

Esha Attiq

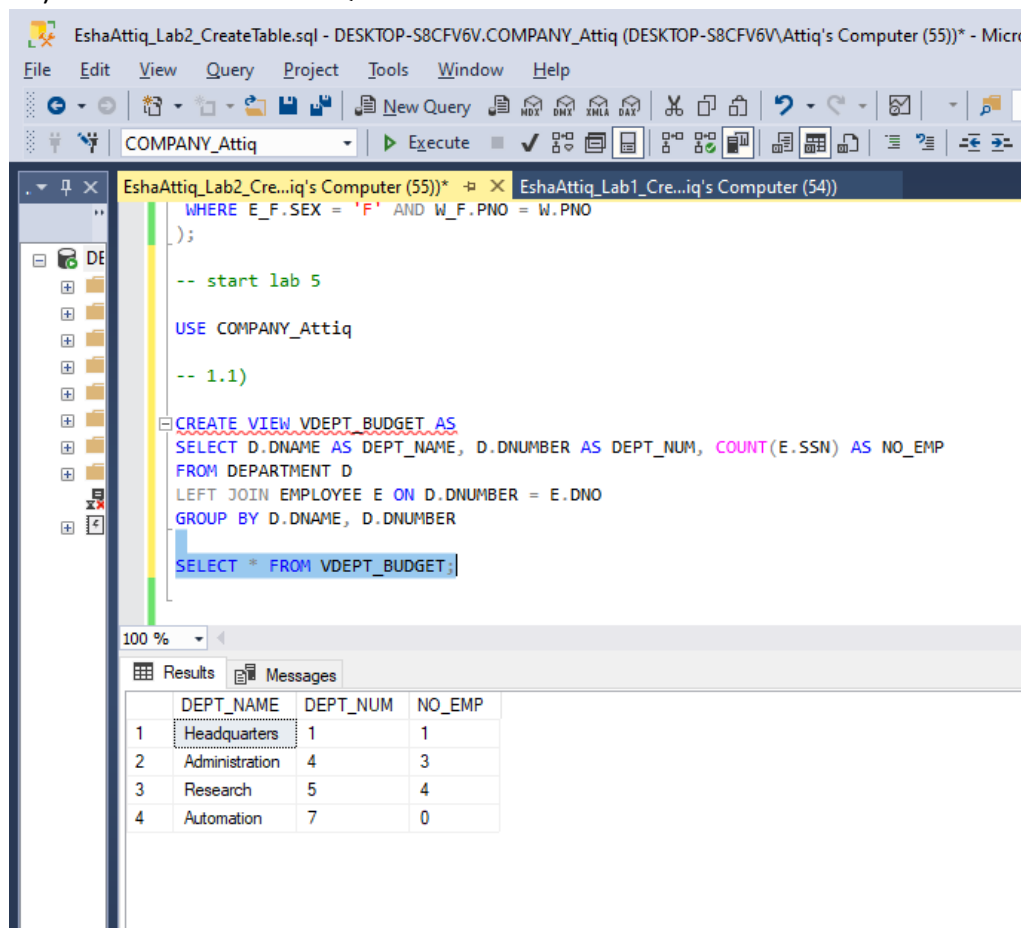
CIS 430

LAB 5: 2 parts; View and Stored Procedure using Cursor

NOTE: I had technical difficulties with my laptop, I had actually just returned it to Best Buy on Friday, and my Lab wasn't be able to be done on time. However, I'm using a laptop at home and had to restart so It'll just look late. [I emailed the TA and tried to email the professor, thank you!]

PART 1

1.1) View creation from SQL



The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The main window displays the SQL script for creating a view named VDEPT_BUDGET. The script includes comments for starting Lab 5 and section 1.1, followed by the SQL code to create the view. The view is created from the DEPARTMENT table, joined with the EMPLOYEE table, and grouped by department name and number. The results of the query are displayed in a table at the bottom of the window.

```
WHERE E_F.SEX = 'F' AND W_F.PNO = W.PNO
);

-- start lab 5

USE COMPANY_Attig

-- 1.1)

CREATE VIEW VDEPT_BUDGET AS
SELECT D.DNAME AS DEPT_NAME, D.DNUMBER AS DEPT_NUM, COUNT(E.SSN) AS NO_EMP
FROM DEPARTMENT D
LEFT JOIN EMPLOYEE E ON D.DNUMBER = E.DNO
GROUP BY D.DNAME, D.DNUMBER

SELECT * FROM VDEPT_BUDGET;
```

	DEPT_NAME	DEPT_NUM	NO_EMP
1	Headquarters	1	1
2	Administration	4	3
3	Research	5	4
4	Automation	7	0

1.2) Table creation from the same SQL as in 1.1

The screenshot shows a SQL Server Enterprise Manager window titled "EshaAttig_Lab2_CreateTable.sql - DESKTOP-S8CFV6V.COMPANY_Attig (DESKTOP-S8CFV6V\Attig's Computer (55))*". The window displays a SQL script with the following content:

```
-- 1.2)
CREATE TABLE DEPT_BUDGET (
    DEPT_NAME CHAR(30),
    DEPT_NUM INT,
    NO_EMP INT
);
INSERT INTO DEPT_BUDGET (DEPT_NAME, DEPT_NUM, NO_EMP)
SELECT D.DNAME AS DEPT_NAME, D.DNUMBER AS DEPT_NUM, COUNT(E.SSN) AS NO_EMP
FROM DEPARTMENT D
LEFT JOIN EMPLOYEE E ON D.DNUMBER = E.DNO
GROUP BY D.DNAME, D.DNUMBER;
SELECT * FROM DEPT_BUDGET
```

The script is executed, and the results are displayed in a table with the following data:

	DEPT_NAME	DEPT_NUM	NO_EMP
1	Headquarters	1	1
2	Administration	4	3
3	Research	5	4
4	Automation	7	0

2.

2.1) adding myself and another employee, then showing the View VDEPT_BUDGET content again.

2.2) I also showed the Table DEPT_BUDGET. Notice how the Table wasn't affected but the View is.

[screenshot on next page]

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the file is 'EshaAttig_Lab2_CreateTable.sql' and the server is 'DESKTOP-S8CFV6V.COMPANY_Attig (DESKTOP-S8CFV6V\Attig's Computer (55))*'. 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 'Query' tab is active, showing the following SQL script:

```
SELECT * FROM DEPT_BUDGET

-- 2.1)

INSERT INTO EMPLOYEE VALUES ('Esha', '', 'Attig', '77777777', '6-Mar-02', '1010 Cleveland, CSU, OH', 'F', 40000, 888665555,5);
INSERT INTO EMPLOYEE VALUES ('Lee', '', 'Seoho', '11111111', '1-Mar-01', '1010 Seoul, Korea', 'M', 80000, 987654321,4);

SELECT * FROM VDEPT_BUDGET
-- 2.2)
SELECT * FROM DEPT_BUDGET
```

The 'Results' pane at the bottom displays two tables. The first table is the result of the first query (SELECT * FROM DEPT_BUDGET), and the second table is the result of the third query (SELECT * FROM VDEPT_BUDGET). Both tables have the same structure: DEPT_NAME, DEPT_NUM, and NO_EMP.

	DEPT_NAME	DEPT_NUM	NO_EMP
1	Headquarters	1	1
2	Administration	4	4
3	Research	5	5
4	Automation	7	0

	DEPT_NAME	DEPT_NUM	NO_EMP
1	Headquarters	1	1
2	Administration	4	3
3	Research	5	4
4	Automation	7	0

3) differences in changing the scheme of existing view and table. Part 1) and 2) is shown.

The screenshot shows the SQL Developer interface with a query window titled 'EshaAttig_Lab2_CreateTable.sql'. The query is as follows:

```
-- 3)
-- for view
DROP VIEW IF EXISTS VDEPT_BUDGET;

CREATE VIEW VDEPT_BUDGET AS
SELECT
    D.DNAME AS DEPT_NAME,
    D.DNUMBER AS DEPT_NUM,
    COUNT(E.SSN) AS NO_EMP,
    SUM(E.SALARY) AS SUM_SALARY,
    AVG(E.SALARY) AS AVE_SALARY
FROM DEPARTMENT D
LEFT JOIN EMPLOYEE E ON D.DNUMBER = E.DNO
GROUP BY D.DNAME, D.DNUMBER;

SELECT * FROM VDEPT_BUDGET
```

The results of the query are displayed in a table with the following columns: DEPT_NAME, DEPT_NUM, NO_EMP, SUM_SALARY, and AVE_SALARY. The data is as follows:

	DEPT_NAME	DEPT_NUM	NO_EMP	SUM_SALARY	AVE_SALARY
1	Headquarters	1	1	55000.00	55000.000000
2	Administration	4	4	173000.00	43250.000000
3	Research	5	5	173000.00	34600.000000
4	Automation	7	0	NULL	NULL

For the View:

The screenshot shows a SQL Server Enterprise Manager window with a query titled 'EshaAttig_Lab2_CreateTable.sql'. The query is executed against a database named 'COMPANY_Attig'. The query window displays the following SQL code:

```
-- for table
ALTER TABLE DEPT_BUDGET
ADD SUM_SALARY DECIMAL(10, 2),
    AVE_SALARY DECIMAL(10, 2);

UPDATE DEPT_BUDGET
SET SUM_SALARY = (
    SELECT SUM(E.SALARY)
    FROM EMPLOYEE E
    WHERE E.DNO = DEPT_BUDGET.DEPT_NUM
),
    AVE_SALARY = (
    SELECT AVG(E.SALARY)
    FROM EMPLOYEE E
    WHERE E.DNO = DEPT_BUDGET.DEPT_NUM
);

SELECT * FROM DEPT_BUDGET;
```

The results grid shows the following data:

	DEPT_NAME	DEPT_NUM	NO_EMP	SUM_SALARY	AVE_SALARY
1	Headquarters	1	1	55000.00	55000.00
2	Administration	4	3	173000.00	43250.00
3	Research	5	4	173000.00	34600.00
4	Automation	7	0	NULL	NULL

For the Table:

#2 Stored Procedure using a Cursor

This 2nd part was giving me errors, some glitches. Although my code is correct. I will provide outputs either way to gain as much points as I can, since some parts work. Thank you.

```
SELECT * FROM DEPT_BUDGET;

-- part 2

CREATE PROCEDURE SP_REPORT_NEW_BUDGET()
BEGIN
    DECLARE done INT DEFAULT FALSE;
    DECLARE dept_name CHAR(30);
    DECLARE dept_number INT;
    DECLARE no_emp INT;
    DECLARE sum_salary INT;
    DECLARE ave_salary INT;

    DROP TABLE IF EXISTS NEW_DEPT_BUDGET;

    CREATE TABLE NEW_DEPT_BUDGET (
        DEPT_NO INT,
        DEPT_NAME CHAR(30),
        COUNT_EMP INT,
        NEW_SUM_SALARY INT,
        NEW_AVE_SALARY INT
    );

    DECLARE CUR CURSOR FOR SELECT DEPT_NAME, DEPT_NUM, NO_EMP, SUM_SALARY, AVE_SALARY FROM VDEPT_BUDGET;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;

100 %
```

Messages

Commands completed successfully.

Completion time: 2023-11-25T21:13:18.3486423-05:00

100 %

COMPANY_Attiq

Execute

```
IF DEPT_NUM = 1 THEN
    SET SUM_SALARY = SUM_SALARY * 1.1;
ELSEIF DEPT_NUM = 4 THEN
    SET SUM_SALARY = SUM_SALARY * 1.2;
ELSEIF DEPT_NUM = 5 THEN
    SET SUM_SALARY = SUM_SALARY * 1.3;
ELSEIF DEPT_NUM = 7 THEN
    SET SUM_SALARY = SUM_SALARY * 1.4;
END IF;

INSERT INTO NEW_DEPT_BUDGET (DEPT_NO, DEPT_NAME, COUNT_EMP, NEW_SUM_SALARY, NEW_AVE_SALARY)
VALUES (DEPT_NUM, DEPT_NAME, NO_EMP, SUM_SALARY, AVE_SALARY);
END LOOP;

CLOSE CUR;

100 %
```

Messages

Msg 102, Level 15, State 1, Procedure SP_REPORT_NEW_BUDGET, Line 3 [Batch Start Line 393]
Incorrect syntax near ')'.
Msg 155, Level 15, State 2, Procedure SP_REPORT_NEW_BUDGET, Line 5 [Batch Start Line 393]
'INT' is not a recognized CURSOR option.
Msg 155, Level 15, State 2, Procedure SP_REPORT_NEW_BUDGET, Line 6 [Batch Start Line 393]
'CHAR' is not a recognized CURSOR option.
Msg 155, Level 15, State 2, Procedure SP_REPORT_NEW_BUDGET, Line 7 [Batch Start Line 393]
'INT' is not a recognized CURSOR option.
Msg 155, Level 15, State 2, Procedure SP_REPORT_NEW_BUDGET, Line 8 [Batch Start Line 393]
'INT' is not a recognized CURSOR option.
Msg 155, Level 15, State 2, Procedure SP_REPORT_NEW_BUDGET, Line 9 [Batch Start Line 393]
'INT' is not a recognized CURSOR option.
Msg 155, Level 15, State 2, Procedure SP_REPORT_NEW_BUDGET, Line 10 [Batch Start Line 393]
'INT' is not a recognized CURSOR option.

100 %

COMPANY_Attiq

Execute

EshaAttq_Lab2_Cre...iq's Computer (54))*

```
VALUES (DEPT_NUM, DEPT_NAME, NO_EMP, SUM_SALARY,
END LOOP;
CLOSE CUR;
SELECT COUNT(*) INTO DONE FROM VDEPT_BUDGET;
IF DONE > 0 THEN
SELECT * FROM VDEPT_BUDGET;
ELSE
SELECT 'VDEPT_BUDGET IS EMPTY.';
END IF;
SELECT * FROM NEW_DEPT_BUDGET;
END;
```

100 %

Messages

Msg 156, Level 15, State 1, Line 448
Incorrect syntax near the keyword 'THEN'.
Msg 156, Level 15, State 1, Line 450
Incorrect syntax near the keyword 'ELSE'.
Msg 156, Level 15, State 1, Line 452
Incorrect syntax near the keyword 'IF'.
Completion time: 2023-11-25T21:19:57.6360845-05:00

COMPANY_Attiq

Execute

EshaAttq_Lab2_Cre...iq's Computer (54))*

```
CLOSE CUR;
SELECT COUNT(*) INTO DONE FROM VDEPT_BUDGET;
IF DONE > 0 THEN
SELECT * FROM VDEPT_BUDGET;
ELSE
SELECT 'VDEPT_BUDGET IS EMPTY.';
END IF;
SELECT * FROM NEW_DEPT_BUDGET;
END;
```

100 %

Results

DEPT_NO	DEPT_NAME	COUNT_EMP	NEW_SUM_SALARY	NEW_AVE_SALARY
---------	-----------	-----------	----------------	----------------

In case you need it: here is

the code for Part 2, some parts, again, work. But I have provided the outputs I am getting. (:
-- part 2

```
CREATE PROCEDURE SP_REPORT_NEW_BUDGET()
```

```
BEGIN
```

```
    DECLARE done INT DEFAULT FALSE;
```

```
    DECLARE dept_name CHAR(30);
```

```
    DECLARE dept_number INT;
```

```
    DECLARE no_emp INT;
```

```
    DECLARE sum_salary INT;
```

```
    DECLARE ave_salary INT;
```

```
DROP TABLE IF EXISTS NEW_DEPT_BUDGET;
```

```
CREATE TABLE NEW_DEPT_BUDGET (
```

```
    DEPT_NO INT,
```

```
    DEPT_NAME CHAR(30),
```

```
    COUNT_EMP INT,
```

```
    NEW_SUM_SALARY INT,
```

```
    NEW_AVE_SALARY INT
```

```
);
```

```
    DECLARE CUR CURSOR FOR SELECT DEPT_NAME, DEPT_NUM, NO_EMP, SUM_SALARY,  
    AVE_SALARY FROM VDEPT_BUDGET;
```

```
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
```


OPEN CUR;

read_loop: LOOP

FETCH CUR INTO DEPT_NAME, DEPT_NUM, NO_EMP, SUM_SALARY, AVE_SALARY;

IF DONE THEN

LEAVE read_loop;

END IF;

IF DEPT_NUM = 1 THEN

SET SUM_SALARY = SUM_SALARY * 1.1;

ELSEIF DEPT_NUM = 4 THEN

SET SUM_SALARY = SUM_SALARY * 1.2;

ELSEIF DEPT_NUM = 5 THEN

SET SUM_SALARY = SUM_SALARY * 1.3;

ELSEIF DEPT_NUM = 7 THEN

SET SUM_SALARY = SUM_SALARY * 1.4;

END IF;

INSERT INTO NEW_DEPT_BUDGET (DEPT_NO, DEPT_NAME, COUNT_EMP,
NEW_SUM_SALARY, NEW_AVE_SALARY)

VALUES (DEPT_NUM, DEPT_NAME, NO_EMP, SUM_SALARY, AVE_SALARY);

END LOOP;

CLOSE CUR;

SELECT COUNT(*) INTO DONE FROM VDEPT_BUDGET;

```
IF DONE > 0 THEN
```

```
    SELECT * FROM VDEPT_BUDGET;
```

```
ELSE
```

```
    SELECT 'VDEPT_BUDGET IS EMPTY.';
```

```
END IF;
```

```
SELECT * FROM NEW_DEPT_BUDGET;
```

```
END;
```