AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

Choose an item.

Assignment Cover Sheet

Project Title:	*Course Management system using file system in python*		
Term :	Mid-term	Date of Submission:	*8 November 2022*
Course Title:	PROGRAMMING IN PYTHON [B]		
Course Code:	*01333*	Section:	B
Semester:	Fall 22-23	Course Teacher:	AKINUL ISLAM JONY

Declaration and Statement of Authorship:

1. I/we hold a copy of this Assignment/Case-Study, which can be produced if the original is lost/damaged.
2. This Assignment/Case-Study is my/our original work and no part of it has been copied from any other student's work or from any other source except where due acknowledgement is made.
3. No part of this Assignment/Case-Study has been written for me/us by any other person except where such collaborationhas been authorized by the concerned teacher and is clearly acknowledged in the assignment.
4. I/we have not previously submitted or currently submitting this work for any other course/unit.
5. This work may be reproduced, communicated, compared and archived for the purpose of detecting plagiarism.
6. I/we give permission for a copy of my/our marked work to be retained by the Faculty for review and comparison, including review by external examiners.
7. I/we understand thatPlagiarism is the presentation of the work, idea or creation of another person as though it is your own. It is a formofcheatingandisaveryseriousacademicoffencethatmayleadtoexpulsionfromtheUniversity. Plagiarized material can be drawn from, and presented in, written, graphic and visual form, including electronic data, and oral presentations. Plagiarism occurs when the origin of them arterial used is not appropriately cited.
8. I/we also understand that enabling plagiarism is the act of assisting or allowing another person to plagiarize or to copy my/our work.

* *Student(s) must complete all details except the faculty use part.*
** Please submit all assignments to your course teacher or the office of the concerned teacher.

Group Name/No.:

No	Name	ID	Program	Signature
1	Rifat Hossain	20-42461-1	Bsc CSE	Rifat

Faculty use only

FACULTYCOMMENTS	Marks Obtained	
	Total Marks	

Course Management system using file system in python

Project summary: In this project various technique were used to keep track of courses in local file system using python programming language. Operations like adding ,delating ,updating ,searching can be performed by this program.

Project requirements: There where several requirements such adding ,delating, updating , searching and displaying all the courses from file. To perform these task csv module is used . also prompt were shown if the necessary steps missing.

Add course: first course name, code ,pre-prequiside and credit is asked from user ,when necessary conditions were being met finally that course is added to the csv file .

Updating an existing Course: From user data following row in csv got updated.

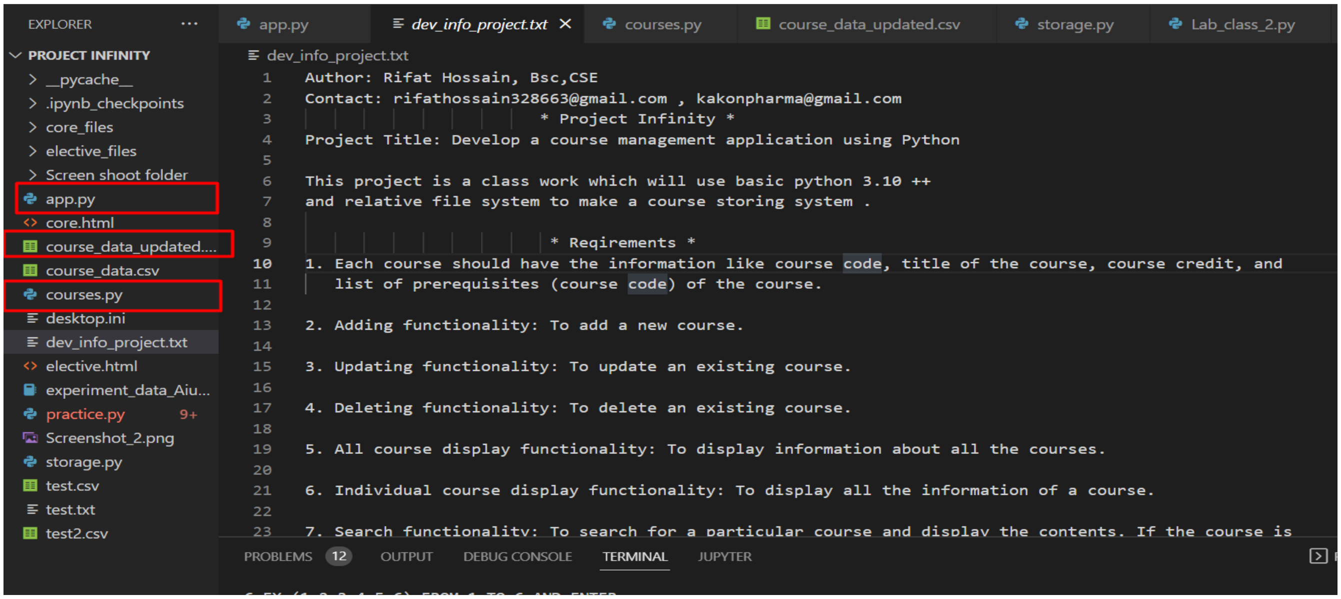
Delating an existing Course: Form user data delatting the course along with course code, index, and prerequisite.

Display courses: All the courses in specified csv file has been showed for the user.

Search Course: This option Shows a given coure exist or not, if that course exist then shows all the data related to that course.

Quit: exit out of the loop ,thus stoping the program.

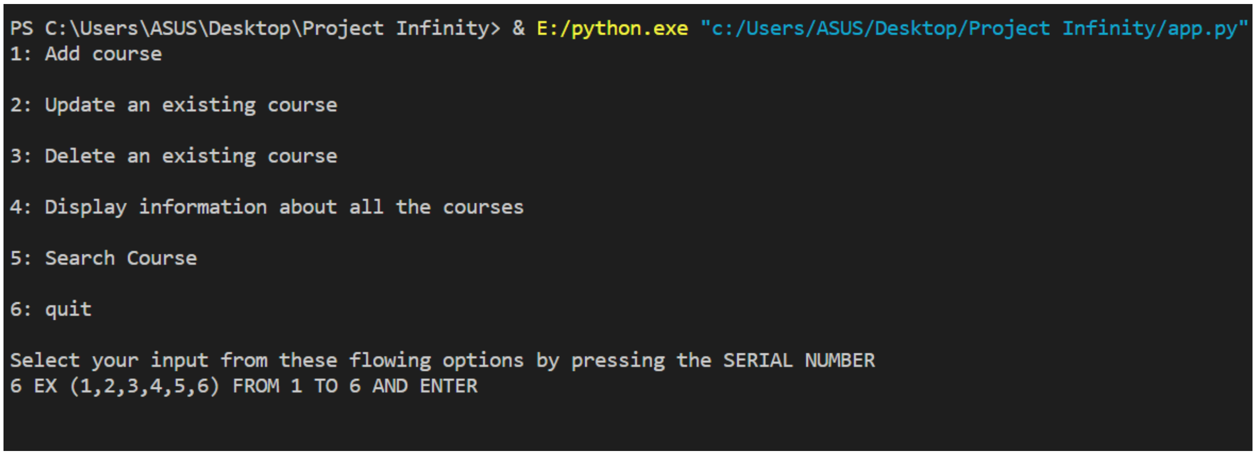
Main Project files



```
Author: Rifat Hossain, Bsc,CSE
Contact: rifathossain328663@gmail.com , kakonpharma@gmail.com
                        * Project Infinity *
Project Title: Develop a course management application using Python

This project is a class work which will use basic python 3.10 ++
and relative file system to make a course storing system .

                         * Reqirements *
1. Each course should have the information like course code, title of the course, course credit, and
   list of prerequisites (course code) of the course.

2. Adding functionality: To add a new course.

3. Updating functionality: To update an existing course.

4. Deleting functionality: To delete an existing course.

5. All course display functionality: To display information about all the courses.

6. Individual course display functionality: To display all the information of a course.

7. Search functionality: To search for a particular course and display the contents. If the course is
```

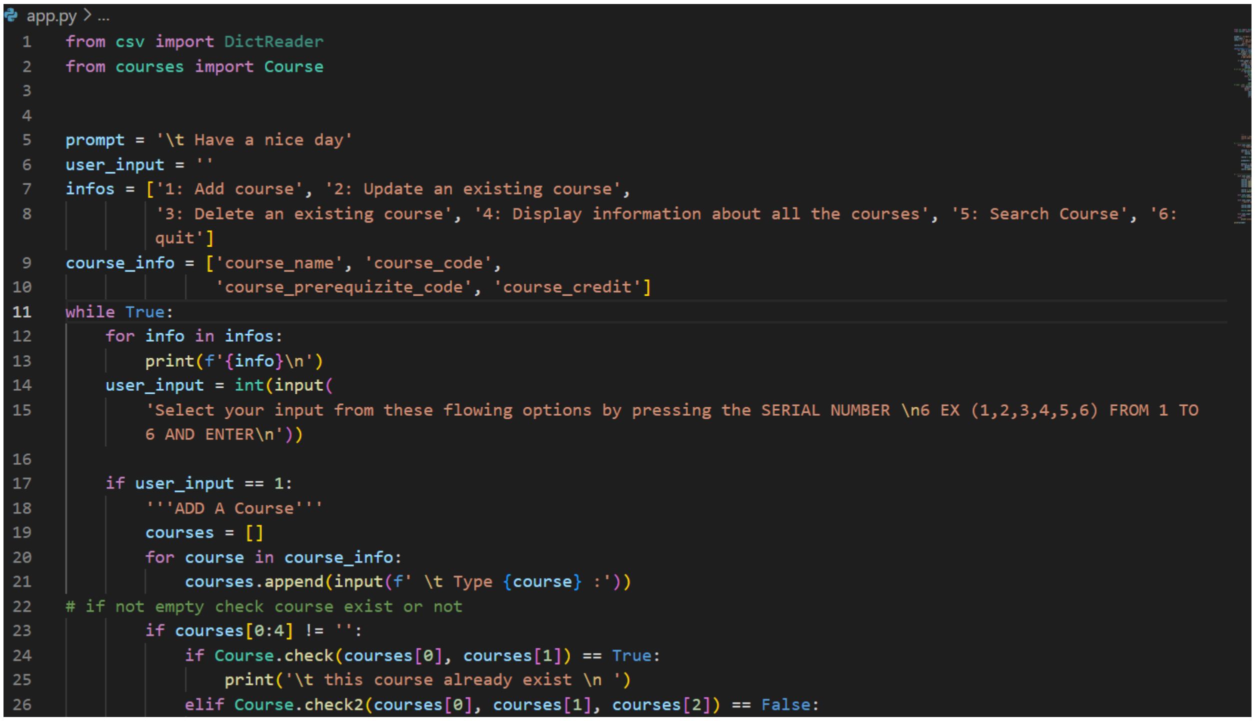
app.py

Shows all the options and call object instances from Course class from Course.py

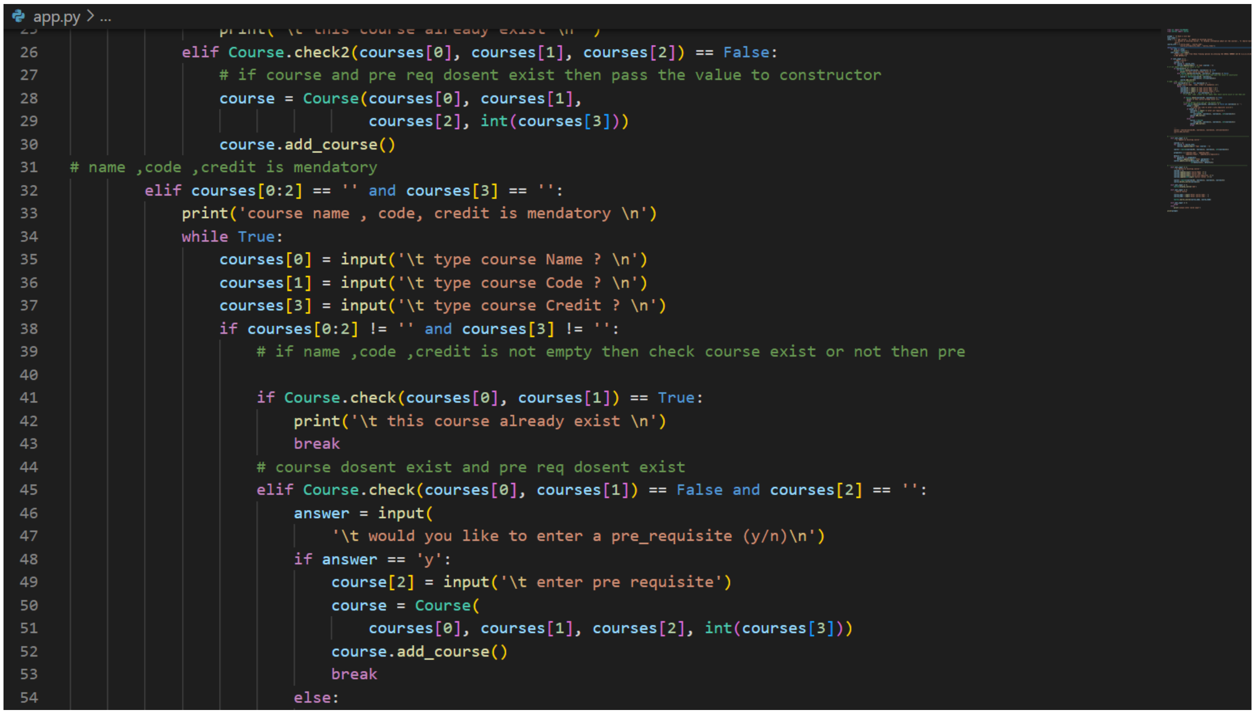


```
PS C:\Users\ASUS\Desktop\Project Infinity> & E:/python.exe "c:/Users/ASUS/Desktop/Project Infinity/app.py"
1: Add course

2: Update an existing course

3: Delete an existing course

4: Display information about all the courses

5: Search Course

6: quit

Select your input from these flowing options by pressing the SERIAL NUMBER
6 EX (1,2,3,4,5,6) FROM 1 TO 6 AND ENTER
```

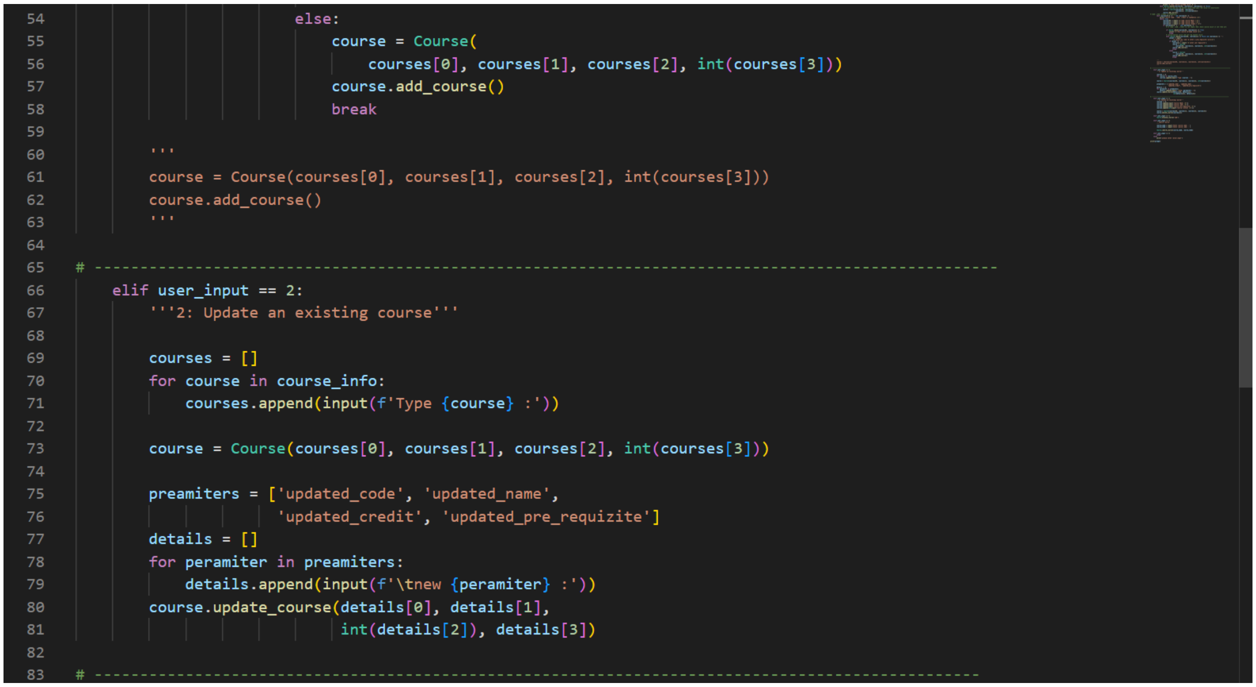
Code:



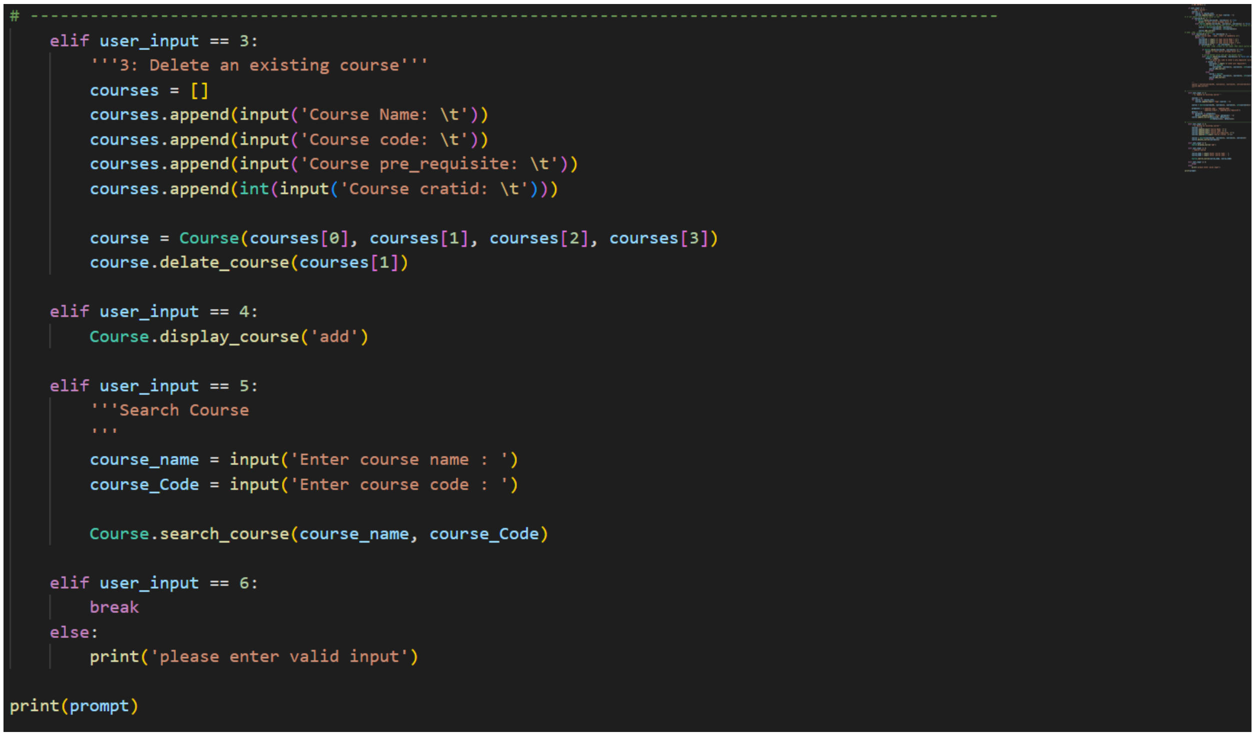
```python
from csv import DictReader
from courses import Course


prompt = '\t Have a nice day'
user_input = ''
infos = ['1: Add course', '2: Update an existing course',
        '3: Delete an existing course', '4: Display information about all the courses', '5: Search Course', '6: quit']
course_info = ['course_name', 'course_code',
            'course_prerequizite_code', 'course_credit']
while True:
    for info in infos:
        print(f'{info}\n')
    user_input = int(input(
        'Select your input from these flowing options by pressing the SERIAL NUMBER \n6 EX (1,2,3,4,5,6) FROM 1 TO 6 AND ENTER\n'))

    if user_input == 1:
        '''ADD A Course'''
        courses = []
        for course in course_info:
            courses.append(input(f' \t Type {course} :'))
# if not empty check course exist or not
        if courses[0:4] != '':
            if Course.check(courses[0], courses[1]) == True:
                print('\t this course already exist \n ')
            elif Course.check2(courses[0], courses[1], courses[2]) == False:
```



```python
                print('\t this course already exist \n')
            elif Course.check2(courses[0], courses[1], courses[2]) == False:
                # if course and pre req dosent exist then pass the value to constructor
                course = Course(courses[0], courses[1],
                                courses[2], int(courses[3]))
                course.add_course()
# name ,code ,credit is mendatory
        elif courses[0:2] == '' and courses[3] == '':
            print('course name , code, credit is mendatory \n')
            while True:
                courses[0] = input('\t type course Name ? \n')
                courses[1] = input('\t type course Code ? \n')
                courses[3] = input('\t type course Credit ? \n')
                if courses[0:2] != '' and courses[3] != '':
                    # if name ,code ,credit is not empty then check course exist or not then pre

                    if Course.check(courses[0], courses[1]) == True:
                        print('\t this course already exist \n')
                        break
                    # course dosent exist and pre req dosent exist
                    elif Course.check(courses[0], courses[1]) == False and courses[2] == '':
                        answer = input(
                            '\t would you like to enter a pre_requisite (y/n)\n')
                        if answer == 'y':
                            course[2] = input('\t enter pre requisite')
                            course = Course(
                                courses[0], courses[1], courses[2], int(courses[3]))
                            course.add_course()
                            break
                        else:
```



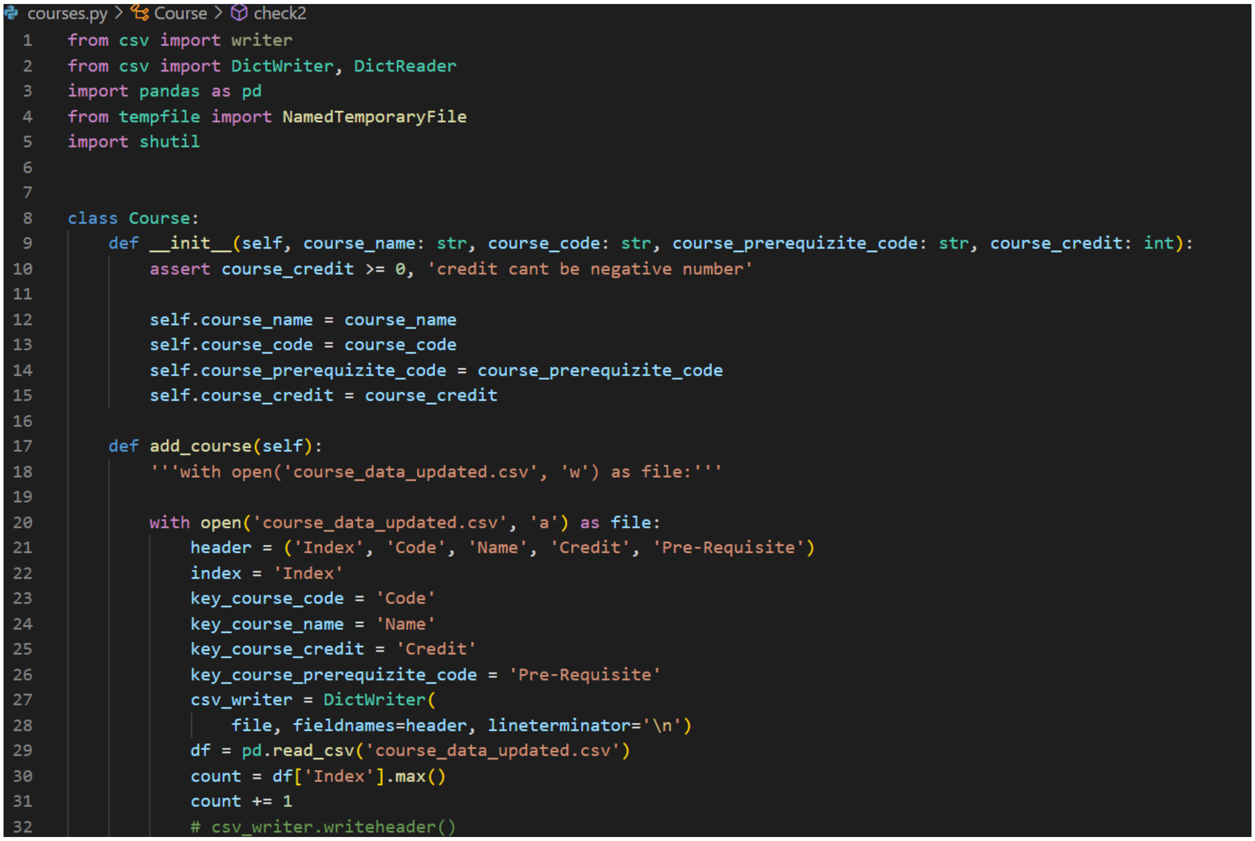
```python
                        else:
                            course = Course(
                                courses[0], courses[1], courses[2], int(courses[3]))
                            course.add_course()
                            break

        '''
        course = Course(courses[0], courses[1], courses[2], int(courses[3]))
        course.add_course()
        '''

# --------------------------------------------------------------------------------------------------
    elif user_input == 2:
        '''2: Update an existing course'''

        courses = []
        for course in course_info:
            courses.append(input(f'Type {course} :'))

        course = Course(courses[0], courses[1], courses[2], int(courses[3]))

        preamiters = ['updated_code', 'updated_name',
                      'updated_credit', 'updated_pre_requizite']
        details = []
        for peramiter in preamiters:
            details.append(input(f'\tnew {peramiter} :'))
        course.update_course(details[0], details[1],
                             int(details[2]), details[3])

# ------------------------------------------------------------------------------------------------
```



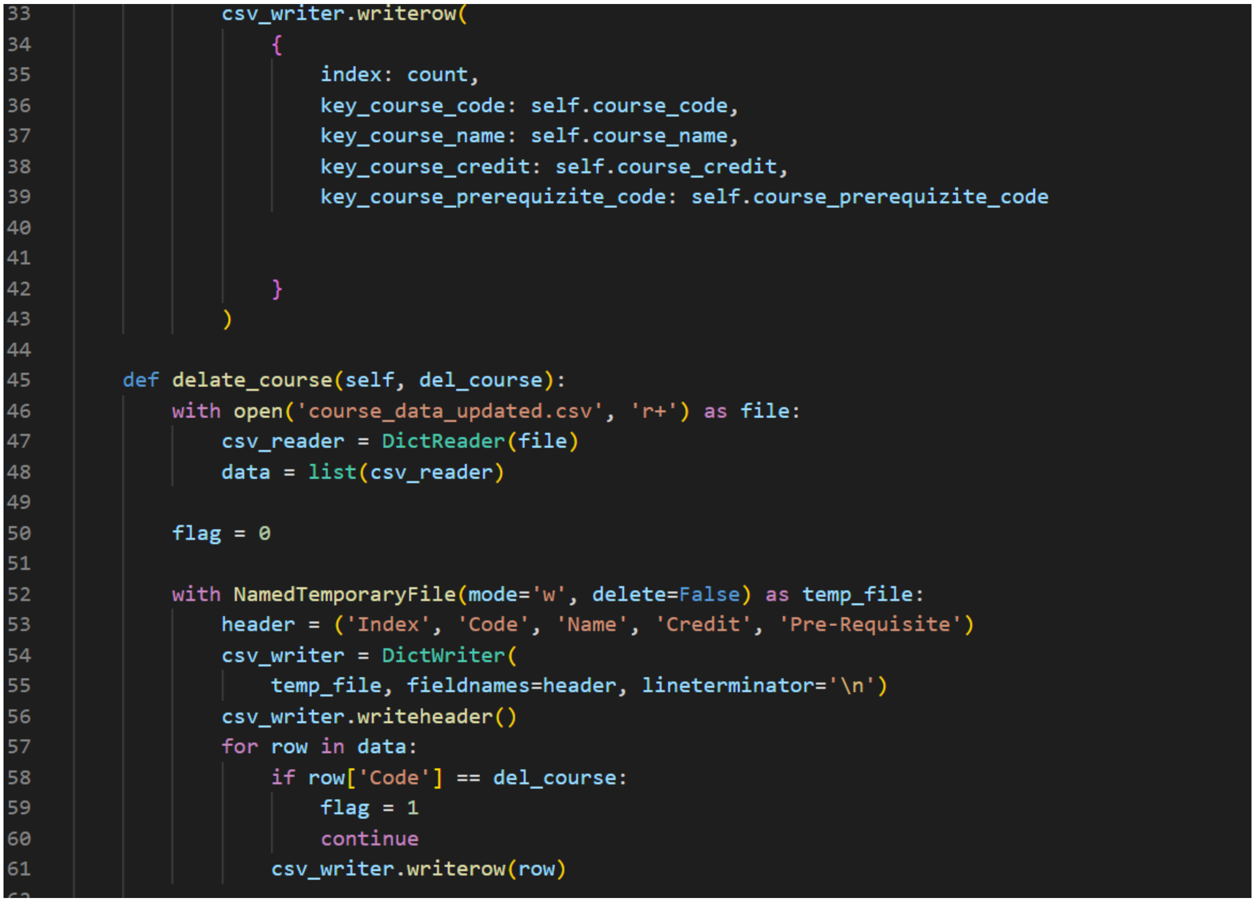
```
# -------------------------------------------------------------------------------------------------
    elif user_input == 3:
        '''3: Delete an existing course'''
        courses = []
        courses.append(input('Course Name: \t'))
        courses.append(input('Course code: \t'))
        courses.append(input('Course pre_requisite: \t'))
        courses.append(int(input('Course cratid: \t')))

        course = Course(courses[0], courses[1], courses[2], courses[3])
        course.delate_course(courses[1])

    elif user_input == 4:
        Course.display_course('add')

    elif user_input == 5:
        '''Search Course
        '''
        course_name = input('Enter course name : ')
        course_Code = input('Enter course code : ')

        Course.search_course(course_name, course_Code)

    elif user_input == 6:
        break
    else:
        print('please enter valid input')

print(prompt)
```

Courses.py

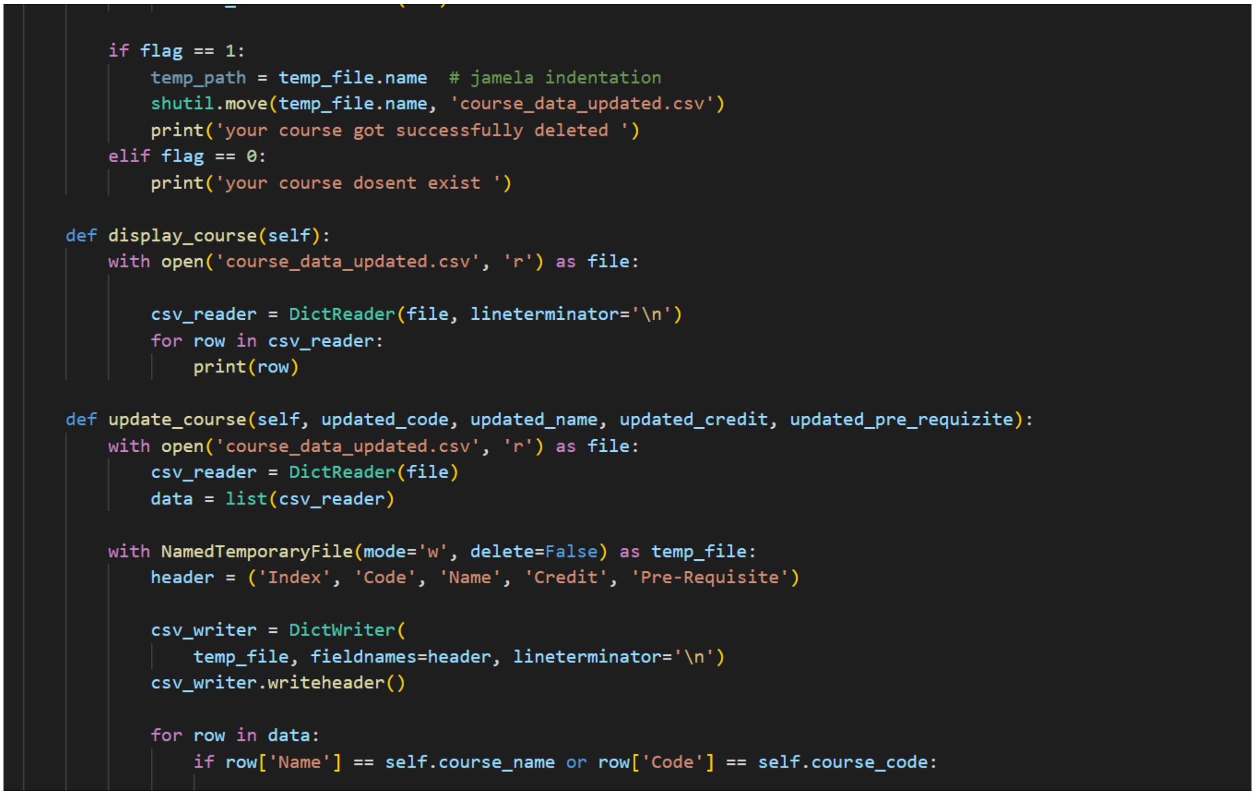
The file contain the Course class, contractor, methods , and decoder such as static method.



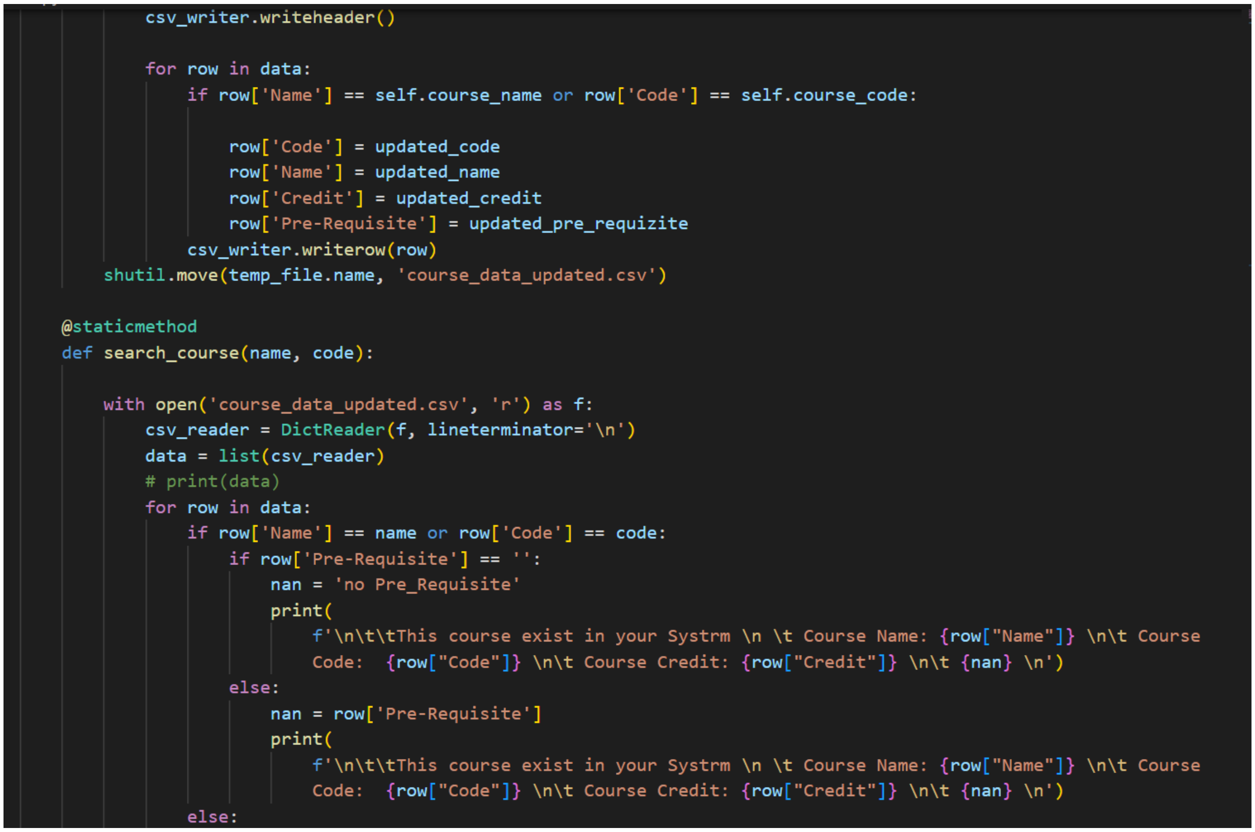
```
from csv import writer
from csv import DictWriter, DictReader
import pandas as pd
from tempfile import NamedTemporaryFile
import shutil


class Course:
    def __init__(self, course_name: str, course_code: str, course_prerequizite_code: str, course_credit: int):
        assert course_credit >= 0, 'credit cant be negative number'

        self.course_name = course_name
        self.course_code = course_code
        self.course_prerequizite_code = course_prerequizite_code
        self.course_credit = course_credit

    def add_course(self):
        '''with open('course_data_updated.csv', 'w') as file:'''

        with open('course_data_updated.csv', 'a') as file:
            header = ('Index', 'Code', 'Name', 'Credit', 'Pre-Requisite')
            index = 'Index'
            key_course_code = 'Code'
            key_course_name = 'Name'
            key_course_credit = 'Credit'
            key_course_prerequizite_code = 'Pre-Requisite'
            csv_writer = DictWriter(
                file, fieldnames=header, lineterminator='\n')
            df = pd.read_csv('course_data_updated.csv')
            count = df['Index'].max()
            count += 1
            # csv_writer.writeheader()
```



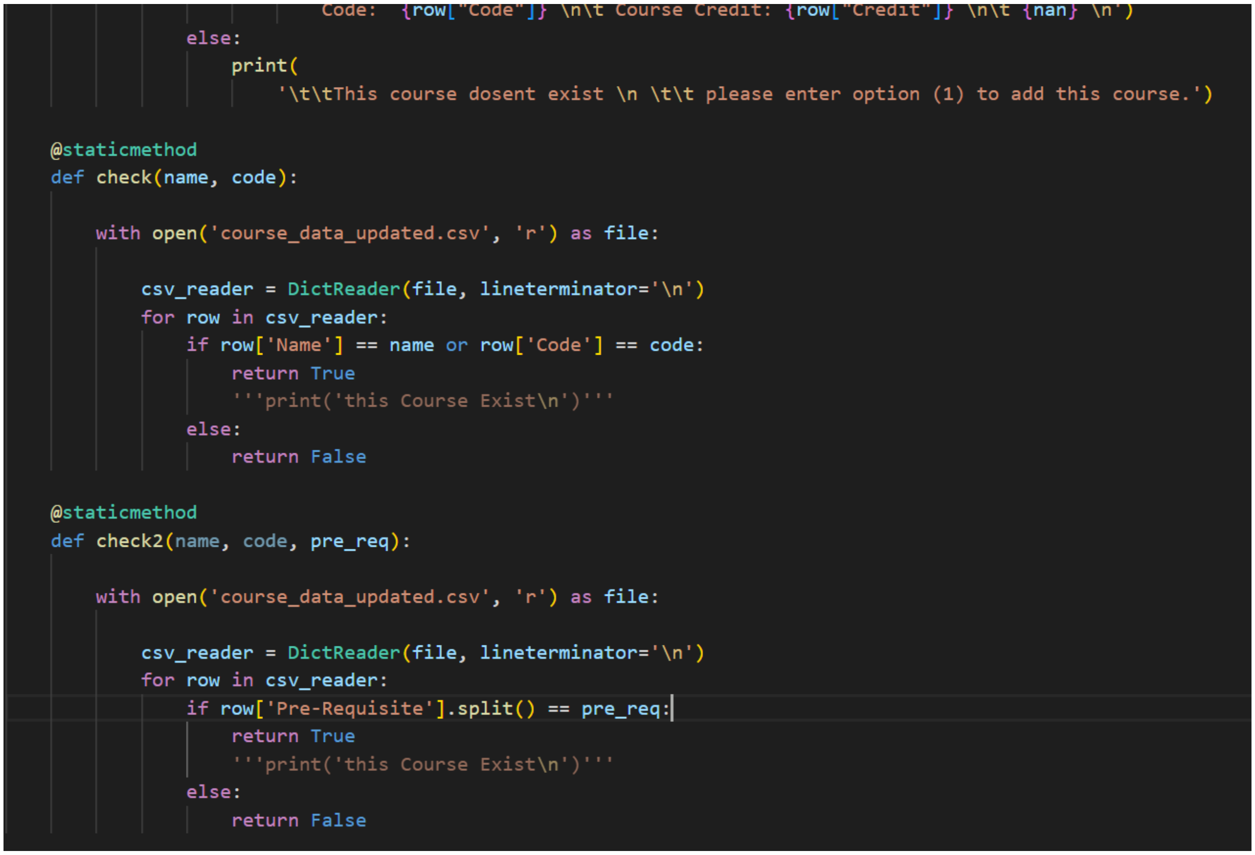
```
            csv_writer.writerow(
                {
                    index: count,
                    key_course_code: self.course_code,
                    key_course_name: self.course_name,
                    key_course_credit: self.course_credit,
                    key_course_prerequizite_code: self.course_prerequizite_code


                }
            )

    def delate_course(self, del_course):
        with open('course_data_updated.csv', 'r+') as file:
            csv_reader = DictReader(file)
            data = list(csv_reader)

        flag = 0

        with NamedTemporaryFile(mode='w', delete=False) as temp_file:
            header = ('Index', 'Code', 'Name', 'Credit', 'Pre-Requisite')
            csv_writer = DictWriter(
                temp_file, fieldnames=header, lineterminator='\n')
            csv_writer.writeheader()
            for row in data:
                if row['Code'] == del_course:
                    flag = 1
                    continue
                csv_writer.writerow(row)
```



```
        if flag == 1:
            temp_path = temp_file.name  # jamela indentation
            shutil.move(temp_file.name, 'course_data_updated.csv')
            print('your course got successfully deleted ')
        elif flag == 0:
            print('your course dosent exist ')

    def display_course(self):
        with open('course_data_updated.csv', 'r') as file:

            csv_reader = DictReader(file, lineterminator='\n')
            for row in csv_reader:
                print(row)

    def update_course(self, updated_code, updated_name, updated_credit, updated_pre_requizite):
        with open('course_data_updated.csv', 'r') as file:
            csv_reader = DictReader(file)
            data = list(csv_reader)

        with NamedTemporaryFile(mode='w', delete=False) as temp_file:
            header = ('Index', 'Code', 'Name', 'Credit', 'Pre-Requisite')

            csv_writer = DictWriter(
                temp_file, fieldnames=header, lineterminator='\n')
            csv_writer.writeheader()

            for row in data:
                if row['Name'] == self.course_name or row['Code'] == self.course_code:
```

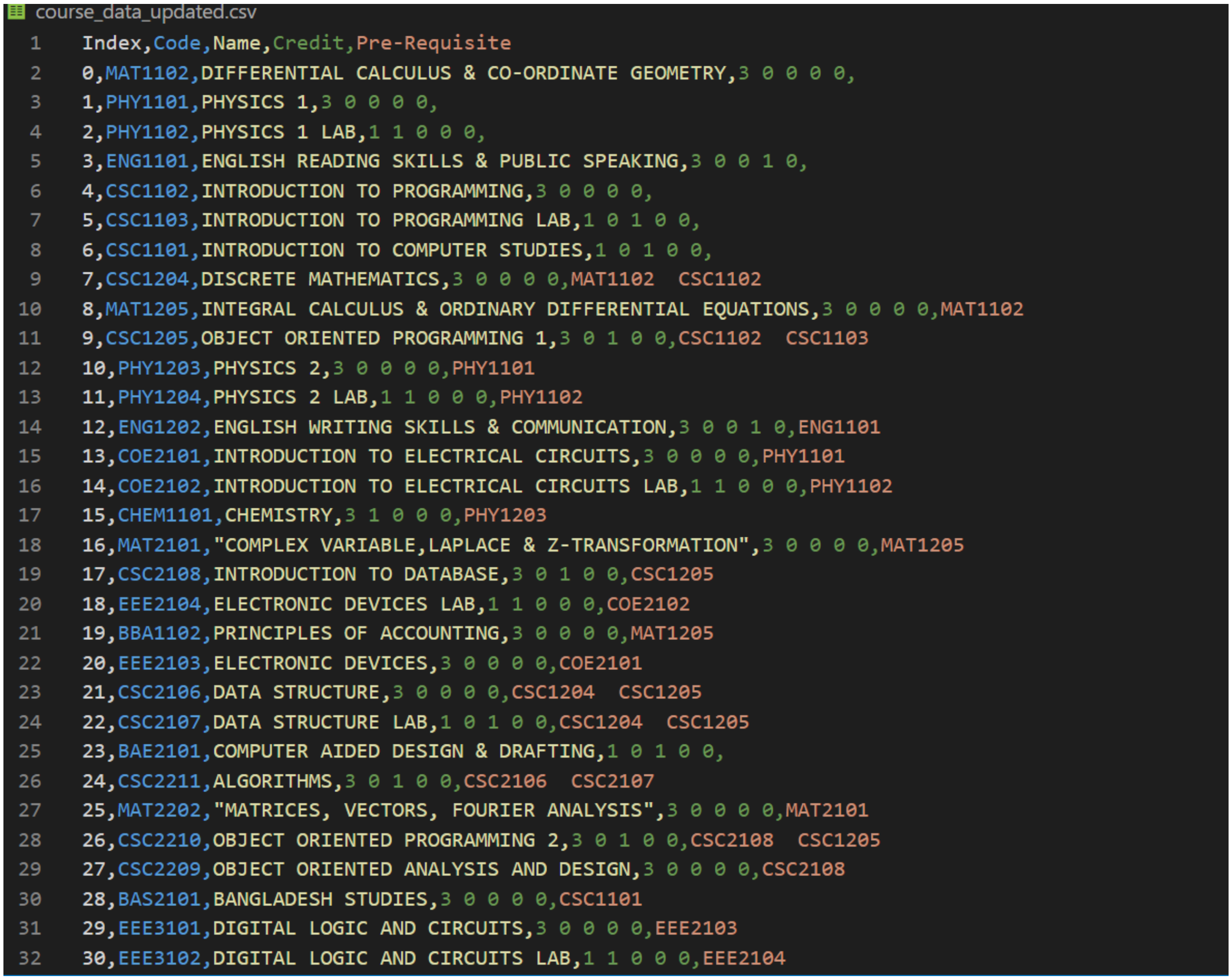


```
            csv_writer.writeheader()

            for row in data:
                if row['Name'] == self.course_name or row['Code'] == self.course_code:

                    row['Code'] = updated_code
                    row['Name'] = updated_name
                    row['Credit'] = updated_credit
                    row['Pre-Requisite'] = updated_pre_requizite
                csv_writer.writerow(row)
        shutil.move(temp_file.name, 'course_data_updated.csv')

    @staticmethod
    def search_course(name, code):

        with open('course_data_updated.csv', 'r') as f:
            csv_reader = DictReader(f, lineterminator='\n')
            data = list(csv_reader)
            # print(data)
            for row in data:
                if row['Name'] == name or row['Code'] == code:
                    if row['Pre-Requisite'] == '':
                        nan = 'no Pre_Requisite'
                        print(
                            f'\n\t\tThis course exist in your Systrm \n \t Course Name: {row["Name"]} \n\t Course
                            Code:  {row["Code"]} \n\t Course Credit: {row["Credit"]} \n\t {nan} \n')
                    else:
                        nan = row['Pre-Requisite']
                        print(
                            f'\n\t\tThis course exist in your Systrm \n \t Course Name: {row["Name"]} \n\t Course
                            Code:  {row["Code"]} \n\t Course Credit: {row["Credit"]} \n\t {nan} \n')
                else:
```



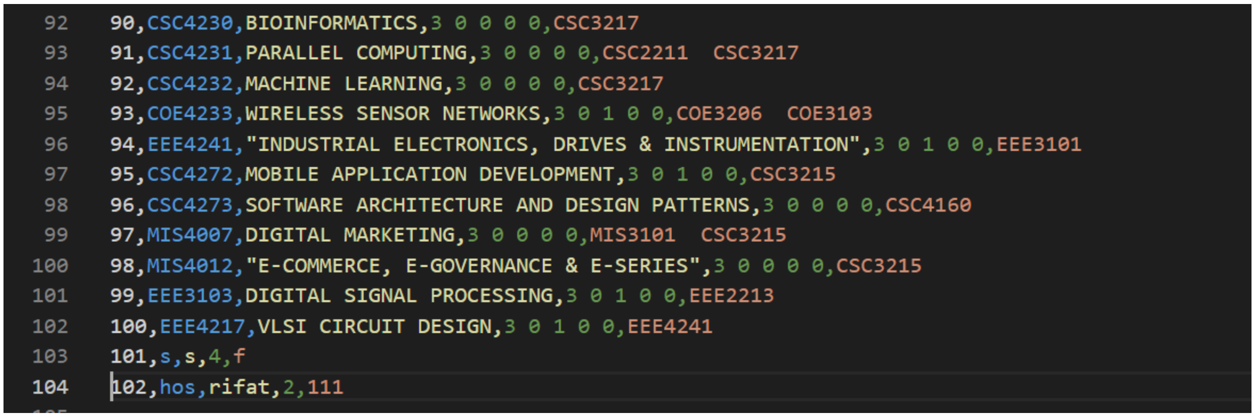
```
                        Code:  {row["Code"]} \n\t Course Credit: {row["Credit"]} \n\t {nan} \n')
                else:
                    print(
                        '\t\tThis course dosent exist \n \t\t please enter option (1) to add this course.')

    @staticmethod
    def check(name, code):

        with open('course_data_updated.csv', 'r') as file:

            csv_reader = DictReader(file, lineterminator='\n')
            for row in csv_reader:
                if row['Name'] == name or row['Code'] == code:
                    return True
                    '''print('this Course Exist\n')'''
                else:
                    return False

    @staticmethod
    def check2(name, code, pre_req):

        with open('course_data_updated.csv', 'r') as file:

            csv_reader = DictReader(file, lineterminator='\n')
            for row in csv_reader:
                if row['Pre-Requisite'].split() == pre_req:
                    return True
                    '''print('this Course Exist\n')'''
                else:
                    return False
```

For Course Storage: `course_data_updated.csv ,many of them were extracted from aiub portal using pandas module.`



course_data_updated.csv

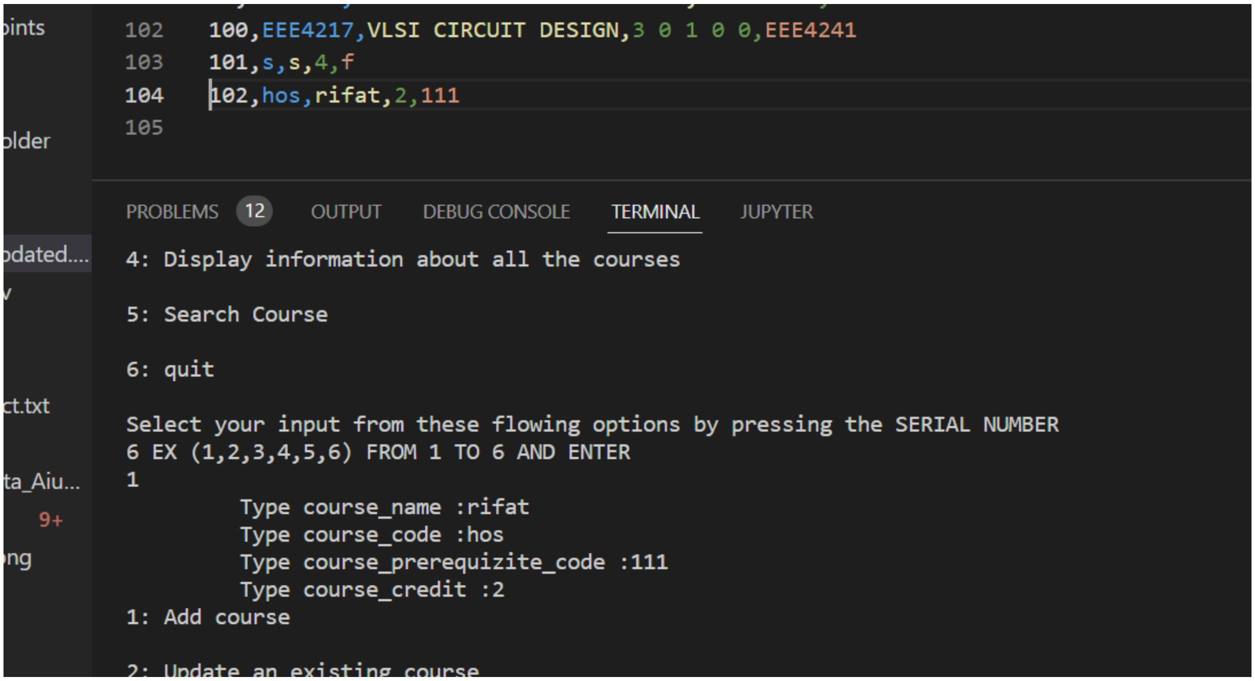
```
Index,Code,Name,Credit,Pre-Requisite
0,MAT1102,DIFFERENTIAL CALCULUS & CO-ORDINATE GEOMETRY,3 0 0 0 0,
1,PHY1101,PHYSICS 1,3 0 0 0 0,
2,PHY1102,PHYSICS 1 LAB,1 1 0 0 0,
3,ENG1101,ENGLISH READING SKILLS & PUBLIC SPEAKING,3 0 0 1 0,
4,CSC1102,INTRODUCTION TO PROGRAMMING,3 0 0 0 0,
5,CSC1103,INTRODUCTION TO PROGRAMMING LAB,1 0 1 0 0,
6,CSC1101,INTRODUCTION TO COMPUTER STUDIES,1 0 1 0 0,
7,CSC1204,DISCRETE MATHEMATICS,3 0 0 0 0,MAT1102  CSC1102
8,MAT1205,INTEGRAL CALCULUS & ORDINARY DIFFERENTIAL EQUATIONS,3 0 0 0 0,MAT1102
9,CSC1205,OBJECT ORIENTED PROGRAMMING 1,3 0 1 0 0,CSC1102  CSC1103
10,PHY1203,PHYSICS 2,3 0 0 0 0,PHY1101
11,PHY1204,PHYSICS 2 LAB,1 1 0 0 0,PHY1102
12,ENG1202,ENGLISH WRITING SKILLS & COMMUNICATION,3 0 0 1 0,ENG1101
13,COE2101,INTRODUCTION TO ELECTRICAL CIRCUITS,3 0 0 0 0,PHY1101
14,COE2102,INTRODUCTION TO ELECTRICAL CIRCUITS LAB,1 1 0 0 0,PHY1102
15,CHEM1101,CHEMISTRY,3 1 0 0 0,PHY1203
16,MAT2101,"COMPLEX VARIABLE,LAPLACE & Z-TRANSFORMATION",3 0 0 0 0,MAT1205
17,CSC2108,INTRODUCTION TO DATABASE,3 0 1 0 0,CSC1205
18,EEE2104,ELECTRONIC DEVICES LAB,1 1 0 0 0,COE2102
19,BBA1102,PRINCIPLES OF ACCOUNTING,3 0 0 0 0,MAT1205
20,EEE2103,ELECTRONIC DEVICES,3 0 0 0 0,COE2101
21,CSC2106,DATA STRUCTURE,3 0 0 0 0,CSC1204  CSC1205
22,CSC2107,DATA STRUCTURE LAB,1 0 1 0 0,CSC1204  CSC1205
23,BAE2101,COMPUTER AIDED DESIGN & DRAFTING,1 0 1 0 0,
24,CSC2211,ALGORITHMS,3 0 1 0 0,CSC2106  CSC2107
25,MAT2202,"MATRICES, VECTORS, FOURIER ANALYSIS",3 0 0 0 0,MAT2101
26,CSC2210,OBJECT ORIENTED PROGRAMMING 2,3 0 1 0 0,CSC2108  CSC1205
27,CSC2209,OBJECT ORIENTED ANALYSIS AND DESIGN,3 0 0 0 0,CSC2108
28,BAS2101,BANGLADESH STUDIES,3 0 0 0 0,CSC1101
29,EEE3101,DIGITAL LOGIC AND CIRCUITS,3 0 0 0 0,EEE2103
30,EEE3102,DIGITAL LOGIC AND CIRCUITS LAB,1 1 0 0 0,EEE2104
```



```
90,CSC4230,BIOINFORMATICS,3 0 0 0 0,CSC3217
91,CSC4231,PARALLEL COMPUTING,3 0 0 0 0,CSC2211  CSC3217
92,CSC4232,MACHINE LEARNING,3 0 0 0 0,CSC3217
93,COE4233,WIRELESS SENSOR NETWORKS,3 0 1 0 0,COE3206  COE3103
94,EEE4241,"INDUSTRIAL ELECTRONICS, DRIVES & INSTRUMENTATION",3 0 1 0 0,EEE3101
95,CSC4272,MOBILE APPLICATION DEVELOPMENT,3 0 1 0 0,CSC3215
96,CSC4273,SOFTWARE ARCHITECTURE AND DESIGN PATTERNS,3 0 0 0 0,CSC4160
97,MIS4007,DIGITAL MARKETING,3 0 0 0 0,MIS3101  CSC3215
98,MIS4012,"E-COMMERCE, E-GOVERNANCE & E-SERIES",3 0 0 0 0,CSC3215
99,EEE3103,DIGITAL SIGNAL PROCESSING,3 0 1 0 0,EEE2213
100,EEE4217,VLSI CIRCUIT DESIGN,3 0 1 0 0,EEE4241
101,s,s,4,f
102,hos,rifat,2,111
```

Operation Results:

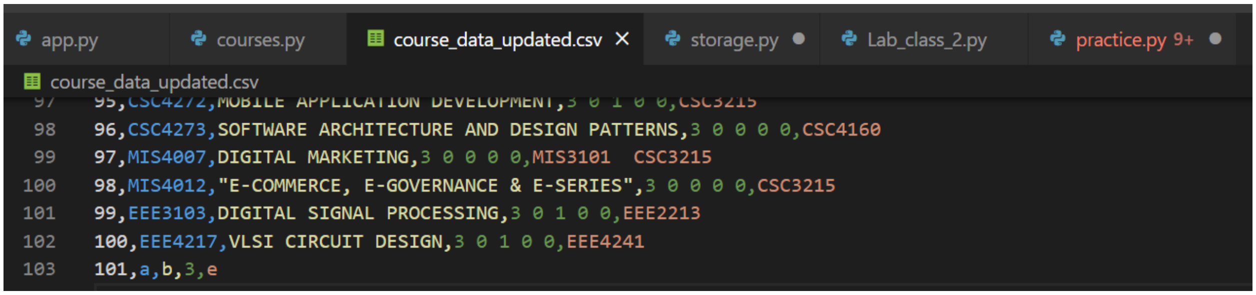
Add course:



```
100,EEE4217,VLSI CIRCUIT DESIGN,3 0 1 0 0,EEE4241
101,s,s,4,f
102,hos,rifat,2,111
```

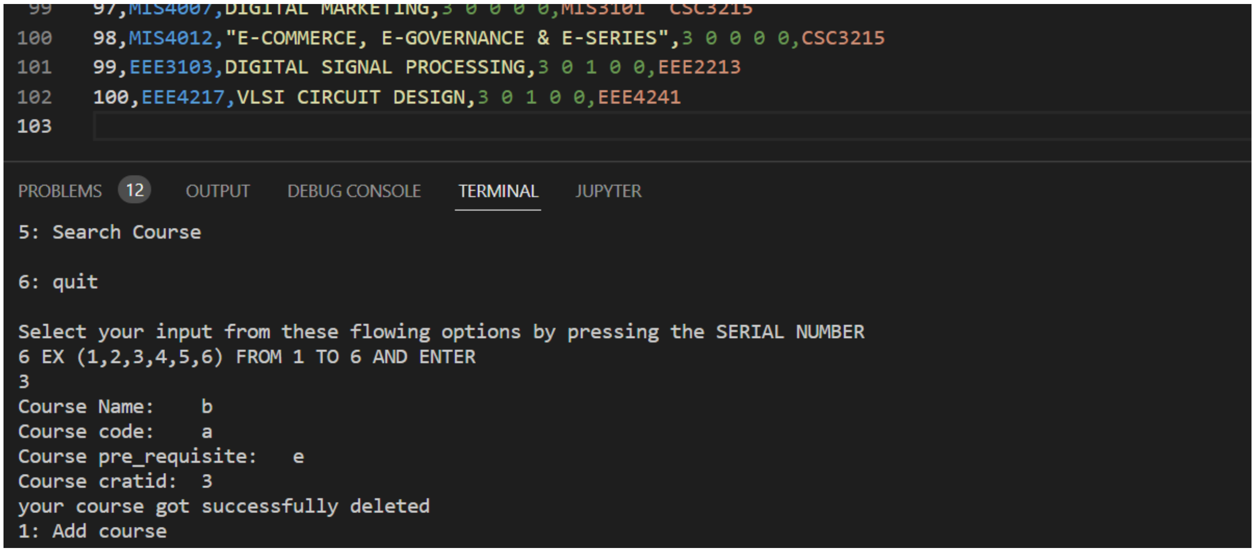
```
4: Display information about all the courses

5: Search Course

6: quit

Select your input from these flowing options by pressing the SERIAL NUMBER
6 EX (1,2,3,4,5,6) FROM 1 TO 6 AND ENTER
1
         Type course_name :rifat
         Type course_code :hos
         Type course_prerequizite_code :111
         Type course_credit :2
1: Add course

2: Update an existing course
```

Delete Course: (before delating course):



```
95,CSC4272,MOBILE APPLICATION DEVELOPMENT,3 0 1 0 0,CSC3215
96,CSC4273,SOFTWARE ARCHITECTURE AND DESIGN PATTERNS,3 0 0 0 0,CSC4160
97,MIS4007,DIGITAL MARKETING,3 0 0 0 0,MIS3101  CSC3215
98,MIS4012,"E-COMMERCE, E-GOVERNANCE & E-SERIES",3 0 0 0 0,CSC3215
99,EEE3103,DIGITAL SIGNAL PROCESSING,3 0 1 0 0,EEE2213
100,EEE4217,VLSI CIRCUIT DESIGN,3 0 1 0 0,EEE4241
101,a,b,3,e
```

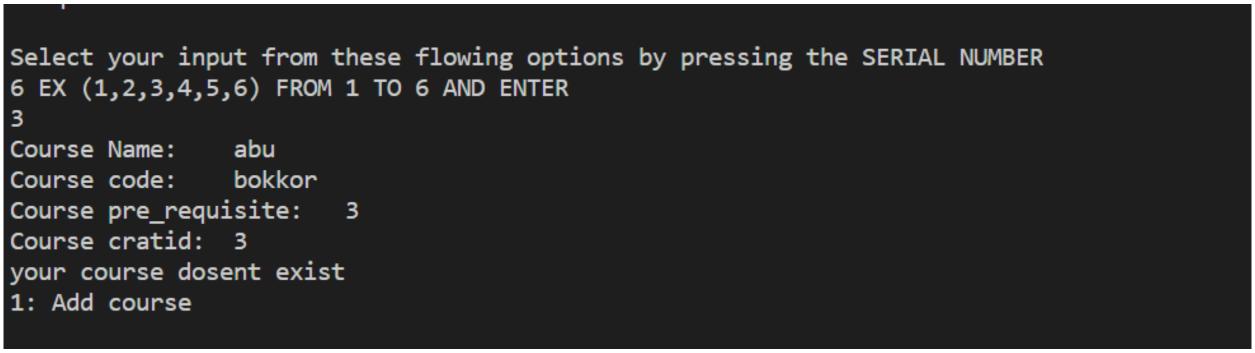
(After deleting course:)



```
97,MIS4007,DIGITAL MARKETING,3 0 0 0 0,MIS3101  CSC3215
98,MIS4012,"E-COMMERCE, E-GOVERNANCE & E-SERIES",3 0 0 0 0,CSC3215
99,EEE3103,DIGITAL SIGNAL PROCESSING,3 0 1 0 0,EEE2213
100,EEE4217,VLSI CIRCUIT DESIGN,3 0 1 0 0,EEE4241
```

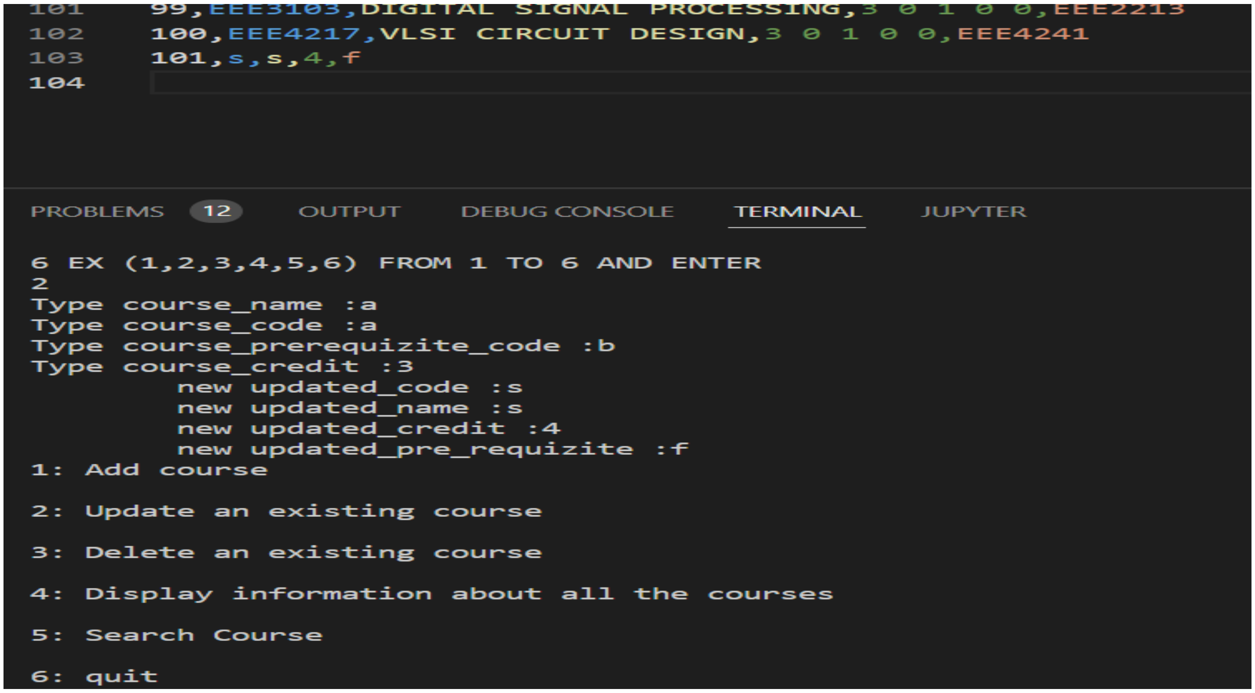
```
5: Search Course

6: quit

Select your input from these flowing options by pressing the SERIAL NUMBER
6 EX (1,2,3,4,5,6) FROM 1 TO 6 AND ENTER
3
Course Name:    b
Course code:    a
Course pre_requisite:   e
Course cratid:  3
your course got successfully deleted
1: Add course
```

If that course does not exist (show prompt)



```
Select your input from these flowing options by pressing the SERIAL NUMBER
6 EX (1,2,3,4,5,6) FROM 1 TO 6 AND ENTER
3
Course Name:    abu
Course code:    bokkor
Course pre_requisite:   3
Course cratid:  3
your course dosent exist
1: Add course
```

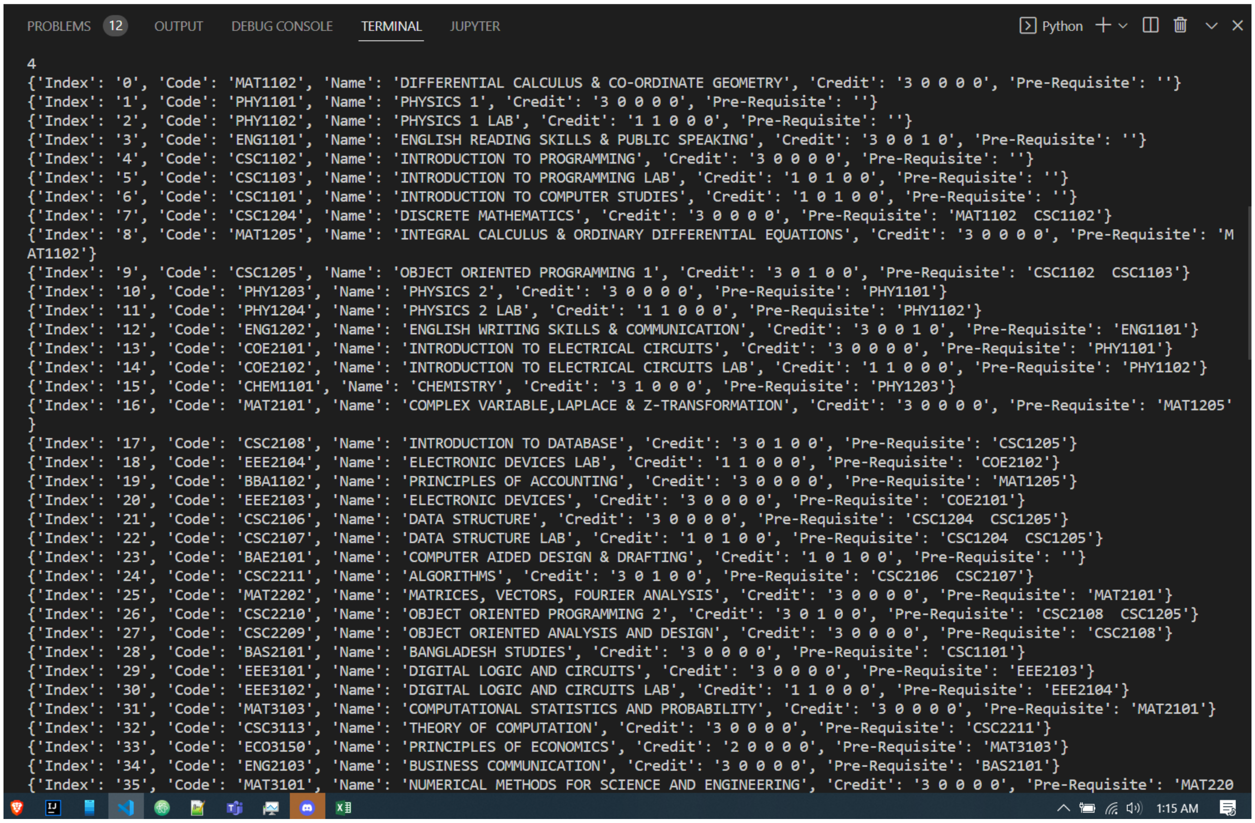
Update Course:



```
99,EEE3103,DIGITAL SIGNAL PROCESSING,3 0 1 0 0,EEE2213
100,EEE4217,VLSI CIRCUIT DESIGN,3 0 1 0 0,EEE4241
101,s,s,4,f
```

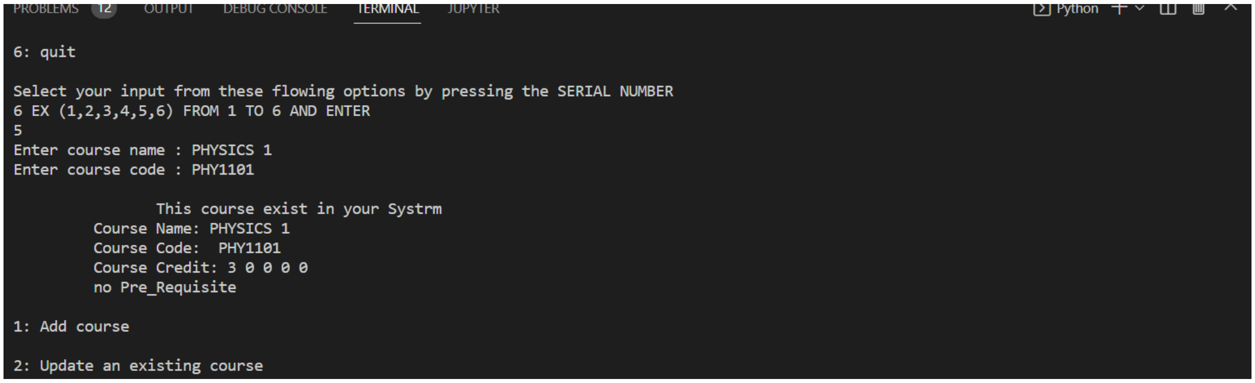
```
6 EX (1,2,3,4,5,6) FROM 1 TO 6 AND ENTER
2
Type course_name :a
Type course_code :a
Type course_prerequizite_code :b
Type course_credit :3
        new updated_code :s
        new updated_name :s
        new updated_credit :4
        new updated_pre_requizite :f
1: Add course

2: Update an existing course

3: Delete an existing course

4: Display information about all the courses

5: Search Course

6: quit
```

Display Courses:



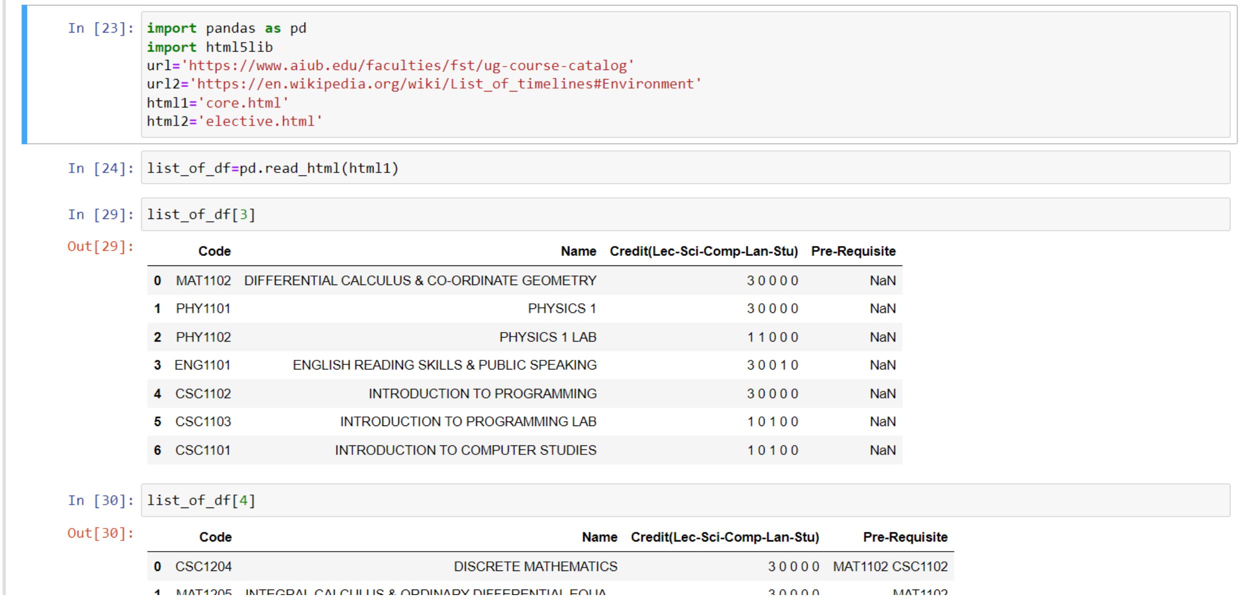
```
4
{'Index': '0', 'Code': 'MAT1102', 'Name': 'DIFFERENTIAL CALCULUS & CO-ORDINATE GEOMETRY', 'Credit': '3 0 0 0 0', 'Pre-Requisite': ''}
{'Index': '1', 'Code': 'PHY1101', 'Name': 'PHYSICS 1', 'Credit': '3 0 0 0 0', 'Pre-Requisite': ''}
{'Index': '2', 'Code': 'PHY1102', 'Name': 'PHYSICS 1 LAB', 'Credit': '1 1 0 0 0', 'Pre-Requisite': ''}
{'Index': '3', 'Code': 'ENG1101', 'Name': 'ENGLISH READING SKILLS & PUBLIC SPEAKING', 'Credit': '3 0 0 1 0', 'Pre-Requisite': ''}
{'Index': '4', 'Code': 'CSC1102', 'Name': 'INTRODUCTION TO PROGRAMMING', 'Credit': '3 0 0 0 0', 'Pre-Requisite': ''}
{'Index': '5', 'Code': 'CSC1103', 'Name': 'INTRODUCTION TO PROGRAMMING LAB', 'Credit': '1 0 1 0 0', 'Pre-Requisite': ''}
{'Index': '6', 'Code': 'CSC1101', 'Name': 'INTRODUCTION TO COMPUTER STUDIES', 'Credit': '1 0 1 0 0', 'Pre-Requisite': ''}
{'Index': '7', 'Code': 'CSC1204', 'Name': 'DISCRETE MATHEMATICS', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'MAT1102  CSC1102'}
{'Index': '8', 'Code': 'MAT1205', 'Name': 'INTEGRAL CALCULUS & ORDINARY DIFFERENTIAL EQUATIONS', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'M
AT1102'}
{'Index': '9', 'Code': 'CSC1205', 'Name': 'OBJECT ORIENTED PROGRAMMING 1', 'Credit': '3 0 1 0 0', 'Pre-Requisite': 'CSC1102  CSC1103'}
{'Index': '10', 'Code': 'PHY1203', 'Name': 'PHYSICS 2', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'PHY1101'}
{'Index': '11', 'Code': 'PHY1204', 'Name': 'PHYSICS 2 LAB', 'Credit': '1 1 0 0 0', 'Pre-Requisite': 'PHY1102'}
{'Index': '12', 'Code': 'ENG1202', 'Name': 'ENGLISH WRITING SKILLS & COMMUNICATION', 'Credit': '3 0 0 1 0', 'Pre-Requisite': 'ENG1101'}
{'Index': '13', 'Code': 'COE2101', 'Name': 'INTRODUCTION TO ELECTRICAL CIRCUITS', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'PHY1101'}
{'Index': '14', 'Code': 'COE2102', 'Name': 'INTRODUCTION TO ELECTRICAL CIRCUITS LAB', 'Credit': '1 1 0 0 0', 'Pre-Requisite': 'PHY1102'}
{'Index': '15', 'Code': 'CHEM1101', 'Name': 'CHEMISTRY', 'Credit': '3 1 0 0 0', 'Pre-Requisite': 'PHY1203'}
{'Index': '16', 'Code': 'MAT2101', 'Name': 'COMPLEX VARIABLE,LAPLACE & Z-TRANSFORMATION', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'MAT1205'
}
{'Index': '17', 'Code': 'CSC2108', 'Name': 'INTRODUCTION TO DATABASE', 'Credit': '3 0 1 0 0', 'Pre-Requisite': 'CSC1205'}
{'Index': '18', 'Code': 'EEE2104', 'Name': 'ELECTRONIC DEVICES LAB', 'Credit': '1 1 0 0 0', 'Pre-Requisite': 'COE2102'}
{'Index': '19', 'Code': 'BBA1102', 'Name': 'PRINCIPLES OF ACCOUNTING', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'MAT1205'}
{'Index': '20', 'Code': 'EEE2103', 'Name': 'ELECTRONIC DEVICES', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'COE2101'}
{'Index': '21', 'Code': 'CSC2106', 'Name': 'DATA STRUCTURE', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'CSC1204  CSC1205'}
{'Index': '22', 'Code': 'CSC2107', 'Name': 'DATA STRUCTURE LAB', 'Credit': '1 0 1 0 0', 'Pre-Requisite': 'CSC1204  CSC1205'}
{'Index': '23', 'Code': 'BAE2101', 'Name': 'COMPUTER AIDED DESIGN & DRAFTING', 'Credit': '1 0 1 0 0', 'Pre-Requisite': ''}
{'Index': '24', 'Code': 'CSC2211', 'Name': 'ALGORITHMS', 'Credit': '3 0 1 0 0', 'Pre-Requisite': 'CSC2106  CSC2107'}
{'Index': '25', 'Code': 'MAT2202', 'Name': 'MATRICES, VECTORS, FOURIER ANALYSIS', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'MAT2101'}
{'Index': '26', 'Code': 'CSC2210', 'Name': 'OBJECT ORIENTED PROGRAMMING 2', 'Credit': '3 0 1 0 0', 'Pre-Requisite': 'CSC2108  CSC1205'}
{'Index': '27', 'Code': 'CSC2209', 'Name': 'OBJECT ORIENTED ANALYSIS AND DESIGN', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'CSC2108'}
{'Index': '28', 'Code': 'BAS2101', 'Name': 'BANGLADESH STUDIES', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'CSC1101'}
{'Index': '29', 'Code': 'EEE3101', 'Name': 'DIGITAL LOGIC AND CIRCUITS', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'EEE2103'}
{'Index': '30', 'Code': 'EEE3102', 'Name': 'DIGITAL LOGIC AND CIRCUITS LAB', 'Credit': '1 1 0 0 0', 'Pre-Requisite': 'EEE2104'}
{'Index': '31', 'Code': 'MAT3103', 'Name': 'COMPUTATIONAL STATISTICS AND PROBABILITY', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'MAT2101'}
{'Index': '32', 'Code': 'CSC3113', 'Name': 'THEORY OF COMPUTATION', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'CSC2211'}
{'Index': '33', 'Code': 'ECO3150', 'Name': 'PRINCIPLES OF ECONOMICS', 'Credit': '2 0 0 0 0', 'Pre-Requisite': 'MAT3103'}
{'Index': '34', 'Code': 'ENG2103', 'Name': 'BUSINESS COMMUNICATION', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'BAS2101'}
{'Index': '35', 'Code': 'MAT3101', 'Name': 'NUMERICAL METHODS FOR SCIENCE AND ENGINEERING', 'Credit': '3 0 0 0 0', 'Pre-Requisite': 'MAT220
```

Search Courses:



```
6: quit

Select your input from these flowing options by pressing the SERIAL NUMBER
6 EX (1,2,3,4,5,6) FROM 1 TO 6 AND ENTER
5
Enter course name : PHYSICS 1
Enter course code : PHY1101

                This course exist in your Systrm
         Course Name: PHYSICS 1
         Course Code:  PHY1101
         Course Credit: 3 0 0 0 0
         no Pre_Requisite

1: Add course

2: Update an existing course
```

Data extraction process:



In [23]:
```
import pandas as pd
import html5lib
url='https://www.aiub.edu/faculties/fst/ug-course-catalog'
url2='https://en.wikipedia.org/wiki/List_of_timelines#Environment'
html1='core.html'
html2='elective.html'
```

In [24]:
```
list_of_df=pd.read_html(html1)
```

In [29]:
```
list_of_df[3]
```

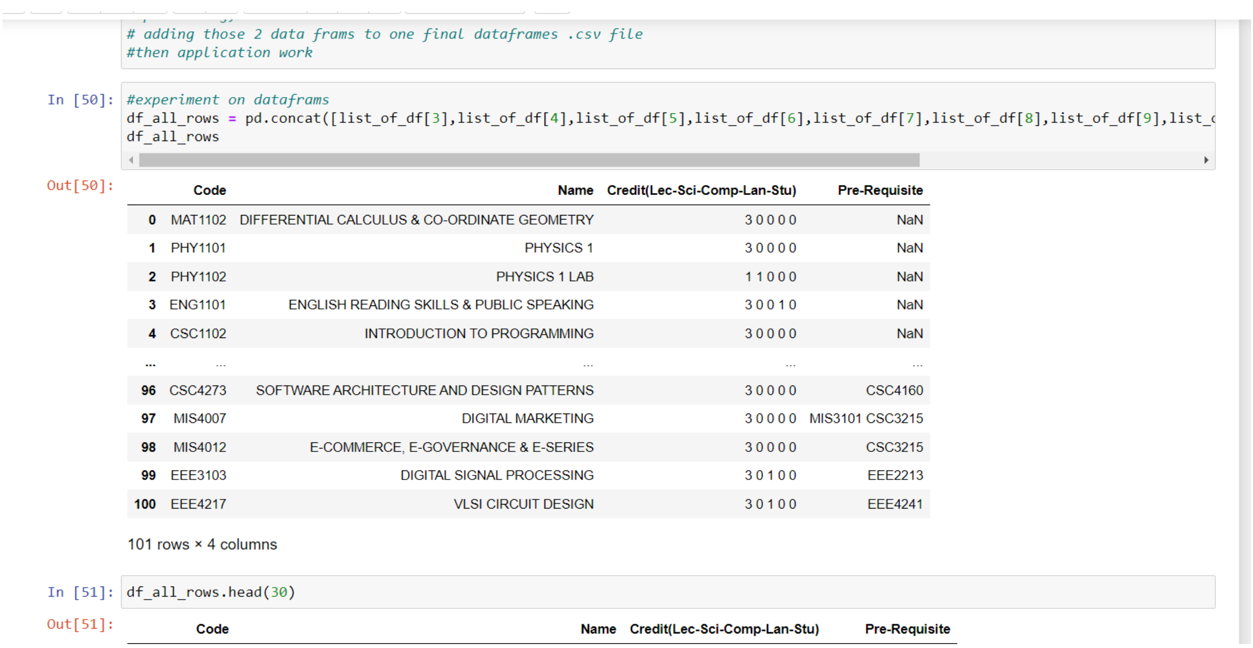
Out[29]:

	Code	Name	Credit(Lec-Sci-Comp-Lan-Stu)	Pre-Requisite
0	MAT1102	DIFFERENTIAL CALCULUS & CO-ORDINATE GEOMETRY	3 0 0 0 0	NaN
1	PHY1101	PHYSICS 1	3 0 0 0 0	NaN
2	PHY1102	PHYSICS 1 LAB	1 1 0 0 0	NaN
3	ENG1101	ENGLISH READING SKILLS & PUBLIC SPEAKING	3 0 0 1 0	NaN
4	CSC1102	INTRODUCTION TO PROGRAMMING	3 0 0 0 0	NaN
5	CSC1103	INTRODUCTION TO PROGRAMMING LAB	1 0 1 0 0	NaN
6	CSC1101	INTRODUCTION TO COMPUTER STUDIES	1 0 1 0 0	NaN

In [30]:
```
list_of_df[4]
```

Out[30]:

	Code	Name	Credit(Lec-Sci-Comp-Lan-Stu)	Pre-Requisite
0	CSC1204	DISCRETE MATHEMATICS	3 0 0 0 0	MAT1102 CSC1102
1	MAT1205	INTEGRAL CALCULUS & ORDINARY DIFFERENTIAL EQUA	3 0 0 0 0	MAT1102



```
# adding those 2 data frams to one final dataframes .csv file
#then application work
```

In [50]:
```
#experiment on dataframs
df_all_rows = pd.concat([list_of_df[3],list_of_df[4],list_of_df[5],list_of_df[6],list_of_df[7],list_of_df[8],list_of_df[9],list_c
df_all_rows
```

Out[50]:

	Code	Name	Credit(Lec-Sci-Comp-Lan-Stu)	Pre-Requisite
0	MAT1102	DIFFERENTIAL CALCULUS & CO-ORDINATE GEOMETRY	3 0 0 0 0	NaN
1	PHY1101	PHYSICS 1	3 0 0 0 0	NaN
2	PHY1102	PHYSICS 1 LAB	1 1 0 0 0	NaN
3	ENG1101	ENGLISH READING SKILLS & PUBLIC SPEAKING	3 0 0 1 0	NaN
4	CSC1102	INTRODUCTION TO PROGRAMMING	3 0 0 0 0	NaN
...	...	...	...	...
96	CSC4273	SOFTWARE ARCHITECTURE AND DESIGN PATTERNS	3 0 0 0 0	CSC4160
97	MIS4007	DIGITAL MARKETING	3 0 0 0 0	MIS3101 CSC3215
98	MIS4012	E-COMMERCE, E-GOVERNANCE & E-SERIES	3 0 0 0 0	CSC3215
99	EEE3103	DIGITAL SIGNAL PROCESSING	3 0 1 0 0	EEE2213
100	EEE4217	VLSI CIRCUIT DESIGN	3 0 1 0 0	EEE4241

101 rows × 4 columns

In [51]:
```
df_all_rows.head(30)
```

Out[51]:

	Code	Name	Credit(Lec-Sci-Comp-Lan-Stu)	Pre-Requisite

Thank you