Assignment 2 Report
Instructor: Adnan A. Khan
CSC 33600
Group members: **Mahfuzul Islam, Nikhil Prajith Kumar, Namgyal Thily**

1) Description of the developed database.

The database consists of three tables which are Users, Faculty and Courses.

Users Table: Contains information about users in the system. It has the following columns:

- Id: An integer column serving as the primary key for the table.
- Username: A unique string column that stores the username of the user.
- Password: A string column that stores the password of the user. Cannot be empty.
- ConfirmPassword: A string column that stores the confirmed password of the user. It cannot be empty.
  MatchPassword Constraint: ConfirmPassword must match Password field
- Age: An integer column that represents the age of the user. It should be between 12 and 100.
- Gender: A character column that represents the gender of the user. It should be either 'F' or 'M'.

Faculty Table: Contains information about faculty members. It has the following columns:

- DepartmentId: The primary key, an integer column representing the department ID associated with the faculty member included in the Users table as well.
- Name: A string column storing the name of the faculty member - cannot be null.
- DepartmentName: A string column that stores the name of the department to which the faculty member belongs. Cannot be null as well.
- IdRef: An integer column serving as a foreign key referencing the Id column in the Users table. It indicates the user associated with the faculty member. Has a Set Null policy for referencing actions.

Courses Table: Contains information about courses offered. It has the following columns:

- CourseId: An integer column serving as the primary key for the table.
- CourseName: A string column storing the name of the course. Cannot be null.
- DepartmentId: An integer column representing the department ID associated with the course. Has a Cascade policy.

This database structure allows for managing users, faculty members, and courses. Users can be associated with faculty members through idRef in the Faculty table, and courses can be associated with specific departments through the departmentID reference in Courses table. The foreign key constraints maintain data consistency and integrity between the tables.

In addition, the referencing between the *Users* and the *Faculty* table is set up with the Set NULL policy for deletion. So if the referenced attribute is deleted in the *Users* table the referencing attribute will be set to NULL in the *Faculty* table, and no errors will be shown.

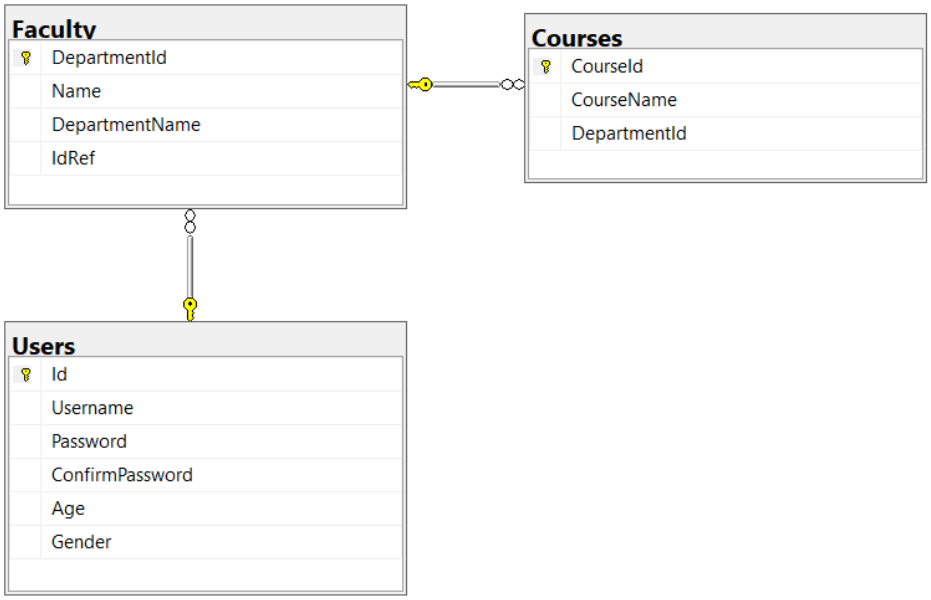
Furthermore, the referencing policy between the *Faculty* and the *Courses* is set to Cascade. If the reference attribute is updated, then the referencing attribute in the *Courses* table is also updated. If a deletion is made in the *Faculty* table that has a referencing with the *Courses* table then, all rows where the referencing attribute is equal to the delete data of the *Faculty* table will be deleted as well.

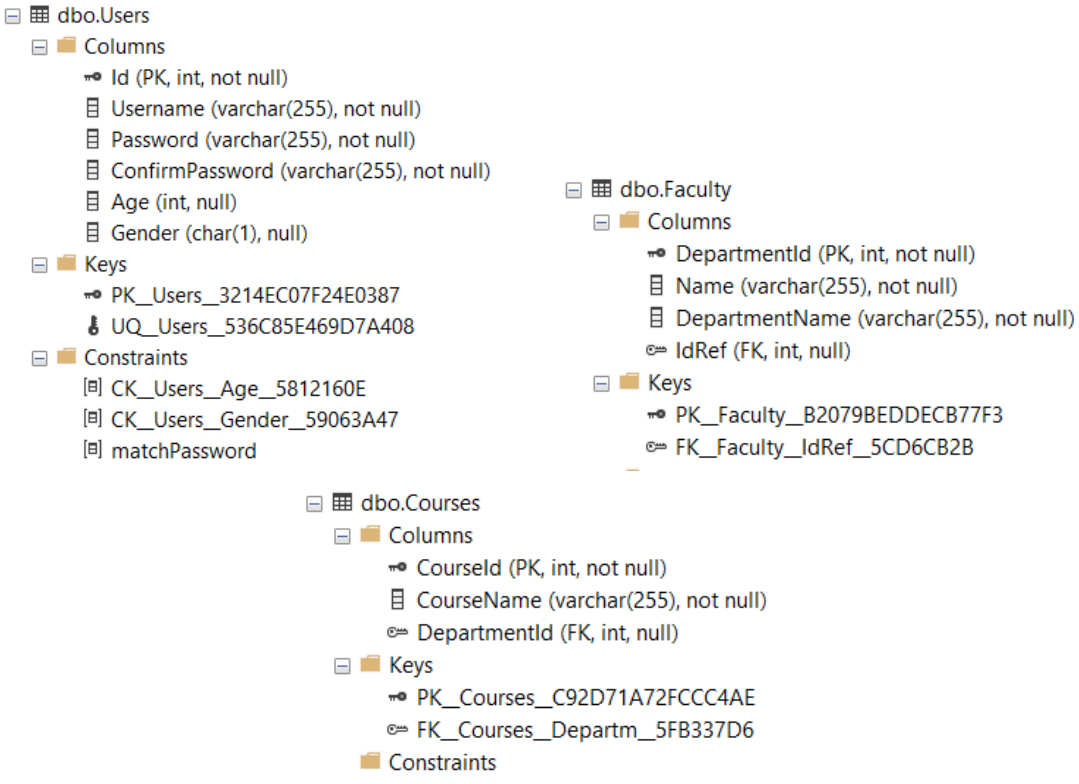
2) Reasoning behind choosing this topic (also include usefulness):

The database consisting of the three tables: Users, Faculty and Courses, allows for the development of a system to manage educational institutions effectively. By incorporating tables for users, consisting of students and staff, along with faculty members and courses, the database enables organization and administration of educational information.

This topic is particularly useful as the database empowers educational institutions to efficiently handle user data, maintain accurate faculty information, and manage course offerings according to department and faculty members available. The database can be used to view which faculty members have no associated courses which will allow the administration to associate them with new courses. In conclusion, this topic helps streamline the overall management of educational institutions.

3) Database design (include screenshot from implemented database showing all columns of all tables, like screenshot from Table Designer window of SQL Management Studio).





The *Users* table allows data with constraints such as a primary key constraint on the Id column, a unique constraint on the Username column, and check constraints on the Password, ConfirmPassword, Age, and Gender columns. Along with a constraint to ensure that Confirm Password equals the Password field.

The *Faculty* table establishes a relationship with the Users table through the foreign key constraint on the IdRef column which references the Id column of the *Users* table. The constraint specifies that the IdRef column must have a valid reference to the Id column in the Users table. The primary key constraint of this table is present in the DepartmentId column.

The Courses table establishes a relationship with the Faculty table through the foreign key constraint on the DepartmentId column. The CourseId is the primary key field of this table; this differentiates a data row from other data rows.

4) All SQL codes (like tables, constraints, can exclude insert)

Note: For some reason there are error lines indicated by the red squiggle on some of the table names and its attributes, but there are no errors present and the SQL code runs successfully.

SQLQueryUsrers.sql...r (MAZ\mahfu (73))

```sql
CREATE TABLE Users (
    Id INT PRIMARY KEY,
    Username VARCHAR(255) UNIQUE NOT NULL,
    Password VARCHAR(255) NOT NULL,
    ConfirmPassword VARCHAR(255) NOT NULL,
    Age INT CHECK (Age BETWEEN 12 AND 100),
    Gender CHAR(1) CHECK (Gender IN ('F', 'M')),
    CONSTRAINT matchPassword CHECK (Password = ConfirmPassword)
);
```

SQL code used for the creation of the *Users* table. Highlights all the constraints present such as primary key, matchPassword, etc.

```
SQLQuery6.sql - M...r (MAZ\mahfu (75))   SQLQueryUsrers.s

CREATE TABLE Faculty (
  DepartmentId INT PRIMARY KEY,
  Name VARCHAR(255) NOT NULL,
  DepartmentName VARCHAR(255) NOT NULL,
  IdRef INT,
  FOREIGN KEY (IdRef)
    REFERENCES Users(Id)
        ON DELETE SET NULL

);
```

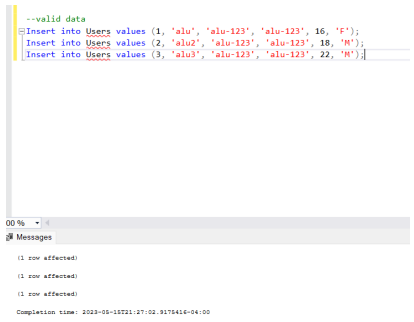
SQL code used for the creation of the *Faculty* table. Included the foreign key and all other constraints put on the columns.

```
SQLQueryCourses.s...r (MAZ\mahfu (76))   SQLQuery6.sql - M...r (MAZ\mahfu (75))   SQLC

CREATE TABLE Courses (
  CourseId INT PRIMARY KEY,
  CourseName VARCHAR(255) NOT NULL,
  DepartmentId INT REFERENCES Faculty(DepartmentId) ON UPDATE CASCADE
);
```

SQL code used to create the *Courses* table which included the referencing code.

Constraint Testing:



	Id	Username	Password	ConfirmPas...	Age	Gender
▶	1	alu	alu-123	alu-123	16	F
	2	alu2	alu-123	alu-123	18	M
	3	alu3	alu-123	alu-123	22	M
*	*NULL*	*NULL*	*NULL*	*NULL*	*NULL*	*NULL*

Valid data entry. 3 rows created successfully.

```
--invalid data: Users
Insert into Users values (4, 'alu4', 'alu-123', 'abc-456', 3, 'F'); -- same user name error, match password error, age error
Insert into Users values (4, 'alu5', 'hshd', 'gghs', 33, 'S'); -- Gender error, match password error
Insert into Users(Id,password,ConfirmPassword,Gender)  values (5, 'hshd', 'gghs', 'M'); -- Username is NULL
Insert into Users values (3, 'alu4', 'alu-123', 'alu-123', 22, 'M');-- same primary key
Insert into Users values(4, 'alu4','alu2222','alu33',25,'M');--Match Password constraint failed
```

Messages

```
Msg 547, Level 16, State 0, Line 8
The INSERT statement conflicted with the CHECK constraint "CK__Users__Age__3899ABB6". The conflict occurred in database "assign2DB", table "dbo.Users", column 'Age'.
The statement has been terminated.
Msg 547, Level 16, State 0, Line 9
The INSERT statement conflicted with the CHECK constraint "CK__Users__Gender__398D8EEE". The conflict occurred in database "assign2DB", table "dbo.Users", column 'Gender'.
The statement has been terminated.
Msg 515, Level 16, State 2, Line 10
Cannot insert the value NULL into column 'Username', table 'assign2DB.dbo.Users'; column does not allow nulls. INSERT fails.
The statement has been terminated.
Msg 2627, Level 14, State 1, Line 11
Violation of PRIMARY KEY constraint 'PK__Users__3214EC074D3E8B82'. Cannot insert duplicate key in object 'dbo.Users'. The duplicate key value is (3).
The statement has been terminated.
Msg 547, Level 16, State 0, Line 12
The INSERT statement conflicted with the CHECK constraint "matchPassword". The conflict occurred in database "assign2DB", table "dbo.Users".
The statement has been terminated.

Completion time: 2023-05-15T21:45:26.9933434-04:00
```

Invalid data entry for Users table. Each test case tests a different constraint check.

```
--Valid data: Faculty
Insert into Faculty values (11, 'Adnan', 'CSC', 1);
Insert into Faculty values (12, 'Abdul', 'CSC', 2);
Insert into Faculty values (13, 'Nikhil', 'CSC', 3);
Update Faculty set DepartmentName='Math' where DepartmentId = 13;
```

Messages

```
(1 row affected)

Completion time: 2023-05-15T22:01:40.3142312-04:00
```

	Departmen...	Name	Departmen...	IdRef
▶	11	Adnan	CSC	1
	12	Abdul	CSC	2
	13	Nikhil	Math	3
*	*NULL*	*NULL*	*NULL*	*NULL*

Valid data entry in Faculty table.

```
--Invalid data: Faculty
Insert into Faculty values (11, 'Kan', 'Eng', 2); --same username
```

```
%
Messages
Msg 2627, Level 14, State 1, Line 22
Violation of PRIMARY KEY constraint 'PK__Faculty__B2079BEDABE3B6AF'. Cannot insert duplicate key in object 'dbo.Faculty'. The duplicate key value is (11).
The statement has been terminated.

Completion time: 2023-05-15T22:06:46.3903217-04:00
```

Invalid data entry into Faculty table.

	Id	Username	Password	ConfirmPas...	Age	Gender
▶	1	alu	alu-123	alu-123	16	F
	3	alu3	alu-123	alu-123	22	M
*	*NULL*	*NULL*	*NULL*	*NULL*	*NULL*	*NULL*

Removes the Id 2

```
--Delete users id
Delete from Users where Id = 2;
```

```
00 %
Messages

(1 row affected)

Completion time: 2023-05-15T22:08:47.7846858-04:00
```

	Departmen...	Name	Departmen...	IdRef
▶	11	Adnan	CSC	1
	12	Abdul	CSC	*NULL*
	13	Nikhil	Math	3
*	*NULL*	*NULL*	*NULL*	*NULL*

Referential Policy check. After removing User Id 2, it sets to Null in faculty table

```
-- Valid Data-- Courses
Insert into Courses values (111, 'Database', 11);
Insert into Courses values (112, 'Algorithm', 12);
Insert into Courses values (113, 'Calc', 13);
```

```
00 %
Messages

(1 row affected)

(1 row affected)

(1 row affected)

Completion time: 2023-05-15T22:15:27.9642127-04:00
```

	CourseId	CourseName	Departmen...
▶	111	Database	11
	112	Algorithm	12
	113	Calc	13
*	NULL	NULL	NULL

Valid data for Courses

```
-- Invalid Data: Courses
Insert into Courses values (111, 'Trig', 13); --same CourseId already exist
```

```
%
Messages
Msg 2627, Level 14, State 1, Line 34
Violation of PRIMARY KEY constraint 'PK__Courses__C92D71A7B5ED85D3'. Cannot insert duplicate key in object 'dbo.Courses'. The duplicate key value is (111).
The statement has been terminated.

Completion time: 2023-05-15T22:18:01.3516413-04:00
```

Invalid data for Courses

```
-- Update row from Faculty
Update Faculty set DepartmentId = 25 where DepartmentName='Math' ;
```

.00 %

Messages

```
(1 row affected)

Completion time: 2023-05-15T22:28:15.5187531-04:00
```

	Departmen...	Name	Departmen...	IdRef
▶	11	Adnan	CSC	1
	12	Abdul	CSC	*NULL*
	25	Nikhil	Math	3
*	*NULL*	*NULL*	*NULL*	*NULL*

	CourseId	CourseName	Departmen...
▶	111	Database	11
	112	Algorithm	12
	113	Calc	25
*	*NULL*	*NULL*	*NULL*

Update Row from Faculty to update also on Courses