SQL LAB – 1

RDBMS,MYSQL

NAME: Keerthana K R

ID: AF0363623

QUESTIONS

Create a database with the name StudentManagementSystem.

1. Create a table with named Student with attributes:
   - StudentID (Primary Key)
   - FirstName
   - LastName
   - DateOfBirth
   - Gender
   - Email
   - Phone

2. Create a table with name Course with attributes:
   - CourseID (Primary Key)
   - CourseTitle
   - Credits

3. Create a table with named Instructor with attributes:
   - InstructorID (Primary Key)
   - FirstName
   - LastName
   - Email

4. Create a table with named Enrollment with attributes:
   - EnrollmentID (Primary Key)
   - EnrollmentDate
   - StudentID(Foreign key)
   - CourseID(Foreign Key)
   - InstructorID(Foreign key)

5. Create a table with named Score with attributes:
   - ScoreID (Primary Key)
   - CourseID (Foreign key)
   - StudentID (Foreign Key)
   - DateOfExam
   - CreditObtained

6. Create a table with named Feedback with attributes:
   - FeedbackID (Primary Key)
   - StudentID (Foreign key)
   - Date
   - InstructorName
   - Feedback

ChatGPT Exercise

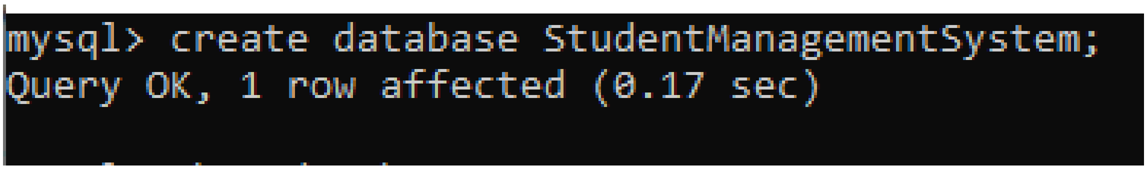
Using ChatGPT generate the Database design

Scenario: Implementing Database Design

The database should store emergency contact information for each employee. This information is crucial for situations where immediate contact with family or emergency contacts is necessary. The design should consider privacy and security measures for sensitive contact details.

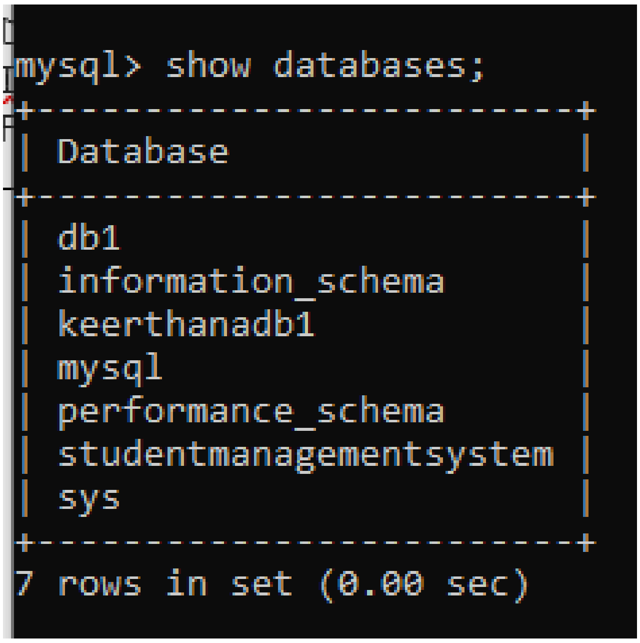
Create a database with the name StudentManagementSystem.

Code :



```
mysql> create database StudentManagementSystem;
Query OK, 1 row affected (0.17 sec)
```

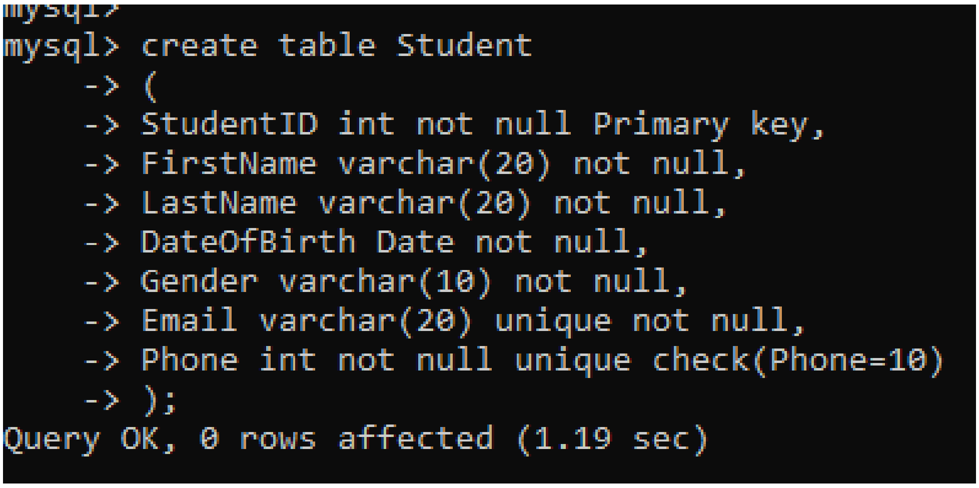
Output :



```
mysql> show databases;
+--------------------------+
| Database                 |
+--------------------------+
| db1                      |
| information_schema       |
| keerthanadb1             |
| mysql                    |
| performance_schema       |
| studentmanagementsystem  |
| sys                      |
+--------------------------+
7 rows in set (0.00 sec)
```

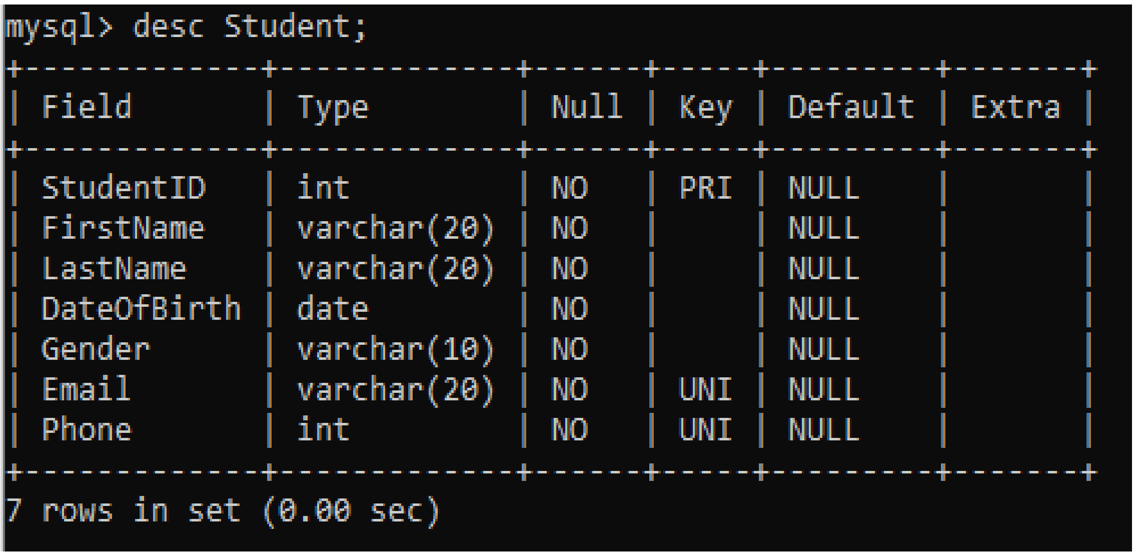
1. Create a table with named Student with attributes: StudentID (Primary Key), FirstName, LastName, DateOfBirth, Gender, Email, Phone

Code :



```
mysql>
mysql> create table Student
    -> (
    -> StudentID int not null Primary key,
    -> FirstName varchar(20) not null,
    -> LastName varchar(20) not null,
    -> DateOfBirth Date not null,
    -> Gender varchar(10) not null,
    -> Email varchar(20) unique not null,
    -> Phone int not null unique check(Phone=10)
    -> );
Query OK, 0 rows affected (1.19 sec)
```

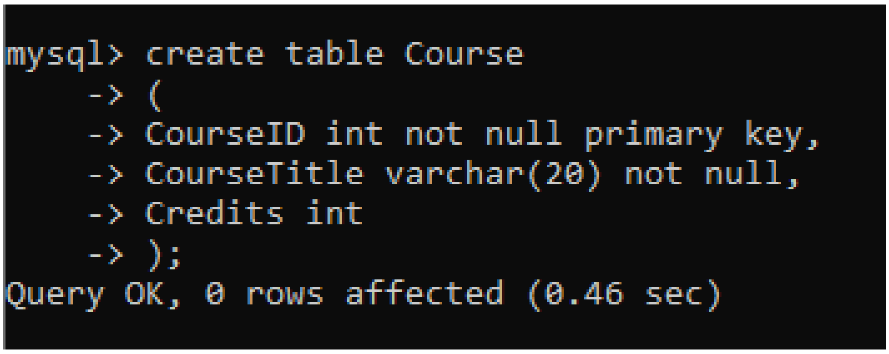
Output :



```
mysql> desc Student;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| StudentID   | int         | NO   | PRI | NULL    |       |
| FirstName   | varchar(20) | NO   |     | NULL    |       |
| LastName    | varchar(20) | NO   |     | NULL    |       |
| DateOfBirth | date        | NO   |     | NULL    |       |
| Gender      | varchar(10) | NO   |     | NULL    |       |
| Email       | varchar(20) | NO   | UNI | NULL    |       |
| Phone       | int         | NO   | UNI | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
7 rows in set (0.00 sec)
```

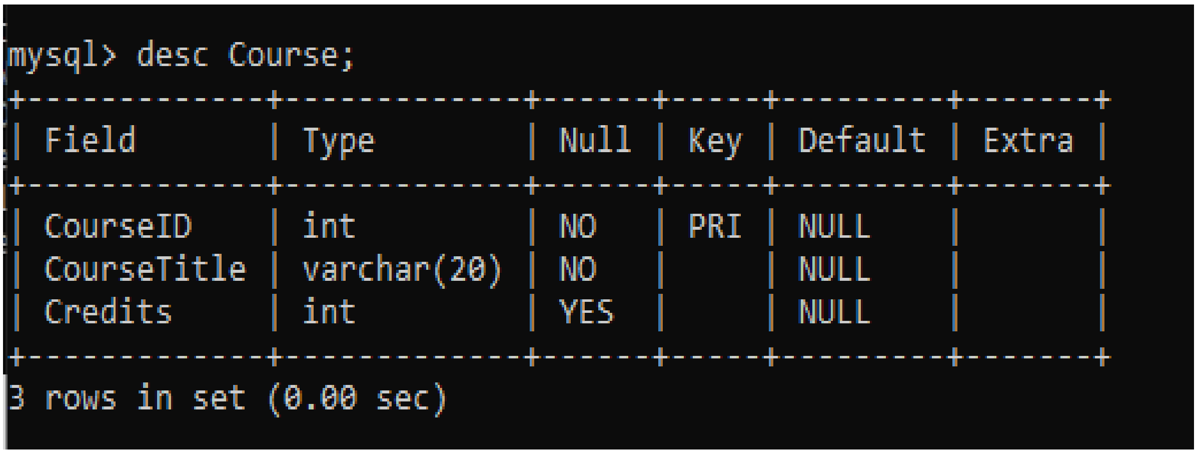
2. Create a table with name Course with attributes:
    - CourseID (Primary Key)
    - CourseTitle
    - Credits

Code ;



```
mysql> create table Course
    -> (
    -> CourseID int not null primary key,
    -> CourseTitle varchar(20) not null,
    -> Credits int
    -> );
Query OK, 0 rows affected (0.46 sec)
```

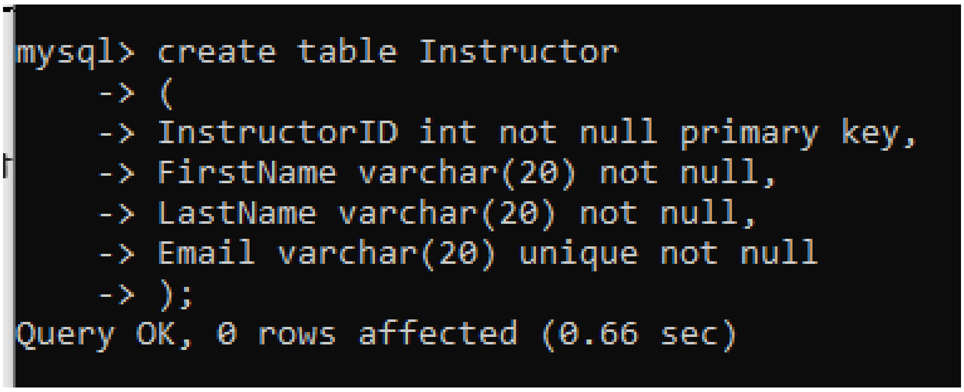
Output :



```
mysql> desc Course;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| CourseID    | int         | NO   | PRI | NULL    |       |
| CourseTitle | varchar(20) | NO   |     | NULL    |       |
| Credits     | int         | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

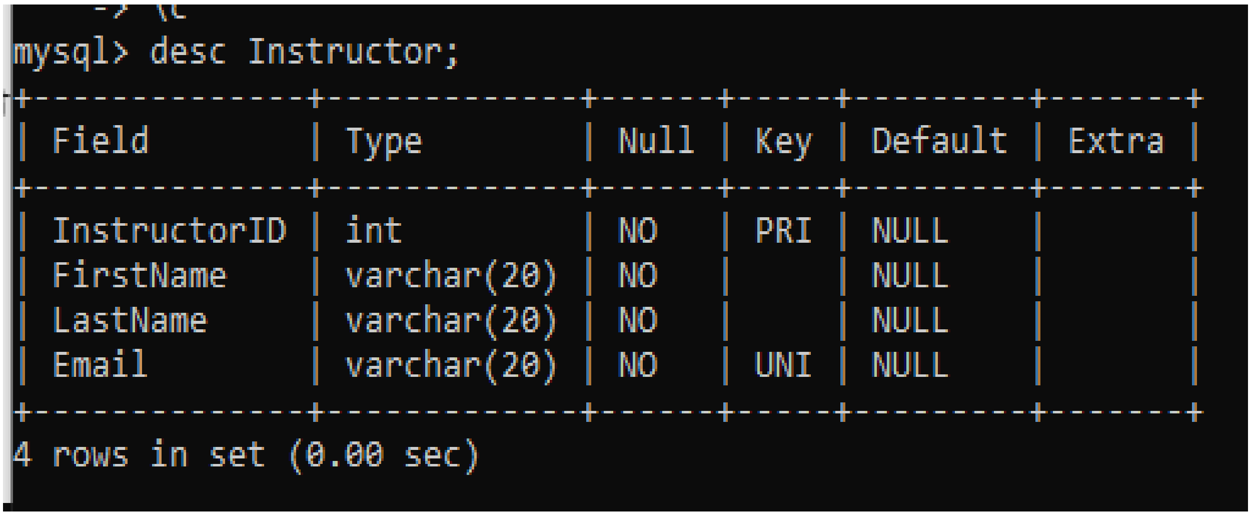
3. Create a table with named Instructor with attributes:
   - InstructorID (Primary Key)
   - FirstName
   - LastName
   - Email

Code :



```
mysql> create table Instructor
    -> (
    -> InstructorID int not null primary key,
    -> FirstName varchar(20) not null,
    -> LastName varchar(20) not null,
    -> Email varchar(20) unique not null
    -> );
Query OK, 0 rows affected (0.66 sec)
```

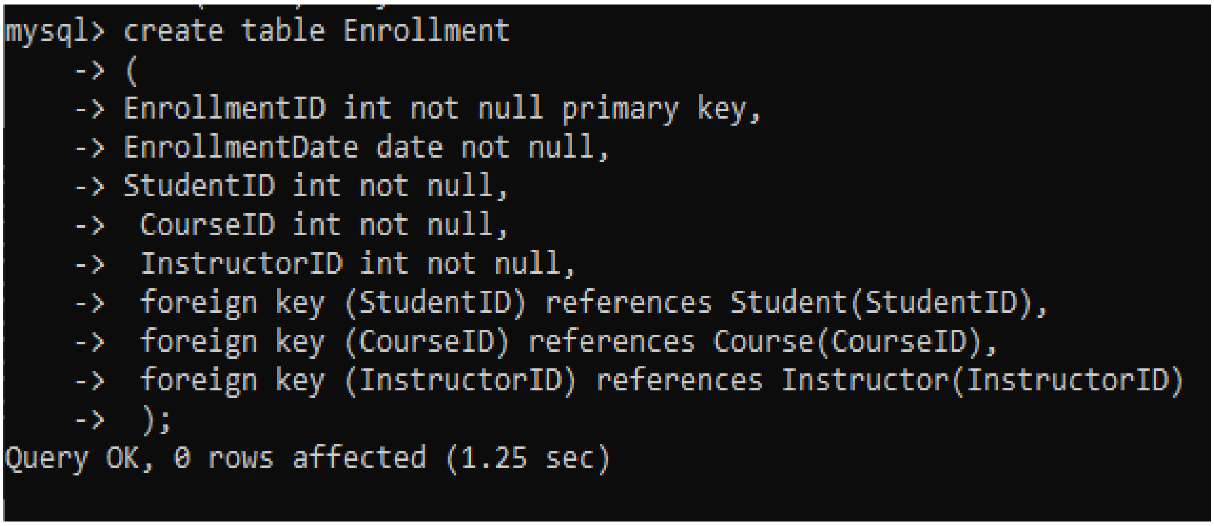
Output :



```
mysql> desc Instructor;
+--------------+-------------+------+-----+---------+-------+
| Field        | Type        | Null | Key | Default | Extra |
+--------------+-------------+------+-----+---------+-------+
| InstructorID | int         | NO   | PRI | NULL    |       |
| FirstName    | varchar(20) | NO   |     | NULL    |       |
| LastName     | varchar(20) | NO   |     | NULL    |       |
| Email        | varchar(20) | NO   | UNI | NULL    |       |
+--------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

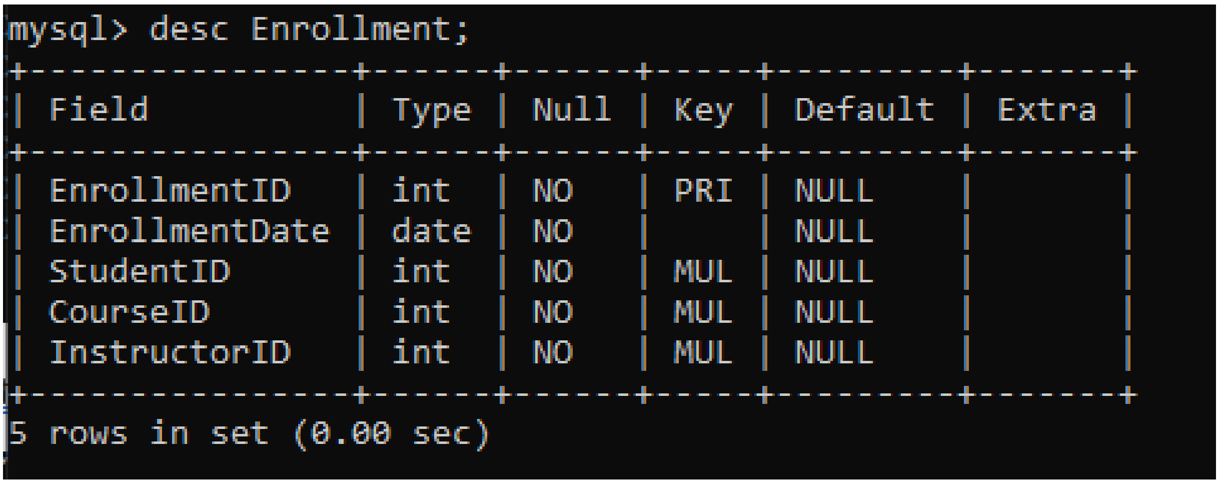
4. Create a table with named Enrollment with attributes:
    - EnrollmentID (Primary Key)
    - EnrollmentDate
    - StudentID(Foreign key)
    - CourseID(Foreign Key)
    - InstructorID(Foreign key)

Code :



```
mysql> create table Enrollment
    -> (
    -> EnrollmentID int not null primary key,
    -> EnrollmentDate date not null,
    -> StudentID int not null,
    ->  CourseID int not null,
    ->  InstructorID int not null,
    ->  foreign key (StudentID) references Student(StudentID),
    ->  foreign key (CourseID) references Course(CourseID),
    ->  foreign key (InstructorID) references Instructor(InstructorID)
    ->  );
Query OK, 0 rows affected (1.25 sec)
```

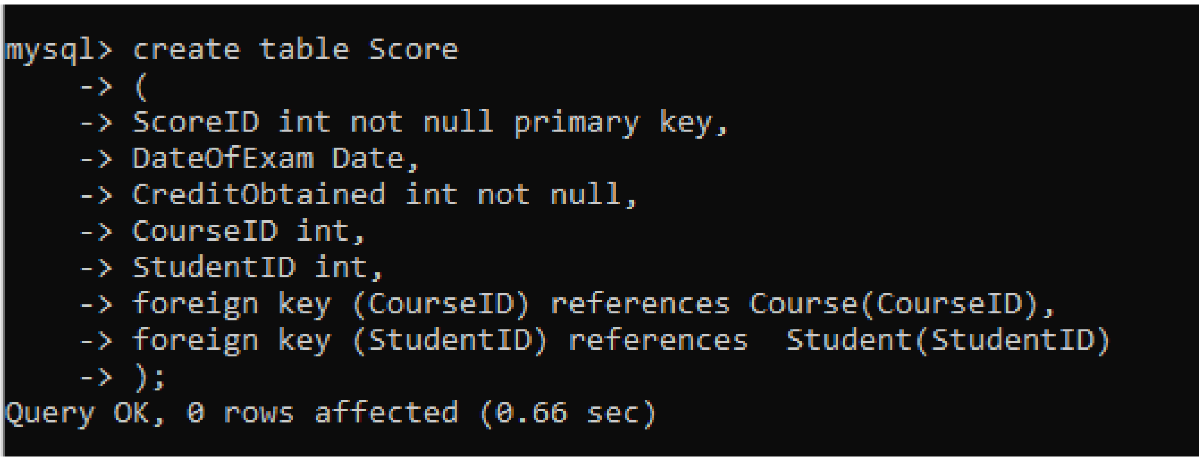
Output :



```
mysql> desc Enrollment;
+----------------+------+------+-----+---------+-------+
| Field          | Type | Null | Key | Default | Extra |
+----------------+------+------+-----+---------+-------+
| EnrollmentID   | int  | NO   | PRI | NULL    |       |
| EnrollmentDate | date | NO   |     | NULL    |       |
| StudentID      | int  | NO   | MUL | NULL    |       |
| CourseID       | int  | NO   | MUL | NULL    |       |
| InstructorID   | int  | NO   | MUL | NULL    |       |
+----------------+------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

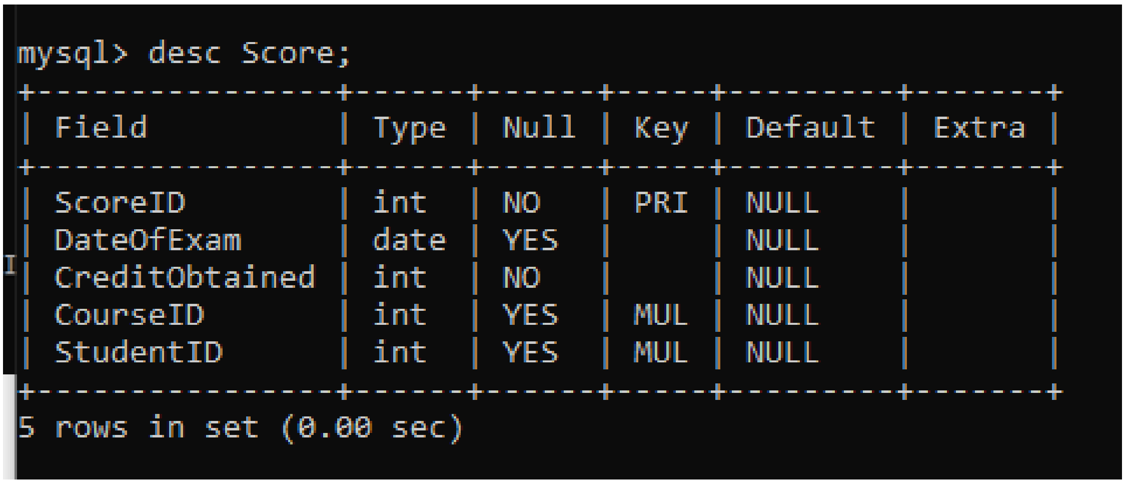
5. Create a table with named Score with attributes:
    - ScoreID (Primary Key)
    - CourseID (Foreign key)
    - StudentID (Foreign Key)
    - DateOfExam
    - CreditObtained

Code:



```
mysql> create table Score
    -> (
    -> ScoreID int not null primary key,
    -> DateOfExam Date,
    -> CreditObtained int not null,
    -> CourseID int,
    -> StudentID int,
    -> foreign key (CourseID) references Course(CourseID),
    -> foreign key (StudentID) references  Student(StudentID)
    -> );
Query OK, 0 rows affected (0.66 sec)
```

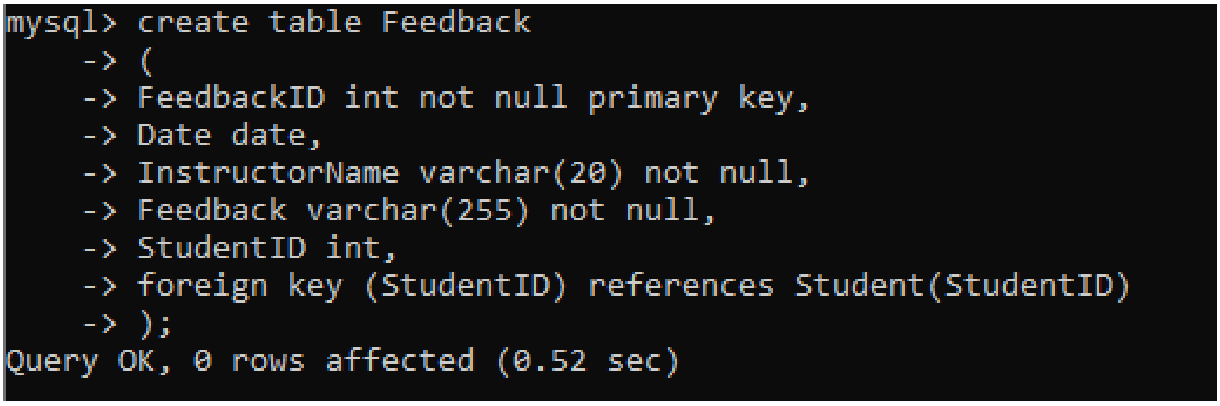
Output;



```
mysql> desc Score;
+----------------+------+------+-----+---------+-------+
| Field          | Type | Null | Key | Default | Extra |
+----------------+------+------+-----+---------+-------+
| ScoreID        | int  | NO   | PRI | NULL    |       |
| DateOfExam     | date | YES  |     | NULL    |       |
| CreditObtained | int  | NO   |     | NULL    |       |
| CourseID       | int  | YES  | MUL | NULL    |       |
| StudentID      | int  | YES  | MUL | NULL    |       |
+----------------+------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

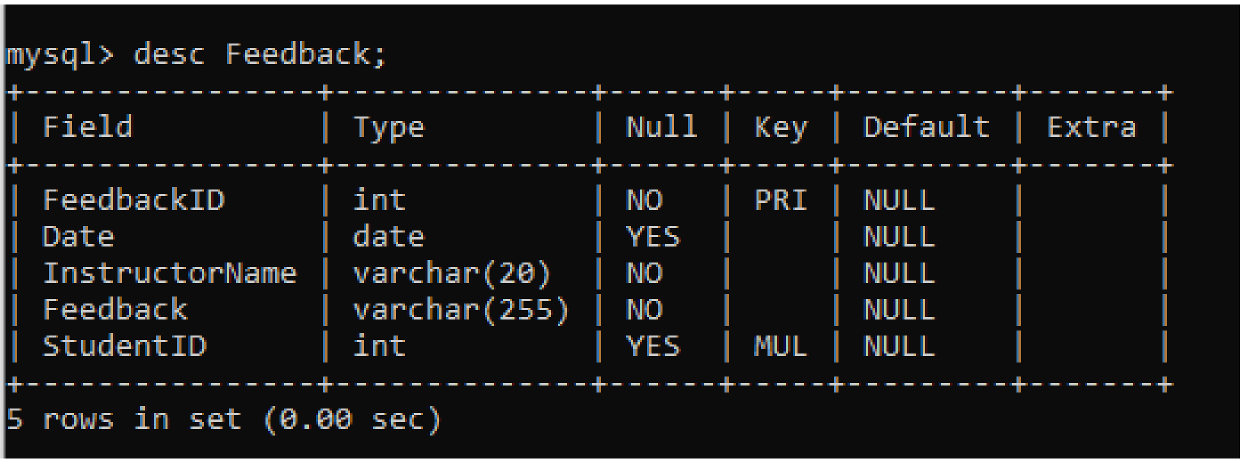
6. Create a table with named Feedback with attributes:
   - FeedbackID (Primary Key)
   - StudentID (Foreign key)
   - Date
   - InstructorName
   - Feedback

Code :



```
mysql> create table Feedback
    -> (
    -> FeedbackID int not null primary key,
    -> Date date,
    -> InstructorName varchar(20) not null,
    -> Feedback varchar(255) not null,
    -> StudentID int,
    -> foreign key (StudentID) references Student(StudentID)
    -> );
Query OK, 0 rows affected (0.52 sec)
```

Output :

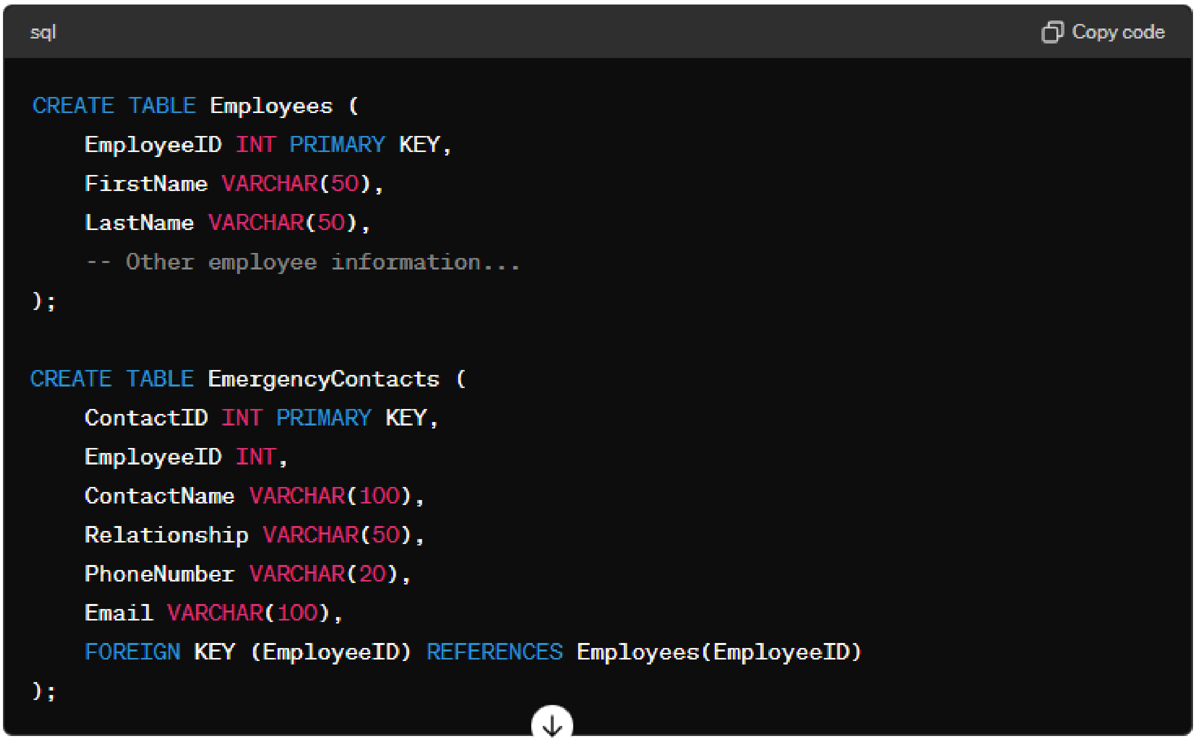


```
mysql> desc Feedback;
+----------------+--------------+------+-----+---------+-------+
| Field          | Type         | Null | Key | Default | Extra |
+----------------+--------------+------+-----+---------+-------+
| FeedbackID     | int          | NO   | PRI | NULL    |       |
| Date           | date         | YES  |     | NULL    |       |
| InstructorName | varchar(20)  | NO   |     | NULL    |       |
| Feedback       | varchar(255) | NO   |     | NULL    |       |
| StudentID      | int          | YES  | MUL | NULL    |       |
+----------------+--------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

ChatGPT exercise

Scenario: Implementing Database Design

The database should store emergency contact information for each employee. This information is crucial for situations where immediate contact with family or emergency contacts is necessary. The design should consider privacy and security measures for sensitive contact details.



```sql
CREATE TABLE Employees (
    EmployeeID INT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    -- Other employee information...
);

CREATE TABLE EmergencyContacts (
    ContactID INT PRIMARY KEY,
    EmployeeID INT,
    ContactName VARCHAR(100),
    Relationship VARCHAR(50),
    PhoneNumber VARCHAR(20),
    Email VARCHAR(100),
    FOREIGN KEY (EmployeeID) REFERENCES Employees(EmployeeID)
);
```