AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH

Assignment Cover Sheet

Students must complete all details except the faculty use part.

Please submit all assignments to your subject lecturers or the office of the concerned lecturer.

Assignment Title: Develop A Course Management Application using Python

Assignment Number: 01 Due Date: 08/11/2022 Semester: 2022-2023,Fall
Subject Code: CSC4162 Subject Name: Programming in Python Section: A

Course Instructor: Akinul Islam Jony Degree Program: BSc CSE

Declaration and Statement of Authorship:

1. I/we hold a copy of this assignment, which can be produced if the original is lost/ damaged.
2. This assignment 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 has been written for me/us by any other person except where such collaboration has been authorized by the lecturer/teacher concerned 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 School for review and comparison, including review by external examiners.

I/we understand that

7. Plagiarism is the presentation of the work, idea or creation of another person as though it is your own. It is a form of cheating and is a very serious academic offence that may lead to expulsion from the University. 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 the material used is not appropriately cited.
8. Enabling plagiarism is the act of assisting or allowing another person to plagiarize or to copy your work

Group Name (if applicable):

No.	Student Name	Student Number	Student Signature	Date
1	A.S.M. FAZLE RABBI	19-39714-1		08/11/22
2				
3				
4				
5				
6				

For faculty use only:

Total Marks: _______ Marks Obtained: _______

Faculty comments____

____

____

Project overview: The project was to develop a course management system using python. This is a console-based application. There are eight functions to accomplish several actions, these are:

1. User Interface
2. Check Prerequisite
3. Add Course
4. Update Course
5. Delete Course
6. Show All Course
7. Search Course
8. Store Course to Text

The user interface displays a menu to the user that they can chose any function from given list or type “quit” in order to exit the system. Check prerequisite checks if the prerequisite is already in the course catalogue or not while adding prerequisite in add course. Add course prompt user to add a new course by taking course information like course code, course title, course credit hour/s and prerequisite/s from the user. Update course update the information of existing course. Delete course can be used to delete any existing course from the system. The show all course function displays the information of all the course in the system. Search course function can be used to search any specific course using the course code. Store course to text function store the course stored in the system to a text file so the user can have a physical copy in their hard drive.

Project solution design: In order to complete this project a class “Course” is created so that object of each course can be created. A dictionary information is taken to store course information such as course code, course title, course credit and course prerequisites. A function called user interface is created so that it can displays list of options available for user to select to perform certain task as well as to take input which option they want to choose or the user can type “quit” in order to exit the system. Check prerequisite check if the prerequisite is in the course catalogue or not it uses a for loop to search in the course catalogue if the course prerequisite is in the course code of any course. The add course function prompt user to give information such course code, course title, course credit hour/s and course prerequisite/s. If the prerequisite given is either “N/A” or something that already exist in the course catalogue then an object of Course class is created and it is stored in course catalogue list and the user is notified with a confirmation text but if the course prerequisite doesn’t exist add course is prompted and user is requested to add the course first before specifying it as prerequisite of any other course, this is done using the check prerequisite function inside the add course function.

For update course function the user first search the course he wants to update with the course code. A for loop is used in order to search the course in course catalogue list, a Boolean flag is used in order to identify if the course is found or not, if the flag returns true the course information is displayed and previous information such as course title, course credit and course prerequisites/s can be updated though prompted user inputs. If the flag shows false the system displays an error message for the user.

For Delete course function the user first search the course he wants to delete with the course code. A for loop is used in order to search the course in course catalogue list, a Boolean flag is used in order to identify if the course is found or not, if the flag returns true the course information is displayed. An input is taken to verify if the user really wants to deleted the course, if 'y' is given as input the course is deleted from the list using built-in remove function. If the course is not found the system displays an error message.

In the show all course function if the length of course catalogue list is zero it displays message to add course first, if the list is not empty with the help of for loop and range function it displays all the stored course in course catalogue.

Search course function take input from user as the course code and search the course catalogue list with a for loop and Boolean flag, if the searched course code matches any course in the course catalogue the flag returns true and information of that individual course is displayed if the course code doesn't match it gives an error message and prompt to add course function.

The store course to text function uses the python file system to store the courses from course catalogue so the user can have a physical copy. First a text file is taken and open in writing mode, then using a for loop the information of course catalogue from top to bottom are written in the file. A conformation message is displayed once all the lines are written in the text file.

Finally using a while loop and if/elif condition user's choice are taken from option one to six, the while loop is terminated once the input is "quit".

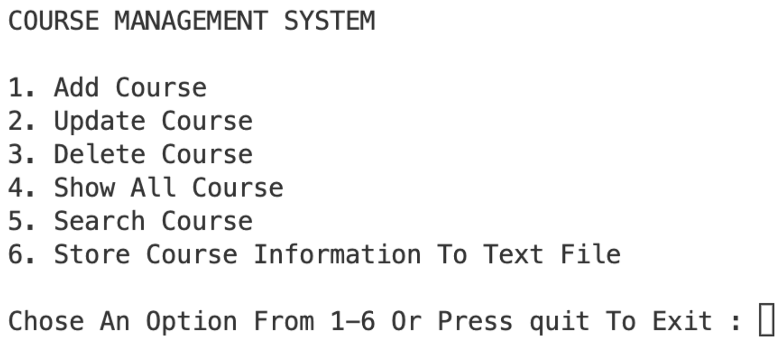
Implementation:

```
class Course:
    """For storing information dictionary as object"""
    def __init__(self,course_code,course_title,course_credit,course_prerequisite):
        self.information = {}
        self.information['course_code'] = course_code
        self.information['course_title'] = course_title
        self.information['course_credit'] = course_credit
        self.information['course_prerequisite'] = course_prerequisite

user_interface_input = 0

def user_interface ():
    """Displaying Menu/User Interface"""
    print ("\nCOURSE MANAGEMENT SYSTEM\n")
    print ("1. Add Course")
    print ("2. Update Course")
    print ("3. Delete Course")
    print ("4. Show All Course")
    print ("5. Search Course")
    print ("6. Store Course Information To Text File\n")
    user_interface_input =input("Chose An Option From 1-6 Or Press quit To Exit : ")
    return user_interface_input

course_catalogue =[]

def check_prequisite(course_prerequisite):
    """Checking if the prerequisite exist or not"""
    for course in course_catalogue:
        if course_prerequisite in course.information['course_code']:
            return True
    return False

def add_course ():
    """To add new course in the course catalogue list"""
    print ("\n*** ADD NEW COURSE ***")
    course_code = input ("\nEnter Course Code : ")
    course_title = input ("Enter Course Name : ")
    course_credit = input ("Enter Course Credit Hour : ")
    course_prerequisite = input ("Enter Course Prerequisite : ")
    if course_prerequisite == 'N/A':
        course_catalogue.append (Course(course_code,course_title,course_credit,course_prerequisite))
        print ("\n*** Course Added ***")
    elif check_prequisite(course_prerequisite):
        course = Course(course_code,course_title,course_credit,course_prerequisite)
        course_catalogue.append(course)
        print ("\n*** Course Added ***")
    else:
        print ("\n*** Course Prerequisite Doesn't Exist, Add It First ***")
        add_course()
```

```python
def update_course ():
    """Update Course Information"""
    print ("\n*** UPDATE COURSE ***")
    found = False
    course_code = input ("\nSearch the Course You want To Update, Search With Course Code: ")
    for course in course_catalogue:
        if course_code in course.information['course_code']:
            found = True
            print ("\n*** Course Found! Course Information ***\n")
            print (f"Course Code: {course.information['course_code']}")
            print (f"Course Name: {course.information['course_title']}")
            print (f"Course Credit Hour: {course.information['course_credit']}")
            print (f"Course Prerequisite: {course.information['course_prerequisite']}\n")
            print ("\n*** UPDATE INFORMATION ***\n")
            course.information['course_title']=input("Enter New Course Name: ")
            course.information['course_credit']=input("Enter New Credit Hour: ")
            course.information['course_prerequisite']=input("Enter New Prerequisite: ")
            print ("\n*** Course Updated ***")

    if(found==False):
        print("\n*** Sorry, Course doesn't exist ***")


def delete_course ():
    """Deleteing Course From Course Catalogue"""
    print ("\n*** DELETE COURSE ***")
    found = False
    course_code = input ("\nEnter Course Code of The Course To Be Deleted: ")
    for course in course_catalogue:
        if course_code in course.information['course_code']:
            found = True
            print ("\n*** Course Found! Course Information ***\n")
            print (f"Course Code: {course.information['course_code']}")
            print (f"Course Name: {course.information['course_title']}")
            print (f"Course Credit Hour: {course.information['course_credit']}")
            print (f"Course Prerequisite: {course.information['course_prerequisite']}\n")
            user_input_delete = input("Do You Want To Delete This Course, Press y To Delete The Course: ")
            if (user_input_delete == "y"):
                course_catalogue.remove (course)
                print ("\n*** Course Deleted ***")
    if(found==False):
        print("\n*** Sorry, Course doesn't exist ***")

def show_all_course ():
    """Display Information of All Course"""
    if len(course_catalogue)== 0:
        print ("\n*** Nothing To Show, Add Course ***")
    else:
        print ("\n*** Course Catalogue ***\n")
    for i in range(len(course_catalogue)):
        print (f"Course Code: {course_catalogue[i].information['course_code']}")
        print (f"Course Name: {course_catalogue[i].information['course_title']}")
        print (f"Course Credit Hour: {course_catalogue[i].information['course_credit']}")
        print (f"Course Prerequisite: {course_catalogue[i].information['course_prerequisite']}\n")
```

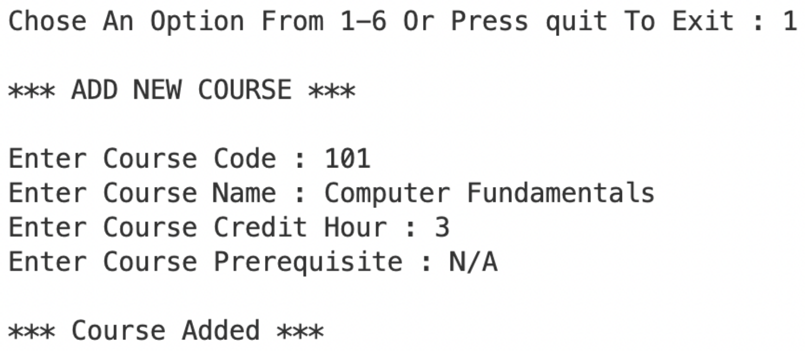
```
def search_course ():
    """Search Course And Displays Information If Found"""
    print ("\n*** SEARCH COURSE ***")
    found = False
    course_code = input ("\nEnter Course Code To Search: ")
    for course in course_catalogue:
        if course_code in course.information['course_code']:
            found = True
            print ("\n*** Course Found! Course Information ***\n")
            print (f"Course Code: {course.information['course_code']}")
            print (f"Course Name: {course.information['course_title']}")
            print (f"Course Credit Hour: {course.information['course_credit']}")
            print (f"Course Prerequisite: {course.information['course_prerequisite']}\n")
    if(found==False):
        print("\n*** Sorry, Course doesn't exist ***")
        add_course ()

def store_course_to_text ():
    """Store Course From System To A Text File"""
    file_name="course_catalogue.txt"
    with open(file_name,'w') as file_obj:
        for i in range(len(course_catalogue)):
            file_obj.writelines("\n")
            file_obj.writelines(f"Course Code: {course_catalogue[i].information['course_code']}\n")
            file_obj.writelines(f"Course Name: {course_catalogue[i].information['course_title']}\n")
            file_obj.writelines(f"Course Credit Hour: {course_catalogue[i].information['course_credit']}\n")
            file_obj.writelines(f"Course Prerequisite: {course_catalogue[i].information['course_prerequisite']}\n")
        print ("\n*** Course Information Stored ***")


user_interface_input = user_interface ()

while(user_interface_input != "quit"):

    if(user_interface_input == "1"):
        add_course()
        user_interface_input = user_interface ()
    elif(user_interface_input == "2"):
        update_course()
        user_interface_input = user_interface ()
    elif(user_interface_input == "3"):
        delete_course()
        user_interface_input = user_interface ()
    elif(user_interface_input == "4"):
        show_all_course()
        user_interface_input = user_interface ()
    elif(user_interface_input == "5"):
        search_course()
        user_interface_input = user_interface ()
    elif(user_interface_input == "6"):
        store_course_to_text()
        user_interface_input = user_interface ()
```

Project Outcomes: When we run the application Course Management System Menu is displayed by user interface function. It offers user six options to choose from and can exit the system if "quite" is typed.



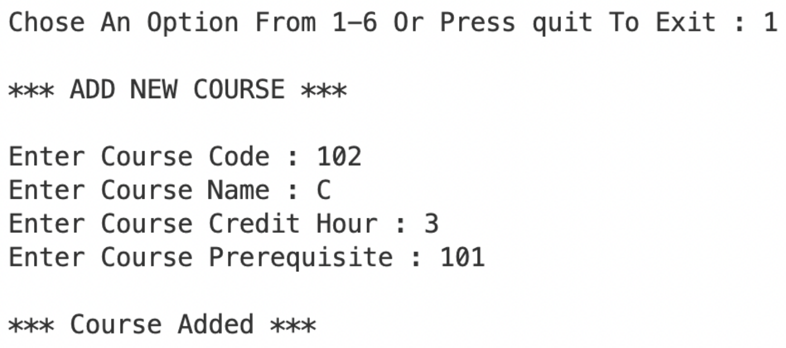
```
COURSE MANAGEMENT SYSTEM

1. Add Course
2. Update Course
3. Delete Course
4. Show All Course
5. Search Course
6. Store Course Information To Text File

Chose An Option From 1-6 Or Press quit To Exit :
```

When the user choose option 1 from the menu add course function is prompted and it takes four inputs such as course code, course name, course credit hour and course prerequisite/s from the user. After user provide all information if the course prerequisite is of existing course code or "N/A" a confirmation message "Course Added" is displayed. Else user is requested to add the course first and add course is prompted from the beginning.



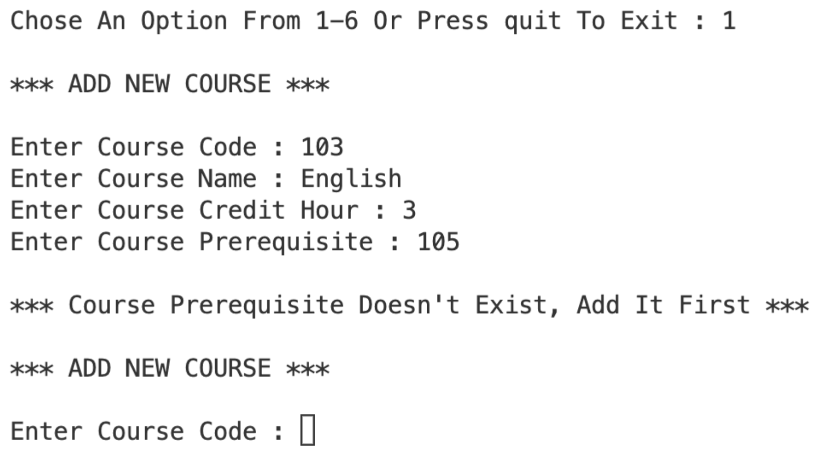
```
Chose An Option From 1-6 Or Press quit To Exit : 1

*** ADD NEW COURSE ***

Enter Course Code : 101
Enter Course Name : Computer Fundamentals
Enter Course Credit Hour : 3
Enter Course Prerequisite : N/A

*** Course Added ***
```

Fig: Adding course with no prerequisite



```
Chose An Option From 1-6 Or Press quit To Exit : 1

*** ADD NEW COURSE ***

Enter Course Code : 102
Enter Course Name : C
Enter Course Credit Hour : 3
Enter Course Prerequisite : 101

*** Course Added ***
```

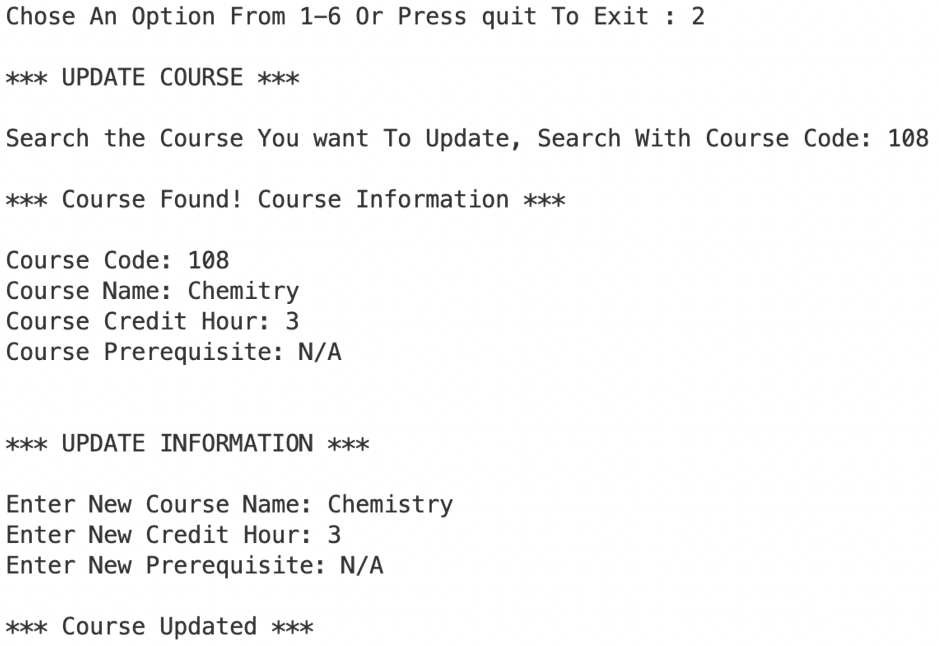
Fig: Adding course with existing course for prerequisite



```
Chose An Option From 1-6 Or Press quit To Exit : 1

*** ADD NEW COURSE ***

Enter Course Code : 103
Enter Course Name : English
Enter Course Credit Hour : 3
Enter Course Prerequisite : 105

*** Course Prerequisite Doesn't Exist, Add It First ***

*** ADD NEW COURSE ***

Enter Course Code :
```

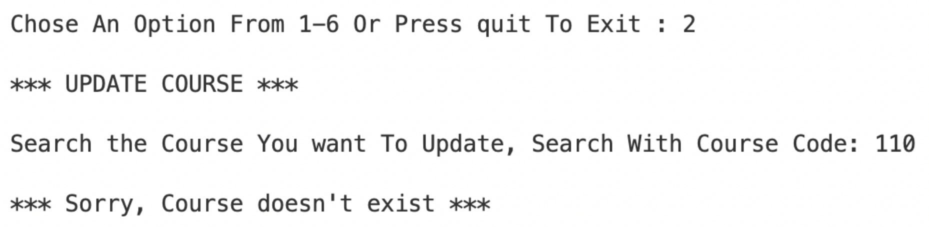
Fig: Adding course with non-existing course for prerequisite

If the user wants to update any course information, he chooses 2 and update course function is prompted. It first asks the user to search the course he wants to update with the course code. If the course is found it gives a confirmation message and displays the course information. Then user input is taken such as course name, course credit and course prerequisite in order to overwrite the previous information, once it is done it displays a
“Course Updated” message, in case the course doesn’t exist it shows an error message “Sorry, Course doesn't exist”.



```
Chose An Option From 1-6 Or Press quit To Exit : 2

*** UPDATE COURSE ***

Search the Course You want To Update, Search With Course Code: 108

*** Course Found! Course Information ***

Course Code: 108
Course Name: Chemitry
Course Credit Hour: 3
Course Prerequisite: N/A


*** UPDATE INFORMATION ***

Enter New Course Name: Chemistry
Enter New Credit Hour: 3
Enter New Prerequisite: N/A

*** Course Updated ***
```

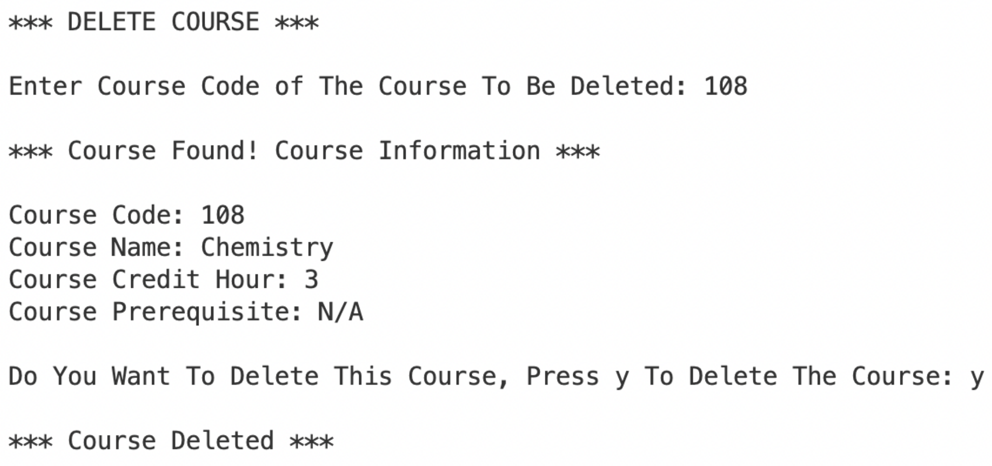
Fig: Updating Course Information



```
Chose An Option From 1-6 Or Press quit To Exit : 2

*** UPDATE COURSE ***

Search the Course You want To Update, Search With Course Code: 110

*** Sorry, Course doesn't exist ***
```

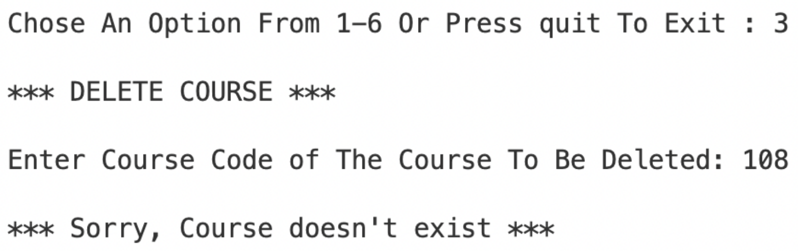
Fig: If the course is not found for updating

In option 3 user can delete course, delete course function is initiated and it ask user to enter the course code of the course he wants to delete, if the course is found it shows course found message and displays the course’s detailed in formation and it takes a confirmation from the user and “Course Deleted” message is shown once the course is deleted.



```
*** DELETE COURSE ***

Enter Course Code of The Course To Be Deleted: 108

*** Course Found! Course Information ***

Course Code: 108
Course Name: Chemistry
Course Credit Hour: 3
Course Prerequisite: N/A

Do You Want To Delete This Course, Press y To Delete The Course: y

*** Course Deleted ***
```

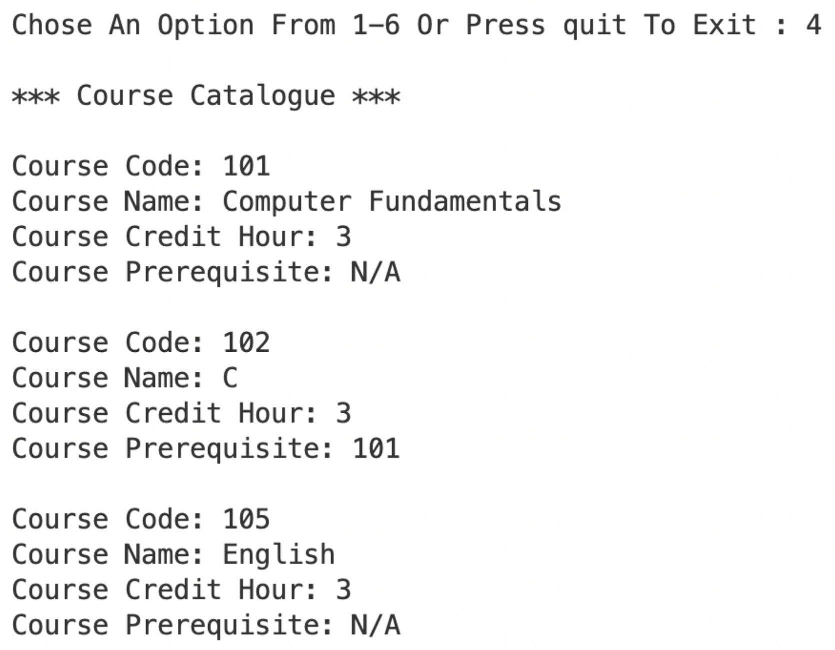
Fig: Deleting Course



```
Chose An Option From 1-6 Or Press quit To Exit : 3

*** DELETE COURSE ***

Enter Course Code of The Course To Be Deleted: 108

*** Sorry, Course doesn't exist ***
```

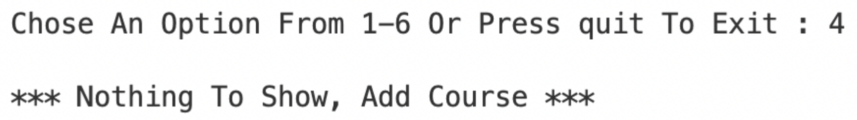
Fig: If the course doesn’t exist for deleting

In the option 4, the user can check all course in the course catalogue and their detailed information unless the catalogue is empty. For this show all course function is used.



```
Chose An Option From 1-6 Or Press quit To Exit : 4

*** Course Catalogue ***

Course Code: 101
Course Name: Computer Fundamentals
Course Credit Hour: 3
Course Prerequisite: N/A

Course Code: 102
Course Name: C
Course Credit Hour: 3
Course Prerequisite: 101

Course Code: 105
Course Name: English
Course Credit Hour: 3
Course Prerequisite: N/A
```

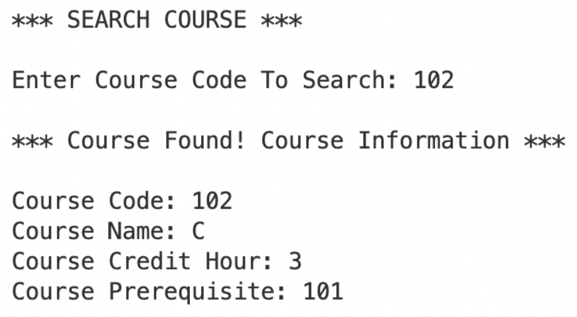
Fig: Show all course



```
Chose An Option From 1-6 Or Press quit To Exit : 4

*** Nothing To Show, Add Course ***
```

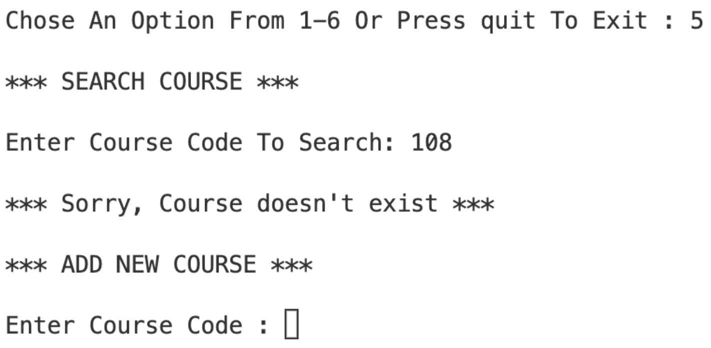
Fig: Empty Course Catalogue

Option 5 is search course option, search course function is used for this feature, it asks user the course they want to search and user gives the course code as the input. If the course is found "Course Found! Course Information" message is displayed along with the detailed information of that course. If the course code doesn't exist it prompt add course function following error message.



```
*** SEARCH COURSE ***

Enter Course Code To Search: 102

*** Course Found! Course Information ***

Course Code: 102
Course Name: C
Course Credit Hour: 3
Course Prerequisite: 101
```

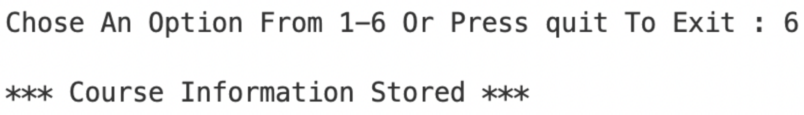
Fig: Search Course



```
Chose An Option From 1-6 Or Press quit To Exit : 5

*** SEARCH COURSE ***

Enter Course Code To Search: 108

*** Sorry, Course doesn't exist ***

*** ADD NEW COURSE ***

Enter Course Code :
```

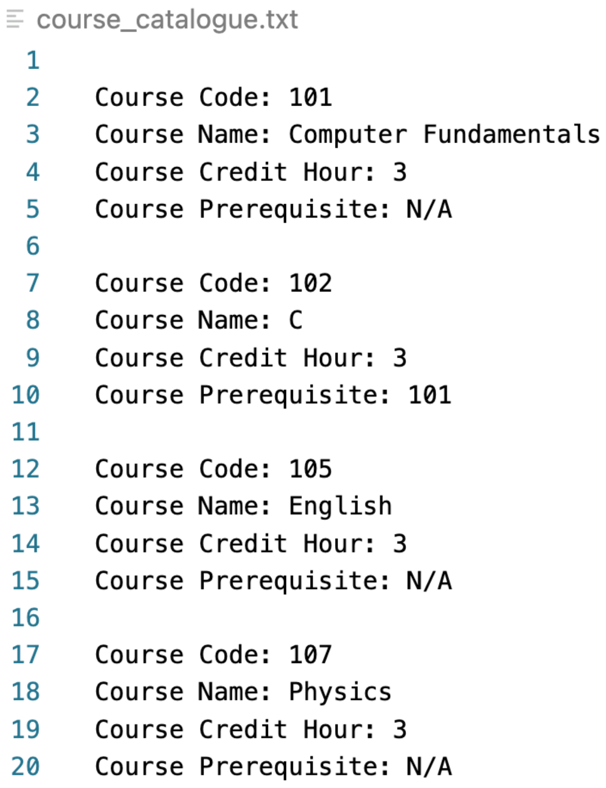
Fig: If the course doesn't exist upon searching

If the user chose option 6, they can store the courses from the course catalogue to a text file so they can have a physical copy for any further use. Store course to text function is used to implement this feature. After the course is store a conformation message "Course Information Stored" is displayed.



```
Chose An Option From 1-6 Or Press quit To Exit : 6

*** Course Information Stored ***
```

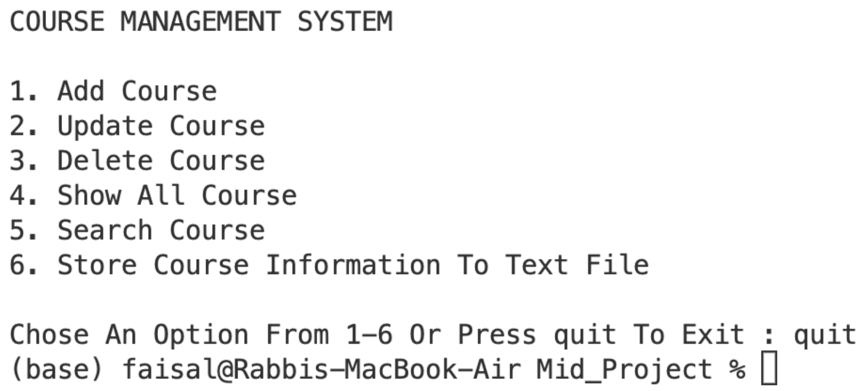
Fig: Storing Course into Text File



course_catalogue.txt

```

Course Code: 101
Course Name: Computer Fundamentals
Course Credit Hour: 3
Course Prerequisite: N/A

Course Code: 102
Course Name: C
Course Credit Hour: 3
Course Prerequisite: 101

Course Code: 105
Course Name: English
Course Credit Hour: 3
Course Prerequisite: N/A

Course Code: 107
Course Name: Physics
Course Credit Hour: 3
Course Prerequisite: N/A
```

Fig: Text File



```
COURSE MANAGEMENT SYSTEM

1. Add Course
2. Update Course
3. Delete Course
4. Show All Course
5. Search Course
6. Store Course Information To Text File

Chose An Option From 1-6 Or Press quit To Exit : quit
(base) faisal@Rabbis-MacBook-Air Mid_Project %
```

Fig: If user input is “quit”