

Esha Attiq

CIS 430

LAB 6: TRIGGER AND STORED PROCEDURE

Note : the formatting is a little weird but all the screenshots & outputs are there!*

1) I recreated my EMPLOYEE & DEPARTMENT tables without any constraints

The image displays two side-by-side screenshots of the SQL Server Enterprise Manager interface, showing the execution of SQL scripts to recreate the EMPLOYEE and DEPARTMENT tables.

Left Screenshot (EshaAttq_Lab5_Cre...iq's Computer (57)):

```
Use master;
if DB_ID('COMPANY_EshaAttq') is NULL
CREATE DATABASE COMPANY_EshaAttq;

Use COMPANY_EshaAttq
if OBJECT_ID('dbo.EMPLOYEE') is not null
drop table dbo.EMPLOYEE

CREATE TABLE EMPLOYEE(
FNAME VARCHAR(25) not null,
MINIT CHAR,
LNAME VARCHAR(25) not null,
SSN CHAR(9) not null,
BDATE DATE,
[ADDRESS] VARCHAR(50) not null,
SEX CHAR not null,
SALARY DECIMAL(18,2),
SUPERSSN CHAR(9),
DNO INT not null
);
```

Right Screenshot (EshaAttq_Lab6_Cre...iq's Co...):

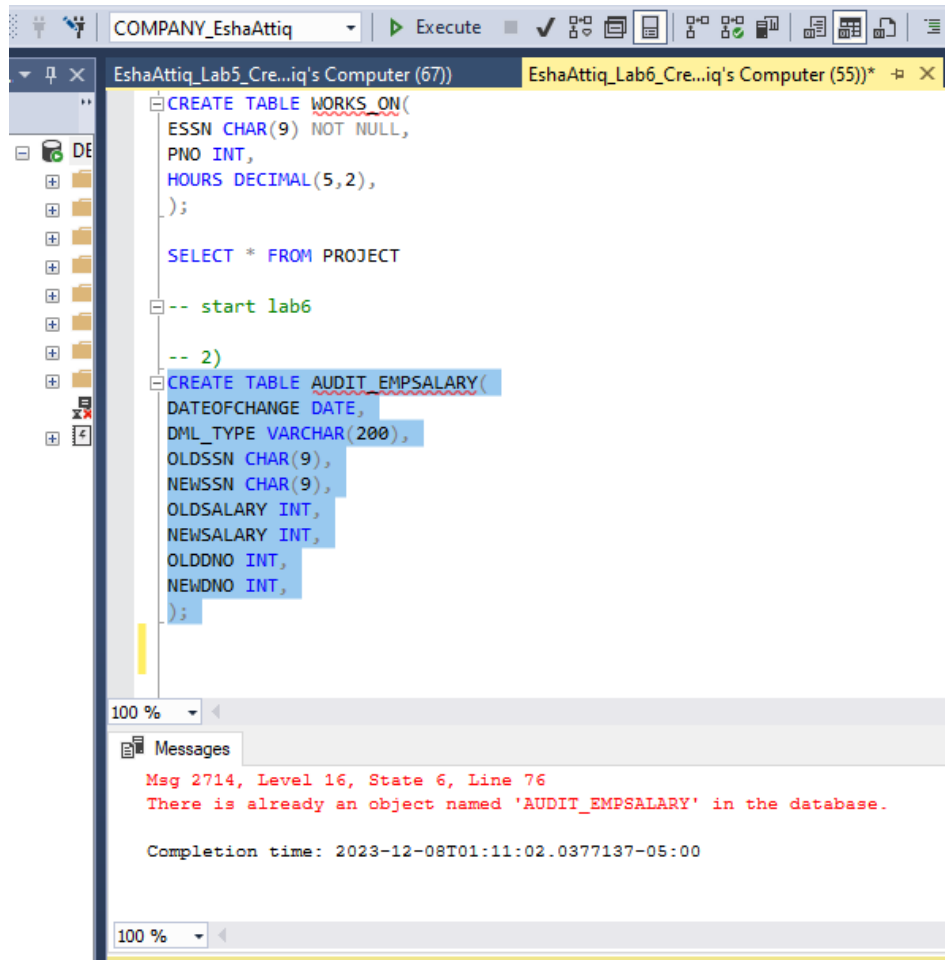
```
Use COMPANY_EshaAttq
if OBJECT_ID('dbo.DEPARTMENT') is not null
drop table dbo.DEPARTMENT

CREATE TABLE DEPARTMENT(
DNAME VARCHAR(25) not null,
DNUMBER CHAR,
MGRSSN CHAR(9) not null,
MGRSTARTDATE DATE,
);

INSERT INTO EMPLOYEE VALUES('John','B','Smith','123456789',
INSERT INTO EMPLOYEE VALUES('Franklin','T','Wong','33344555
INSERT INTO EMPLOYEE VALUES('Joyce','A','English','45345345
INSERT INTO EMPLOYEE VALUES('Ramesh','K','Narayan','6668844
INSERT INTO EMPLOYEE VALUES('James','E','Borg','888665555',
INSERT INTO EMPLOYEE VALUES('Jennifer','S','Wallace','98765
INSERT INTO EMPLOYEE VALUES('Ahmad','V','Jabbar','987987987
INSERT INTO EMPLOYEE VALUES('Alicia','J','Zelaya','99988777
Insert Into DEPARTMENT Values ('Headquarters','1','88866555
Insert Into DEPARTMENT Values ('Administration','4','987654
```

Both screenshots show the "Messages" pane at the bottom, indicating "Commands completed successfully." and "Completion time: 2023-12-08T21:24:47.7260424-05:00".

3) Created an AUDIT_EMPSALARY table



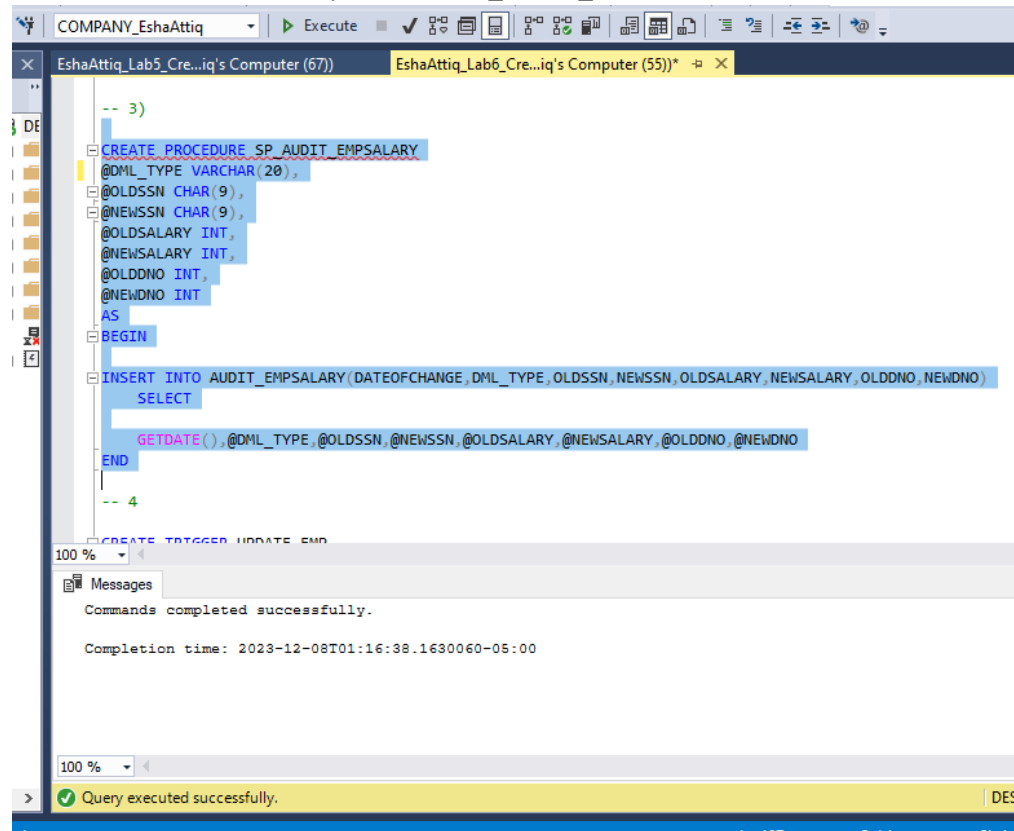
The screenshot shows the SQL Developer interface. The top toolbar includes an 'Execute' button. The main editor displays a SQL script with the following content:

```
CREATE TABLE WORKS_ON(  
  ESSN CHAR(9) NOT NULL,  
  PNO INT,  
  HOURS DECIMAL(5,2),  
);  
  
SELECT * FROM PROJECT  
  
-- start lab6  
  
-- 2)  
CREATE TABLE AUDIT_EMPSALARY(  
  DATEOFCHANGE DATE,  
  DML_TYPE VARCHAR(200),  
  OLDSSN CHAR(9),  
  NEWSSN CHAR(9),  
  OLDSALARY INT,  
  NEWSALARY INT,  
  OLDDNO INT,  
  NEWDNO INT,  
);
```

Below the script, the 'Messages' pane shows an error message:

```
Msg 2714, Level 16, State 6, Line 76  
There is already an object named 'AUDIT_EMPSALARY' in the database.  
  
Completion time: 2023-12-08T01:11:02.0377137-05:00
```

And then created a stored procedure SP_AUDIT_EMPSALARY:



The screenshot displays the SQL Server Enterprise Manager interface. The top toolbar includes buttons for 'Execute', 'Format', 'Copy', 'Paste', 'Print', 'Save', 'Undo', 'Redo', 'Zoom In', 'Zoom Out', 'Full Screen', and 'Help'. The main window shows a script for creating a stored procedure named SP_AUDIT_EMPSALARY. The script is as follows:

```
-- 3)
CREATE PROCEDURE SP_AUDIT_EMPSALARY
@DML_TYPE VARCHAR(20),
@OLDSSN CHAR(9),
@NEWSSN CHAR(9),
@OLDSALARY INT,
@NEWSALARY INT,
@OLDDNO INT,
@NEWDNO INT
AS
BEGIN
INSERT INTO AUDIT_EMPSALARY (DATEOFCHANGE, DML_TYPE, OLDSSN, NEWSSN, OLDSALARY, NEWSALARY, OLDDNO, NEWDNO)
SELECT
GETDATE(), @DML_TYPE, @OLDSSN, @NEWSSN, @OLDSALARY, @NEWSALARY, @OLDDNO, @NEWDNO
END
-- 4

CREATE TRIGGER UPDATE_EMP
```

Below the script, the 'Messages' pane shows the following output:

```
Commands completed successfully.
Completion time: 2023-12-08T01:16:38.1630060-05:00
```

At the bottom, a status bar indicates 'Query executed successfully.' with a green checkmark icon.

4 & 2) Call stored procedures SP_AUDIT_EMPSALARY at end of body of each Trigger to record all history of the changes by UPDATE/INSERT/DELETE.

2.1 UPDATE:

[screenshot on next page]

```
COMPANY_EshaAttiq
Execute
EshaAttiq_Lab5_Cre...iq's Computer (67)
EshaAttiq_Lab6_Cre...iq's Computer (55)*
-- 4) creating triggers
-- update trigger
CREATE TRIGGER UPDATE_EMP
ON EMPLOYEE
FOR UPDATE
AS
BEGIN
    DECLARE @OLDSSN CHAR(9);
    DECLARE @NEWSSN CHAR(9);
    DECLARE @OLDSALARY INT;
    DECLARE @NEWSALARY INT;
    DECLARE @OLDDNO INT;
    DECLARE @NEWDNO INT;

    SELECT * INTO #TEMP1 FROM INSERTED;
    SELECT * INTO #TEMP2 FROM DELETED;
    IF UPDATE(SALARY)
    BEGIN
        WHILE(EXISTS(SELECT SSN FROM #TEMP1))
        BEGIN
            SELECT TOP 1 @OLDSSN=SSN FROM #TEMP2;
            SELECT TOP 1 @NEWSSN=SSN FROM #TEMP2;
            SELECT TOP 1 @OLDSALARY=SALARY FROM #TEMP2;
        END
    END
END
```

100 %

Messages

Commands completed successfully.

Completion time: 2023-12-08T01:21:30.8510064-05:00

100 %

Query executed successfully.

```
COMPANY_EshaAttiq
Execute
EshaAttiq_Lab5_Cre...iq's Computer (67)
EshaAttiq_Lab6_Cre...iq's Computer (55)*
BEGIN
    SELECT TOP 1 @OLDSSN=SSN FROM #TEMP2;
    SELECT TOP 1 @NEWSSN=SSN FROM #TEMP2;
    SELECT TOP 1 @OLDSALARY=SALARY FROM #TEMP2;
    SELECT TOP 1 @NEWSALARY=SALARY FROM #TEMP1;
    SELECT TOP 1 @OLDDNO=DNO FROM #TEMP2;
    SELECT TOP 1 @NEWDNO=DNO FROM #TEMP1;

    EXEC SP_AUDIT_EMPSALARY 'UPDATE',
        @OLDSSN, @NEWSSN, @OLDSALARY, @NEWSALARY, @OLDDNO, @NEWDNO

    DELETE TOP (1) FROM #TEMP1;
    DELETE TOP (1) FROM #TEMP2;
END
END
END
```

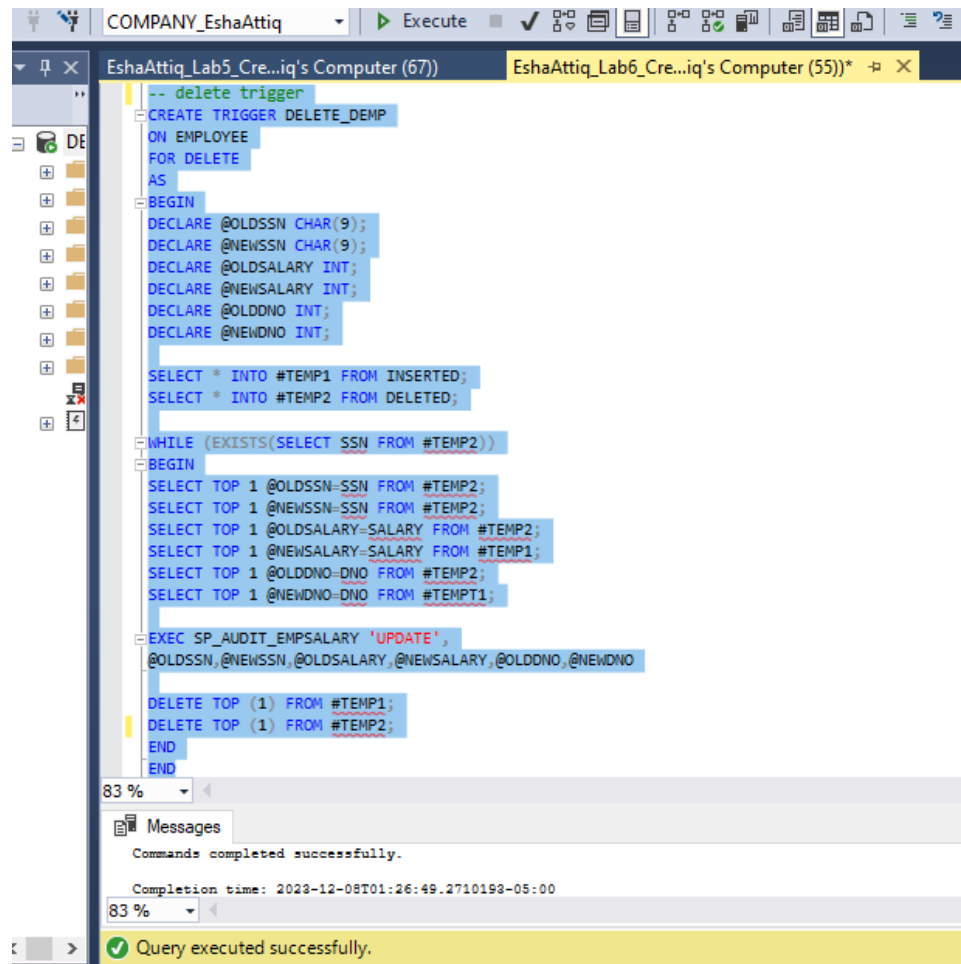
100 %

Messages

Commands completed successfully.

Completion time: 2023-12-08T01:21:30.8510064-05:00

2.2 DELETE:



```
-- delete trigger
CREATE TRIGGER DELETE_DEMP
ON EMPLOYEE
FOR DELETE
AS
BEGIN
    DECLARE @OLDSSN CHAR(9);
    DECLARE @NEWSSN CHAR(9);
    DECLARE @OLDSALARY INT;
    DECLARE @NEWSALARY INT;
    DECLARE @OLDDNO INT;
    DECLARE @NEWDDNO INT;

    SELECT * INTO #TEMP1 FROM INSERTED;
    SELECT * INTO #TEMP2 FROM DELETED;

    WHILE (EXISTS(SELECT SSN FROM #TEMP2))
    BEGIN
        SELECT TOP 1 @OLDSSN=SSN FROM #TEMP2;
        SELECT TOP 1 @NEWSSN=SSN FROM #TEMP2;
        SELECT TOP 1 @OLDSALARY=SALARY FROM #TEMP2;
        SELECT TOP 1 @NEWSALARY=SALARY FROM #TEMP1;
        SELECT TOP 1 @OLDDNO=DNO FROM #TEMP2;
        SELECT TOP 1 @NEWDDNO=DNO FROM #TEMP1;

        EXEC SP_AUDIT_EMPSALARY 'UPDATE',
            @OLDSSN,@NEWSSN,@OLDSALARY,@NEWSALARY,@OLDDNO,@NEWDDNO

        DELETE TOP (1) FROM #TEMP1;
        DELETE TOP (1) FROM #TEMP2;
    END
END
```

83 %

Messages

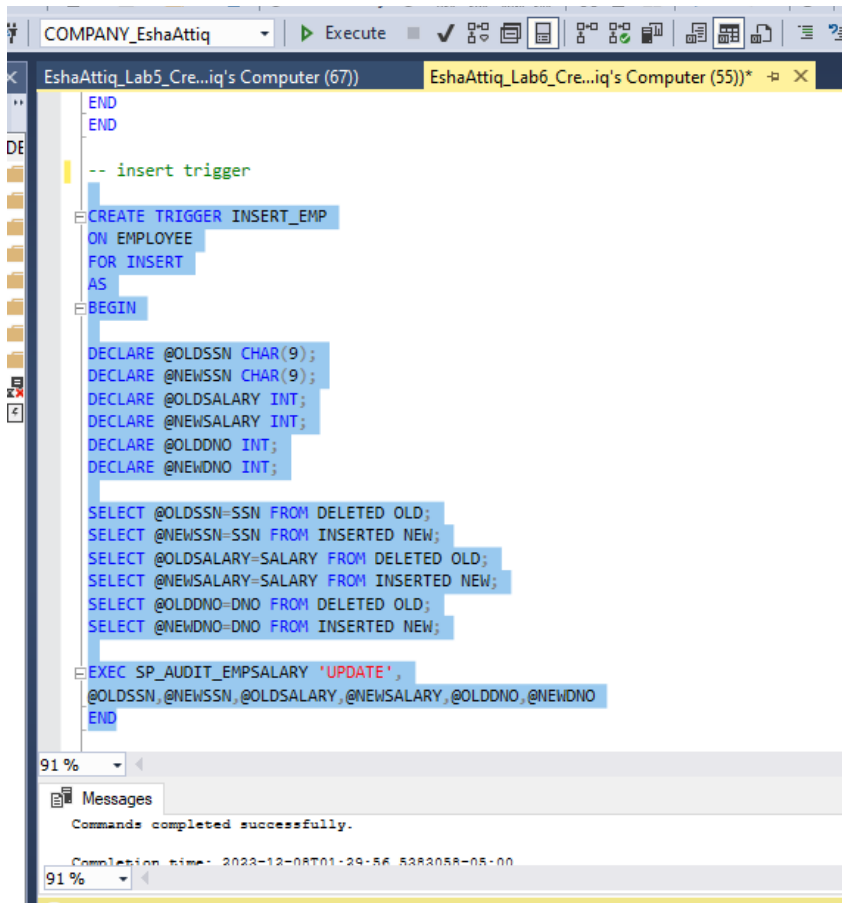
Commands completed successfully.

Completion time: 2023-12-08T01:26:49.2710193-05:00

83 %

Query executed successfully.

2.3 INSERT:



The screenshot shows the SQL Server Enterprise Manager interface. The top toolbar includes buttons for 'Execute', 'Execute with Results', 'Execute with Messages', 'Execute with Log', 'Execute with Statistics', 'Execute with Profiler', 'Execute with SQL Profiler', 'Execute with SQL Server Profiler', and 'Execute with SQL Server Profiler'. The main window displays a T-SQL script for creating an insert trigger named INSERT_EMP on the EMPLOYEE table. The script includes variable declarations for SSN, SALARY, and DNO, and uses SELECT statements to retrieve values from the DELETED and INSERTED tables. The trigger is then executed using the SP_AUDIT_EMPSALARY stored procedure. The bottom status bar shows '91 %' completion and a message: 'Commands completed successfully.' The completion time is 2023-12-08T01:29:56.5383055-05:00.

```
END
END
-- insert trigger
CREATE TRIGGER INSERT_EMP
ON EMPLOYEE
FOR INSERT
AS
BEGIN
DECLARE @OLDSSN CHAR(9);
DECLARE @NEWSSN CHAR(9);
DECLARE @OLDSALARY INT;
DECLARE @NEWSALARY INT;
DECLARE @OLDDNO INT;
DECLARE @NEWDNO INT;

SELECT @OLDSSN=SSN FROM DELETED OLD;
SELECT @NEWSSN=SSN FROM INSERTED NEW;
SELECT @OLDSALARY=SALARY FROM DELETED OLD;
SELECT @NEWSALARY=SALARY FROM INSERTED NEW;
SELECT @OLDDNO=DNO FROM DELETED OLD;
SELECT @NEWDNO=DNO FROM INSERTED NEW;

EXEC SP_AUDIT_EMPSALARY 'UPDATE',
@OLDSSN,@NEWSSN,@OLDSALARY,@NEWSALARY,@OLDDNO,@NEWDNO
END
```

91 %

Messages

Commands completed successfully.

Completion time: 2023-12-08T01:29:56.5383055-05:00

91 %

3 & 5) Letting each event happen to get each trigger created fired by executing these DML on EMPLOYEE

Displaying EMP TABLE and AUDIT_EMPSALARY

```

SELECT @NEWSN=SSN FROM INSERTED NEW;
SELECT @OLDSALARY=SALARY FROM DELETED OLD;
SELECT @NEWSALARY=SALARY FROM INSERTED NEW;
SELECT @OLDDNO=DNO FROM DELETED OLD;
SELECT @NEWDNO=DNO FROM INSERTED NEW;

EXEC SP_AUDIT_EMPSALARY 'UPDATE',
@OLDSSN,@NEWSN,@OLDSALARY,@NEWSALARY,@OLDDNO,@NEWDNO
END

-- 5) a. display the emp table and audit_emp_salary

SELECT * FROM EMPLOYEE
SELECT * FROM AUDIT_EMPSALARY
    
```

91 %

Results Messages

	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
1	John	B	Smith	123456789	1955-01-09	731 Fondren, Houston, TX	M	30000.00	987654321	5
2	Franklin	T	Wong	333445555	2045-12-08	638 Voss, Houston, TX	M	40000.00	888665555	5
3	Joyce	A	English	453453453	1962-07-31	5631 Rice, Houston, TX	F	25000.00	333445555	5
4	Ramesh	K	Narayan	666884444	1952-09-15	975 Fire Oak, Humble, TX	M	38000.00	333445555	5
5	James	E	Borg	888665555	2027-11-10	450 Stone, Houston, TX	M	55000.00		1
6	Jennifer	S	Wallace	987654321	2031-06-20	291 Berry, Bellaire, TX	F	43000.00	888665555	4
7	Ahmad	V	Jabbar	987987987	1959-03-29	980 Dallas, Houston, TX	M	25000.00	987654321	4
8	Alicia	J	Zelaya	999887777	1958-07-19	3321 Castle, SPrint, TX	F	25000.00	987654321	4

-- 5) a. display the emp table and audit_emp_salary

SELECT * FROM EMPLOYEE

SELECT * FROM AUDIT_EMPSALARY

%

Results Messages

DATEOFCHANGE	DML_TYPE	OLDSSN	NEWSN	OLDSALARY	NEWSALARY	OLDDNO	NEWDNO
--------------	----------	--------	-------	-----------	-----------	--------	--------

Executed the UPDATE query

```
COMPANY_EshaAttiq
Execute
EshaAttiq_Lab5_Cre...iq's Computer (67)*
EshaAttiq_Lab6_Cre...iq's Computer (55)*
SELECT @NEWSALARY=SALARY FROM INSERTED NEW;
SELECT @OLDDNO=DNO FROM DELETED OLD;
SELECT @NEWDDNO=DNO FROM INSERTED NEW;
EXEC SP_AUDIT_EMP_SALARY 'UPDATE',
@OLDSSN,@NEWSSN,@OLDSALARY,@NEWSALARY,@OLDDNO,@NEWDDNO
END

-- 5) a. display the emp table and audit_emp_salary

SELECT * FROM EMPLOYEE
SELECT * FROM AUDIT_EMP_SALARY

-- 5) b. executing the update query

UPDATE EMPLOYEE SET SALARY=99000 WHERE DNO=5;
```


Displaying table EMPLOYEE and AUDIT_EMP_SALARY

COMPANY_EshaAttig

Execute

EshaAttig_Lab5_Cre...iq's Computer (67))* EshaAttig_Lab6_Cre...iq's Computer (55))*

```
-- 5) b. executing the update query  
  
UPDATE EMPLOYEE SET SALARY=99000 WHERE DNO=5;  
  
-- 5) c. display table employee & audit_emp_salary  
  
SELECT * FROM EMPLOYEE  
  
SELECT * FROM AUDIT_EMP_SALARY
```

91 %

Results Messages

	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
1	John	B	Smith	123456789	1955-01-09	731 Fondren, Houston, TX	M	99000.00	987654321	5
2	Franklin	T	Wong	333445555	2045-12-08	638 Voss, Houston, TX	M	99000.00	888665555	5
3	Joyce	A	English	453453453	1962-07-31	5631 Rice, Houston, TX	F	99000.00	333445555	5
4	Ramesh	K	Narayan	666884444	1952-09-15	975 Fire Oak, Humble, TX	M	99000.00	333445555	5
5	James	E	Borg	888665555	2027-11-10	450 Stone, Houston, TX	M	55000.00		1
6	Jennifer	S	Wallace	987654321	2031-06-20	291 Beny, Bellaire, TX	F	43000.00	888665555	4
7	Ahmad	V	Jabbar	987987987	1959-03-29	980 Dallas, Houston, TX	M	25000.00	987654321	4
8	Alicia	J	Zelaya	999887777	1958-07-19	3321 Castle, SPrint, TX	F	25000.00	987654321	4

EshaAttig_Lab5_Cre...iq's Computer (67))* EshaAttig_Lab6_Cre...iq's Computer (55))*

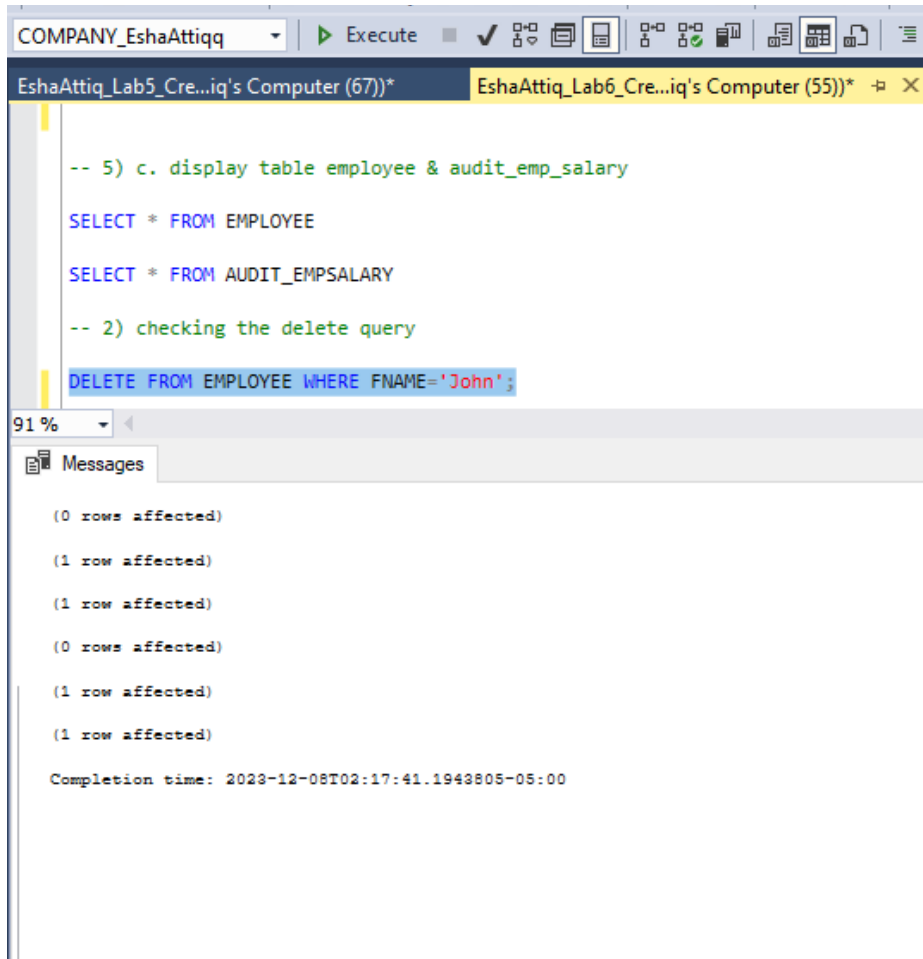
```
-- 5) c. display table employee & audit_emp_salary  
  
SELECT * FROM EMPLOYEE  
  
SELECT * FROM AUDIT_EMP_SALARY
```

91 %

Results Messages

	DATEOFCHANGE	DML_TYPE	OLDSSN	NEWSSN	OLDSALARY	NEWSALARY	OLDDNO	NEWDNO
1	2023-12-08	UPDATE	666884444	666884444	38000	99000	5	5
2	2023-12-08	UPDATE	453453453	453453453	25000	99000	5	5
3	2023-12-08	UPDATE	333445555	333445555	40000	99000	5	5
4	2023-12-08	UPDATE	123456789	123456789	30000	99000	5	5

Checking the DELETE query



The screenshot shows a SQL IDE window with a toolbar at the top containing icons for Execute, Save, and other functions. The main editor area contains the following SQL code:

```
-- 5) c. display table employee & audit_emp_salary  
  
SELECT * FROM EMPLOYEE  
  
SELECT * FROM AUDIT_EMPSALARY  
  
-- 2) checking the delete query  
DELETE FROM EMPLOYEE WHERE FNAME='John';
```

Below the editor, a status bar indicates "91 %". A "Messages" tab is active, displaying the following output:

```
(0 rows affected)  
  
(1 row affected)  
  
(1 row affected)  
  
(0 rows affected)  
  
(1 row affected)  
  
(1 row affected)  
  
Completion time: 2023-12-08T02:17:41.1943805-05:00
```

Then displaying the table EMPLOYEE and AUDIT_EMPSALARY

COMPANY_EshaAttigq Execute

EshaAttigq_Lab5_Cre...iq's Computer (67))* EshaAttigq_Lab6_Cre...iq's Computer (55))*

```
-- 5) c. display table employee & audit_emp_salary  
  
SELECT * FROM EMPLOYEE  
  
SELECT * FROM AUDIT_EMPSALARY  
  
-- 2) checking the delete query  
  
DELETE FROM EMPLOYEE WHERE FNAME='John';  
  
SELECT * FROM EMPLOYEE
```

91 %

Results Messages

	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
1	Franklin	T	Wong	333445555	2045-12-08	638 Voss, Houston, TX	M	99000.00	888665555	5
2	Joyce	A	English	453453453	1962-07-31	5631 Rice, Houston, TX	F	99000.00	333445555	5
3	Ramesh	K	Narayan	666884444	1952-09-15	975 Fire Oak, Humble, TX	M	99000.00	333445555	5
4	James	E	Borg	888665555	2027-11-10	450 Stone, Houston, TX	M	55000.00		1
5	Jennifer	S	Wallace	987654321	2031-06-20	291 Berry, Bellaire, TX	F	43000.00	888665555	4
6	Ahmad	V	Jabbar	987987987	1959-03-29	980 Dallas, Houston, TX	M	25000.00	987654321	4
7	Alicia	J	Zelaya	999887777	1958-07-19	3321 Castle, SPrint, TX	F	25000.00	987654321	4

COMPANY_EshaAttigq Execute

EshaAttigq_Lab5_Cre...iq's Computer (67))* EshaAttigq_Lab6_Cre...iq's Computer (55))*

```
-- 5) c. display table employee & audit_emp_salary  
  
SELECT * FROM EMPLOYEE  
  
SELECT * FROM AUDIT_EMPSALARY  
  
-- 2) checking the delete query  
  
DELETE FROM EMPLOYEE WHERE FNAME='John';  
  
SELECT * FROM EMPLOYEE  
  
SELECT * FROM AUDIT_EMPSALARY
```

91 %

Results Messages

	DATEOFCHANGE	DML_TYPE	OLDSSN	NEWSSN	OLDSALARY	NEWSALARY	OLDDNO	NEWDNO
1	2023-12-08	UPDATE	666884444	666884444	38000	99000	5	5
2	2023-12-08	UPDATE	453453453	453453453	25000	99000	5	5
3	2023-12-08	UPDATE	333445555	333445555	40000	99000	5	5
4	2023-12-08	UPDATE	123456789	123456789	30000	99000	5	5
5	2023-12-08	UPDATE	123456789	123456789	99000	NULL	5	NULL

INSERTING my own data into EMPLOYEE

COMPANY_EshaAttiiq Execute

EshaAttiiq_Lab5_Cre...iq's Computer (67)* EshaAttiiq_Lab6_Cre...iq's Computer (55)*

```
SELECT * FROM AUDIT_EMPLOYEE
```

-- INSERTING MY OWN DATA

```
INSERT INTO EMPLOYEE VALUES ('ESHA','A','ATTIQ','123456789','07-Jul-55','731 Fondern, Houston, TX','F',30000,'987654321',5);
```

```
SELECT * FROM EMPLOYEE
```

```
SELECT * FROM AUDIT_EMPLOYEE
```

91 %

Messages

(1 row affected)

(1 row affected)

Completion time: 2023-12-08T02:21:52.4741593-05:00

Then displaying tables EMPLOYEE and AUDIT_EMPLOYEE

-- INSERTING MY OWN DATA

```
INSERT INTO EMPLOYEE VALUES ('ESHA','A','ATTIQ','123456789','07-Jul-55','731 Fondern, Houston, TX','F',30000,'987654321',5);
```

```
SELECT * FROM EMPLOYEE
```

```
SELECT * FROM AUDIT_EMPLOYEE
```

91 %

Results Messages

	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
1	ESHA	A	ATTIQ	123456789	1955-07-07	731 Fondern, Houston, TX	F	30000.00	987654321	5
2	Franklin	T	Wong	333445555	2045-12-08	638 Voss, Houston, TX	M	99000.00	888665555	5
3	Joyce	A	English	453453453	1962-07-31	5631 Rice, Houston, TX	F	99000.00	333445555	5
4	Ramesh	K	Narayan	666884444	1952-09-15	975 Fire Oak, Humble, TX	M	99000.00	333445555	5
5	James	E	Borg	888665555	2027-11-10	450 Stone, Houston, TX	M	55000.00		1
6	Jennifer	S	Wallace	987654321	2031-06-20	291 Berry, Bellaire, TX	F	43000.00	888665555	4
7	Ahmad	V	Jabbar	987987987	1959-03-29	980 Dallas, Houston, TX	M	25000.00	987654321	4
8	Alicia	J	Zelaya	999887777	1958-07-19	3321 Castle, Sprint, TX	F	25000.00	987654321	4

-- INSERTING MY OWN DATA

```
INSERT INTO EMPLOYEE VALUES ('ESHA','A','ATTIQ','123456789','07-Jul-55','731 Fondern, Houston, TX','F',30000,'987654321',5);
```

```
SELECT * FROM EMPLOYEE
```

```
SELECT * FROM AUDIT_EMPLOYEE
```

91 %

Results Messages

	DATEOFCHANGE	DML_TYPE	OLDSSN	NEWSSN	OLDSALARY	NEWSALARY	OLDNO	NEWNO
1	2023-12-08	UPDATE	666884444	666884444	38000	99000	5	5
2	2023-12-08	UPDATE	453453453	453453453	25000	99000	5	5
3	2023-12-08	UPDATE	333445555	333445555	40000	99000	5	5
4	2023-12-08	UPDATE	123456789	123456789	30000	99000	5	5
5	2023-12-08	UPDATE	123456789	123456789	99000	NULL	5	NULL
6	2023-12-08	UPDATE	NULL	123456789	NULL	30000	NULL	5