | 00:00:00 | 2 rows".

Q1) Getting female managers names & addresses

-- getting female managers names & addresses

```
SELECT E.FNAME, E.LNAME, E.ADDRESS
FROM EMPLOYEE E
INNER JOIN DEPARTMENT D ON E.SSN = D.MGRSSN
WHERE E.SEX = 'F';
```



Q2) A list of all project number's for projects that involve an employee whos last name is smith, either as a worker or as a manager of the dept. that controls the project.

-- list for Smith as worker or manager of dept. that controls proj

```
SELECT DISTINCT P.PNUMBER
FROM PROJECT P
INNER JOIN WORKS_ON W ON P.PNUMBER = W.PNO
INNER JOIN EMPLOYEE E ON W.ESSN = E.SSN
WHERE E.LNAME = 'Smith'
```

UNION

```
SELECT DISTINCT P.PNUMBER
```

```

FROM PROJECT P
INNER JOIN DEPARTMENT D ON P.DNUM = D.DNUMBER
INNER JOIN EMPLOYEE E ON D.MGRSSN = E.SSN
WHERE E.LNAME = 'Smith';

```

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

```

-- getting female managers names & addresses
SELECT E.FNAME, E.LNAME, E.ADDRESS
FROM EMPLOYEE E
INNER JOIN DEPARTMENT D ON E.SSN = D.MGRSSN
WHERE E.SEX = 'F';

-- list for Smith as worker or manager of dept. that controls proj
SELECT DISTINCT P.PNUMBER
FROM PROJECT P
INNER JOIN WORKS_ON W ON P.PNUMBER = W.PNO
INNER JOIN EMPLOYEE E ON W.ESSN = E.SSN
WHERE E.LNAME = 'Smith'

UNION

SELECT DISTINCT P.PNUMBER
FROM PROJECT P
INNER JOIN DEPARTMENT D ON P.DNUM = D.DNUMBER
INNER JOIN EMPLOYEE E ON D.MGRSSN = E.SSN
WHERE E.LNAME = 'Smith';

-- highest ranked employee doesnt report to anyone in the company

```

The results pane shows the output of the query, displaying a table with the following data:

	PNUMBER
1	1
2	2

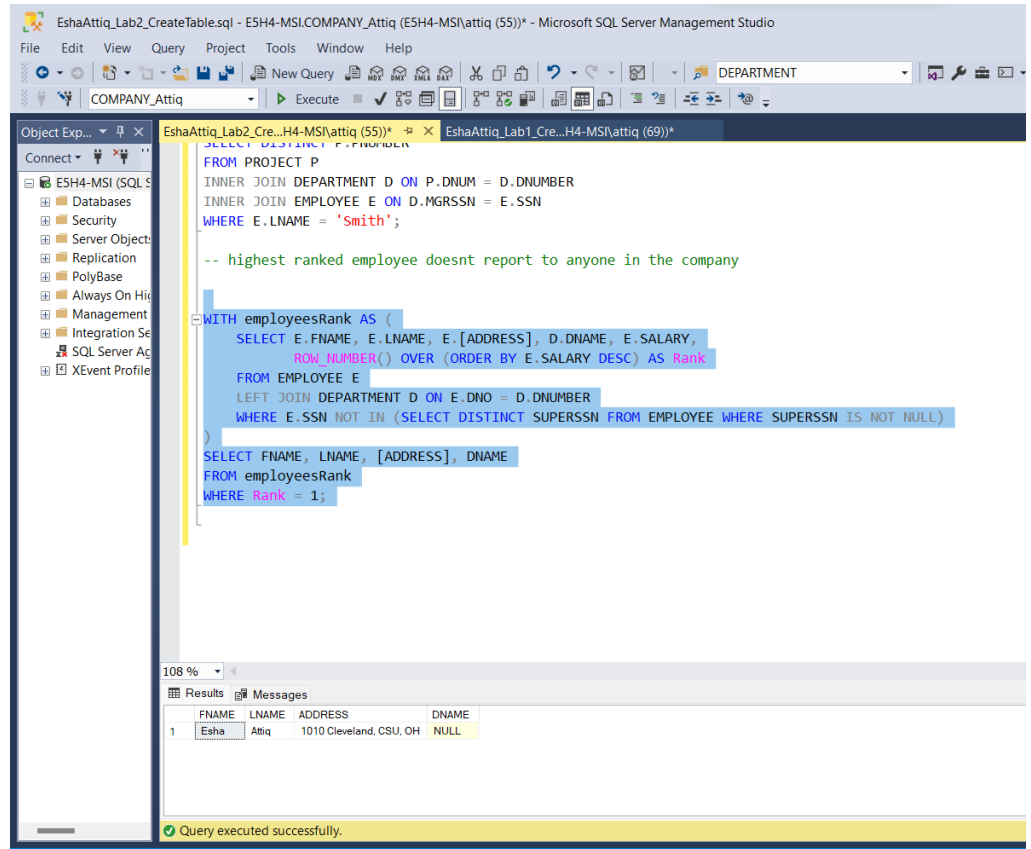
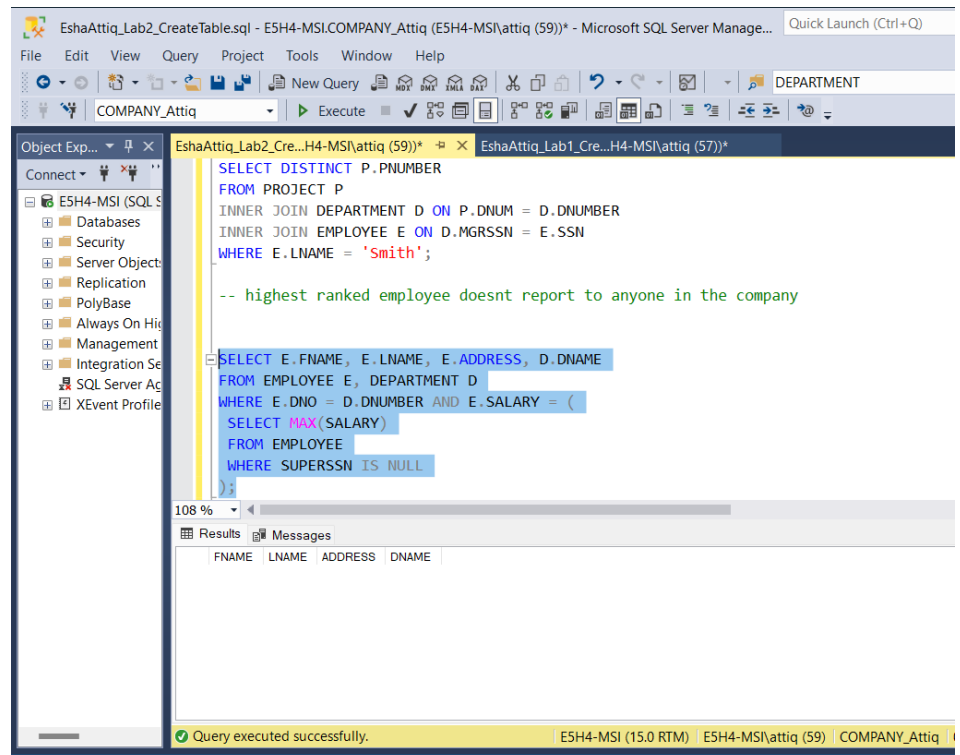
The status bar at the bottom indicates: "Query executed successfully. ESH4-MSI (15.0 RTM) | ESH4-MSI\attiq (59) | COMPANY_Attig | 00:00:00 | 2 rows".

Q3) : Retrieve the name and address and his/her department name of the highest ranked employee who does not report to anybody in the company.

(I tried 2 approaches but I keep getting odd outputs, I'll have both screenshots)

-- highest ranked employee doesnt report to anyone in the company

```
SELECT E.FNAME, E.LNAME,
E.ADDRESS, D.DNAME
FROM EMPLOYEE E,
DEPARTMENT D
WHERE E.DNO = D.DNUMBER
AND E.SALARY = (
    SELECT MAX(SALARY)
    FROM EMPLOYEE
    WHERE SUPERSSN IS NULL
);
```



Q4) For each department, list all the employees who are working in the department with the employee's first and last name and first and last name of his or her immediate supervisor. List the result in the order of each department number and department name. (it works fine, I am doing it correct. but my James Borg Headquarters is glitching)

```
-- List employees in each DEPARTMENT with their supervisor's first and last name
SELECT
    D.DNUMBER ,
    D.DNAME ,
    E1.FNAME AS EMPLOYEEFNAME,
    E1.LNAME AS EMPLOYEEELNAME,
    E2.FNAME AS SUPERVISORFNAME,
    E2.LNAME AS SUPERVISORLNAME
FROM DEPARTMENT D
LEFT JOIN EMPLOYEE E1
ON E1.DNO = D.DNUMBER
LEFT JOIN EMPLOYEE E2
ON E1.SUPERSSN = E2.SSN
ORDER BY
    D.DNUMBER, D.DNAME;
```

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor displays the SQL query for listing employees by department with their supervisors. The query has been executed successfully, and the results are shown in the Results pane below.

Query:

```
-- List employees in each DEPARTMENT with their supervisor's first and last name
SELECT
    D.DNUMBER ,
    D.DNAME ,
    E1.FNAME AS EMPLOYEEFNAME,
    E1.LNAME AS EMPLOYEEELNAME,
    E2.FNAME AS SUPERVISORFNAME,
    E2.LNAME AS SUPERVISORLNAME
FROM DEPARTMENT D
LEFT JOIN EMPLOYEE E1
ON E1.DNO = D.DNUMBER
LEFT JOIN EMPLOYEE E2
ON E1.SUPERSSN = E2.SSN
ORDER BY
    D.DNUMBER, D.DNAME;
```

Results:

	DNUMBER	DNAME	EMPLOYEEFNAME	EMPLOYEEELNAME	SUPERVISORFNAME	SUPERVISORLNAME
1	1	Headquarters	NULL	NULL	NULL	NULL
2	4	Administration	Jennifer	Wallace	James	Borg
3	4	Administration	Ahmad	Jabbar	Jennifer	Wallace
4	4	Administration	Alicia	Zelaya	Jennifer	Wallace
5	5	Research	John	Smith	Jennifer	Wallace
6	5	Research	Franklin	Wong	James	Borg
7	5	Research	Joyce	English	Franklin	Wong
8	5	Research	Ramesh	Narayan	Franklin	Wong
9	5	Research	James	Borg	NULL	NULL
10	7	Automation	NULL	NULL	NULL	NULL

Query executed successfully.

EXTRA BONUS POINT!!!

```
-- List employees in each DEPARTMENT with their supervisor's first and last name
SELECT
    D.DNUMBER ,
    D.DNAME ,
    E1.FNAME AS EMPLOYEEFNAME,
    E1.LNAME AS EMPLOYEEELNAME,
    E2.FNAME AS SUPERVISORFNAME,
    E2.LNAME AS SUPERVISORLNAME
FROM DEPARTMENT D
JOIN EMPLOYEE E1 -- changed left join to just join to remove the DEPARTMENTS without any employees
ON E1.DNO = D.DNUMBER
LEFT JOIN EMPLOYEE E2
ON E1.SUPERSSN = E2.SSN
ORDER BY
    D.DNUMBER, D.DNAME;
```

We just changed that **LEFT JOIN** to **JOIN**

The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The query window displays the SQL code from the previous block. The query has been executed successfully, and the results are shown in the Results pane at the bottom. The results pane shows a table with 8 rows and 6 columns: DNUMBER, DNAME, EMPLOYEEFNAME, EMPLOYEEELNAME, SUPERVISORFNAME, and SUPERVISORLNAME. The data is as follows:

	DNUMBER	DNAME	EMPLOYEEFNAME	EMPLOYEEELNAME	SUPERVISORFNAME	SUPERVISORLNAME
1	4	Administration	Jennifer	Wallace	James	Borg
2	4	Administration	Ahmad	Jabbar	Jennifer	Wallace
3	4	Administration	Alicia	Zelaya	Jennifer	Wallace
4	5	Research	John	Smith	Jennifer	Wallace
5	5	Research	Franklin	Wong	James	Borg
6	5	Research	Joyce	English	Franklin	Wong
7	5	Research	Ramesh	Narayan	Franklin	Wong
8	5	Research	James	Borg	NULL	NULL

The status bar at the bottom indicates "Query executed successfully." and "8 rows".