1. SQL to create three tables:

CREATE TABLE Students

( studentId INT AUTO\_INCREMENT,

stName VARCHAR(30) NOT NULL,

email VARCHAR(40),

gpa FLOAT DEFAULT 0.0,

PRIMARY KEY (studentId),

CONSTRAINT CHECK\_GPA

CHECK (gpa BETWEEN 0 AND 4.3)

);

CREATE TABLE Courses

( courseId INT AUTO\_INCREMENT,

courseNumber VARCHAR(10) NOT NULL,

courseName VARCHAR(50) NOT NULL,

courseDescription VARCHAR(200),

creditHours INT DEFAULT 3,

PRIMARY KEY (courseId),

CONSTRAINT CHECK\_CREDIT\_HOUR

CHECK (creditHours BETWEEN 0 AND 6)

);

CREATE TABLE Enrollments

( studentId INT,

courseId INT,

letterGrade CHAR(2),

PRIMARY KEY (studentId, courseId),

FOREIGN KEY (studentId)

REFERENCES Students(studentId)

ON DELETE NO ACTION

ON UPDATE CASCADE,

FOREIGN KEY (courseId)

REFERENCES Courses(courseId)

ON DELETE NO ACTION

ON UPDATE CASCADE

);

1. Create Students table:

文本

描述已自动生成

1. Create Courses table:

文本

描述已自动生成

1. Create Enrollments table:

文本

描述已自动生成

1. Test Students table
2. test student gpa constraint:

INSERT INTO Students (stName, email, gpa)

VALUE ('hui', '456@unb.ca', 5.0);

表格

描述已自动生成

1. test student not null constraint:

INSERT INTO Students (stName, email, gpa)

VALUE (NULL, '456@unb.ca', 2.0);

文本

低可信度描述已自动生成

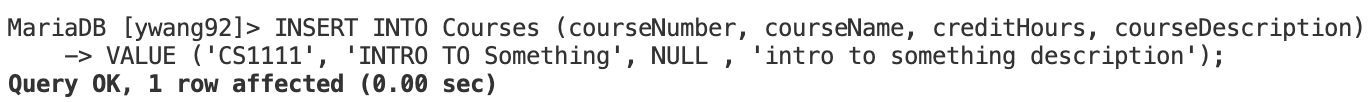
1. Test Courses table:
2. test creditHours constraint:

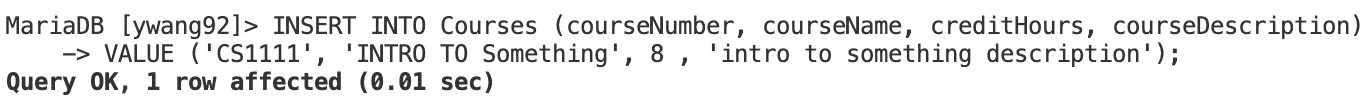
INSERT INTO Courses (courseNumber, courseName, creditHours, courseDescription)

VALUE ('CS1111', 'INTRO TO Something', NULL , 'intro to something description');

INSERT INTO Courses (courseNumber, courseName, creditHours, courseDescription)

VALUE ('CS1111', 'INTRO TO Something', 8 , 'intro to something description');





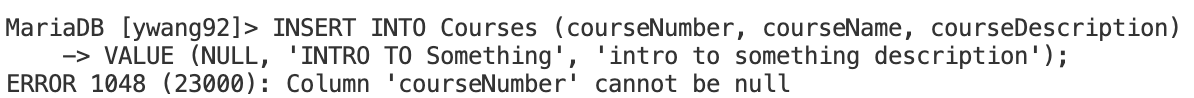
1. test course not null constraint:

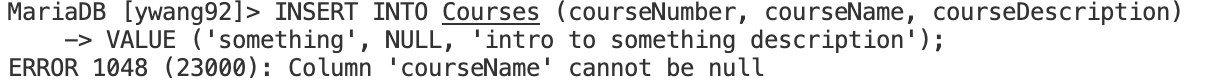
INSERT INTO Courses (courseNumber, courseName, courseDescription)

VALUE (NULL, 'INTRO TO Something', 'intro to something description');

INSERT INTO Courses (courseNumber, courseName, courseDescription)

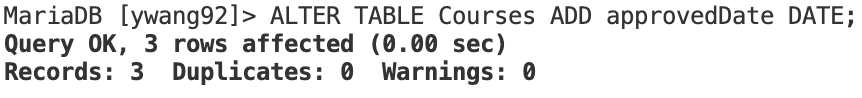
VALUE ('something', NULL, 'intro to something description');





1. Add an approvedDate attribute to the Course table:

ALTER TABLE Courses ADD approvedDate DATE;



1. Use Insert (or Update) statements to add (or update) data to the table with the extra attribute:

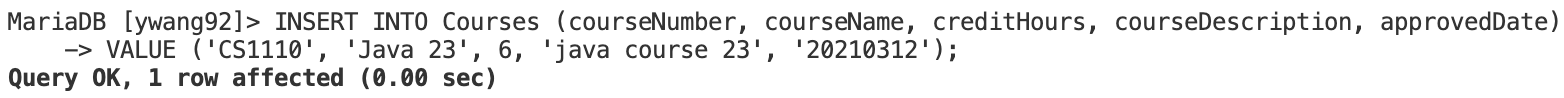
INSERT INTO Courses (courseNumber, courseName, creditHours, courseDescription, approvedDate)

VALUE ('CS1110', 'Java 23', 6, 'java course 23', '20210312');

UPDATE Courses

SET approvedDate = '20210520'

WHERE courseId = 2;



文本

描述已自动生成

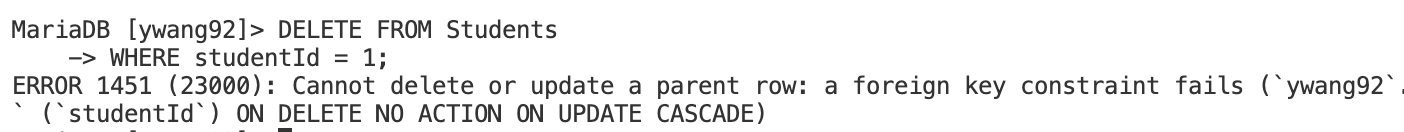
表格

描述已自动生成

1. test deleting student that has enrolled in a course

DELETE FROM Students

WHERE studentId = 1;



1. for each student lists the information for courses they’ve taken and their letter grade

select \* from Courses;

select \* from Students;

select \* from Enrollments;

SELECT \*

FROM Students AS A

LEFT JOIN Enrollments AS B ON A.studentId = B.studentId

LEFT JOIN Courses AS C ON B.courseId = C.courseId;

表格

描述已自动生成

表格

描述已自动生成

表格

描述已自动生成

表格

描述已自动生成