

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 application

Source code

SMS.py

```
sms.py > Student > _init_
1  import sqlite3
2
3
4  class Student:
5      def __init__(self,roll,first,last,dob,course_id):
6          self.conn=sqlite3.connect('college.db')
7          self.c=self.conn.cursor()
8          self.roll=roll
9          self.first=first
10         self.last=last
11         self.dob=dob
12         self.course_id=course_id
13         self.c.execute('''CREATE TABLE IF NOT EXISTS students
14             (roll INTEGER PRIMARY KEY ,
15              first text,
16              last text,
17              dob text,
18              course_id INTEGER,
19              FOREIGN KEY (course_id) REFERENCES courses (course_id))''')
20
21     def add_student(self):
22         try:
23             self.c.execute('''INSERT INTO students VALUES (?, ?, ?, ?)''',(self.roll, self.first, self.last, self.dob, self.course_id))
24             self.conn.commit()
25             print('You have successfully registered')
26             self.conn.close()
27         except Exception:
28             print('ERROR: Could not add to student table')
29
30
31
32
33     class Course:
34         def __init__(self,course_id,course_name,duration,fees):
35             self.conn=sqlite3.connect('college.db')
36             self.c=self.conn.cursor()
37             self.course_id=course_id
38             self.course_name=course_name
39             self.duration=duration
40             self.fees=fees
```

```

sms.py > Student > _init_
40     self.fees=fees
41     self.c.execute(''''CREATE TABLE IF NOT EXISTS courses
42                     (course_id INTEGER PRIMARY KEY,
43                      course_name text,
44                      duration INTEGER,
45                      fees INTEGER)''')
46
47     def add_course(self):
48         try:
49             self.c.execute(''''INSERT INTO courses(course_id,course_name,duration,fees) VALUES (?,?,?,?)''',(self.course_id,self.course_name,self.duration,self.fees))
50             self.conn.commit()
51             print('Successfully added course')
52             self.conn.close()
53         except Exception:
54             print('ERROR: could not add to courses table')
55
56     def view_table_students():
57         conn=sqlite3.connect('college.db')
58         c=conn.cursor()
59         c.execute("SELECT * FROM students")
60         rows=c.fetchall()
61         for row in rows:
62             print(row)
63         conn.close()
64
65     def view_table_courses():
66         conn=sqlite3.connect('college.db')
67         c=conn.cursor()
68         c.execute("SELECT * FROM courses")
69         rows=c.fetchall()
70         for row in rows:
71             print(row)
72         conn.close()
73
74     def apply_to_course(roll,course_id):
75         conn=sqlite3.connect('college.db')
76         c=conn.cursor()
77         c.execute(''''UPDATE students SET course_id = ? WHERE roll = ?''',(course_id,roll))
78         conn.commit()
79         print('Successfully updated the course')
80         conn.close()
81
82     #except Exception:
83     #    print('there war a problem updating')
84
85
86 if __name__=="__main__":
87     print("Welcome to SMS (Student Managment System)")
88     print("\nTell us who you are :\n 1.Student\n 2.Admin\n")
89     choice=int(input('Enter your choice :'))
90
91     if(choice==1):
92         print('Welcome Student')
93         print('Choose what to do')
94         print('\n1.Register \n2.View courses \n3.Apply for course')
95         choice = int(input())
96
97         if (choice==1):
98             print('Enter your Roll number : ')
99             roll_num=int(input())
100             print('Enter your first name :')
101             first_name = input()
102             print('Enter your last name : ')
103             last_name = input()
104             print('Enter your Date of Birth : ')
105             DOB = input()
106             print('Enter the course ID : ')
107             Course_id=int(input())
108             student = Student(roll_num,first_name,last_name,DOB,Course_id)
109             student.add_student()
110
111         elif(choice==2):
112             view_table_courses()
113
114         elif(choice==3):
115             roll=int(input('Enter your roll : \n'))
116             course_id=int(input('Enter the new course ID : \n'))
117             apply_to_course(roll,course_id)
118
119

```

```

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     else:
147         print('Incorrect choice, User. Try again')
148

```

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-2021.8.110588891\pythonFiles\lib\python\debugpy\launcher" "55312" "--" "d:\codes\web_development\case_study_3\sms.py"
Welcome to SMS (Student Management 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 Management 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:

DB Browser for SQLite - D:\codes\web_development\case_study_3\college.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database

Database Structure Browse Data Edit Pragmas Execute SQL

Table: students

	roll	first	last	dob	course_id
1	110	Vatsal	Kumar	28/08/2000	2
2	111	Kumar	Kaustubh	31/08/2003	2

1 2

Type of data currently in cell: Text / Numeric
1 character(s)

Apply

Remote

Identity Select an identity to connect

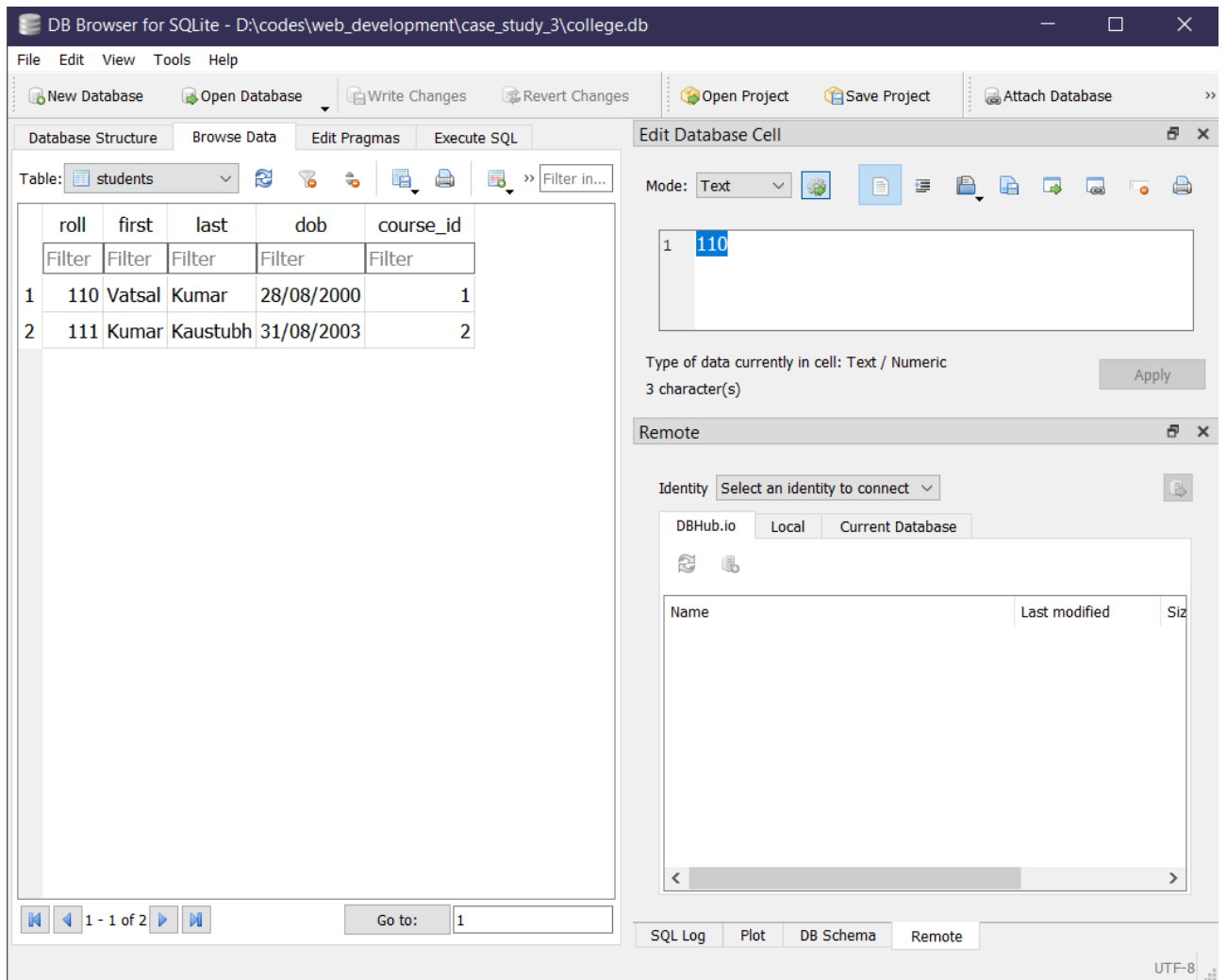
DBHub.io Local Current Database

Name	Last modified	Size
------	---------------	------

SQL Log Plot DB Schema Remote

UTF-8

Changes after Updation :



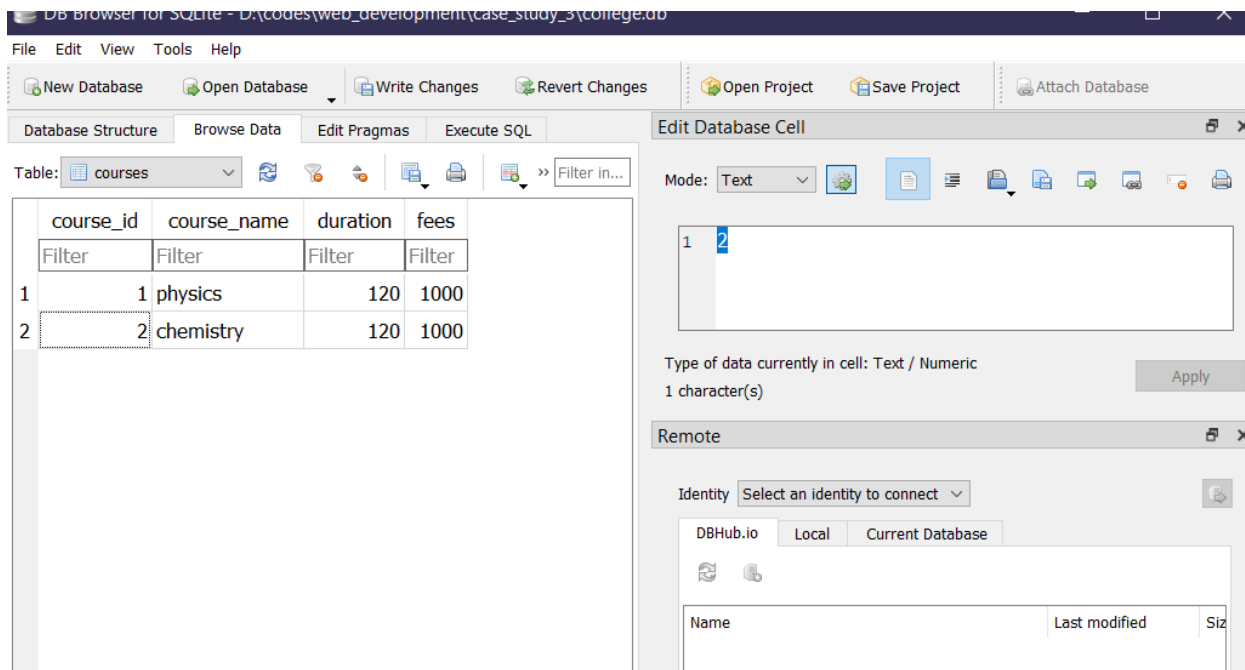
Admin adds a Course :

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

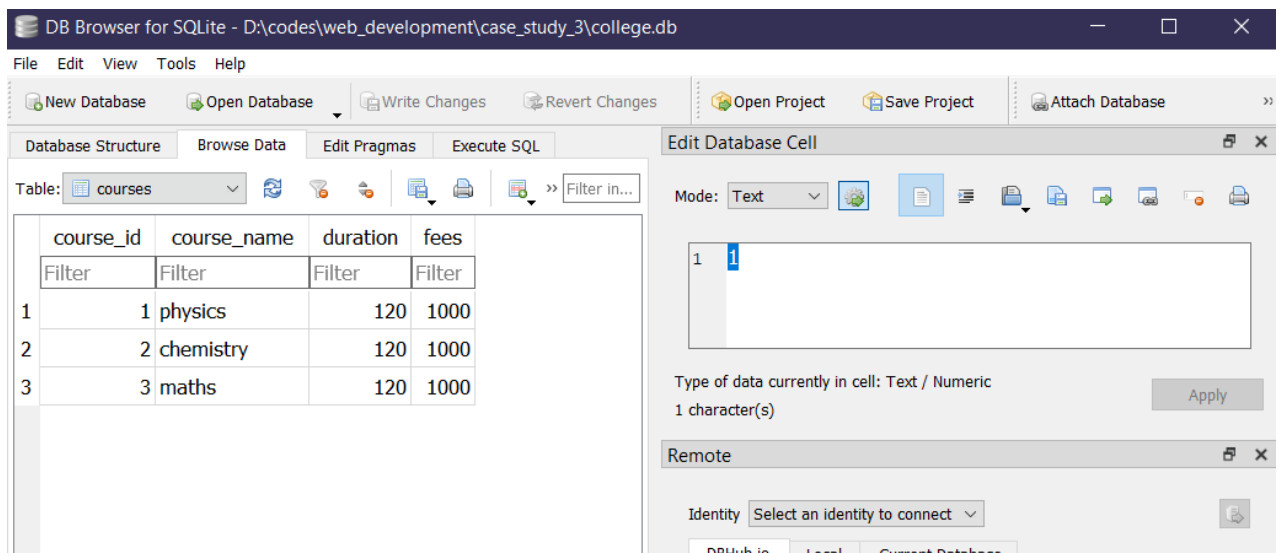
Tell us who you are :
1.Student
2.Admin

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 :

```

2021.8.1105858891\pythonFiles\lib\python\debugpy\launcher --49546 -- -- "d:\codes\web_development\case_study_3\sms.py"
Welcome to SMS (Student Management 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
3
(1, 'physics', 120, 1000)
(2, 'chemistry', 120, 1000)
(3, 'maths', 120, 1000)
(base) PS D:\codes\web_development\case_study_3>

```

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>