RollCall Solutions

Proposal/Design:-

We will be helping the Master's Program Office (MPO) at Robert H. Smith School of Business to improve their database design. Currently, some faculty take attendance on pen and paper. So we came up with the idea to ease the process of taking attendance. Further, the team decided to make it more efficient by collecting feedback from students after each class and storing them.

We planned to update the attendance process. Currently, students sign the attendance sheets on paper during class sessions.

Mail to MPO (Amy Swann- Director of Business Master's Programs):-

We are planning a new database or incorporating changes into the existing system where a student will mark his presence using an access code provided by the lecturer in class. To avoid proxy cases, we can probably have time limits. For example, the faculty starts this attendance process just before the end of his class session, and students must finish within a minute.

At the same time, we can collect feedback from students. Was the class helpful? Any changes/suggestions for improvement in the future?
What were the reasons for students not attending a particular class? Are there any conflicts with interviews?

Storing these details and then gaining insights will help to improve the services.
We got a confirmation mail to go ahead with the project and to take data from the canvas website.

Data Resources:-

https://umd.instructure.com/
https://www.testudo.umd.edu/
https://www.rhsmith.umd.edu/directory
https://www.rhsmith.umd.edu/office-career-services/employers/recruitment/specialty-masters

Tables Description:-

Department:
Identify different departments in the specialty Master’s program at Smith school from the official website.

Student:
We have collected students' information such as UID, name, email, and username from the canvas website.

Faculty:
We identified faculty details such as name and email from the UMD faculty directory.

Course:
We searched it on Testudo to determine which faculty teaches what course.

Work:
This table has details of which faculty are working in which department.

Teach:
This table will have details of which faculty are teaching what course for the students.

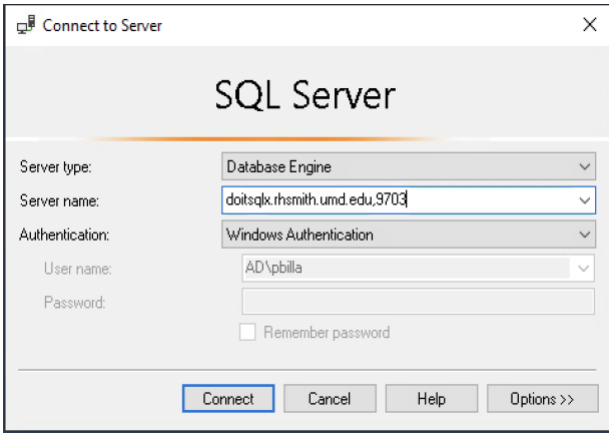
Attendance and Review:
We collected data from students for these two tables.

References:-

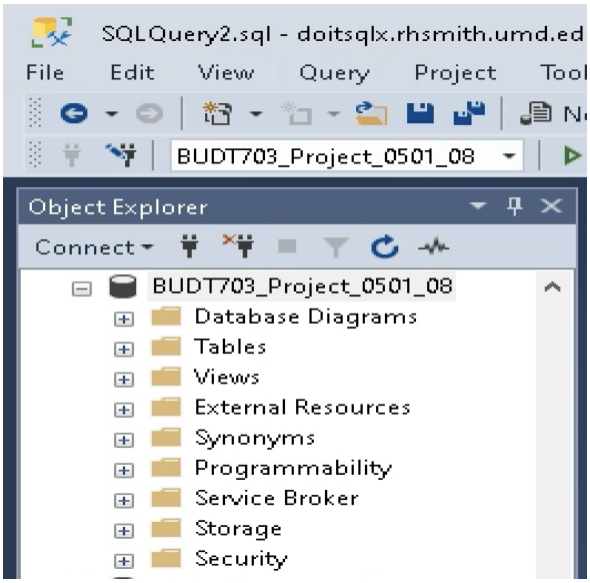
We referred to class notes, recordings on canvas, and the textbook for building our database model.

Steps to test the project:-

Step1:- Connect to the SQL Server on SQL Server Management Studio.



Step2:- After connecting to the server, switch to the database BUDT703_Project_0501_08

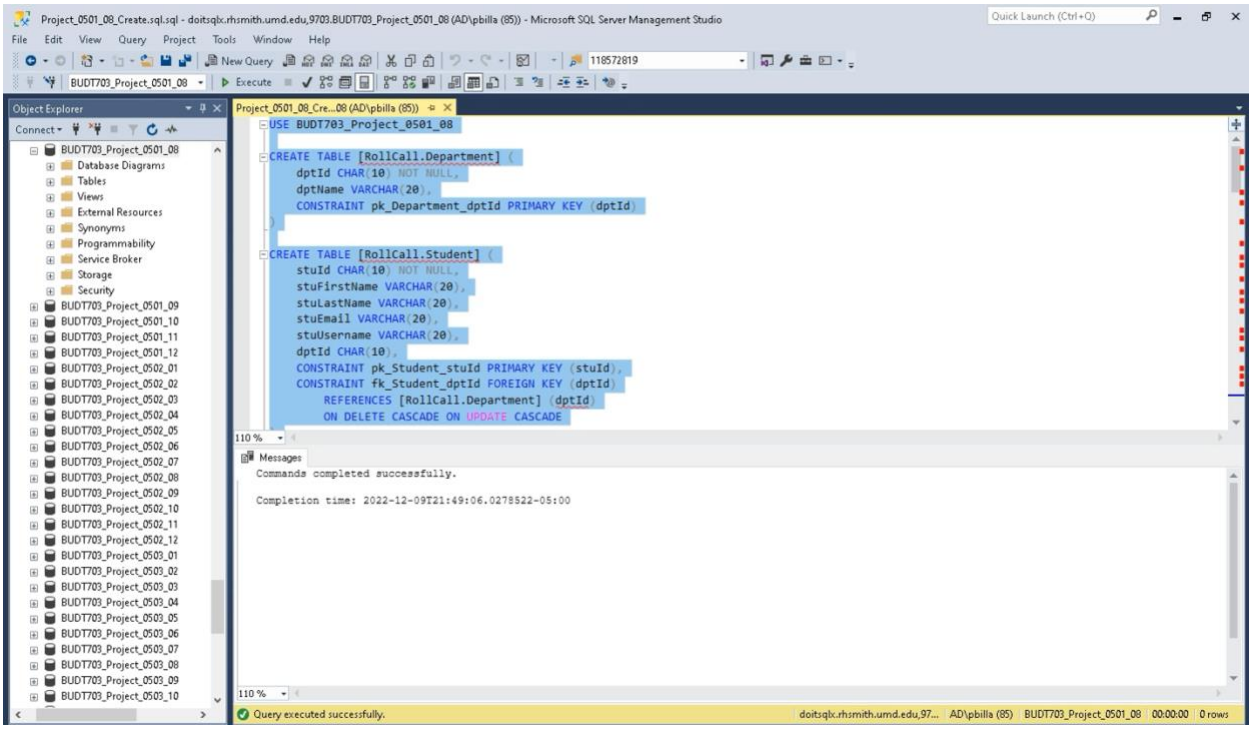


Step3:- After switching the database, open and execute Project_0501_08_Drop.sql

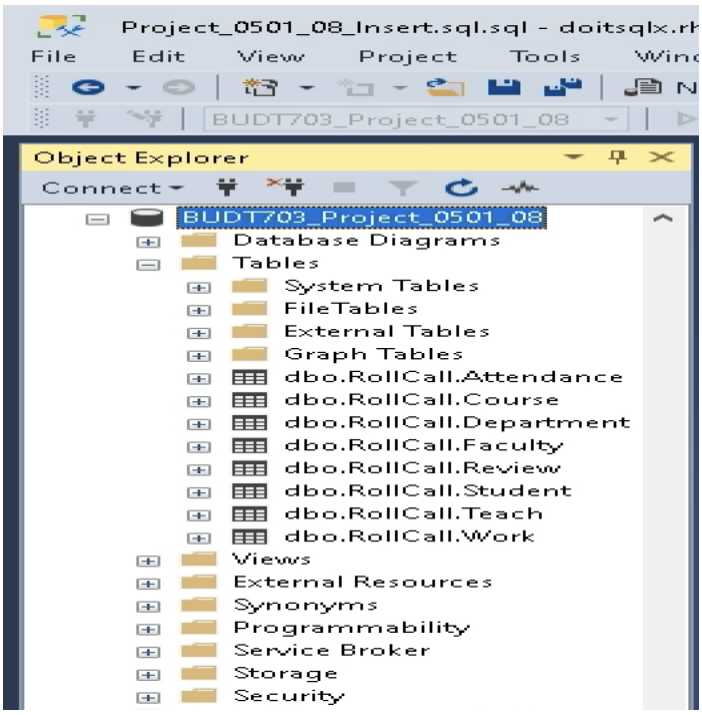
```
USE BUDT703_Project_0501_08

DROP TABLE IF EXISTS [RollCall.Review];
DROP TABLE IF EXISTS [RollCall.Attendance];
DROP TABLE IF EXISTS [RollCall.Teach];
DROP TABLE IF EXISTS [RollCall.Work];
DROP TABLE IF EXISTS [RollCall.Faculty];
DROP TABLE IF EXISTS [RollCall.Course];
DROP TABLE IF EXISTS [RollCall.Student];
DROP TABLE IF EXISTS [RollCall.Department];

DROP PROCEDURE getCourseonDepartment
```

```
Messages
Commands completed successfully.

Completion time: 2022-12-10T10:00:41.7676203+05:30
```

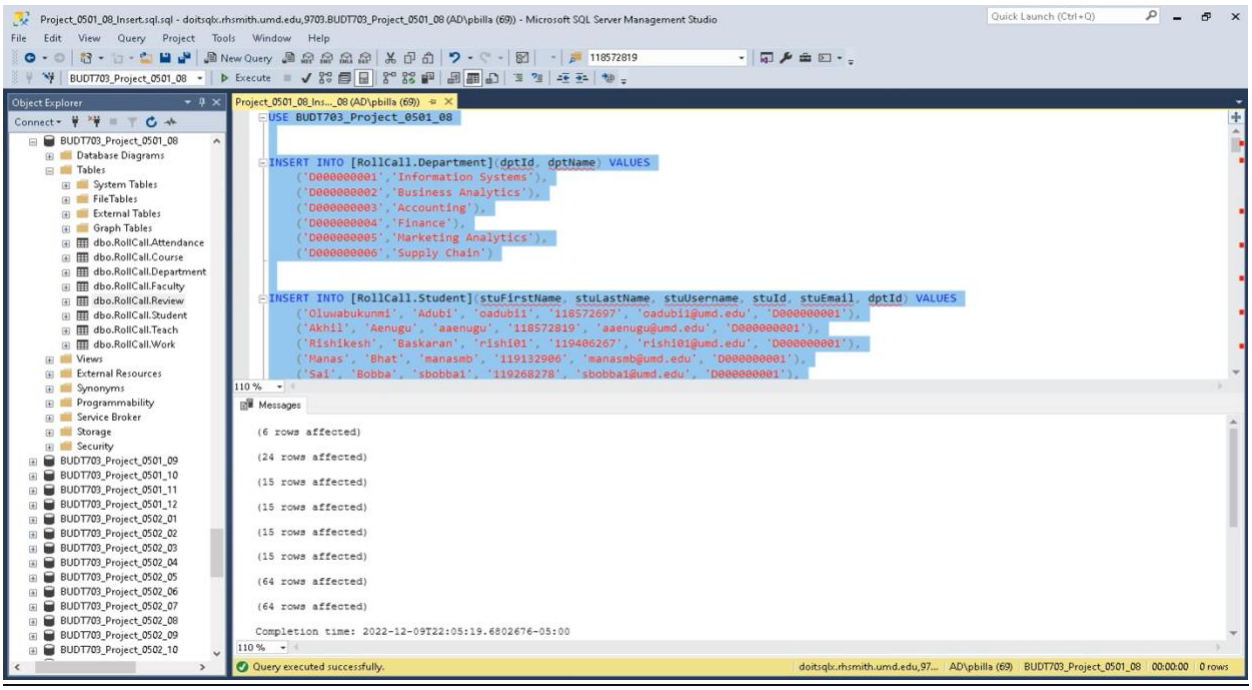
Step4:- Now, open and execute Project_0501_08_Create.sql



Verify if tables are created:



Step5:- After creating the tables, open and execute Project_0501_08_Insert.sql



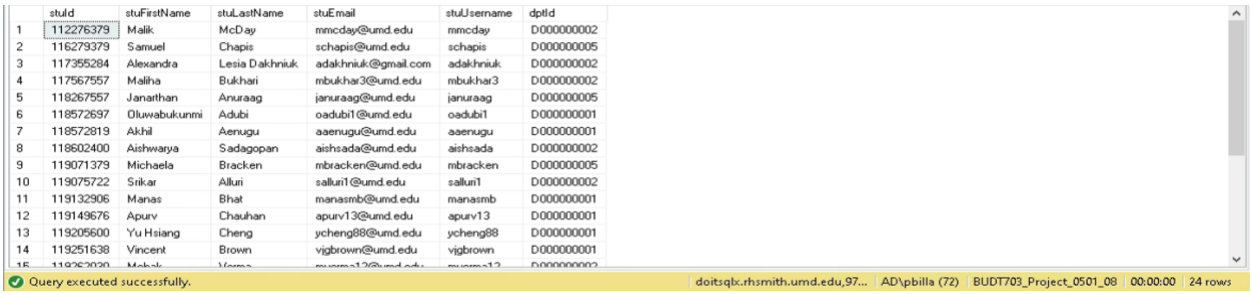
Step6:- Check contents of each table after insertion.

Department:

	dptId	dptName
1	D000000001	Information Systems
2	D000000002	Business Analytics
3	D000000003	Accounting
4	D000000004	Finance
5	D000000005	Marketing Analytics
6	D000000006	Supply Chain

Query executed successfully. doitsqlx.rhsmith.umd.edu,97... AD\pbilla (131) BUDT703_Project_0501_08 00:00:00 6 rows

Student:



	stuId	stuFirstName	stuLastName	stuEmail	stuUsername	dptId
1	112276379	Malik	McDay	mmcday@umd.edu	mmcday	D000000002
2	116279379	Samuel	Chapis	schapis@umd.edu	schapis	D000000005
3	117355284	Alexandra	Lesia Dakhniuk	adakhniuk@gmail.com	adakhniuk	D000000002
4	117567557	Maliha	Bukhari	mbukhar3@umd.edu	mbukhar3	D000000002
5	118267557	Janarthan	Anuraag	januraag@umd.edu	januraag	D000000005
6	118572697	Oluwabukunmi	Adubi	oadubi1@umd.edu	oadubi1	D000000001
7	118572819	Akhil	Aenugu	aaenugu@umd.edu	aaenugu	D000000001
8	118602400	Aishwarya	Sadagopan	aishsada@umd.edu	aishsada	D000000002
9	119071379	Michaela	Bracken	mbracken@umd.edu	mbracken	D000000005
10	119075722	Srikar	Alluri	salluri1@umd.edu	salluri1	D000000002
11	119132906	Manas	Bhat	manasmb@umd.edu	manasmb	D000000001
12	119149676	Apurv	Chauhan	apurv13@umd.edu	apurv13	D000000001
13	119205600	Yu Hsiang	Cheng	ycheng88@umd.edu	ycheng88	D000000001
14	119251638	Vincent	Brown	vjgbrown@umd.edu	vjgbrown	D000000001
15	119262039	Mehak	Verma	mverma12@umd.edu	mverma12	D000000002

Query executed successfully. | doitsqlx.rhsmith.umd.edu,97... | AD\pbilla (72) | BUDT703_Project_0501_08 | 00:00:00 | 24 rows

Course:

	corId	corName	corStrength
1	C000000001	Database Management Systems	39
2	C000000002	Data Models And Decisions	39
3	C000000003	Data Processing And Analytics in Python	39
4	C000000004	Digital Business Transformation	40
5	C000000005	Managing Digital Business Markets	40
6	C000000006	Decision Operation And Information Techonology	39
7	C000000007	Data Processing And Analytics in Python	39
8	C000000008	Advanced Marketing Analytics	39
9	C000000009	Business Policies And Ethics	40
10	C000000010	Maketing Strategy	40
11	C000000011	Supply Chain Strategy	39
12	C000000012	Purchasing and Inbound Logistics	39
13	C000000013	International Supply Chain Management	39
14	C000000014	Accounting and Information Assurance	40
15	C000000015	Introduction to Financial Markets and Financial ...	40

Query executed successfully. | doitsqlx.rhsmith.umd.edu,97... | AD\pbilla (135) | BUDT703_Project_0501_08 | 00:00:00 | 15 rows

Faculty:

	facId	facFirstName	facLastName	facEmail
1	F000000001	Adam	Lee	adamlee@umd.edu
2	F000000002	Sujin	Kim	sujinkim@umd.edu
3	F000000003	John	Bono	jbono@umd.edu
4	F000000004	Tejwansh	Anand	tejanand@umd.edu
5	F000000005	Siva	Viswanathan	sviswan1@umd.edu
6	F000000006	Suresh	Acharya	suresh12@umd.edu
7	F000000007	Kunpeng	Zhang	kpzhang@umd.edu
8	F000000008	Michel	Wedel	mwedel@umd.edu
9	F000000009	Amna	Kirmani	akirmani@umd.edu
10	F000000010	Judy	Frels	jfrels@umd.edu
11	F000000011	Martin	Dresner	mdresner@umd.edu
12	F000000012	Phil	Evers	pevers@umd.edu
13	F000000013	Niratcha	Tungtisanont	niratcha@umd.edu
14	F000000014	Lawrence	Gordon	lagordon@umd.edu
15	F000000015	Albert	Kyle	akyle@rhsmith.umd.edu

Query executed successfully. | doitsqlx.rhsmith.umd.edu,97... | AD\pbilla (138) | BUDT703_Project_0501_08 | 00:00:00 | 15 rows

Work:

	dptId	facId
1	D000000001	F000000001
2	D000000001	F000000002
3	D000000001	F000000003
4	D000000001	F000000004
5	D000000001	F000000005
6	D000000002	F000000006
7	D000000002	F000000007
8	D000000003	F000000008
9	D000000003	F000000009
10	D000000003	F000000010
11	D000000004	F000000011
12	D000000004	F000000012
13	D000000004	F000000013
14	D000000005	F000000014
15	D000000006	F000000015

Query executed successfully. | doitsqlx.rhsmith.umd.edu,97... | AD\pbilla (59) | BUDT703_Project_0501_08 | 00:00:00 | 15 rows

Teach:

	facId	corId
1	F000000001	C000000001
2	F000000002	C000000002
3	F000000003	C000000003
4	F000000004	C000000004
5	F000000005	C000000005
6	F000000006	C000000006
7	F000000007	C000000007
8	F000000008	C000000008
9	F000000009	C000000009
10	F000000010	C000000010
11	F000000011	C000000011
12	F000000012	C000000012
13	F000000013	C000000013
14	F000000014	C000000014
15	F000000015	C000000015

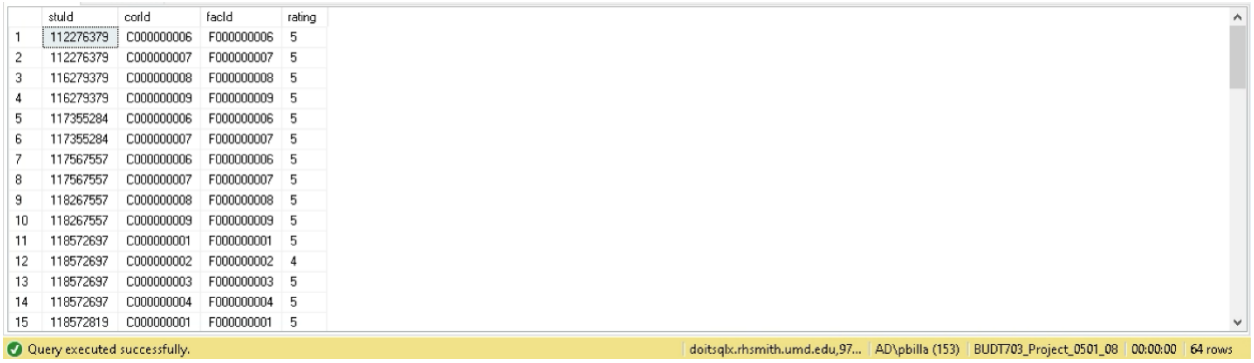
Query executed successfully. | doitsqlx.rhsmith.umd.edu,97... | AD\pbilla (152) | BUDT703_Project_0501_08 | 00:00:00 | 15 rows

Attendance:

	stuId	corId	atdstatus	atddate
1	112276379	C000000006	Present	2022-08-29
2	112276379	C000000007	Absent	2022-08-31
3	116279379	C000000008	Present	2022-08-29
4	116279379	C000000009	Present	2022-08-31
5	117355284	C000000006	Present	2022-08-29
6	117355284	C000000007	Present	2022-08-31
7	117567557	C000000006	Present	2022-08-29
8	117567557	C000000007	Present	2022-08-31
9	118267557	C000000008	Present	2022-08-29
10	118267557	C000000009	Present	2022-08-31
11	118572697	C000000001	Present	2022-08-29
12	118572697	C000000002	Absent	2022-08-29
13	118572697	C000000003	Present	2022-08-30
14	118572697	C000000004	Present	2022-08-30
15	118572819	C000000001	Absent	2022-08-29

Query executed successfully. | doitsqlx.rhsmith.umd.edu,97... | AD\pbilla (155) | BUDT703_Project_0501_08 | 00:00:00 | 64 rows

Review:



	stuId	corId	facId	rating
1	112276379	C000000006	F000000006	5
2	112276379	C000000007	F000000007	5
3	116279379	C000000008	F000000008	5
4	116279379	C000000009	F000000009	5
5	117355284	C000000006	F000000006	5
6	117355284	C000000007	F000000007	5
7	117567557	C000000006	F000000006	5
8	117567557	C000000007	F000000007	5
9	118267557	C000000008	F000000008	5
10	118267557	C000000009	F000000009	5
11	118572697	C000000001	F000000001	5
12	118572697	C000000002	F000000002	4
13	118572697	C000000003	F000000003	5
14	118572697	C000000004	F000000004	5
15	118572819	C000000001	F000000001	5

Query executed successfully. | doitsqlx.rhsmith.umd.edu,97... | AD\pbilla (153) | BUDT703_Project_0501_08 | 00:00:00 | 64 rows

Step7:- After creating tables and inserting data into it, review the database tables and their information before performing the required business transactions.

Business Transaction 1:

To find the list of students who have least regularly attended the courses in each department

```
-- 1. To find the students who have attended atleast one class for the registered courses but have least attendance in each department.

SELECT ac.stuId AS 'Student Id',
       ac.[Student Name] AS 'Student Name',
       ac.attendanceCount AS 'Attendance Count',
       ac.dptName AS 'Department Name'
       FROM (SELECT s.stuId,
                    CONCAT(s.stuFirstName,' ',s.stuLastName) AS 'Student Name',
                    COUNT(a.atdstatus) AS attendanceCount,
                    d.dptId,
                    d.dptName,
                    DENSE_RANK() OVER (PARTITION BY d.dptName ORDER BY COUNT(a.atdstatus)) as atd_rnk
                FROM [RollCall.Student] s, [RollCall.Attendance] a, [RollCall.Department] d
                WHERE a.stuId = s.stuId AND s.dptId=d.dptId AND a.atdstatus = 'Present'
                GROUP BY s.stuId, s.stuFirstName, s.stuLastName, d.dptId, d.dptName
       )ac
WHERE ac.atd_rnk=1 -- rank is used to determine the students with least attendance
ORDER BY [Attendance Count]
```

Output:

Student Id	Student Name	Attendance Count	Department Name
112276379	Malik McDay	1	Business Analytics
119262030	Mohak Verma	1	Business Analytics
119340639	Lohith Maddula	1	Business Analytics
119419847	Mohit Buddha	1	Business Analytics
119071379	Michaela Bracken	1	Marketing Analytics
119597517	Anirudh Anandh	1	Marketing Analytics
119718278	Papitha Lakshmanan	1	Supply Chain
119406267	Rishikesh Baskaran	2	Information Systems

Business Transaction 2:

```
-- 2. To find the list of students present for the courses for the first day of the class.

SELECT s.stuId AS 'Student Id',
        CONCAT(s.stuFirstName,' ',s.stuLastName) AS 'Student Name',
        s.stuEmail AS 'Student Email',
        s.stuUsername AS 'Student Username',
        s.dptId AS 'Department Id'
FROM [RollCall.Student] s
WHERE s.stuId IN (
                SELECT a.stuId
                FROM [RollCall.Attendance] a, (
                                                SELECT MIN(a.atddate) AS firstDay
                                                FROM [RollCall.Attendance] a) fd
                WHERE a.atddate = fd.firstDay)
```

Output:

Student Id	Student Name	Student Email	Student Username	Department Id
112276379	Malik McDay	mmcday@umd.edu	mmcday	D000000002
116279379	Samuel Chapis	schapis@umd.edu	schapis	D000000005
117355284	Alexandra Lesia Dakhniuk	adakhniuk@gmail.com	adakhniuk	D000000002
117567557	Maliha Bukhari	mbukhar3@umd.edu	mbukhar3	D000000002
118267557	Janarthan Anuraag	januraag@umd.edu	januraag	D000000005
118572697	Oluwabukunmi Adubi	oadubi1@umd.edu	oadubi1	D000000001
118572819	Akhil Aenugu	aaenugu@umd.edu	aaenugu	D000000001
118602400	Aishwarya Sadagopan	aishsada@umd.edu	aishsada	D000000002
119071379	Michaela Bracken	mbracken@umd.edu	mbracken	D000000005
119075722	Srikar Alluri	salluri1@umd.edu	salluri1	D000000002
119132906	Manas Bhat	manasmb@umd.edu	manasmb	D000000001
119149676	Apurv Chauhan	apurv13@umd.edu	apurv13	D000000001
119205600	Yu Hsiang Cheng	ycheng88@umd.edu	ycheng88	D000000001
119251638	Vincent Brown	vjgbrown@umd.edu	vjgbrown	D000000001
119262030	Mohak Verma	mverma12@umd.edu	mverma12	D000000002
119268278	Sai Bobba	sbobba1@umd.edu	sbobba1	D000000001
119340639	Lohith Maddula	lohith88@umd.edu	lohith88	D000000002
119381311	Parichay Bajaj	pbajaj@umd.edu	pbajaj	D000000002
119406267	Rishikesh Baskaran	rishi01@umd.edu	rishi01	D000000001
119410134	Sai Varanasi	ssvaran9@umd.edu	svaran9	D000000002
119419847	Mohit Buddha	mbuddha@umd.edu	mbuddha	D000000002
119597517	Anirudh Anandh	aananth@umd.edu	aananth	D000000005
119718078	Akhib Ahmed	aakhib@umd.edu	aakhib	D000000006
119718278	Papitha Lakshmanan	lpapitha@umd.edu	lpapitha	D000000006

Business Transaction 3:

```
-- 3. To find the average rating given by all students for each faculty.

SELECT r.facId AS 'Faculty Id',
        CONCAT(f.facFirstName,' ',f.facLastName) AS 'Faculty Name',
        AVG(r.rating) AS 'Average Rating'
FROM [RollCall.Review] r, [RollCall.Faculty] f
WHERE r.facId = f.facId
GROUP BY r.facId, f.facFirstName, f.facLastName, r.rating
ORDER BY r.facId ASC
```

Output:

Faculty Id	Faculty Name	Average Rating
F000000001	Adam Lee	5
F000000002	Sujin Kim	4
F000000003	John Bono	5
F000000004	Tejwansh Anand	5
F000000006	Suresh Acharya	5
F000000007	Kunpeng Zhang	5
F000000008	Michel Wedel	5
F000000009	Amna Kirmani	5
F000000010	Judy Frels	5
F000000011	Martin Dresner	5

Business Transaction 4:

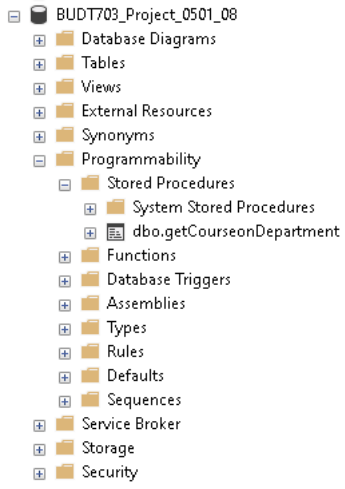
```
-- 4. To find the department having the highest positive feedback given from their students.

SELECT d.dptId AS 'Department Id',
        d.dptName AS 'Department Name',
        AVG(r.rating) AS 'Average Rating'
FROM [RollCall.Department] d, [RollCall.Review] r, [RollCall.Student] s
WHERE s.dptId = d.dptId AND r.stuId = s.stuId
GROUP BY d.dptId, d.dptName
ORDER BY 'Average Rating' DESC
```

Department Id	Department Name	Average Rating
D000000002	Business Analytics	5
D000000005	Marketing Analytics	5
D000000006	Supply Chain	5
D000000001	Information Systems	4

Business Transaction 5:

```
--5. Finding the average rating of courses offered based on department(eg Department Name = Information Systems).

CREATE OR ALTER PROCEDURE getCourseonDepartment
    @DepartmentName nvarchar(50)
AS

    SET NOCOUNT ON;
    SELECT ar.dptName AS 'Department Name',
           ar.corName AS 'Course Name',
           ar.avg_rating as 'Average Rating' FROM
           (SELECT d.dptName,
                   c.corName,
                   AVG(r.rating) as avg_rating,
                   DENSE_RANK() OVER (PARTITION BY d.dptName order by AVG(r.rating) DESC) as course_rnk
           FROM [RollCall.Department] d, [RollCall.Review] r, [RollCall.Student] s, [RollCall.Course] c
           WHERE s.dptId = d.dptId AND r.stuId = s.stuId and r.corId = c.corId
           GROUP BY d.dptName, c.corName)ar
    WHERE ar.dptName = @DepartmentName
    ORDER BY ar.avg_rating DESC

EXECUTE getCourseonDepartment @DepartmentName = N'Information Systems';
```

Verify if procedure is created:



Output:

Department Name	Course Name	Average Rating
Information Systems	Data Processing And Analytics in Python	5
Information Systems	Database Management Systems	5
Information Systems	Digital Business Transformation	5
Information Systems	Data Models And Decisions	4

Queries:

DML.sql

Step8:- Lastly, disconnect from the server and close the connection.