Language used

Python

Approach

Import all the required modules - **SQLITE3**

Create classes - **Student , Course**

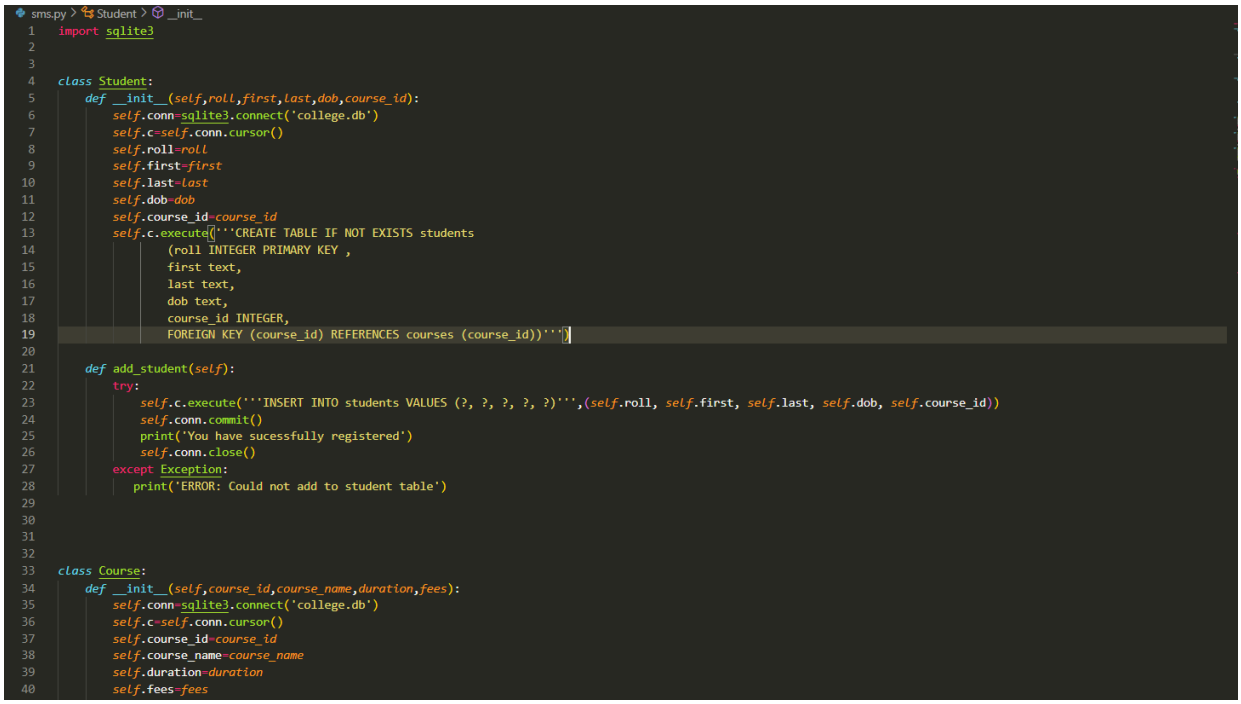
Created necessary functions for student - **Register, View Courses, Apply for a course**

Created necessary functions for Admin - **Add new course , View Students , View Courses**

Created driver function to run the apllication

Source code

SMS.py



```python
import sqlite3


class Student:
    def __init__(self,roll,first,last,dob,course_id):
        self.conn=sqlite3.connect('college.db')
        self.c=self.conn.cursor()
        self.roll=roll
        self.first=first
        self.last=last
        self.dob=dob
        self.course_id=course_id
        self.c.execute('''CREATE TABLE IF NOT EXISTS students
                (roll INTEGER PRIMARY KEY ,
                first text,
                last text,
                dob text,
                course_id INTEGER,
                FOREIGN KEY (course_id) REFERENCES courses (course_id))''')

    def add_student(self):
        try:
            self.c.execute('''INSERT INTO students VALUES (?, ?, ?, ?, ?)''',(self.roll, self.first, self.last, self.dob, self.course_id))
            self.conn.commit()
            print('You have sucessfully registered')
            self.conn.close()
        except Exception:
            print('ERROR: Could not add to student table')



class Course:
    def __init__(self,course_id,course_name,duration,fees):
        self.conn=sqlite3.connect('college.db')
        self.c=self.conn.cursor()
        self.course_id=course_id
        self.course_name=course_name
        self.duration=duration
        self.fees=fees
```

sms.py > Student > __init__

```
        self.fees=fees
        self.c.execute('''CREATE TABLE IF NOT EXISTS courses
                    (course_id INTEGER PRIMARY KEY,
                    course_name text,
                    duration INTEGER,
                    fees INTEGER)''')

    def add_course(self):
        try:
            self.c.execute('''INSERT INTO courses(course_id,course_name,duration,fees) VALUES (?,?,?,?)''',(self.course_id,self.course_name,self.duration,self.fees))
            self.conn.commit()
            print('Sucessfully added course')
            self.conn.close()
        except Exception:
            print('ERROR: could not add to courses table')

def view_table_students():
    conn=sqlite3.connect('college.db')
    c=conn.cursor()
    c.execute("SELECT * FROM students")
    rows=c.fetchall()
    for row in rows:
        print(row)
    conn.close()

def view_table_courses():
    conn=sqlite3.connect('college.db')
    c=conn.cursor()
    c.execute("SELECT * FROM courses")
    rows=c.fetchall()
    for row in rows:
        print(row)
    conn.close()

def apply_to_course(roll,course_id):
    conn=sqlite3.connect('college.db')
    c=conn.cursor()
    c.execute('''UPDATE students SET course_id = ? WHERE roll = ?''',(course_id,roll))
    conn.commit()
    print('Successfully updated the course')
    conn.close()
    #except Exception:
     #   print('there war a problem updating')



if __name__=="__main__":
    print("Welcome to SMS (Student Managment System)")
    print('\nTell us who you are :\n 1.Student\n 2.Admin\n')
    choice=int(input('Enter your choice :'))

    if(choice==1):
        print('Welcome Student')
        print('Choose what to do')
        print('\n1.Register \n2.View courses \n3.Apply for course')
        choice = int(input())

        if (choice==1):
            print('Enter your Roll number : ')
            roll_num=int(input())
            print('Enter your first name :')
            first_name = input()
            print('Enter your last name : ')
            last_name = input()
            print('Enter your Date of Birth : ')
            DOB = input()
            print('Enter the course ID : ')
            Course_id=int(input())
            student = Student(roll_num,first_name,last_name,DOB,Course_id)
            student.add_student()

        elif(choice==2):
            view_table_courses()

        elif(choice==3):
            roll=int(input('Enter your roll : \n'))
            course_id=int(input('Enter the new course ID : \n'))
            apply_to_course(roll,course_id)
```

```python
sms.py > Student > __init__
119
120         else:
121             print('Inncorrect choice,Student. Try again')
122
123
124     elif(choice==2):
125         print('Welcome Admin')
126         print('tell us what to do\n 1.Add a new course to the list \n 2.View all students table\n 3.view all Courses table')
127         choice=int(input('tell us your choice Admin \n'))
128         if(choice==1):
129             print('Enter the Course ID : ')
130             course_id=int(input())
131             print('Enter the Course name :')
132             course_name = input()
133             print('Enter the course duration : ')
134             duration = input()
135             print('Enter the fees for the course : ')
136             fees = int(input())
137             course=Course(course_id,course_name,duration,fees)
138             course.add_course()
139         elif(choice==2):
140             view_table_students()
141         elif(choice==3):
142             view_table_courses()
143         else:
144             print('Incorrect choice, Admin. Try again')
145
146
147     else:
148         print('Incorrect choice, User. Try again')
```

outputs :

Registering as a student :

```
(base) PS D:\codes\web_development\case_study_3>  d:; cd 'd:\codes\web_development\case_study_3'; & 'C:\Users\Vatsal\anaconda3\python.exe' 'c:\Users\Vatsal\.vscode\extensions\ms-python.python-
2021.8.1105858891\pythonFiles\lib\python\debugpy\launcher' '55312' '--' 'd:\codes\web_development\case_study_3\sms.py'
Welcome to SMS (Student Managment System)

Tell us who you are :
 1.Student
 2.Admin

Enter your choice :1
Welcome Student
Choose what to do

1.Register
2.View courses
3.Apply for course
1
Enter your Roll number :
111
Enter your first name :
Kumar
Enter your last name :
Kaustubh
Enter your Date of Birth :
31/08/2003
Enter the course ID :
2
You have sucessfully registered
(base) PS D:\codes\web_development\case_study_3>
```

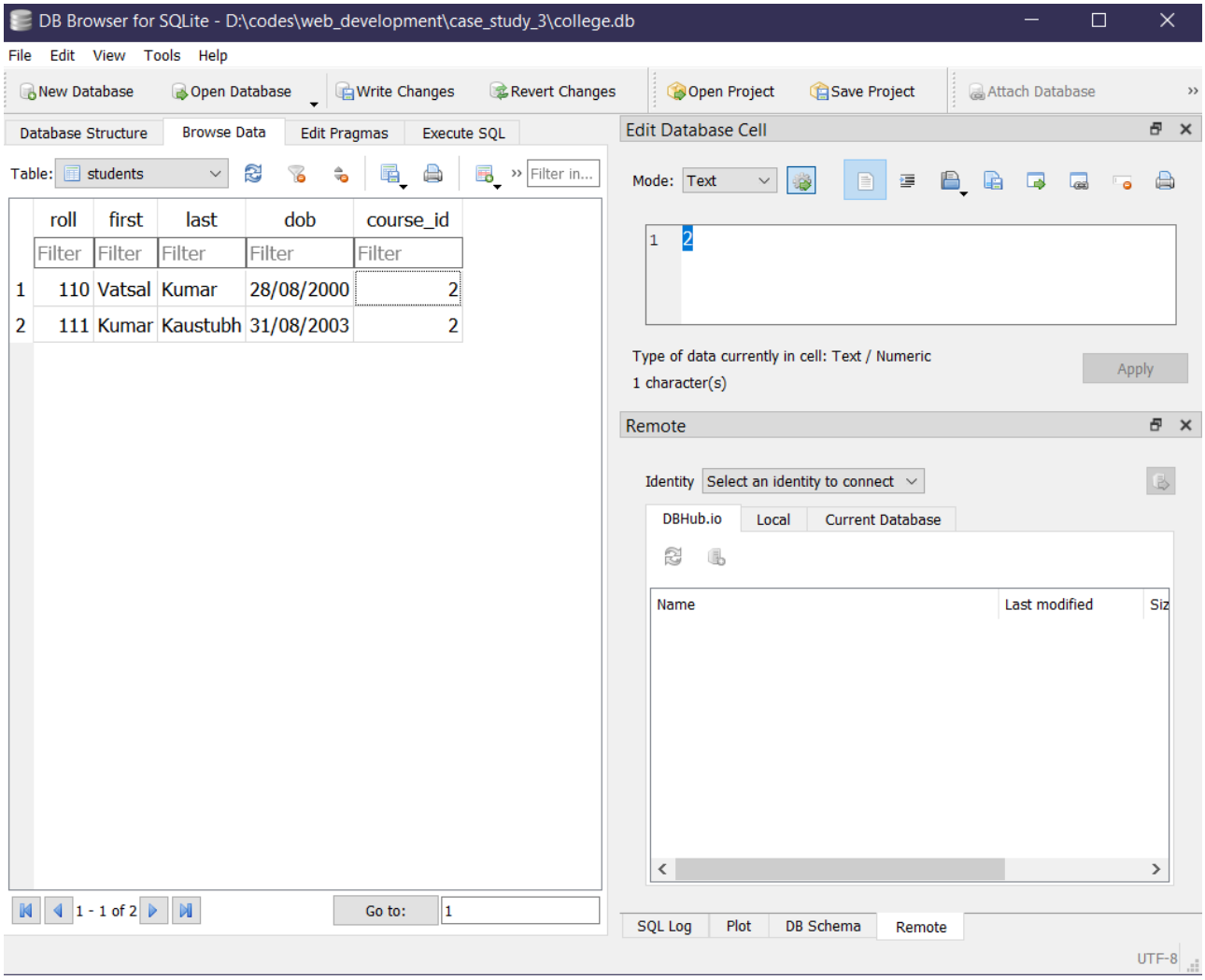
Viewing Courses as Student :

```
PROBLEMS 3   OUTPUT   DEBUG CONSOLE   TERMINAL

Tell us who you are :
 1.Student
 2.Admin

Enter your choice :1
Welcome Student
Choose what to do

1.Register
2.View courses
3.Apply for course
2
(1, 'physics', 120, 1000)
(2, 'chemistry', 120, 1000)
(base) PS D:\codes\web_development\case_study_3>
```

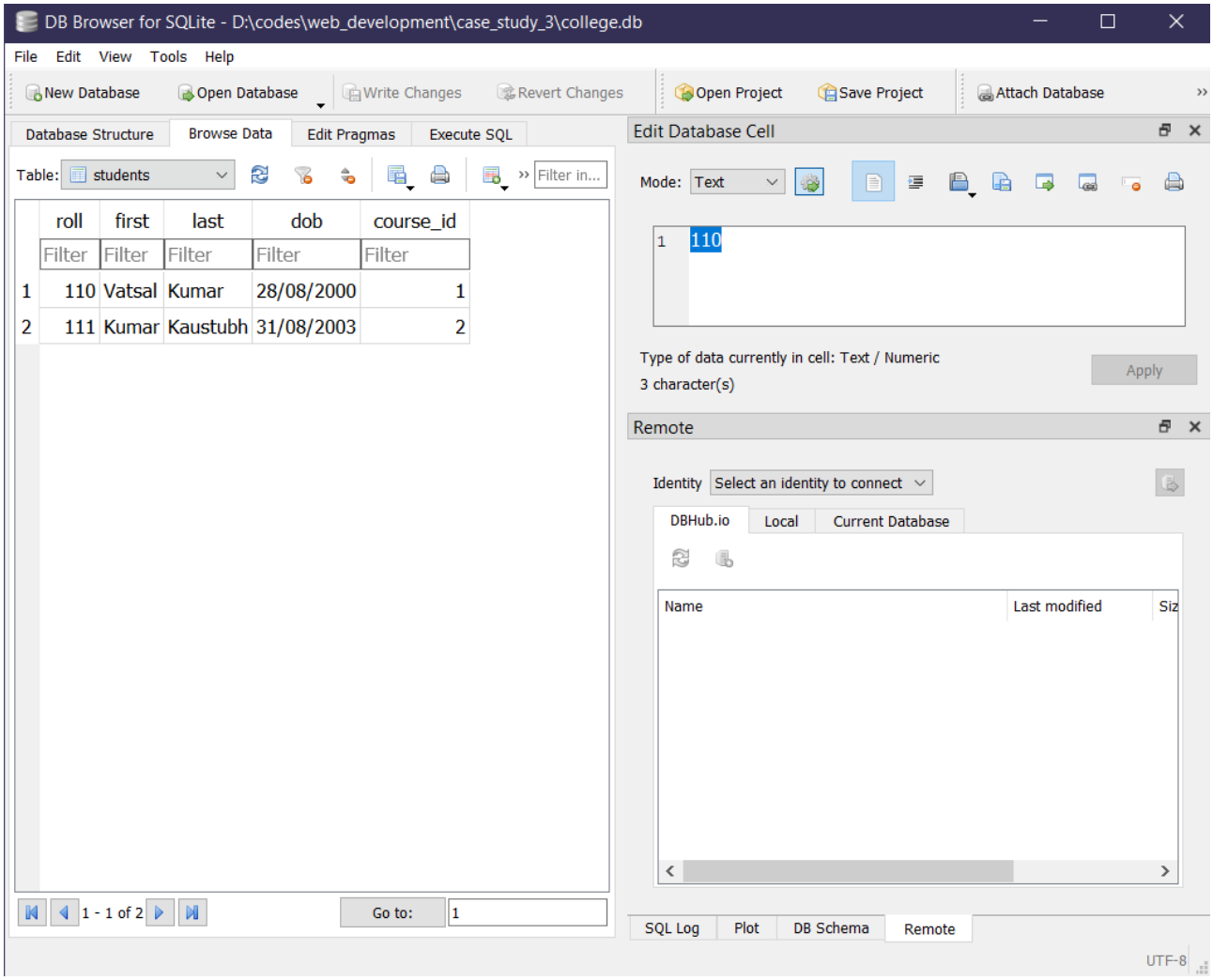
Applying for a new course :

```
PROBLEMS 3  OUTPUT  DEBUG CONSOLE  TERMINAL

2021.8.1105858891\pythonFiles\lib\python\debugpy\launcher' '55377' '--' 'd:\codes\web_development\case_study_3\sms.py'
Welcome to SMS (Student Managment System)

Tell us who you are :
 1.Student
 2.Admin

Enter your choice :1
Welcome Student
Choose what to do

1.Register
2.View courses
3.Apply for course
3
Enter your roll :
1
Enter the new course ID :
1
Successfully updated the course
(base) PS D:\codes\web_development\case_study_3> 
```

Results before Updation:



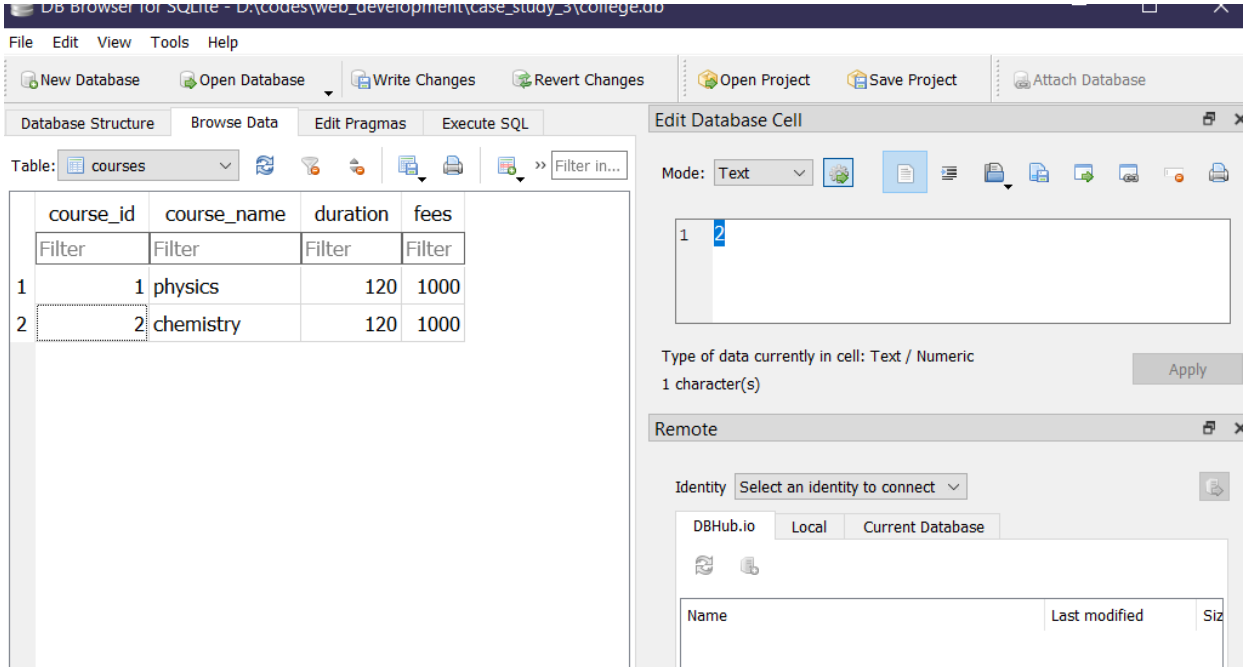
Changes after Updation :



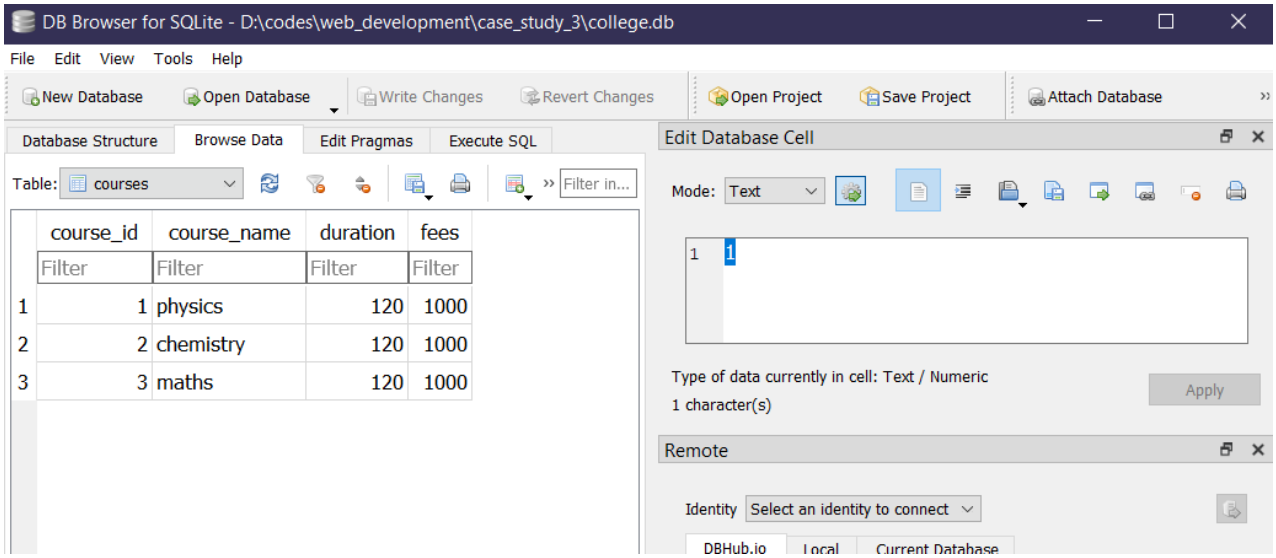
Admin adds a Course :

```
2021.8.1105858891\pythonFiles\lib\python\debugpy\launcher' '55751' '--' 'd:\codes\web_development\case_study_3\sms.py'
Welcome to SMS (Student Managment System)

Tell us who you are :
 1.Student
 2.Admin

Enter your choice :2
Welcome Admin
tell us what to do
 1.Add a new course to the list
 2.View all students table
 3.view all Courses table
tell us your choice Admin
1
Enter the Course ID :
3
Enter the Course name :
maths
Enter the course duration :
120
Enter the fees for the course :
1000
Sucessfully added course
(base) PS D:\codes\web_development\case_study_3>
```

Before changes in courses table :



After changes in Courses table :



Admin views Courses Table :

Admin views Student table :

```
2021.8.1105858891\pythonFiles\lib\python\debugpy\launcher' '50738' '--' 'd:\codes\web_development\case_study_3\sms.py'
Welcome to SMS (Student Managment System)

Tell us who you are :
 1.Student
 2.Admin

Enter your choice :2
Welcome Admin
tell us what to do
 1.Add a new course to the list
 2.View all students table
 3.view all Courses table
tell us your choice Admin
2
(110, 'Vatsal', 'Kumar', '28/08/2000', 1)
(111, 'Kumar', 'Kaustubh', '31/08/2003', 2)
(base) PS D:\codes\web_development\case_study_3>
```

Github link :

https://github.com/captain7oxic/Student-Management-system-python