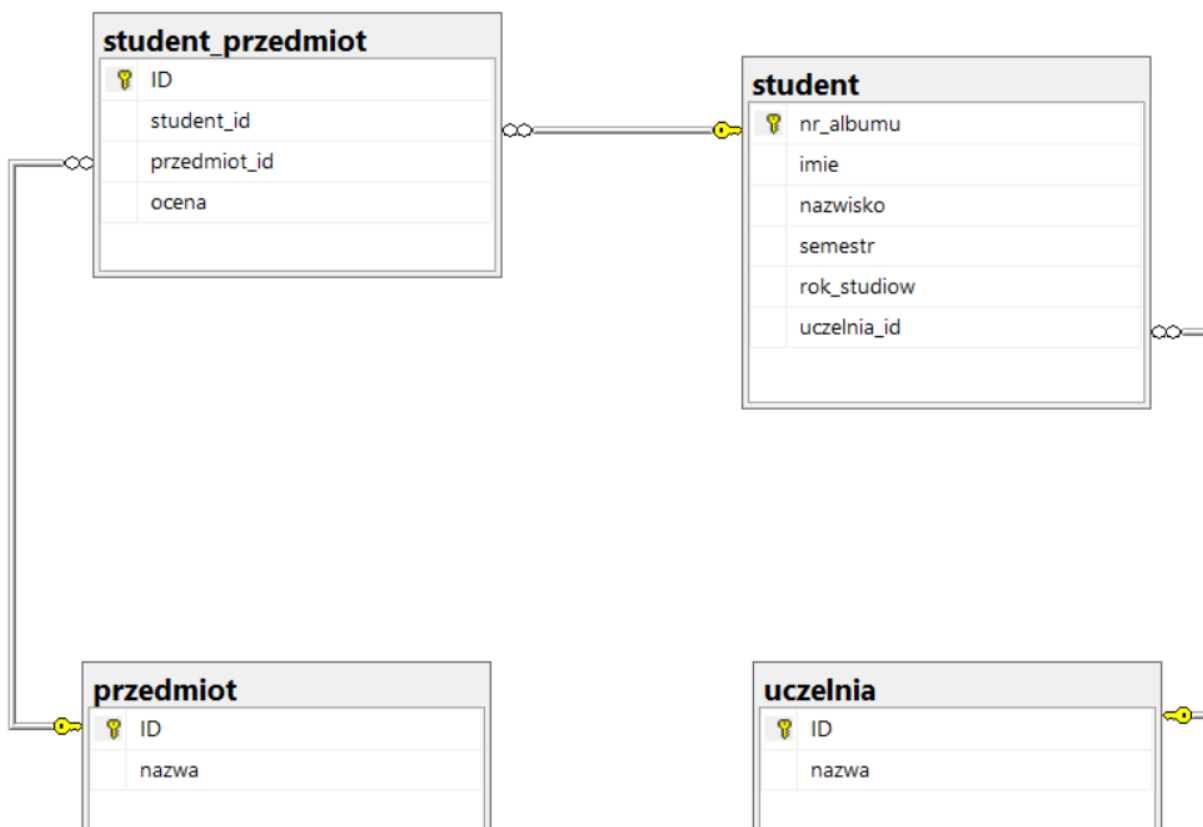


### ***Zadania:***

1. Stworzyć procedurę, która stworzy nowego studenta, zapisze go na wybraną uczelnię i trzy wybrane przedmioty. Użytkownik podaje wszystkie dane dotyczące studenta, nazwę uczelni oraz nazwy przedmiotów. W przypadku podania złej nazwy uczelni lub przedmiotu, należy obsłużyć błąd i wyświetlić użytkownikowi informację co zrobił źle. Po każdym kroku (stworzenie studenta, zapisanie na uczelnie i na przedmioty) proszę o wypisanie stosownego komunikatu do konsoli.
2. Przeprowadzono konkurs na popularność przedmiotów prowadzonych na uczelniach. Zdecydowano, że wyłączność do prowadzenia przedmiotu będzie miała tylko uczelnia, która ma największą liczbę studentów zapisanych na przedmiot. Do tabeli student\_przedmiot dodać kolumnę "uczelnia\_id", a następnie zapełnić ją odpowiednimi wartościami. Proszę w sprawozdaniu załączyć potrzebne altery/select'y/update'y.
3. Stworzyć procedurę, która po podaniu nazwy przedmiotu porówna jego popularność na wszystkich uczelniach.

### ***Rozwiązywanie zadania:***

Po pierwsze zrobię diagram bazy danych, dla wygody robienia zadań (rysunek 1):



(rysunek 1)

1. Stworzona procedura, która stworzy nowego studenta, zapisze go na wybraną uczelnię i trzy wybrane przedmioty. Użytkownik podaje wszystkie dane dotyczące studenta, nazwę uczelni oraz nazwy przedmiotów. W przypadku podania złej nazwy uczelni lub przedmiotu, należy obsłużyć błąd i wyświetlić użytkownikowi informację co zrobił źle. Po każdym kroku (stworzenie studenta, zapisanie na uczelnię i na przedmioty) proszę o wypisanie stosownego komunikatu do konsoli.

```
CREATE PROCEDURE zadaine1 @imie nvarchar(20), @nazwisko nvarchar(20), @nazwa_uczelni
nvarchar(20), @przedmiot nvarchar(40)
AS
BEGIN
    BEGIN TRY
        INSERT INTO student_przedmiot ( student_id, przedmiot_id )
        VALUES ( (SELECT nr_albumu, imie, nazwisko FROM INSERTED) ,
        (SELECT 1 + CONVERT(INT, ((SELECT COUNT(*) FROM przedmiot)) * (3 +
przedmiot.ID))),
        (SELECT 1 + CONVERT(INT, ((SELECT COUNT(*) FROM uczelnia)) * (uczelnia.ID)))
        )
        print 'Student zostal zapisany do uczelni i do 3 przedmiotow';
    END TRY
    BEGIN CATCH
        IF @nazwa_uczelni IS NULL
            print 'Nie isntenie takiej uczelni';
        ELSE IF @przedmiot IS NULL
            print 'Nie podano trzech przedmiotow';
        print ERROR_MESSAGE()
        RETURN 0
    END CATCH
END
```

2. Przeprowadzono konkurs na popularność przedmiotów prowadzonych na uczelniach. Zdecydowano, że wyłączność do prowadzenia przedmiotu będzie miała tylko uczelnia, która ma największą liczbę studentów zapisanych na przedmiot. Do tabeli student\_przedmiot dodać kolumnę “uczelnia\_id”, a następnie zapełnić ją odpowiednimi wartościami. Proszę w sprawozdaniu załączyć potrzebne altery/select’y/update’y.

```
ALTER TABLE student_przedmiot ADD uczelnia_id int NULL

CREATE TRIGGER zadanie2_trigger ON student_przedmiot FOR UPDATE
AS
UPDATE student_przedmiot SET uczelnia_id = 1
WHERE student_id IN (SELECT i.student_id FROM INSERTED i)
```

3. Stworzona procedura, która po podaniu nazwy przedmiotu porówna jego popularność na wszystkich uczelniach. W tym zadaniu trzeba połączyć trzy tabeli *dbo.przedmoit*, *dbo.student*, *dbo.uczelnia*, po połączeniu trzech tabel, trzeba pogrupować za pomocą polecenia *GROUP BY przedmiot i uczelnia* po tym trzeba zrobić *COUNT(\*)przedmiot i COUNT(\*) uczelnia*. To jest rozwiązanie tego zadania.

```
CREATE PROCEDURE zadanie3 @nazwa_przedmiotu nvarchar(40)
AS
BEGIN
    SELECT u.nazwa
    FROM student s
        JOIN uczelnia u ON s.uczelnia_id = u.ID
        JOIN przedmiot p ON s.uczelnia_id = p.ID
    GROUP BY u.nazwa
    SELECT COUNT(*) FROM uczelnia
    SELECT COUNT(*) FROM przedmiot

    INSERT INTO przedmiot ( nazwa )
    VALUES ('matematyka');
```

END

# CALY KOD:

```
CREATE PROCEDURE zadaine1 @imie nvarchar(20), @nazwisko nvarchar(20), @nazwa_uczelni
nvarchar(20), @przedmiot nvarchar(40)
AS
BEGIN
    BEGIN TRY
        INSERT INTO student_przedmiot ( student_id, przedmiot_id )
        VALUES ( (SELECT nr_albumu, imie, nazwisko FROM INSERTED) ,
        (SELECT 1 + CONVERT(INT, ((SELECT COUNT(*) FROM przedmiot)) * (3 +
przedmiot.ID))),
        (SELECT 1 + CONVERT(INT, ((SELECT COUNT(*) FROM uczelnia)) * (uczelnia.ID)))
        )
        print 'Student zostal zapisany do uczelni i do 3 przedmiotow';
    END TRY
    BEGIN CATCH
        IF @nazwa_uczelni IS NULL
            print 'Nie isntenie takiej uczelni';
        ELSE IF @przedmiot IS NULL
            print 'Nie podano trzech przedmiotow';
        print ERROR_MESSAGE()
        RETURN 0
    END CATCH
END
```

```
ALTER TABLE student_przedmiot ADD uczelnia_id int NULL
CREATE TRIGGER zadanie2_trigger ON student_przedmiot FOR UPDATE
AS
UPDATE student_przedmiot SET uczelnia_id = 1
WHERE student_id IN (SELECT i.student_id FROM INSERTED i)
```

```
CREATE PROCEDURE zadanie3 @nazwa_przedmiotu nvarchar(40)
AS
BEGIN
    SELECT u.nazwa
    FROM student s
    JOIN uczelnia u ON s.uczelnia_id = u.ID
    JOIN przedmiot p ON s.uczelnia_id = p.ID
    GROUP BY u.nazwa
    SELECT COUNT(*) FROM uczelnia
    SELECT COUNT(*) FROM przedmiot
    INSERT INTO przedmiot ( nazwa )
    VALUES ('matematyka');
END
```