promt: Ok, till now I have learned many things by taking instructions from you, now test me. From what I've learned, you can tell me to build on any database that's hard.

**ChatGPT said:**

ChatGPT:

Great! Let's put your skills to the test with a more challenging task. Here's what I want you to do:

**Task: Create a Library Management System Database**

You'll need to build a database for managing books, students, and book lending/return activities in a library. Here's what you need to do:

**1. Create Tables**

* **Students** table:
  + student\_id: INT (Auto Increment, Primary Key)
  + name: VARCHAR(50)
  + email: VARCHAR(100) (Ensure it contains '@gmail.com')
  + phone: CHAR(11) (Unique constraint)
  + department: VARCHAR(50)
* **Books** table:
  + book\_id: INT (Auto Increment, Primary Key)
  + title: VARCHAR(100)
  + author: VARCHAR(50)
  + purchase\_date: DATE
  + available: BOOLEAN (True if available, False if lent out)
* **Lendings** table (Tracks which student borrowed which book and when):
  + lending\_id: INT (Auto Increment, Primary Key)
  + student\_id: INT (Foreign Key from Students table)
  + book\_id: INT (Foreign Key from Books table)
  + borrow\_date: DATE (Default to current date)
  + return\_date: DATE (Can be NULL if the book hasn’t been returned yet)

**2. Constraints**

* Ensure email in the Students table must contain @gmail.com.
* Use AUTO\_INCREMENT for IDs.
* Add UNIQUE constraints where necessary.
* Use a FOREIGN KEY to link the student\_id and book\_id in the Lendings table to their respective tables.

**3. Insert Data**

* Insert at least 3 students, 3 books, and 2 lending records (one of the books should be lent out, so its available status will be False).

**4. Queries**

* **Query 1**: Get a list of all students who have borrowed books along with the book titles.
* **Query 2**: Get a list of all books currently available.
* **Query 3**: Update a student's phone number.
* **Query 4**: Mark a book as returned (update return\_date and available status).

Let me know how it goes, and feel free to ask questions if you get stuck!

TasK:

**CREATE DATABASE CGPT;**

**USE CGPT;**

**CREATE** **TABLE** students

(

Student\_id **INT PRIMARY KEY,**

Name **VARCHAR(50),**

Email **CHAR(100),**

PHONE **CHAR(11),**

Department **VARCHAR(50)**

);

**CREATE** **TABLE** BOOKS

(

Book\_id **INT PRIMARY KEY,**

Title **VARCHAR(100),**

author **VARCHAR(50),**

Purchase\_date **DATE,**

available **BOOLEAN DEFAULT TRUE**

);

**CREATE** **TABLE** LENDINGS

(

lending\_id **INT PRIMARY KEY,**

Student\_id I**NT ,**

Book\_id **INT**,

Borrow\_date **DATETIME DEFAULT CURRENT\_TIMESTAMP,**

return\_date **DATE,**

**FOREIGN** **KEY**(student\_id) **REFERENCES** students(student\_id),

**FOREIGN** **KEY**(book\_id) **REFERENCES** books(Book\_id)

);

**SELECT \*FROM** STUDENTS;

**INSERT INTO** STUDENTS

**VALUES**(1,'AN Mamun','almamun20044@gmail.com','01702079049','CSE'),

(2,'Ashik','abc@gmail.com','01902086234','CSE'),

(3,'Jon','xyz@gmail.com','01902038979','EEE');

**SELECT** \***FROM BOOKS;**

**INSERT** **INTO** BOOKS (book\_id,title,author,purchase\_date)

**VALUES**(1,'GOOGLE','Sundar\_pichai',CURRENT\_DATE());

**INSERT INTO** BOOKS(book\_id,title,author,purchase\_date)

**VALUES**(2,'AMAZON','Andy\_Jassy',CURRENT\_DATE()),

(3,'Algorithms','AN Mamun',CURRENT\_DATE());

**START** **TRANSACTION**;

**SELECT** \***FROM** LENDINGS;

**INSERT** **INTO** LENDINGS(lending\_id,student\_id,book\_id,borrow\_date,return\_date)

**VALUES**(1,2,1,**CURRENT\_DATE(),DATE\_ADD(CURRENT\_DATE(),INTERVAL 7 DAY**));

**SELECT DISTINCT** book\_id **FROM** lendings;

**SELECT \*FROM** BOOKS;

**SET SQL\_SAFE\_UPDATES = 0;**

**UPDATE** BOOKS

**SET** available = **FALSE**

**WHERE** book\_id = 1;

**SET SQL\_SAFE\_UPDATES = 1;**

**COMMIT;**