# LU06.A03 - SQL-DDL: Constraint Management

## Requirements

- Work type: individual
- Timeframe: 30 Minutes
- Means of aid:
  - Only teaching materials, no websearch, no use of ai.
  - W3Schools | SQL Editor
- Expected result: Semantically and syntactically correct SQL statements according to the requirements of the case studies.

## Case studies / Assignments

Here are five assignments, each covering a specific MySQL constraint. As usual, along with the solutions at the end (link).

## Assignments

### A: PRIMARY KEY

Create a table books that has a book_id as a unique identifier for each book, with book_id as the primary key. Include columns for book_title (VARCHAR) and author_name (VARCHAR). The book_id should be an integer and cannot be NULL.

### B: FOREIGN KEY

Create two tables: one called departments and the other called employees. Each department has a department_id as its primary key. In the employees table, include a column called department_id as a foreign key that references the departments table. Ensure that every employee is linked to a department.

### C: NOT NULL

Create a table students that includes a student_id (INT) and a student_name (VARCHAR). Ensure that the student_name column cannot have a NULL value by applying the NOT NULL constraint.

### D: AUTO INCREMENT

Create a table products where each product has an automatically generated, unique product_id using the AUTO_INCREMENT feature. Include columns for product_name and price.

### E: UNIQUE

Create a table users that has a user_id (INT) and email (VARCHAR). Ensure that no two users can have the same email address by applying the UNIQUE constraint to the email column.

## Solution

Lösung

## Vocabulary

| English | German |
|---|---|
| … | … |
| … | … |

---