**University Course Management System: A Story**

At Elmwood University, the administration faced challenges managing its ever-growing student body, course offerings, and academic records. To streamline operations and ensure data integrity, they decided to develop a robust database management system. The database has been created with the appropriate structure, but the data is getting bad very quickly! You have been hired as a consultant for $$$$$$$ to add constraints to the database.

**The Student Table**

The first step in this endeavor was to organize the student records. Every student at Elmwood University was given a unique student ID, a number that distinguished them from their peers. The administration also wanted to make sure that each student had their email addresses stored uniquely in the database to avoid any confusion in communication. Along with these details, each student's name and date of birth were also recorded.

**The Course Table**

Next, the focus shifted to the courses offered by the university. Each course needed course ID, a name, and a specified number of credits. The administration was adamant that the credits for any course should be between 0 and 5, reflecting the academic value of the course.

**The Enrollment Table**

With students and courses now neatly organized, it was essential to record the enrollments—detailing which students were taking which courses. Each enrollment was given a unique enrollment ID. The Enrollment table needed to utilize both the Student and Course tables to ensure that there were no phantom enrollments. Additionally, the grades should only fall in the values 'A', 'B', 'C', 'D', or 'F'.

**The Instructor Table**

The university also needed to keep track of its faculty. Each instructor was assigned a unique instructor ID, and their names, email addresses, and departments were recorded. As with students, each instructor's email had to be unique to prevent any mix-ups.

**The CourseInstructor Table**

Finally, the university needed to document which instructors were teaching which courses. The CourseInstructor table was created for this purpose. It linked the course ID from the Course table and the instructor ID from the Instructor table, ensuring that each instructor could be assigned to multiple courses and each course could have multiple instructors. To prevent instructors from getting more credit than they deserve, the university wants to make sure that the combination of instructor ID and course ID is unique.