Assignment_2_Question_2

```
import mysql.connector
conn = mysql.connector.connect(host='localhost', password='Aby@0210',
user='root', database = "db_1")
mycursor=conn.cursor()
Table_1="""
create table if not exists Student(
    StudentID int PRIMARY KEY,
    Name varchar(255),
    Email varchar(255),
    Phone varchar(255),
    Address text
)
"""
Table_2="""
create table if not exists Course(
    CourseID int PRIMARY KEY,
    CourseName varchar(255),
    Credits int
)
"""
Table_3="""
create table if not exists Exam(
    ExamID int PRIMARY KEY,
    ExamDate Date,
    ExamTime Time,
    Location varchar(255)
)
"""
Table_4="""
create table if not exists Faculty(
    FacultyID int PRIMARY KEY,
    Name varchar(255),
    Email varchar(255),
    Phone varchar(20),
    Department varchar(255)
)
"""
Table_5="""
create table if not exists Enrollment(
    EnrollmentID int PRIMARY KEY,
    StudentID int,
    CourseID int,
    EnrollmentDate date,
    Foreign key(StudentID) references Student(StudentID),
    Foreign key(CourseID) references Course(CourseID)
)
"""
```

```
Table_6="""
create table if not exists Teaching(
    TeachingID int PRIMARY KEY,
    FacultyID int,
    CourseID int,
    Foreign key(FacultyID) references Faculty(FacultyID),
    Foreign key(CourseID) references Course(CourseID)
)
"""
Table_7="""
create table if not exists ExamRegistration(
    RegistrationID int PRIMARY KEY,
    StudentID int,
    ExamID int,
    RegistrationDate Date,
    Foreign key(StudentID) references Student(StudentID),
    Foreign key(ExamID) references Exam(ExamID)
)
"""
Table_8="""
create table if not exists ExamResults(
    ResultID int PRIMARY KEY,
    StudentID int,
    ExamID int,
    Score Decimal(5,2),
    Foreign key(StudentID) references Student(StudentID),
    Foreign key(ExamID) references Exam(ExamID)
)
"""

mycursor.execute(Table_1)
mycursor.execute(Table_2)
mycursor.execute(Table_3)
mycursor.execute(Table_4)
mycursor.execute(Table_5)
mycursor.execute(Table_6)
mycursor.execute(Table_7)
mycursor.execute(Table_8)
conn.commit()
conn.close()
```