Benjamin Tisinger IST659
Homework 7
11/27/23

Questions

Answer these questions using the problem set submission template For any screen shots provided, please follow the guidelines for submitting a screen shot.

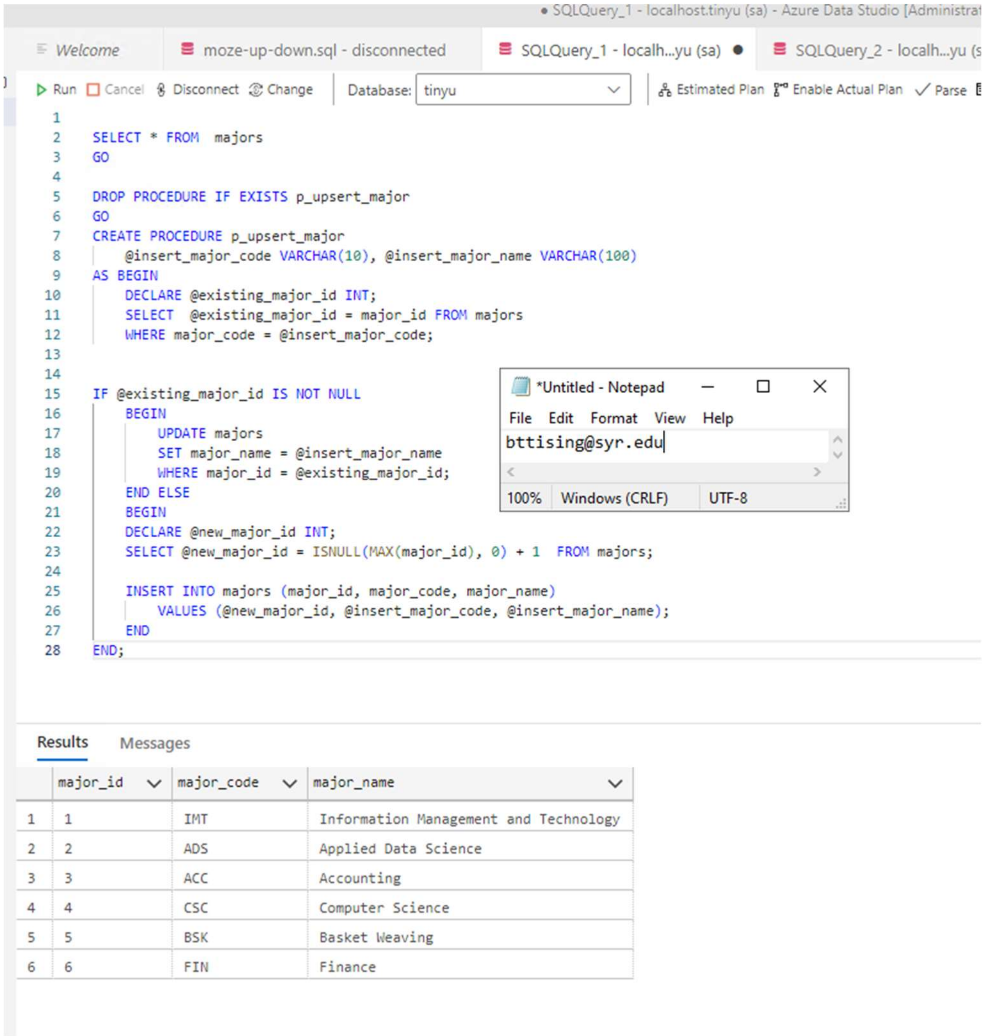
Write the following as SQL programs. For each, include the SQL as a screen shot with the output of the query.

1. In the **TinyU** database:
    a. Write an SQL Stored procedure called **p_upsert_major**, which, given a major_code (business key) and a major_name, does an Upsert, which is the following:
        i. Checks if the major_code exists in the table already.
        ii. If yes, updates the table and makes the major_name match the new major name.
        iii. If no, inserts the new major_name and major_code into the table. HINT: major_id is not a surrogate key, so you will need to determine the next ID yourself in code!
    b. Test your stored procedure by executing it to make these changes:
        i. Change : CSC—Computer Sciences to CSC—Computer Science
        ii. Add: FIN—Finance

Make sure your screen shot captures all up/down code in 1.a AND another screen shot captures 1.b—the output of your code execution—to show that it works. SELECT the table before and after!p

Resources Used to Assist:

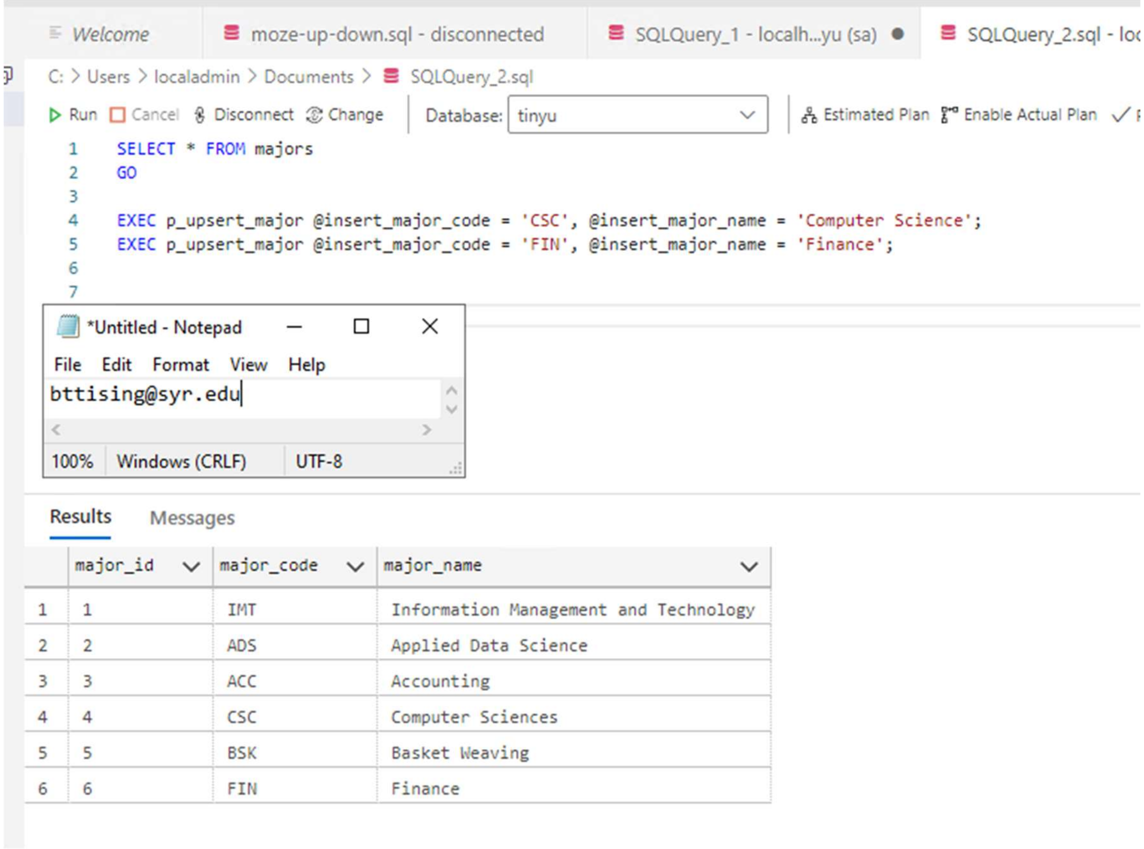
- https://www.w3schools.com/sql/sql_stored_procedures.asp
- https://stackoverflow.com/questions/459457/what-is-a-stored-procedure
- https://www.tutorialspoint.com/sql/sql-stored-procedures.htm
- https://stackoverflow.com/questions/51306691/select-maxid-in-an-empty-table-returns-null-instead-of-0
- https://www.youtube.com/watch?v=NrBJmtD0kEw
- https://chat.openai.com/ **- For Select Verification and Tweaking of Code**



```

SELECT * FROM  majors
GO

DROP PROCEDURE IF EXISTS p_upsert_major
GO
CREATE PROCEDURE p_upsert_major
    @insert_major_code VARCHAR(10), @insert_major_name VARCHAR(100)
AS BEGIN
    DECLARE @existing_major_id INT;
    SELECT  @existing_major_id = major_id FROM majors
    WHERE major_code = @insert_major_code;


IF @existing_major_id IS NOT NULL
    BEGIN
        UPDATE majors
        SET major_name = @insert_major_name
        WHERE major_id = @existing_major_id;
    END ELSE
    BEGIN
    DECLARE @new_major_id INT;
    SELECT @new_major_id = ISNULL(MAX(major_id), 0) + 1  FROM majors;

    INSERT INTO majors (major_id, major_code, major_name)
        VALUES (@new_major_id, @insert_major_code, @insert_major_name);
    END
END;
```

bttising@syr.edu

	major_id	major_code	major_name
1	1	IMT	Information Management and Technology
2	2	ADS	Applied Data Science
3	3	ACC	Accounting
4	4	CSC	Computer Science
5	5	BSK	Basket Weaving
6	6	FIN	Finance



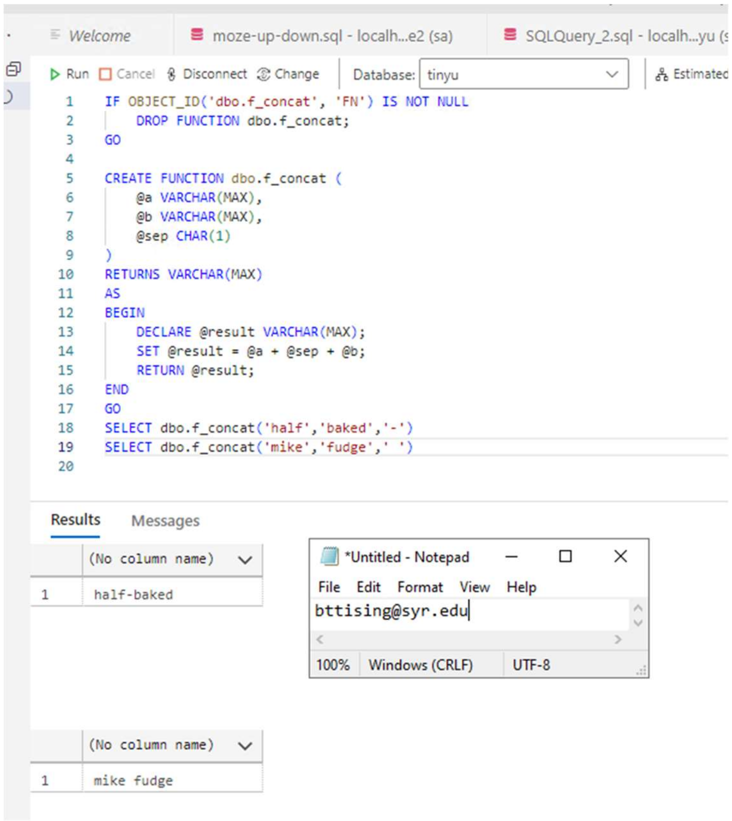
```
SELECT * FROM majors
GO

EXEC p_upsert_major @insert_major_code = 'CSC', @insert_major_name = 'Computer Science';
EXEC p_upsert_major @insert_major_code = 'FIN', @insert_major_name = 'Finance';
```

bttising@syr.edu

	major_id	major_code	major_name
1	1	IMT	Information Management and Technology
2	2	ADS	Applied Data Science
3	3	ACC	Accounting
4	4	CSC	Computer Sciences
5	5	BSK	Basket Weaving
6	6	FIN	Finance

2. In the **TinyU** database:
    a. Write a user-defined function called **f_concat** that combines the any two varchars @a and @b together with a one-character @sep in between.
    For example:

```
select dbo.f_concat('half','baked','-') -- 'half-baked'
select dbo.f_concat('mike', 'fudge', ' ') -- 'mike fudge'
```



```
IF OBJECT_ID('dbo.f_concat', 'FN') IS NOT NULL
    DROP FUNCTION dbo.f_concat;
GO

CREATE FUNCTION dbo.f_concat (
    @a VARCHAR(MAX),
    @b VARCHAR(MAX),
    @sep CHAR(1)
)
RETURNS VARCHAR(MAX)
AS
BEGIN
    DECLARE @result VARCHAR(MAX);
    SET @result = @a + @sep + @b;
    RETURN @result;
END
GO
SELECT dbo.f_concat('half','baked','-')
SELECT dbo.f_concat('mike','fudge',' ')
```

	(No column name)
1	half-baked

bttising@syr.edu

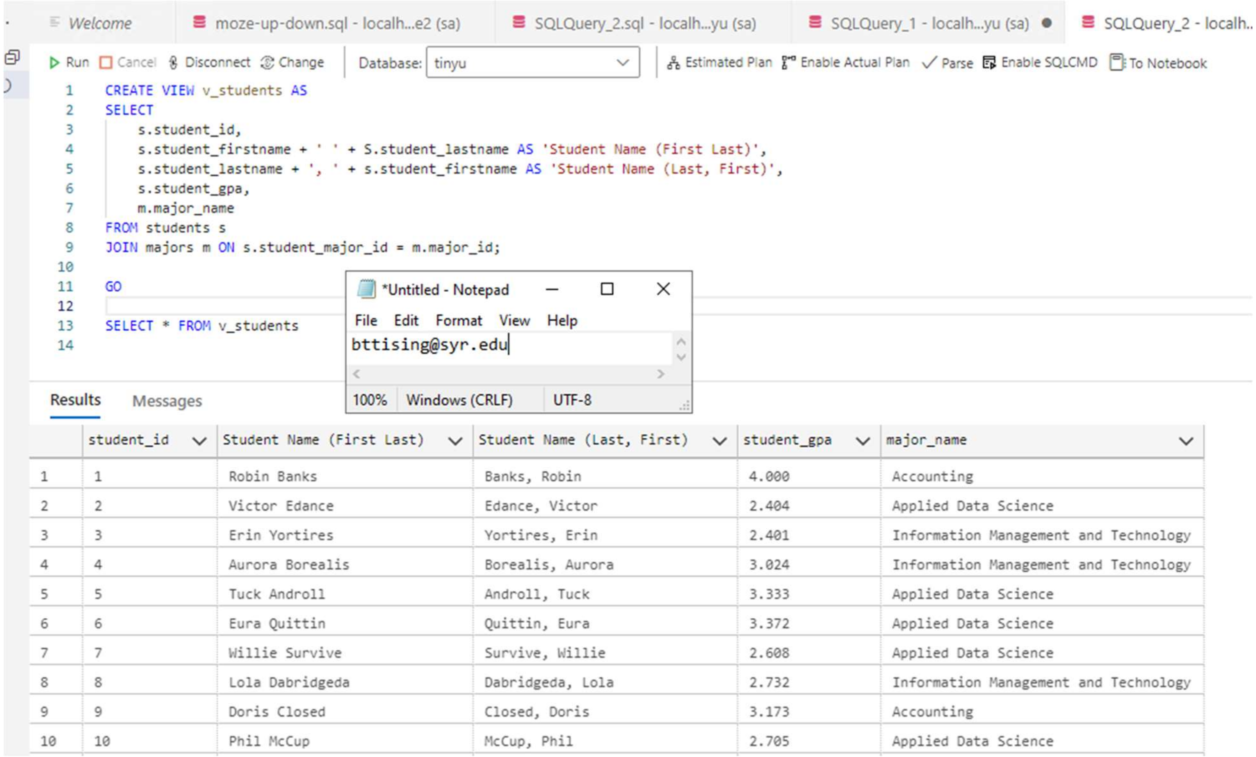
	(No column name)
1	mike fudge

Resources Used to Assist:

- https://learn.microsoft.com/en-us/sql/relational-databases/user-defined-functions/create-user-defined-functions-database-engine?view=sql-server-ver16
- https://hasura.io/learn/database/microsoft-sql-server/create-function/
- https://chat.openai.com/ **- For Select Verification of Code and adding IF OBJECT_ID Statement**

b. Now create a view called **v_students** that displays the student_id, student name (first last), student name (last, first), GPA, and name of major. You should call the function you created in 2.a. After you create the view, execute it with a SELECT statement.

Make sure your screen shot captures all up/down code in 2.a AND another screen shot captures 2.b, along with the output of the SELECT statement on the view (first few rows is fine).



Database: tinyu

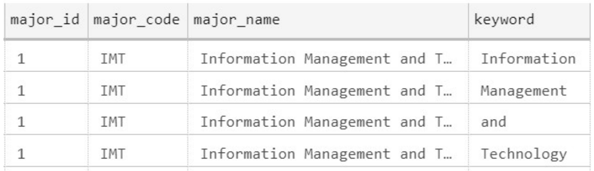
```
CREATE VIEW v_students AS
SELECT
    s.student_id,
    s.student_firstname + ' ' + S.student_lastname AS 'Student Name (First Last)',
    s.student_lastname + ', ' + s.student_firstname AS 'Student Name (Last, First)',
    s.student_gpa,
    m.major_name
FROM students s
JOIN majors m ON s.student_major_id = m.major_id;

GO

SELECT * FROM v_students
```

bttising@syr.edu

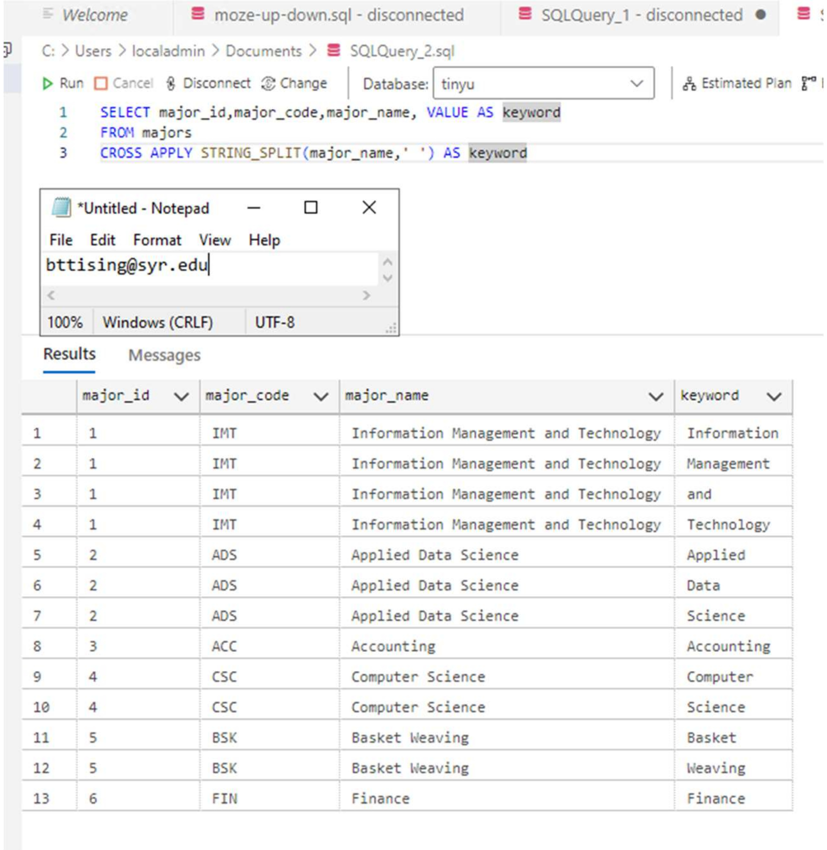
	student_id	Student Name (First Last)	Student Name (Last, First)	student_gpa	major_name
1	1	Robin Banks	Banks, Robin	4.000	Accounting
2	2	Victor Edance	Edance, Victor	2.404	Applied Data Science
3	3	Erin Yortires	Yortires, Erin	2.401	Information Management and Technology
4	4	Aurora Borealis	Borealis, Aurora	3.024	Information Management and Technology
5	5	Tuck Androll	Androll, Tuck	3.333	Applied Data Science
6	6	Eura Quittin	Quittin, Eura	3.372	Applied Data Science
7	7	Willie Survive	Survive, Willie	2.608	Applied Data Science
8	8	Lola Dabridgeda	Dabridgeda, Lola	2.732	Information Management and Technology
9	9	Doris Closed	Closed, Doris	3.173	Accounting
10	10	Phil McCup	McCup, Phil	2.705	Applied Data Science

3. In the **TinyU** database:

    a. Write a query on the **majors** table so that the major_name is broken up into keywords, one per row. HINT: You must use string_split() with cross-apply.



major_id	major_code	major_name	keyword
1	IMT	Information Management and T…	Information
1	IMT	Information Management and T…	Management
1	IMT	Information Management and T…	and
1	IMT	Information Management and T…	Technology



Welcome
moze-up-down.sql - disconnected
SQLQuery_1 - disconnected
C: > Users > localadmin > Documents > SQLQuery_2.sql
Run
Cancel
Disconnect
Change
Database: tinyu
Estimated Plan
*Untitled - Notepad
File Edit Format View Help
100% Windows (CRLF) UTF-8
Results
Messages

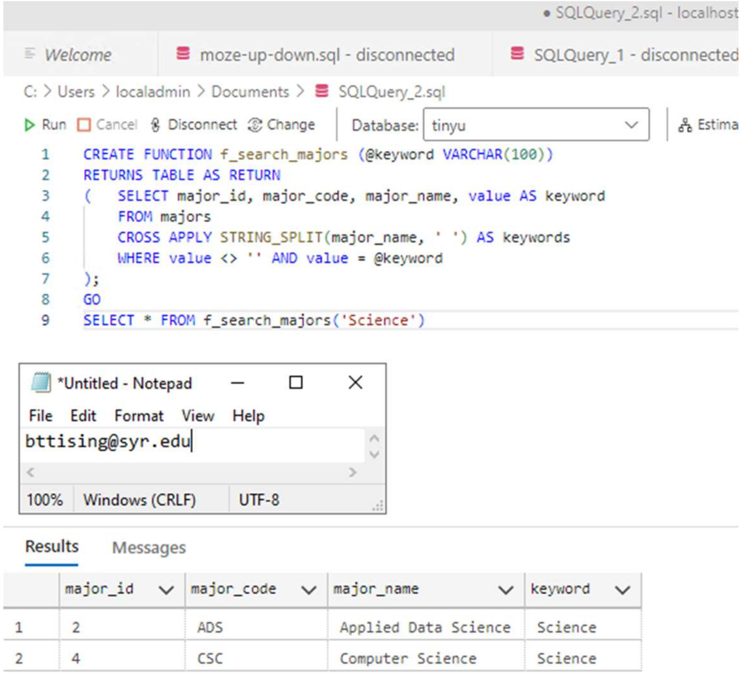
```
SELECT major_id,major_code,major_name, VALUE AS keyword
FROM majors
CROSS APPLY STRING_SPLIT(major_name,' ') AS keyword
```

bttising@syr.edu

	major_id	major_code	major_name	keyword
1	1	IMT	Information Management and Technology	Information
2	1	IMT	Information Management and Technology	Management
3	1	IMT	Information Management and Technology	and
4	1	IMT	Information Management and Technology	Technology
5	2	ADS	Applied Data Science	Applied
6	2	ADS	Applied Data Science	Data
7	2	ADS	Applied Data Science	Science
8	3	ACC	Accounting	Accounting
9	4	CSC	Computer Science	Computer
10	4	CSC	Computer Science	Science
11	5	BSK	Basket Weaving	Basket
12	5	BSK	Basket Weaving	Weaving
13	6	FIN	Finance	Finance

    b. Then use the query in 3.a to create a table-valued function **f_search_majors** that allows you to search the majors by keyword. Demonstrate calling the TVF by querying all majors with the “Science” keyword.

Your screen shot should include the query in 3.a Another screen shot should show the TVF in 3.b and the sample output from the SELECT statement calling the TVF.


• SQLQuery_2.sql - localhost
Welcome
moze-up-down.sql - disconnected
SQLQuery_1 - disconnected
C: > Users > localadmin > Documents > SQLQuery_2.sql
Run
Cancel
Disconnect
Change
Estima
*Untitled - Notepad
File
Edit
Format
View
Help
100%
Windows (CRLF)
UTF-8
Results
Messages

Database: tinyu

```sql
CREATE FUNCTION f_search_majors (@keyword VARCHAR(100))
RETURNS TABLE AS RETURN
(   SELECT major_id, major_code, major_name, value AS keyword
    FROM majors
    CROSS APPLY STRING_SPLIT(major_name, ' ') AS keywords
    WHERE value <> '' AND value = @keyword
);
GO
SELECT * FROM f_search_majors('Science')
```

bttising@syr.edu

	major_id	major_code	major_name	keyword
1	2	ADS	Applied Data Science	Science
2	4	CSC	Computer Science	Science

Resources Used to Assist:

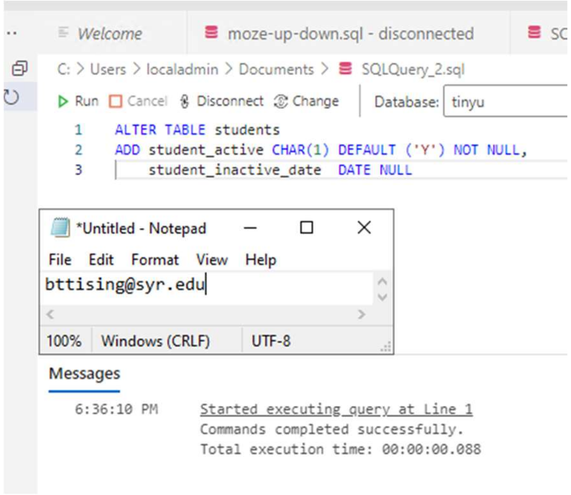
- https://www.codeproject.com/Articles/167399/Using-Table-Valued-Functions-in-SQL-Server
- https://www.databasestar.com/sql-cross-apply/

4. In the **TinyU** database:

   a. Alter the **students** table and add the following columns:

      i. student_active char(1) default (‘Y’) not null

      ii. student_inactive_date date null

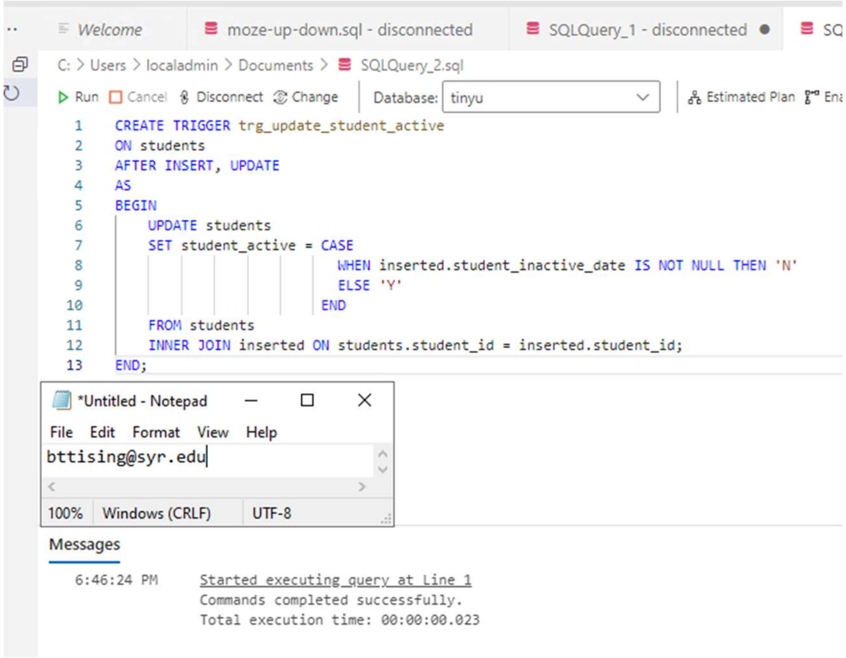

Welcome
moze-up-down.sql - disconnected
SC
C: > Users > localadmin > Documents > SQLQuery_2.sql
Run
Cancel
Disconnect
Change
Database: tinyu
*Untitled - Notepad
File Edit Format View Help
100% Windows (CRLF) UTF-8
Messages

```
ALTER TABLE students
ADD student_active CHAR(1) DEFAULT ('Y') NOT NULL,
    student_inactive_date  DATE NULL
```

bttising@syr.edu

```
6:36:10 PM      Started executing query at Line 1
                Commands completed successfully.
                Total execution time: 00:00:00.088
```

   b. Create a trigger on the **students** table: when there is an student_inactive_date set, set student_active to ‘N’, and whenever there is not a student_inactive_date, then student_active is set to ‘Y’.


Welcome
moze-up-down.sql - disconnected
SQLQuery_1 - disconnected
SQ
C: > Users > localadmin > Documents > SQLQuery_2.sql
Run
Cancel
Disconnect
Change
Database: tinyu
Estimated Plan
Ena
*Untitled - Notepad
File Edit Format View Help
100% Windows (CRLF) UTF-8
Messages

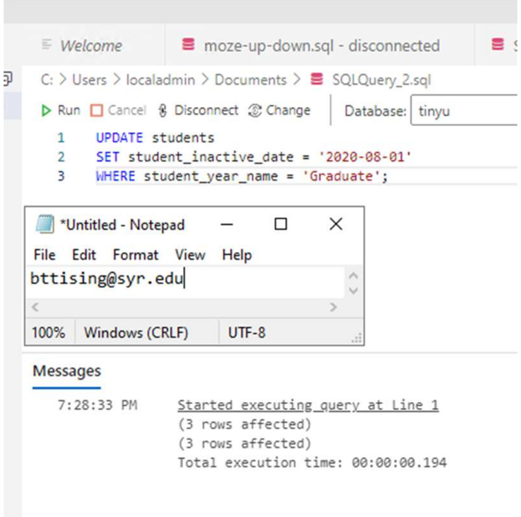
```
CREATE TRIGGER trg_update_student_active
ON students
AFTER INSERT, UPDATE
AS
BEGIN
    UPDATE students
    SET student_active = CASE
                            WHEN inserted.student_inactive_date IS NOT NULL THEN 'N'
                            ELSE 'Y'
                          END
    FROM students
    INNER JOIN inserted ON students.student_id = inserted.student_id;
END;
```

bttising@syr.edu

```
6:46:24 PM      Started executing query at Line 1
                Commands completed successfully.
                Total execution time: 00:00:00.023
```

Resources Used to Assist: https://chat.openai.com/ - Data Verification

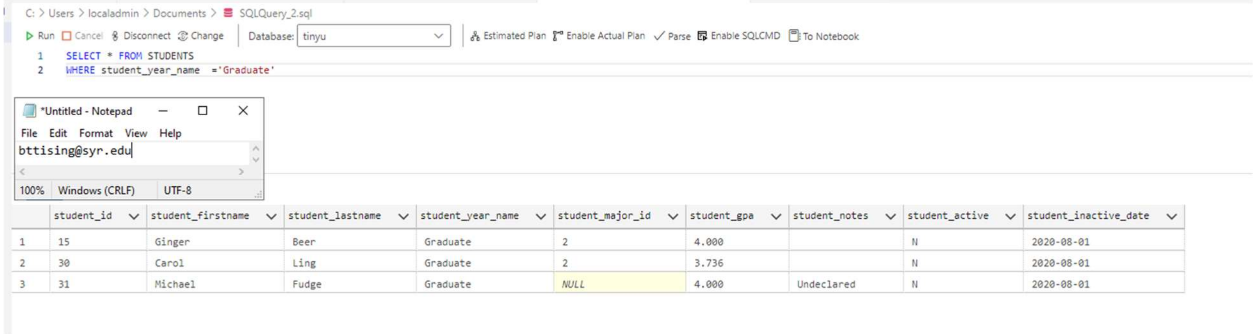
c. Write SQL code to deactivate all the ‘Graduate’ students with a date of ‘2020-08-01’.


Welcome
moze-up-down.sql - disconnected
C: > Users > localadmin > Documents > SQLQuery_2.sql
Run
Cancel
Disconnect
Change
Database: tinyu
*Untitled - Notepad
File Edit Format View Help
100% Windows (CRLF) UTF-8
Messages

```sql
UPDATE students
SET student_inactive_date = '2020-08-01'
WHERE student_year_name = 'Graduate';
```

bttising@syr.edu

```
7:28:33 PM    Started executing query at Line 1
              (3 rows affected)
              (3 rows affected)
              Total execution time: 00:00:00.194
```



```sql
SELECT * FROM STUDENTS
WHERE student_year_name  ='Graduate'
```

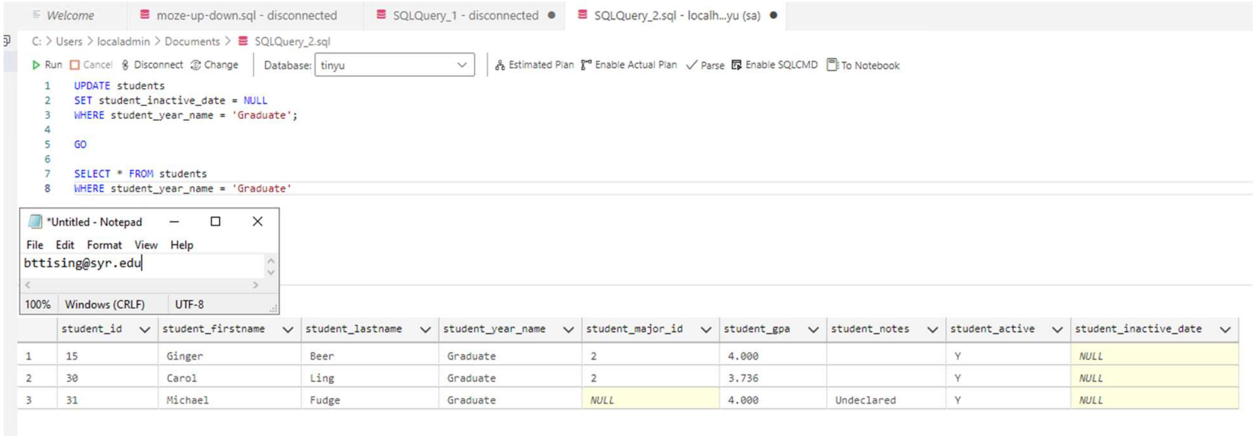
bttising@syr.edu

	student_id	student_firstname	student_lastname	student_year_name	student_major_id	student_gpa	student_notes	student_active	student_inactive_date
1	15	Ginger	Beer	Graduate	2	4.000		N	2020-08-01
2	30	Carol	Ling	Graduate	2	3.736		N	2020-08-01
3	31	Michael	Fudge	Graduate	NULL	4.000	Undeclared	N	2020-08-01

Graduate Students are now inactive.

d. Write SQL code to reactivate all the ‘Graduate’ students.

Provide a screenshot of your code from 4.a. and 4.b working. Provide another screen shot demonstrating 4.c worked. Then, provide a final screen shot of code and demonstration of 4.d working.


Welcome
moze-up-down.sql - disconnected
SQLQuery_1 - disconnected
SQLQuery_2.sql - localh...yu (sa)
C: > Users > localadmin > Documents > SQLQuery_2.sql
Run
Cancel
Disconnect
Change
Database: tinyu
Estimated Plan
Enable Actual Plan
Parse
Enable SQLCMD
To Notebook
*Untitled - Notepad
File Edit Format View Help
100% Windows (CRLF) UTF-8

```sql
UPDATE students
SET student_inactive_date = NULL
WHERE student_year_name = 'Graduate';

GO

SELECT * FROM students
WHERE student_year_name = 'Graduate'
```

bttising@syr.edu

	student_id	student_firstname	student_lastname	student_year_name	student_major_id	student_gpa	student_notes	student_active	student_inactive_date
1	15	Ginger	Beer	Graduate	2	4.000		Y	NULL
2	30	Carol	Ling	Graduate	2	3.736		Y	NULL
3	31	Michael	Fudge	Graduate	NULL	4.000	Undeclared	Y	NULL

Reflection

Use this section to reflect on your learning. To achieve the highest grade on the assignment you must be as descriptive and personal as possible with your reflection. Take time to consider these questions before you answer them. Your reflection should be personal. I consider it just as important as the work itself.

1. Reflect upon 3 things you learned this week.

    - **How to Create Functions**
    - **How to Create Triggers**
    - **How to do Stored Procedures**

2. What do you feel is still unclear about the topics covered this week?

    N/A

3. Do you feel you were prepared for this assignment? What can you do to be better prepared?

    Yes, I think I had to do a little outside digging and research but It wasn't too difficult.

4. Now that you have completed this topic, rate your comfort level with this week's material. This should be an honest assessment: (choose one)

    4 ==> I understand this material and can explain it to others.
    [✓ 3 ==> I understand this material.]
    2 ==> I somewhat understand the material but sometimes need guidance from others.
    1 ==> I understand very little of this material and need extra help.

5. Please provide any additional thoughts you have regarding your learning journey in this course to date.

Enjoying it so far!