तत् त्वं पूषन् अपावृणु
केन्द्रीय विद्यालय संगठन

CENTRAL BOARD OF SECONDARY EDUCATION

PM SHRI Kendriya Vidyalaya-3 9 B.R.D.

A.F.S. Pune- 411014

A PROJECT RECORD FILE IS SUBMITTED FOR THE COMPUTER SCIENCE, CLASS 12, SESSION 2024-25

SUBMITTED BY:- AMAN MONDAL

CLASS:- 12th A

ROLL NUMBER:-

SUBJECT TEACHER(CS):- SUSHMA SINGH CHOUHAN

CERTIFICATE

This is to certify that **Aman Mondal** student of class XI, **Kendriya Vidyalaya-3 9 B.R.D. A.F.S. Pune- 411014** has completed the **Practical File** during the academic year 2024-25 towards partial fulfilment of credit for the **Computer Science** project evaluation of 2024-25 and submitted satisfactory report, as compiled in the following pages, under my supervision.

Internal Examiner Signature

External Examiner Signature

Date:

School Seal

Principal Signature

ACKNOWLEDGEMENT

I wish to express my deep sense of gratitude and indebtedness to our learned teacher **Sushma Singh Chouhan, C.S. Teacher, Kendriya Vidyalaya-3 9 B.R.D. A.F.S. Pune- 411014** for her invaluable help, advice and guidance in the preparation of this project.

I am also greatly indebted to our Principal
Mr. Md. Nassimuddin and school authorities for providing me with the facilities and requisite laboratory conditions for making this Project file.

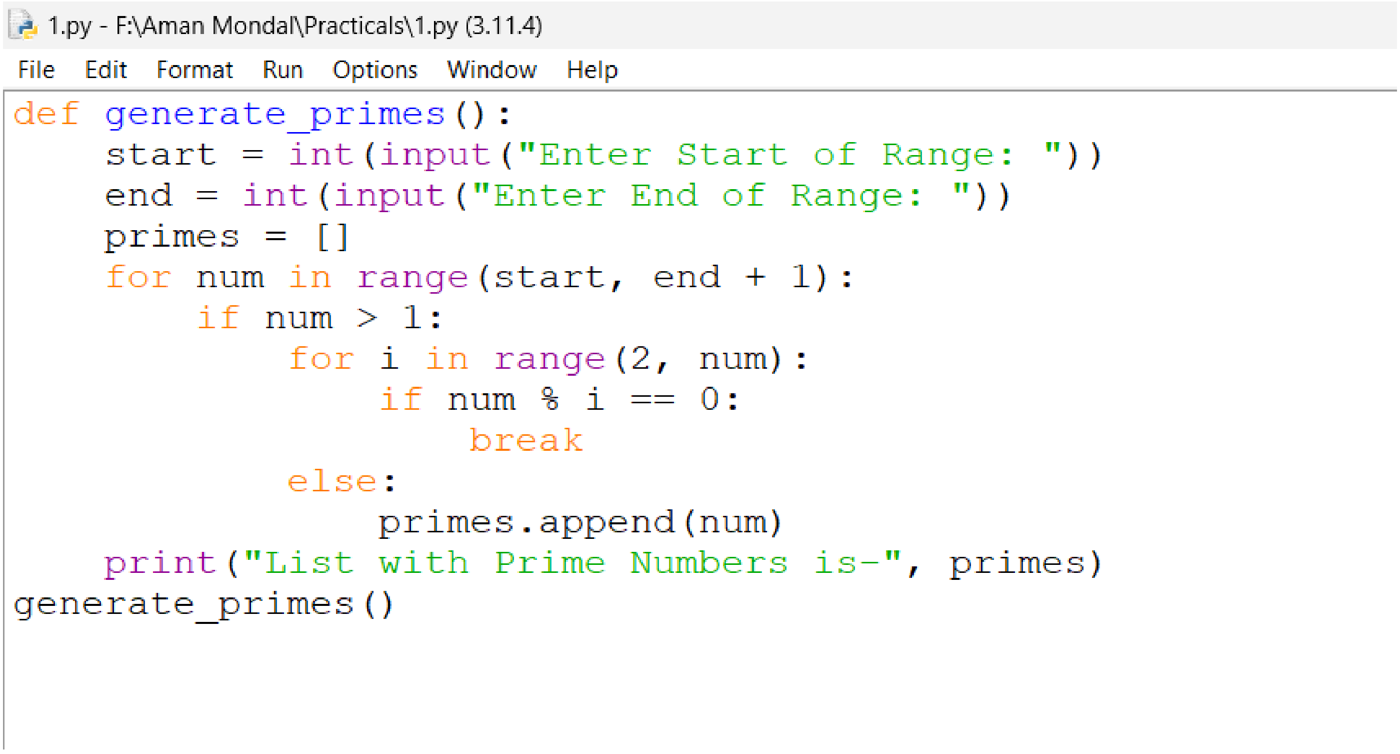
I also extend my thanks to a number of teachers,
my classmates and my friends who helped me to complete this Project file successfully.

Aman Mondal

Index

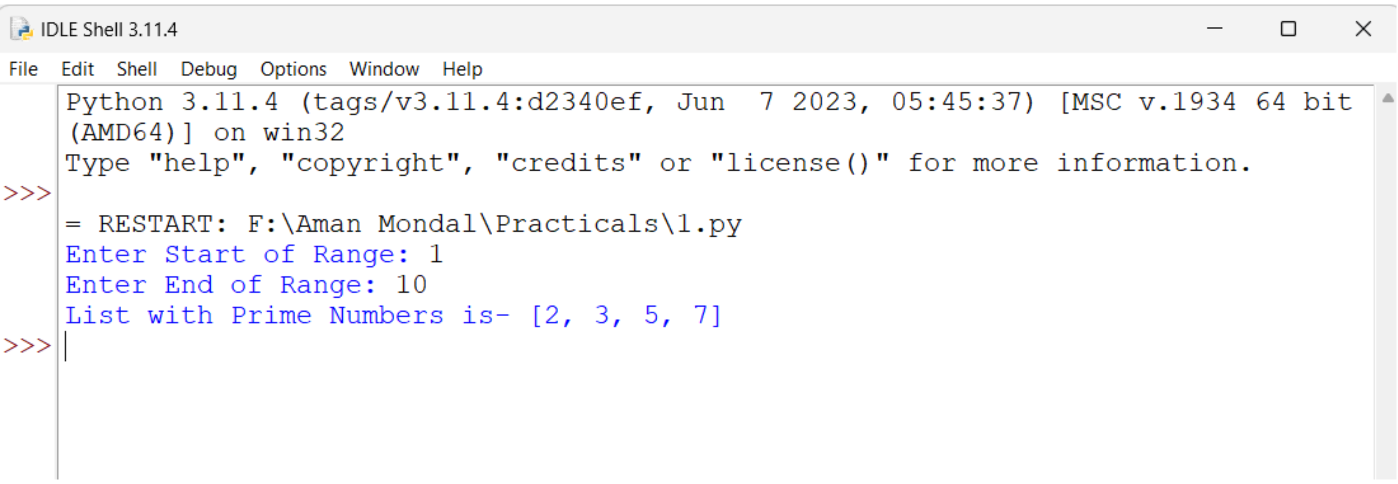
Q1. Write a Python program to generate prime numbers for a given range.

Source Code:-


1.py - F:\Aman Mondal\Practicals\1.py (3.11.4)
File Edit Format Run Options Window Help

```python
def generate_primes():
    start = int(input("Enter Start of Range: "))
    end = int(input("Enter End of Range: "))
    primes = []
    for num in range(start, end + 1):
        if num > 1:
            for i in range(2, num):
                if num % i == 0:
                    break
            else:
                primes.append(num)
    print("List with Prime Numbers is-", primes)
generate_primes()
```

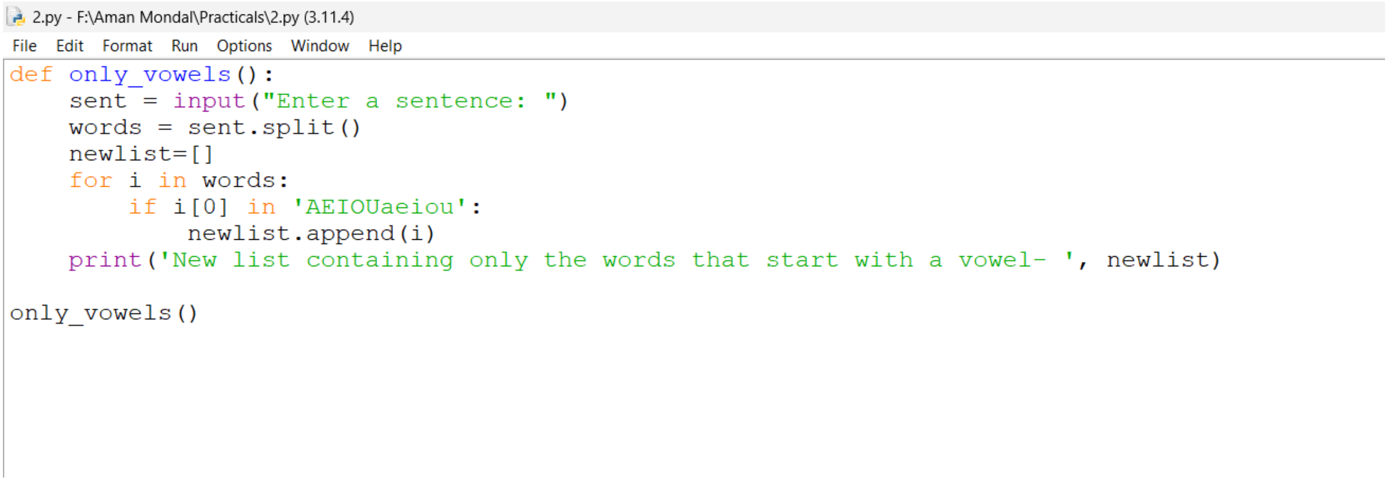
Output:-



```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: F:\Aman Mondal\Practicals\1.py
Enter Start of Range: 1
Enter End of Range: 10
List with Prime Numbers is- [2, 3, 5, 7]
>>>
```

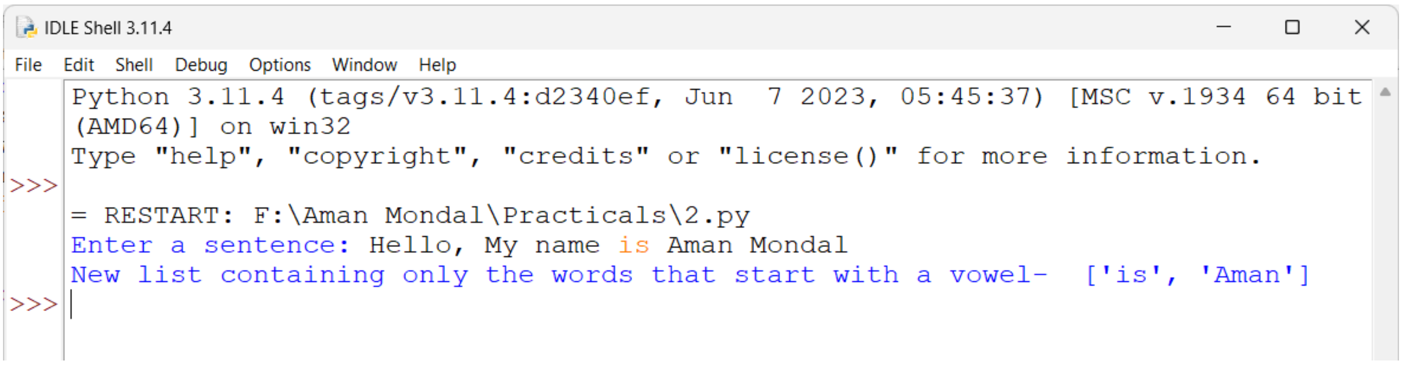
Q2. Develop a program that takes a list of words and returns a new list containing only the words that start with a vowel.

Source Code:-


2.py - F:\Aman Mondal\Practicals\2.py (3.11.4)
File Edit Format Run Options Window Help

```python
def only_vowels():
    sent = input("Enter a sentence: ")
    words = sent.split()
    newlist=[]
    for i in words:
        if i[0] in 'AEIOUaeiou':
            newlist.append(i)
    print('New list containing only the words that start with a vowel- ', newlist)

only_vowels()
```

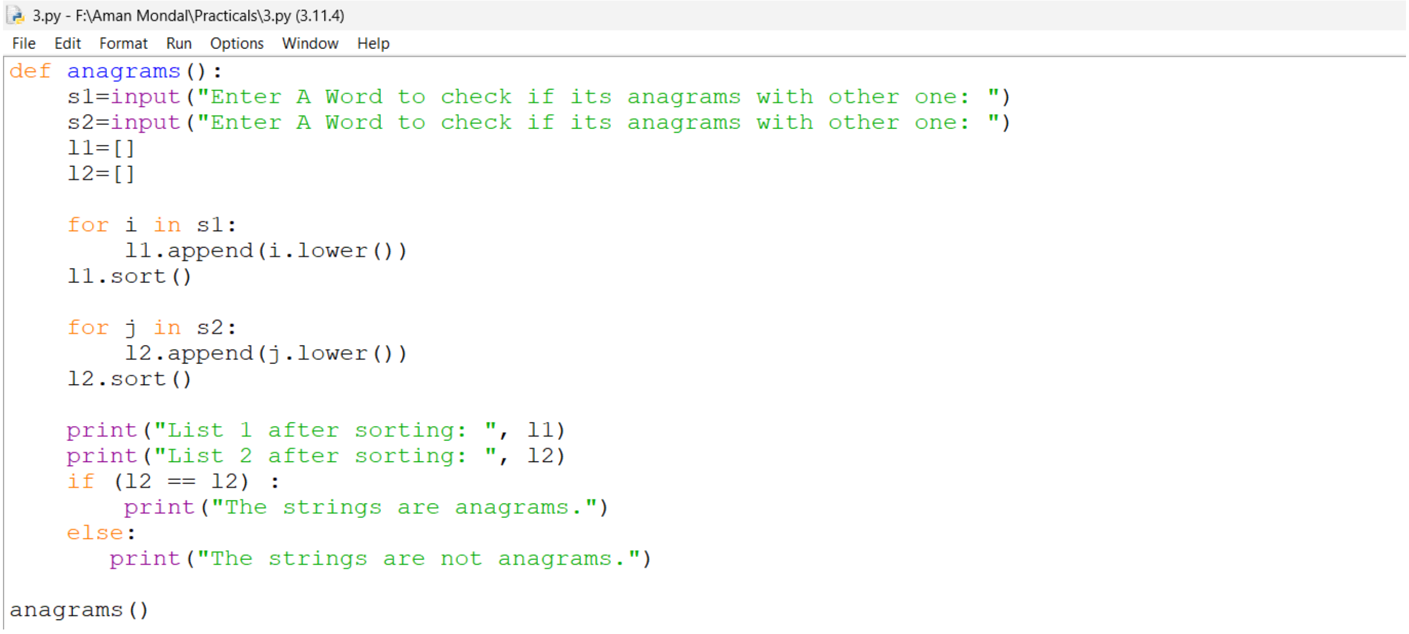
Output:-



```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: F:\Aman Mondal\Practicals\2.py
Enter a sentence: Hello, My name is Aman Mondal
New list containing only the words that start with a vowel-  ['is', 'Aman']
>>> 
```

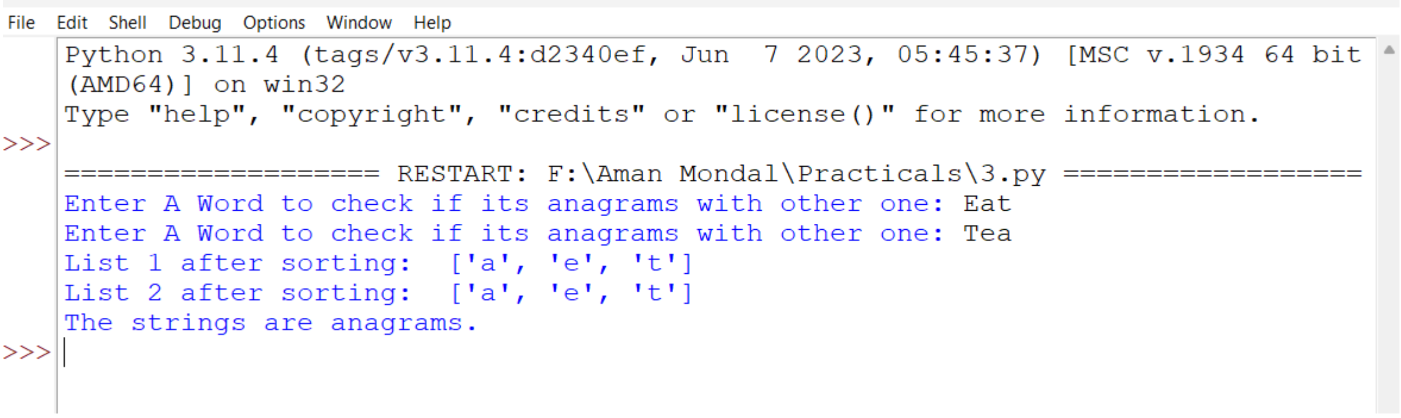
Q3. Write a function that takes two strings as input and checks whether they are anagrams of each other

Source Code:-


3.py - F:\Aman Mondal\Practicals\3.py (3.11.4)
File Edit Format Run Options Window Help

```
def anagrams():
    s1=input("Enter A Word to check if its anagrams with other one: ")
    s2=input("Enter A Word to check if its anagrams with other one: ")
    l1=[]
    l2=[]

    for i in s1:
        l1.append(i.lower())
    l1.sort()

    for j in s2:
        l2.append(j.lower())
    l2.sort()

    print("List 1 after sorting: ", l1)
    print("List 2 after sorting: ", l2)
    if (l2 == l2) :
        print("The strings are anagrams.")
    else:
       print("The strings are not anagrams.")

anagrams()
```

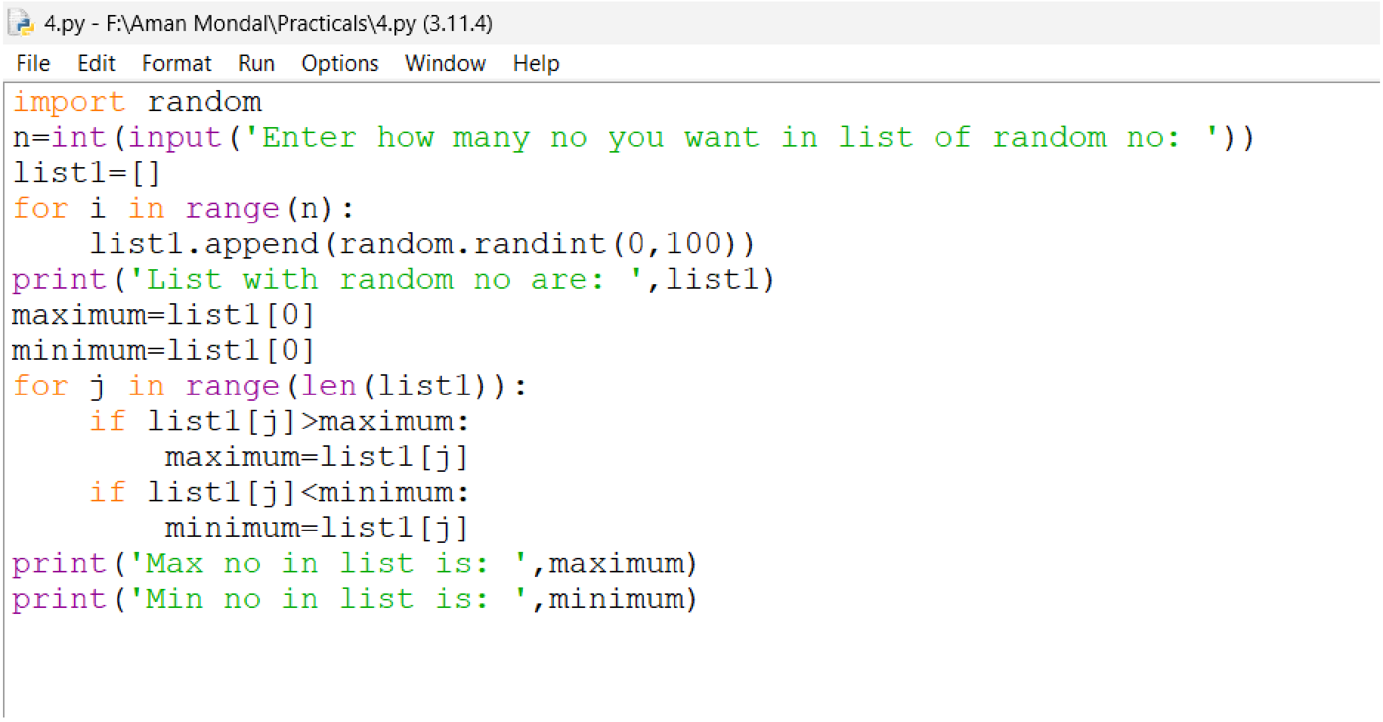
Output:-



```
    Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit
    (AMD64)] on win32
    Type "help", "copyright", "credits" or "license()" for more information.
>>>
    =================== RESTART: F:\Aman Mondal\Practicals\3.py ==================
    Enter A Word to check if its anagrams with other one: Eat
    Enter A Word to check if its anagrams with other one: Tea
    List 1 after sorting:  ['a', 'e', 't']
    List 2 after sorting:  ['a', 'e', 't']
    The strings are anagrams.
>>>
```

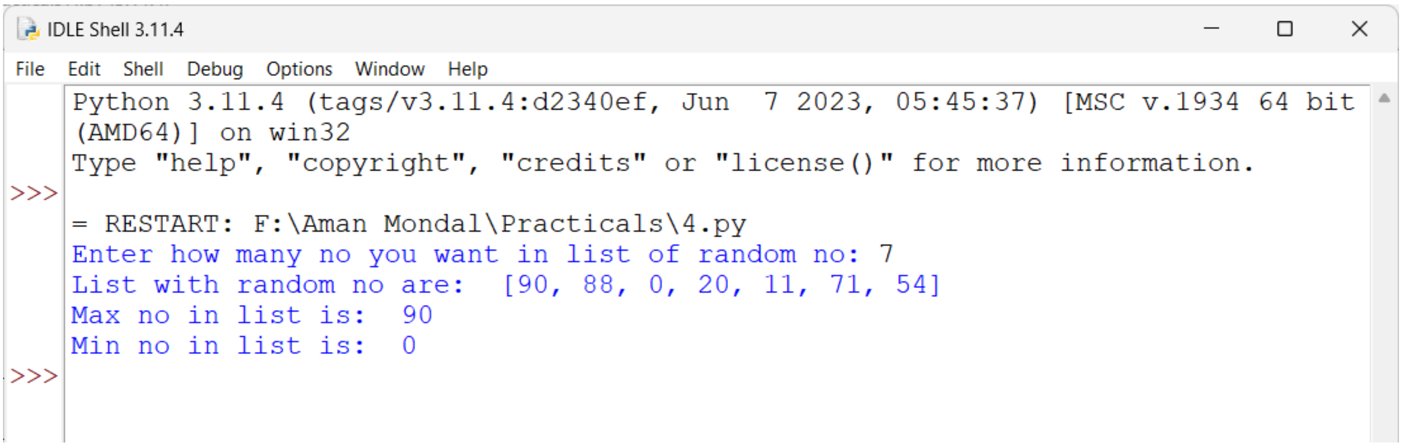
Q4. Create a program that generates a random list of numbers and finds the largest and smallest numbers without using built-in functions like max() and min().

Source Code:-


4.py - F:\Aman Mondal\Practicals\4.py (3.11.4)
File Edit Format Run Options Window Help

```
import random
n=int(input('Enter how many no you want in list of random no: '))
list1=[]
for i in range(n):
    list1.append(random.randint(0,100))
print('List with random no are: ',list1)
maximum=list1[0]
minimum=list1[0]
for j in range(len(list1)):
    if list1[j]>maximum:
        maximum=list1[j]
    if list1[j]<minimum:
        minimum=list1[j]
print('Max no in list is: ',maximum)
print('Min no in list is: ',minimum)
```

Output:-

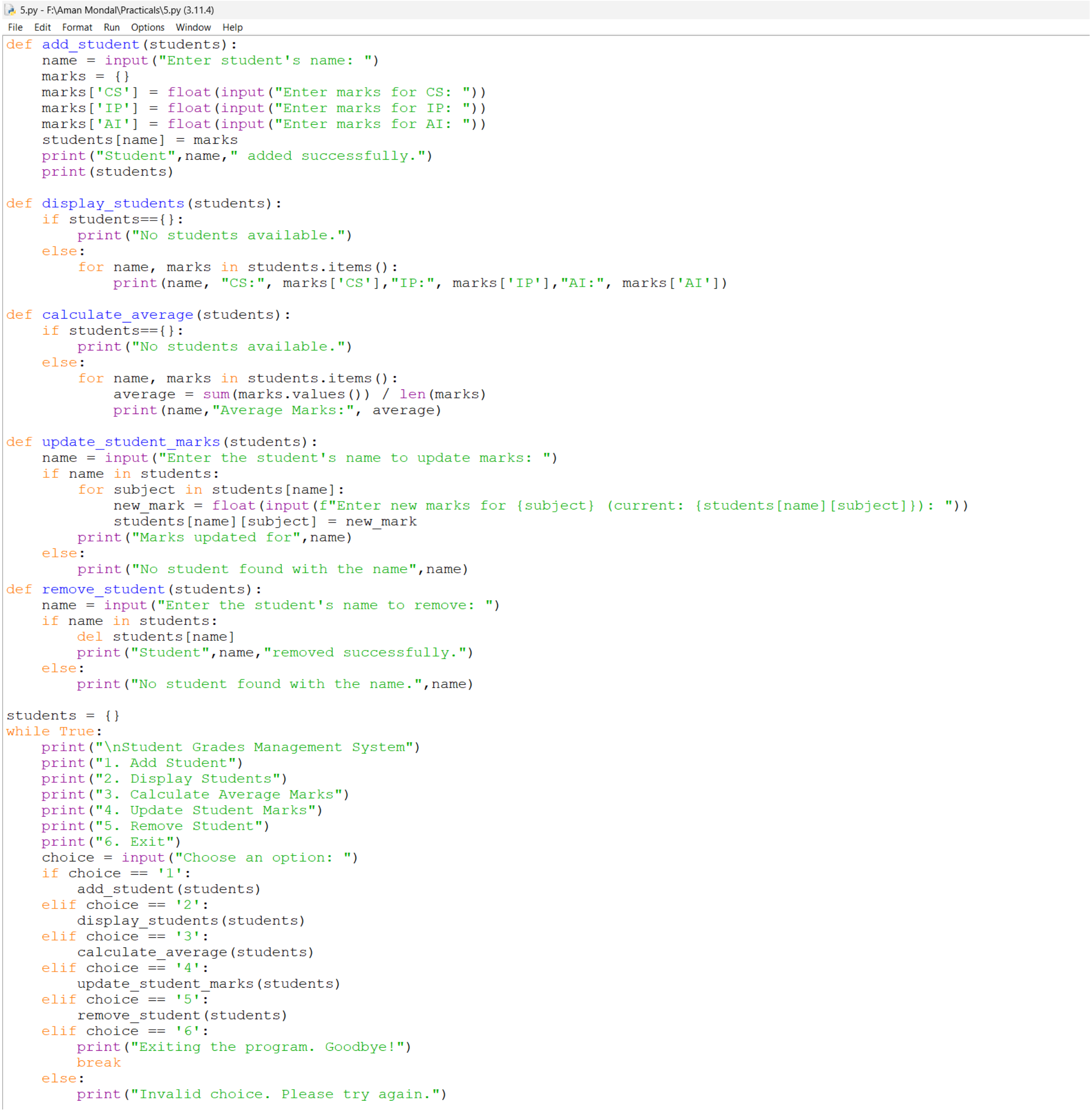


```
    Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit
    (AMD64)] on win32
    Type "help", "copyright", "credits" or "license()" for more information.
>>>
    = RESTART: F:\Aman Mondal\Practicals\4.py
    Enter how many no you want in list of random no: 7
    List with random no are:  [90, 88, 0, 20, 11, 71, 54]
    Max no in list is:  90
    Min no in list is:  0
>>>
```

Q5. Write a menu-based Python program that allows you to manage a simple student grades system using Dictionary. Your program should perform the following tasks:

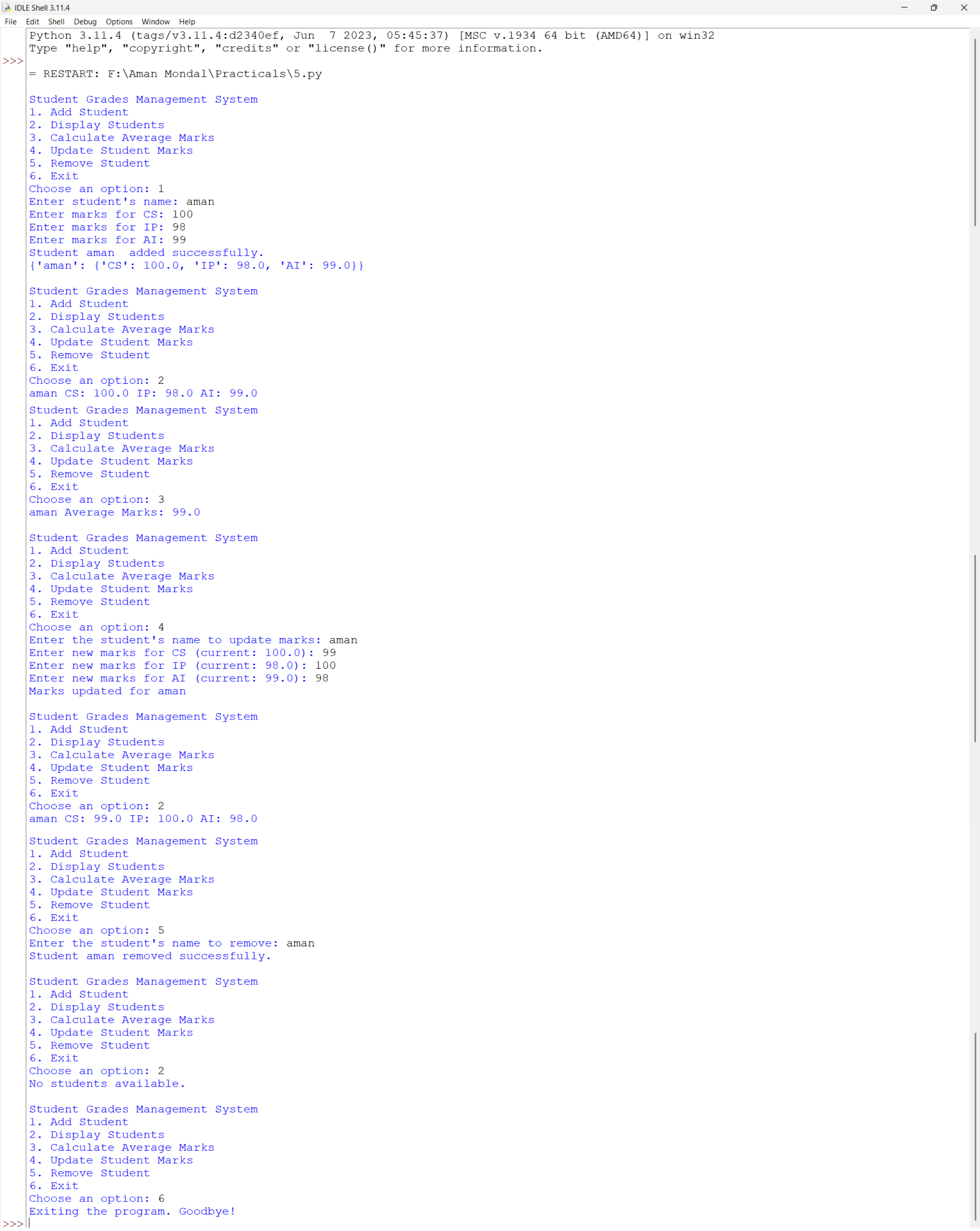
1. Add few student with their name and marks for three subjects (CS,IP,AI).
2. Display all students and their marks.
3. For each student, calculate and display their average marks.
4. Allow the user to update the marks for a particular student.
5. Allow the user to remove a student from the system.
6. Allow the user to exit the program.

Source Code:-


5.py - F:\Aman Mondal\Practicals\5.py (3.11.4)
File Edit Format Run Options Window Help

```
def add_student(students):
    name = input("Enter student's name: ")
    marks = {}
    marks['CS'] = float(input("Enter marks for CS: "))
    marks['IP'] = float(input("Enter marks for IP: "))
    marks['AI'] = float(input("Enter marks for AI: "))
    students[name] = marks
    print("Student",name," added successfully.")
    print(students)

def display_students(students):
    if students=={}:
        print("No students available.")
    else:
        for name, marks in students.items():
            print(name, "CS:", marks['CS'],"IP:", marks['IP'],"AI:", marks['AI'])

def calculate_average(students):
    if students=={}:
        print("No students available.")
    else:
        for name, marks in students.items():
            average = sum(marks.values()) / len(marks)
            print(name,"Average Marks:", average)

def update_student_marks(students):
    name = input("Enter the student's name to update marks: ")
    if name in students:
        for subject in students[name]:
            new_mark = float(input(f"Enter new marks for {subject} (current: {students[name][subject]}): "))
            students[name][subject] = new_mark
        print("Marks updated for",name)
    else:
        print("No student found with the name",name)
def remove_student(students):
    name = input("Enter the student's name to remove: ")
    if name in students:
        del students[name]
        print("Student",name,"removed successfully.")
    else:
        print("No student found with the name.",name)

students = {}
while True:
    print("\nStudent Grades Management System")
    print("1. Add Student")
    print("2. Display Students")
    print("3. Calculate Average Marks")
    print("4. Update Student Marks")
    print("5. Remove Student")
    print("6. Exit")
    choice = input("Choose an option: ")
    if choice == '1':
        add_student(students)
    elif choice == '2':
        display_students(students)
    elif choice == '3':
        calculate_average(students)
    elif choice == '4':
        update_student_marks(students)
    elif choice == '5':
        remove_student(students)
    elif choice == '6':
        print("Exiting the program. Goodbye!")
        break
    else:
        print("Invalid choice. Please try again.")
```

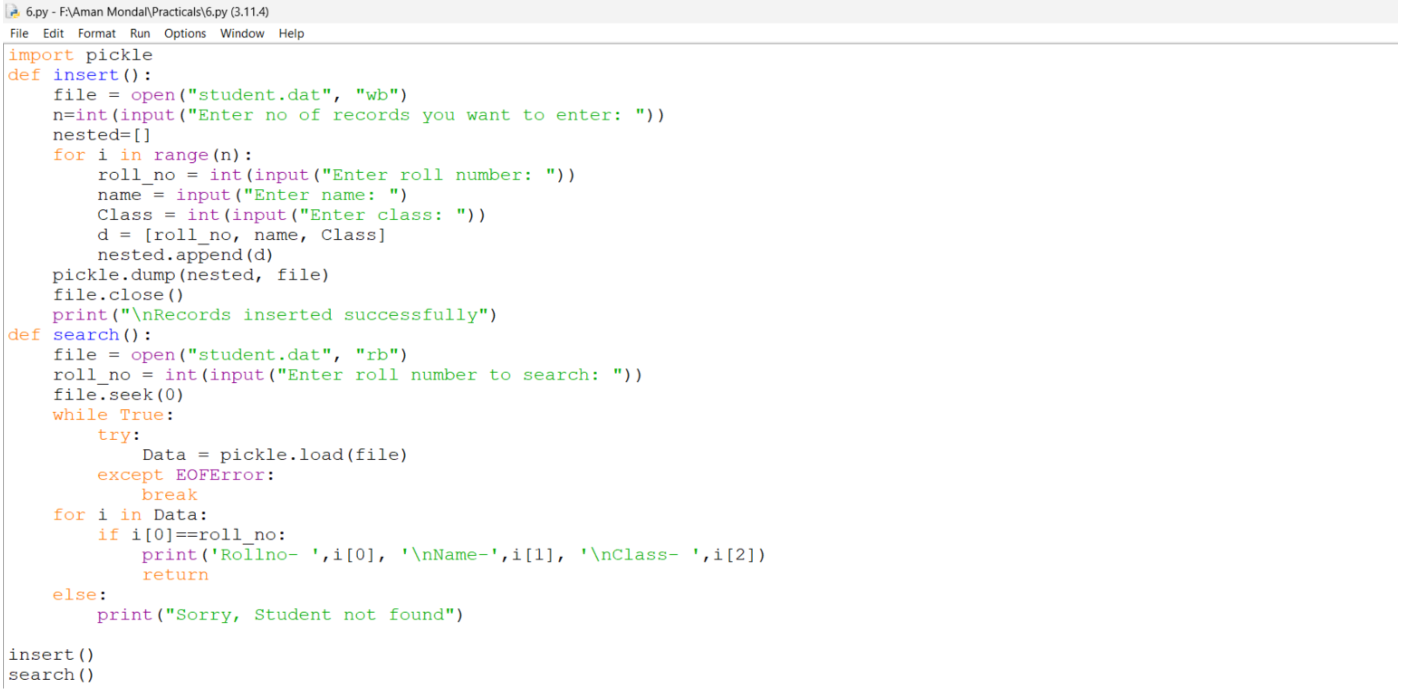
Output:-



```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: F:\Aman Mondal\Practicals\5.py

Student Grades Management System
1. Add Student
2. Display Students
3. Calculate Average Marks
4. Update Student Marks
5. Remove Student
6. Exit
Choose an option: 1
Enter student's name: aman
Enter marks for CS: 100
Enter marks for IP: 98
Enter marks for AI: 99
Student aman  added successfully.
{'aman': {'CS': 100.0, 'IP': 98.0, 'AI': 99.0}}

Student Grades Management System
1. Add Student
2. Display Students
3. Calculate Average Marks
4. Update Student Marks
5. Remove Student
6. Exit
Choose an option: 2
aman CS: 100.0 IP: 98.0 AI: 99.0
Student Grades Management System
1. Add Student
2. Display Students
3. Calculate Average Marks
4. Update Student Marks
5. Remove Student
6. Exit
Choose an option: 3
aman Average Marks: 99.0

Student Grades Management System
1. Add Student
2. Display Students
3. Calculate Average Marks
4. Update Student Marks
5. Remove Student
6. Exit
Choose an option: 4
Enter the student's name to update marks: aman
Enter new marks for CS (current: 100.0): 99
Enter new marks for IP (current: 98.0): 100
Enter new marks for AI (current: 99.0): 98
Marks updated for aman

Student Grades Management System
1. Add Student
2. Display Students
3. Calculate Average Marks
4. Update Student Marks
5. Remove Student
6. Exit
Choose an option: 2
aman CS: 99.0 IP: 100.0 AI: 98.0
Student Grades Management System
1. Add Student
2. Display Students
3. Calculate Average Marks
4. Update Student Marks
5. Remove Student
6. Exit
Choose an option: 5
Enter the student's name to remove: aman
Student aman removed successfully.

Student Grades Management System
1. Add Student
2. Display Students
3. Calculate Average Marks
4. Update Student Marks
5. Remove Student
6. Exit
Choose an option: 2
No students available.

Student Grades Management System
1. Add Student
2. Display Students
3. Calculate Average Marks
4. Update Student Marks
5. Remove Student
6. Exit
Choose an option: 6
Exiting the program. Goodbye!
>>>
```

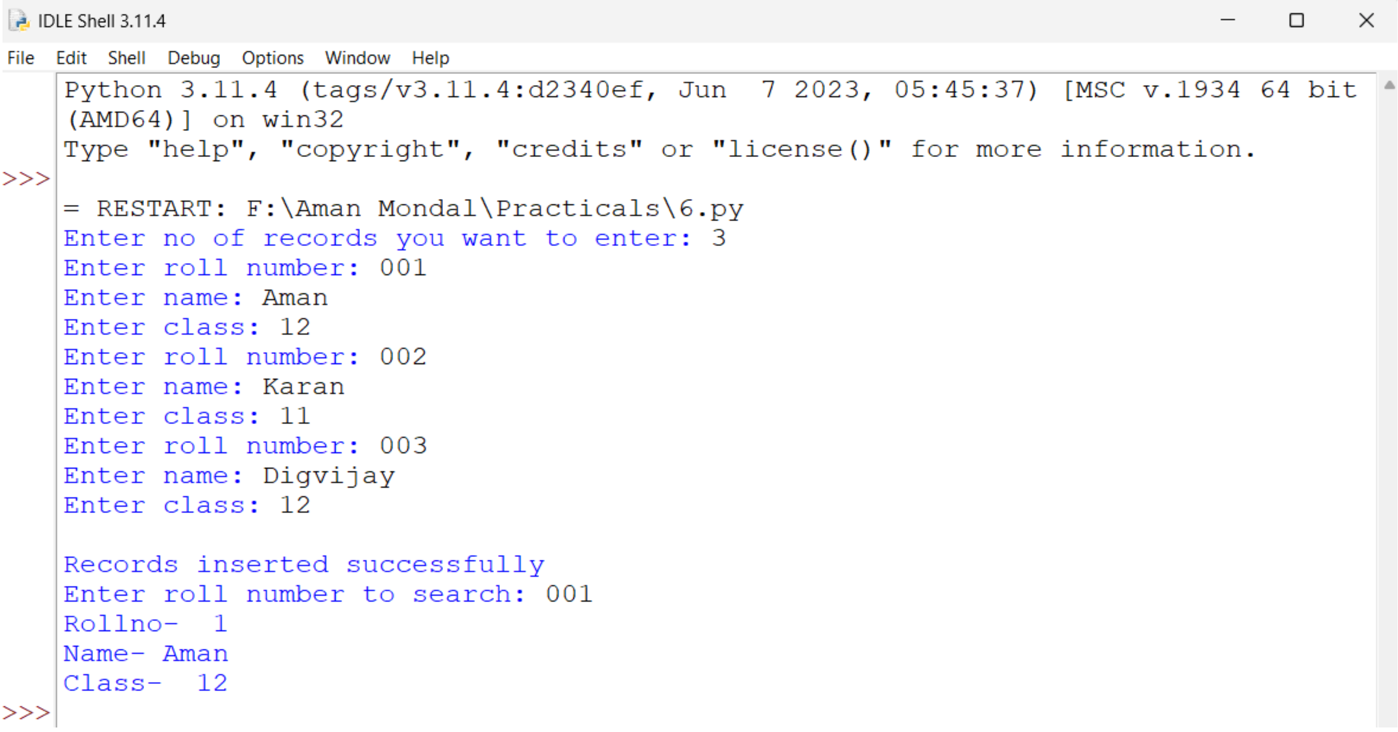
Q6. Create a binary file with roll number, name and class and insert 3 records in it in the format [roll-no, name, class]. Search for a given roll number and display the name, if not found display appropriate message.

Source Code:-


6.py - F:\Aman Mondal\Practicals\6.py (3.11.4)
File Edit Format Run Options Window Help

```
import pickle
def insert():
    file = open("student.dat", "wb")
    n=int(input("Enter no of records you want to enter: "))
    nested=[]
    for i in range(n):
        roll_no = int(input("Enter roll number: "))
        name = input("Enter name: ")
        Class = int(input("Enter class: "))
        d = [roll_no, name, Class]
        nested.append(d)
    pickle.dump(nested, file)
    file.close()
    print("\nRecords inserted successfully")
def search():
    file = open("student.dat", "rb")
    roll_no = int(input("Enter roll number to search: "))
    file.seek(0)
    while True:
        try:
            Data = pickle.load(file)
        except EOFError:
            break
    for i in Data:
        if i[0]==roll_no:
            print('Rollno- ',i[0], '\nName-',i[1], '\nClass- ',i[2])
            return
    else:
        print("Sorry, Student not found")

insert()
search()
```

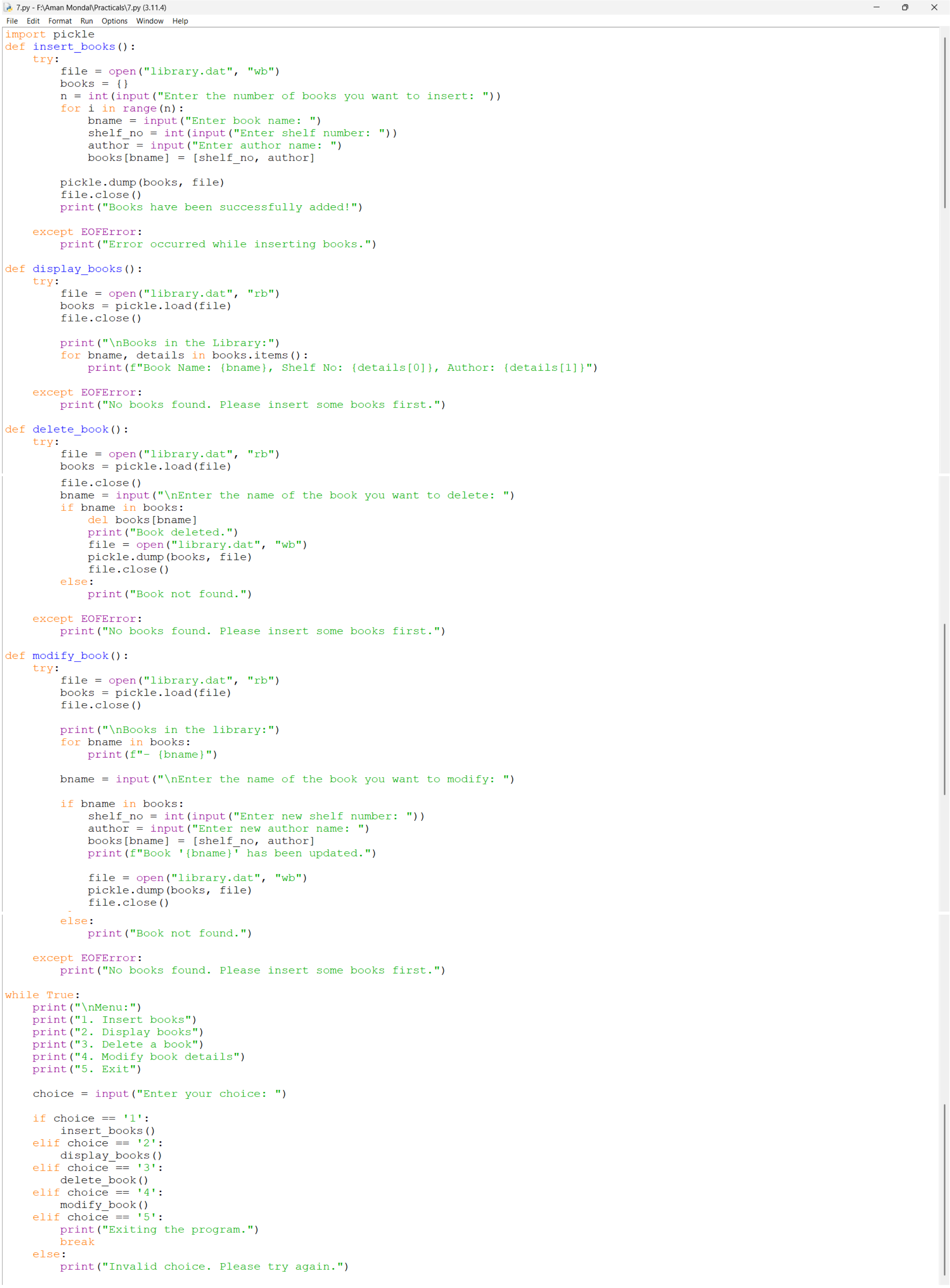
Output:-



```
    Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit
    (AMD64)] on win32
    Type "help", "copyright", "credits" or "license()" for more information.
>>>
    = RESTART: F:\Aman Mondal\Practicals\6.py
    Enter no of records you want to enter: 3
    Enter roll number: 001
    Enter name: Aman
    Enter class: 12
    Enter roll number: 002
    Enter name: Karan
    Enter class: 11
    Enter roll number: 003
    Enter name: Digvijay
    Enter class: 12

    Records inserted successfully
    Enter roll number to search: 001
    Rollno-  1
    Name- Aman
    Class-  12
>>>
```

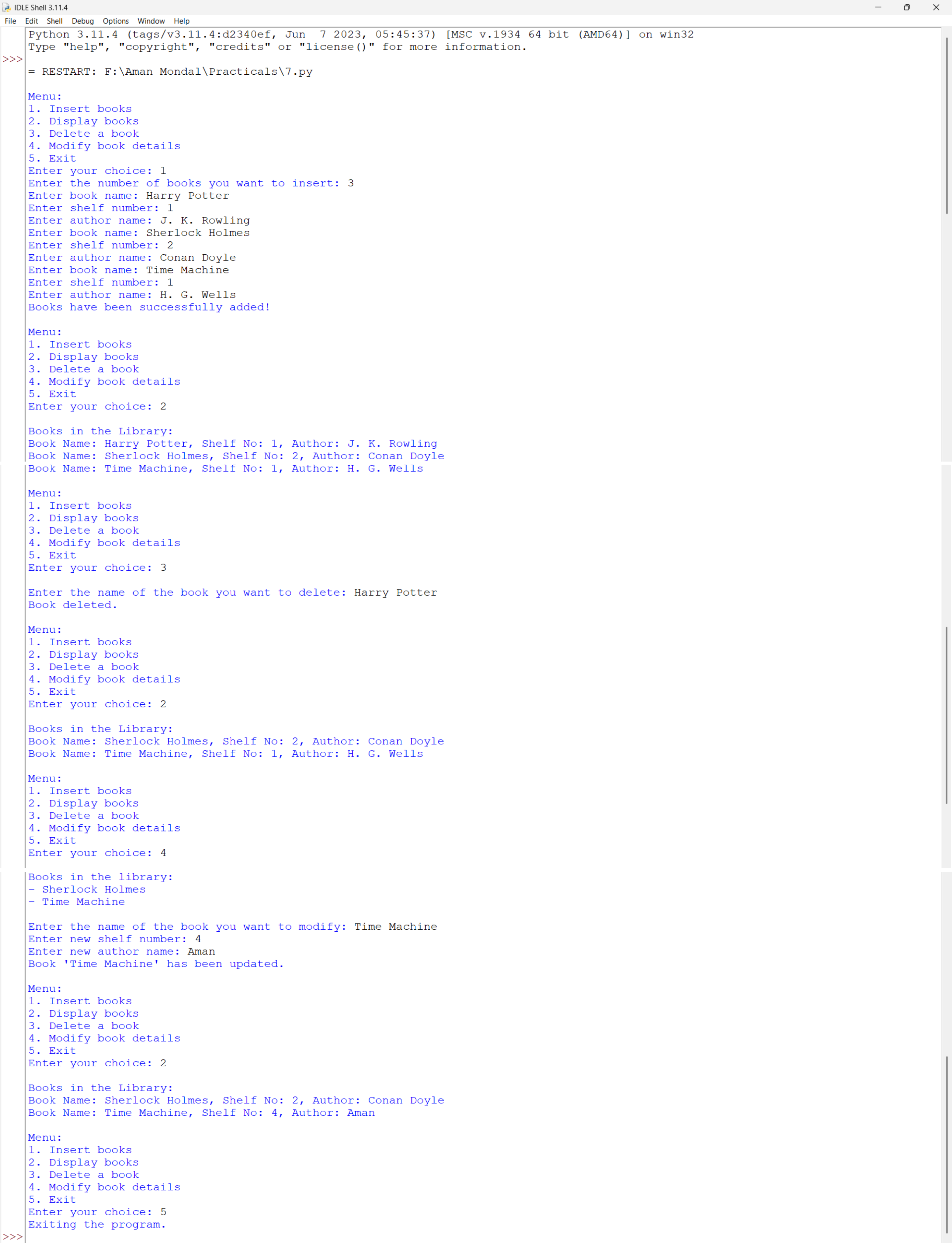
Q7- Create a binary file with book,shelf_no and author.
Insert 3 values- {“Harry Potter”:[1, “J.K. Rowling”]},
{“Sherlock Holmes”:[2, “Conan Doyle”]},
{“Time Machine”:[1, “H.G. Wells”]}
The program should display all the book names to the user and prompt them to enter the name of a book they wish to delete. After that, the program should ask the user to modify the details of one of the remaining book entries.

Source Code:-


7.py - F:\Aman Mondal\Practicals\7.py (3.11.4)
File Edit Format Run Options Window Help

```
import pickle
def insert_books():
    try:
        file = open("library.dat", "wb")
        books = {}
        n = int(input("Enter the number of books you want to insert: "))
        for i in range(n):
            bname = input("Enter book name: ")
            shelf_no = int(input("Enter shelf number: "))
            author = input("Enter author name: ")
            books[bname] = [shelf_no, author]

        pickle.dump(books, file)
        file.close()
        print("Books have been successfully added!")

    except EOFError:
        print("Error occurred while inserting books.")

def display_books():
    try:
        file = open("library.dat", "rb")
        books = pickle.load(file)
        file.close()

        print("\nBooks in the Library:")
        for bname, details in books.items():
            print(f"Book Name: {bname}, Shelf No: {details[0]}, Author: {details[1]}")

    except EOFError:
        print("No books found. Please insert some books first.")

def delete_book():
    try:
        file = open("library.dat", "rb")
        books = pickle.load(file)
        file.close()
        bname = input("\nEnter the name of the book you want to delete: ")
        if bname in books:
            del books[bname]
            print("Book deleted.")
            file = open("library.dat", "wb")
            pickle.dump(books, file)
            file.close()
        else:
            print("Book not found.")

    except EOFError:
        print("No books found. Please insert some books first.")

def modify_book():
    try:
        file = open("library.dat", "rb")
        books = pickle.load(file)
        file.close()

        print("\nBooks in the library:")
        for bname in books:
            print(f"- {bname}")

        bname = input("\nEnter the name of the book you want to modify: ")

        if bname in books:
            shelf_no = int(input("Enter new shelf number: "))
            author = input("Enter new author name: ")
            books[bname] = [shelf_no, author]
            print(f"Book '{bname}' has been updated.")

            file = open("library.dat", "wb")
            pickle.dump(books, file)
            file.close()
        else:
            print("Book not found.")

    except EOFError:
        print("No books found. Please insert some books first.")

while True:
    print("\nMenu:")
    print("1. Insert books")
    print("2. Display books")
    print("3. Delete a book")
    print("4. Modify book details")
    print("5. Exit")

    choice = input("Enter your choice: ")

    if choice == '1':
        insert_books()
    elif choice == '2':
        display_books()
    elif choice == '3':
        delete_book()
    elif choice == '4':
        modify_book()
    elif choice == '5':
        print("Exiting the program.")
        break
    else:
        print("Invalid choice. Please try again.")
```

Output:-



```
    Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
    Type "help", "copyright", "credits" or "license()" for more information.
>>>
    = RESTART: F:\Aman Mondal\Practicals\7.py

    Menu:
    1. Insert books
    2. Display books
    3. Delete a book
    4. Modify book details
    5. Exit
    Enter your choice: 1
    Enter the number of books you want to insert: 3
    Enter book name: Harry Potter
    Enter shelf number: 1
    Enter author name: J. K. Rowling
    Enter book name: Sherlock Holmes
    Enter shelf number: 2
    Enter author name: Conan Doyle
    Enter book name: Time Machine
    Enter shelf number: 1
    Enter author name: H. G. Wells
    Books have been successfully added!

    Menu:
    1. Insert books
    2. Display books
    3. Delete a book
    4. Modify book details
    5. Exit
    Enter your choice: 2

    Books in the Library:
    Book Name: Harry Potter, Shelf No: 1, Author: J. K. Rowling
    Book Name: Sherlock Holmes, Shelf No: 2, Author: Conan Doyle
    Book Name: Time Machine, Shelf No: 1, Author: H. G. Wells

    Menu:
    1. Insert books
    2. Display books
    3. Delete a book
    4. Modify book details
    5. Exit
    Enter your choice: 3

    Enter the name of the book you want to delete: Harry Potter
    Book deleted.

    Menu:
    1. Insert books
    2. Display books
    3. Delete a book
    4. Modify book details
    5. Exit
    Enter your choice: 2

    Books in the Library:
    Book Name: Sherlock Holmes, Shelf No: 2, Author: Conan Doyle
    Book Name: Time Machine, Shelf No: 1, Author: H. G. Wells

    Menu:
    1. Insert books
    2. Display books
    3. Delete a book
    4. Modify book details
    5. Exit
    Enter your choice: 4

    Books in the library:
    - Sherlock Holmes
    - Time Machine

    Enter the name of the book you want to modify: Time Machine
    Enter new shelf number: 4
    Enter new author name: Aman
    Book 'Time Machine' has been updated.

    Menu:
    1. Insert books
    2. Display books
    3. Delete a book
    4. Modify book details
    5. Exit
    Enter your choice: 2

    Books in the Library:
    Book Name: Sherlock Holmes, Shelf No: 2, Author: Conan Doyle
    Book Name: Time Machine, Shelf No: 4, Author: Aman

    Menu:
    1. Insert books
    2. Display books
    3. Delete a book
    4. Modify book details
    5. Exit
    Enter your choice: 5
    Exiting the program.
>>>
```

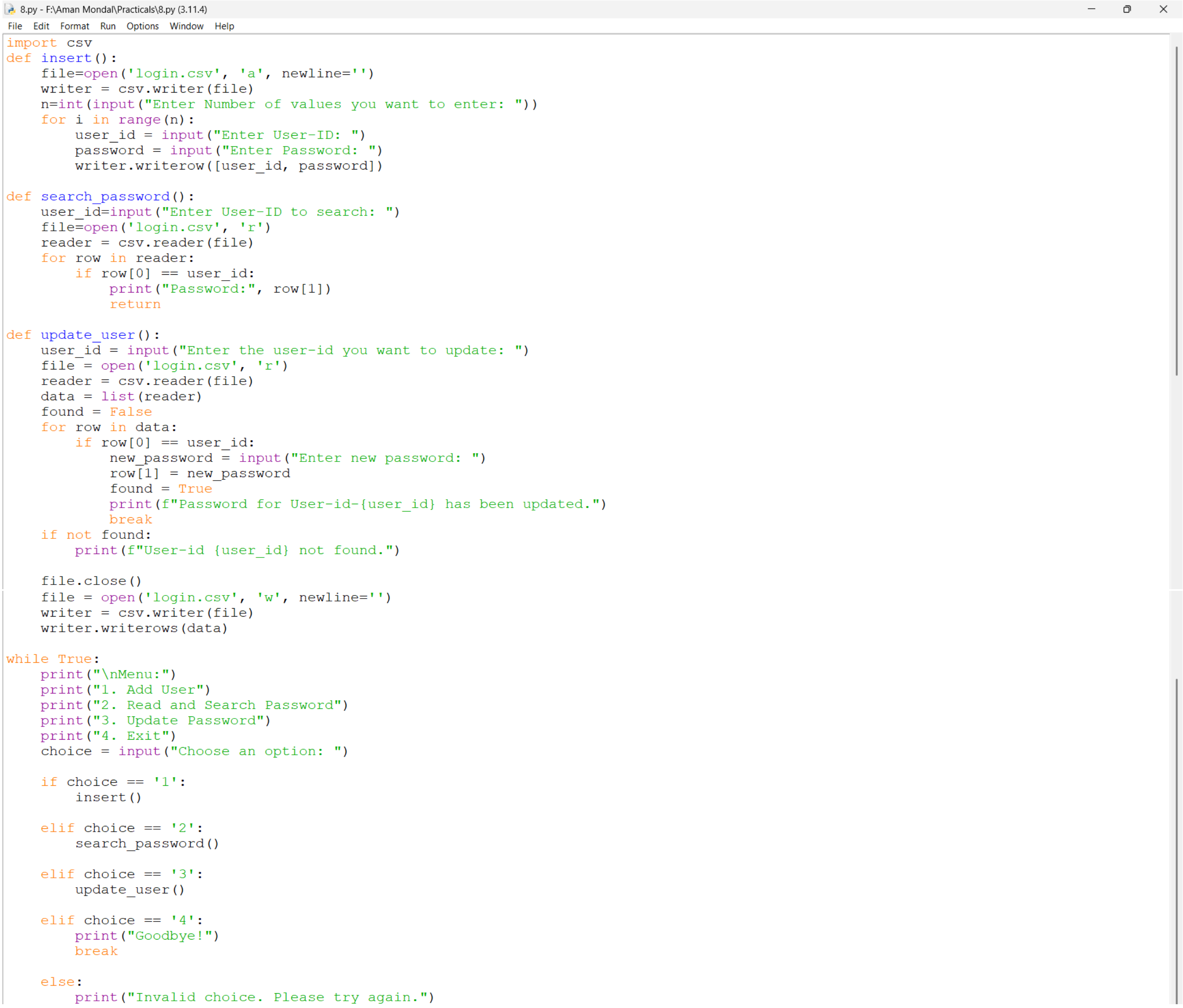
Q8. Create a CSV file (login.csv) by entering user-id and password and write a menu- based python program to perform the following task.

Read- Read and search the password for given user-id.

Update- The program asks the user to modify the details of the given user-id.

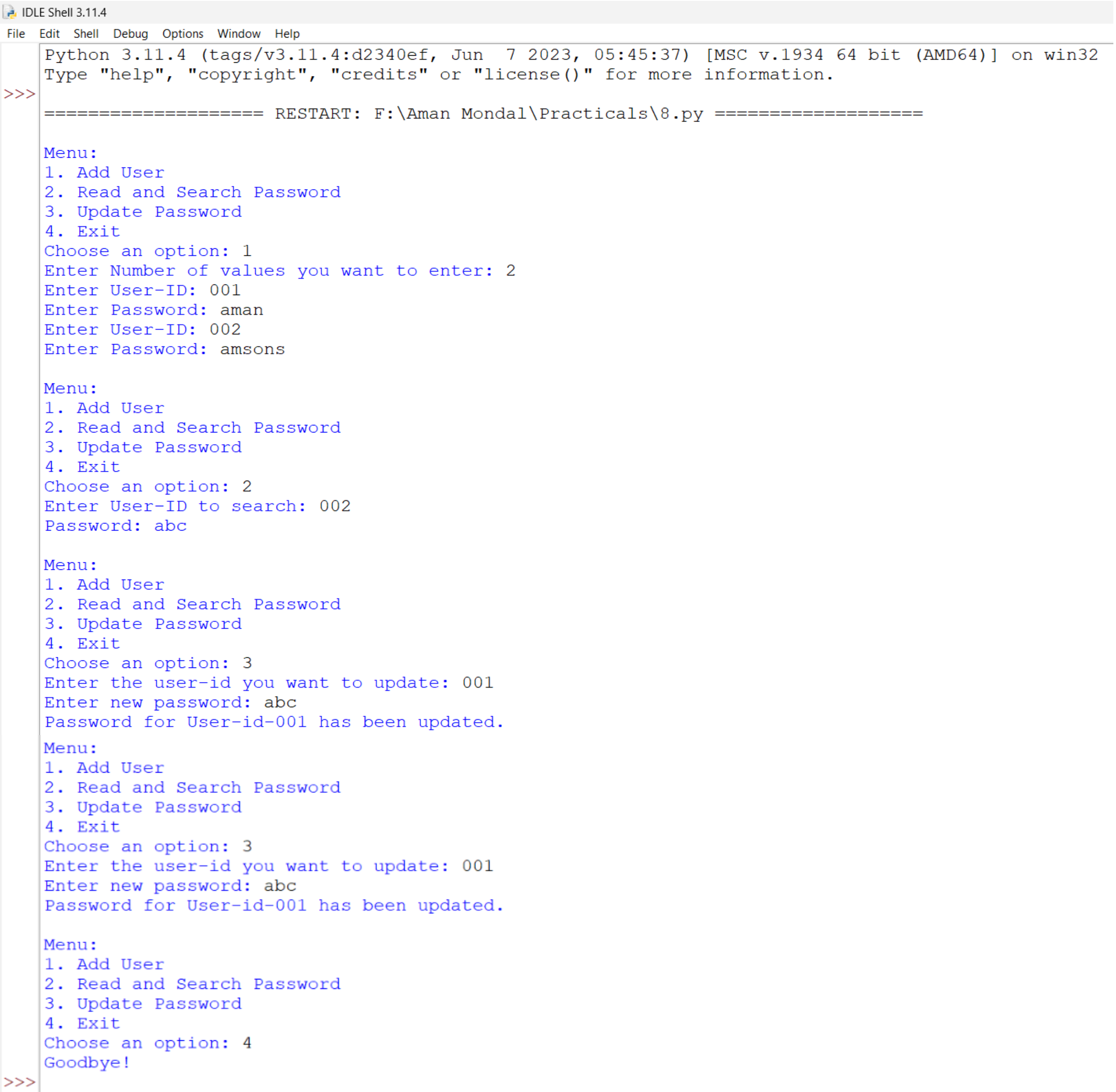
Delete- Prompt user to enter the user-id they wish to delete.

Source Code:-


8.py - F:\Aman Mondal\Practicals\8.py (3.11.4)
File Edit Format Run Options Window Help

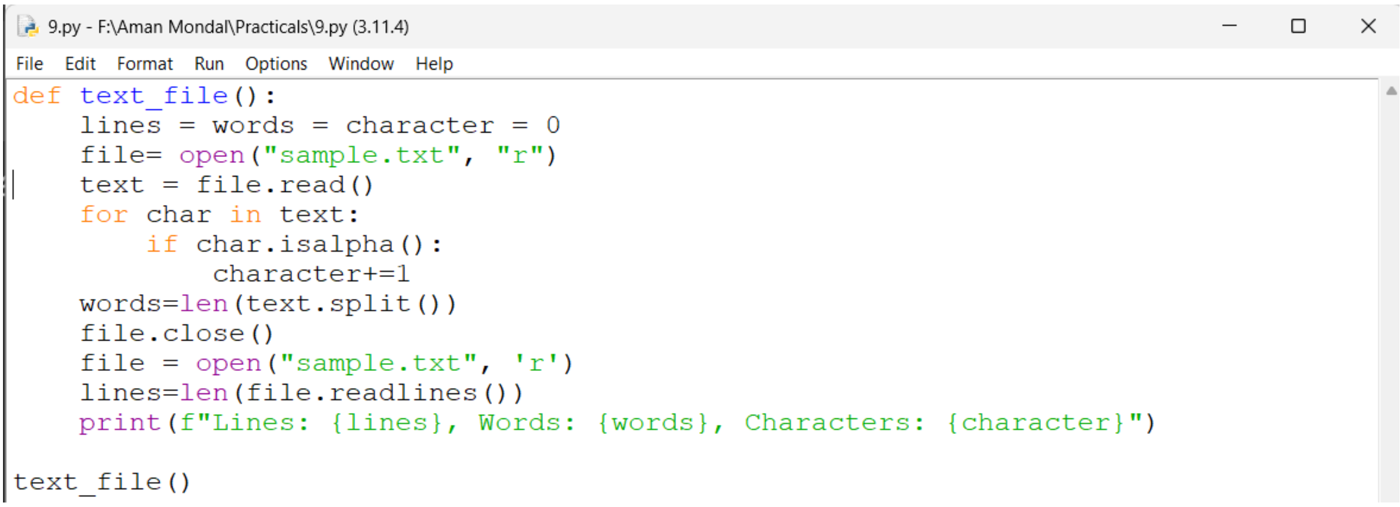
```
import csv
def insert():
    file=open('login.csv', 'a', newline='')
    writer = csv.writer(file)
    n=int(input("Enter Number of values you want to enter: "))
    for i in range(n):
        user_id = input("Enter User-ID: ")
        password = input("Enter Password: ")
        writer.writerow([user_id, password])

def search_password():
    user_id=input("Enter User-ID to search: ")
    file=open('login.csv', 'r')
    reader = csv.reader(file)
    for row in reader:
        if row[0] == user_id:
            print("Password:", row[1])
            return

def update_user():
    user_id = input("Enter the user-id you want to update: ")
    file = open('login.csv', 'r')
    reader = csv.reader(file)
    data = list(reader)
    found = False
    for row in data:
        if row[0] == user_id:
            new_password = input("Enter new password: ")
            row[1] = new_password
            found = True
            print(f"Password for User-id-{user_id} has been updated.")
            break
    if not found:
        print(f"User-id {user_id} not found.")

    file.close()
    file = open('login.csv', 'w', newline='')
    writer = csv.writer(file)
    writer.writerows(data)

while True:
    print("\nMenu:")
    print("1. Add User")
    print("2. Read and Search Password")
    print("3. Update Password")
    print("4. Exit")
    choice = input("Choose an option: ")

    if choice == '1':
        insert()

    elif choice == '2':
        search_password()

    elif choice == '3':
        update_user()

    elif choice == '4':
        print("Goodbye!")
        break

    else:
        print("Invalid choice. Please try again.")
```

Output:-



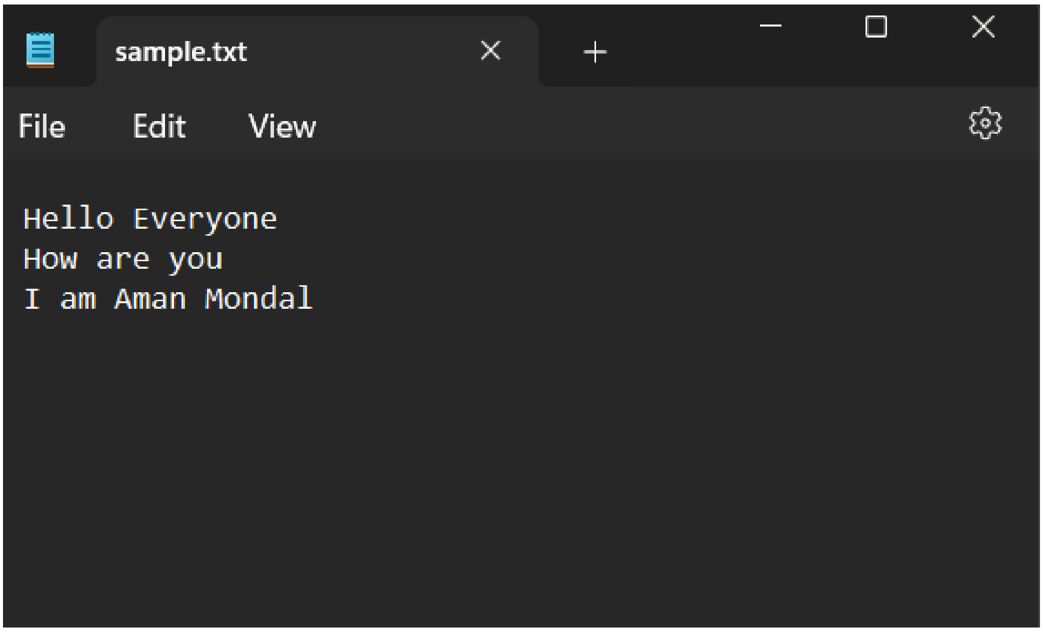
```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==================== RESTART: F:\Aman Mondal\Practicals\8.py ===================

Menu:
1. Add User
2. Read and Search Password
3. Update Password
4. Exit
Choose an option: 1
Enter Number of values you want to enter: 2
Enter User-ID: 001
Enter Password: aman
Enter User-ID: 002
Enter Password: amsons

Menu:
1. Add User
2. Read and Search Password
3. Update Password
4. Exit
Choose an option: 2
Enter User-ID to search: 002
Password: abc

Menu:
1. Add User
2. Read and Search Password
3. Update Password
4. Exit
Choose an option: 3
Enter the user-id you want to update: 001
Enter new password: abc
Password for User-id-001 has been updated.
Menu:
1. Add User
2. Read and Search Password
3. Update Password
4. Exit
Choose an option: 3
Enter the user-id you want to update: 001
Enter new password: abc
Password for User-id-001 has been updated.

Menu:
1. Add User
2. Read and Search Password
3. Update Password
4. Exit
Choose an option: 4
Goodbye!
>>>
```

Q9. Write a Python program that reads a text file and counts the number of lines, words, and characters in the file. Display the counts in the following format:
Lines: X, Words: Y, Characters: Z.
Source Code:-


9.py - F:\Aman Mondal\Practicals\9.py (3.11.4)
File Edit Format Run Options Window Help

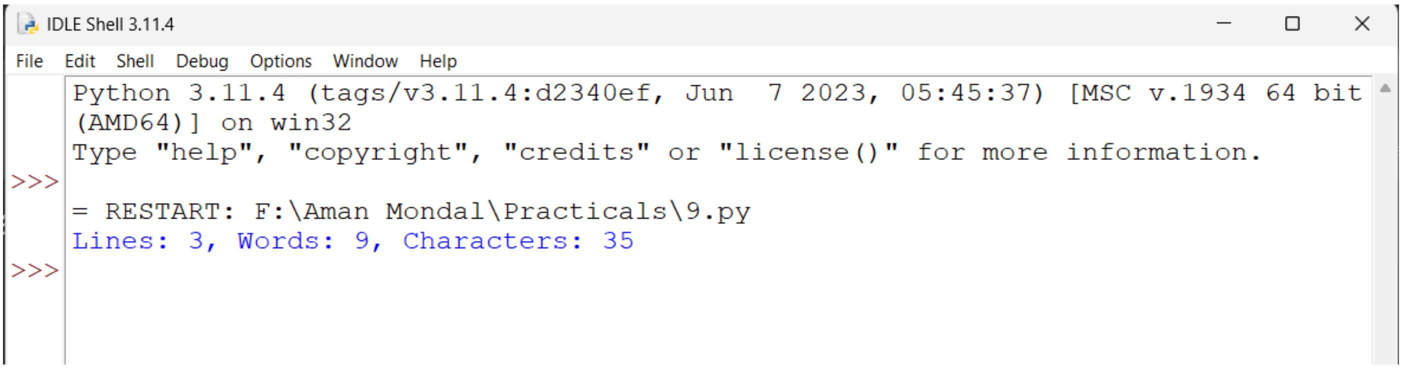
```python
def text_file():
    lines = words = character = 0
    file= open("sample.txt", "r")
    text = file.read()
    for char in text:
        if char.isalpha():
            character+=1
    words=len(text.split())
    file.close()
    file = open("sample.txt", 'r')
    lines=len(file.readlines())
    print(f"Lines: {lines}, Words: {words}, Characters: {character}")

text_file()
```

Sample.txt:-



```
Hello Everyone
How are you
I am Aman Mondal
```

Output:-

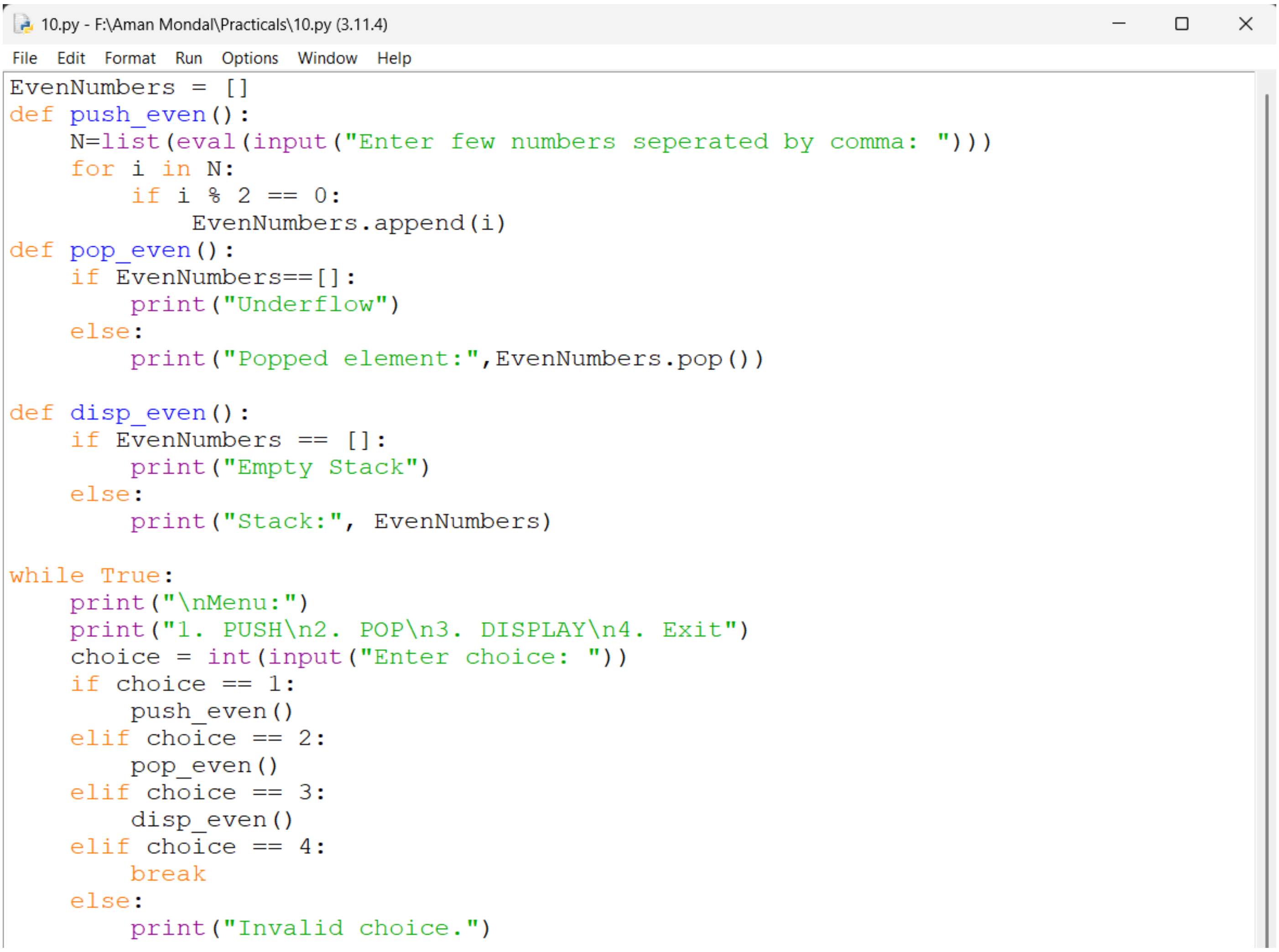


```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: F:\Aman Mondal\Practicals\9.py
Lines: 3, Words: 9, Characters: 35
>>>
```

Q10. Write a menu-based python program to perform the following questions.

- Write the definition of a user-defined function ‘push_even(N)’ which accepts a list of integers in a parameter ‘N’ and pushes all those integers which are even from the list ‘N’ into a Stack named ‘EvenNumbers’.
- Write function pop_even() to pop the topmost number from the stack and returns it. If the stack is already empty, the function should display "Underflow".
- Write function disp_even() to display all element of the stack without deleting them. If the stack is empty, the function should display 'Empty Stack’.

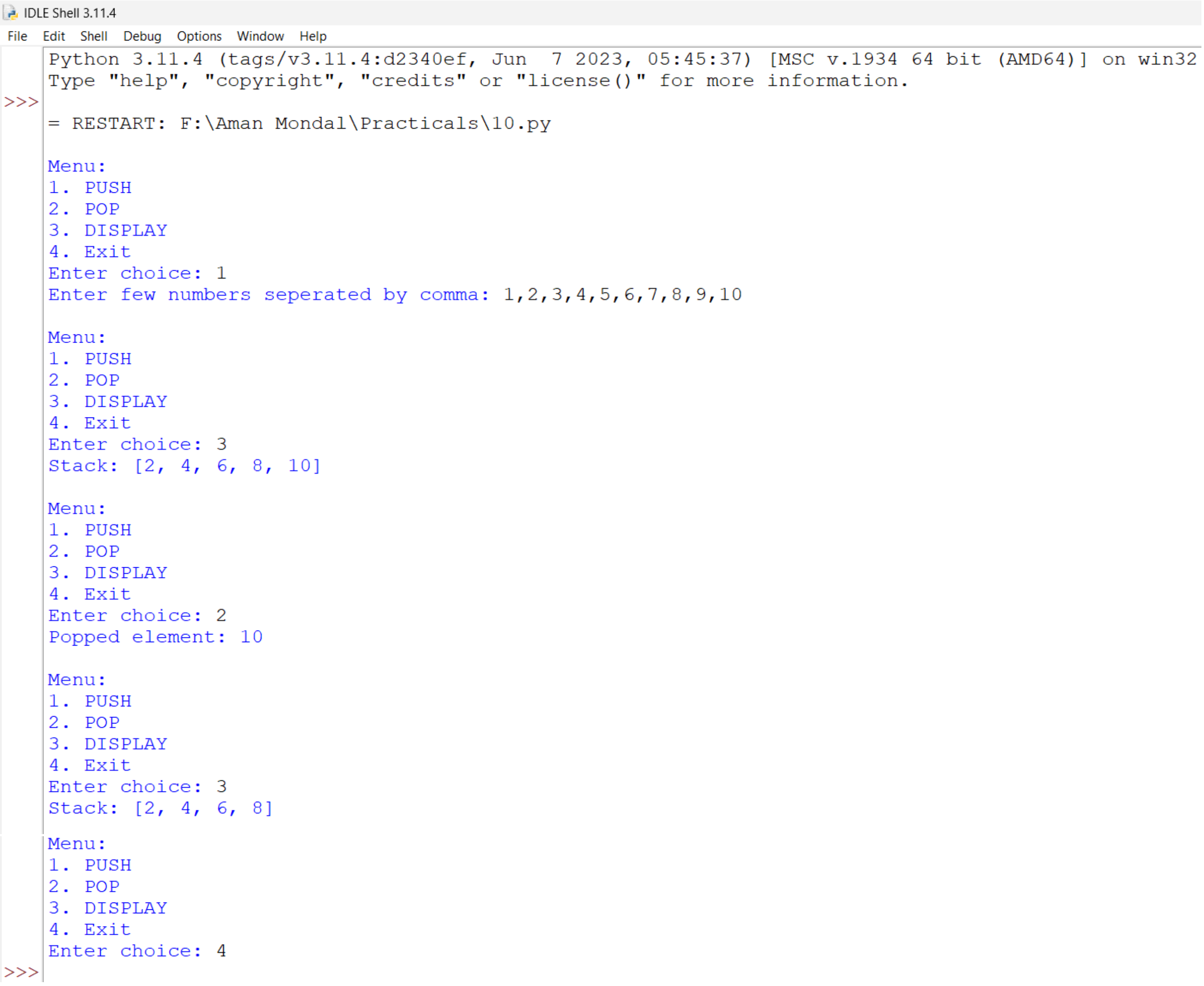
Source Code:-



10.py - F:\Aman Mondal\Practicals\10.py (3.11.4)
File Edit Format Run Options Window Help

```
EvenNumbers = []
def push_even():
    N=list(eval(input("Enter few numbers seperated by comma: ")))
    for i in N:
        if i % 2 == 0:
            EvenNumbers.append(i)
def pop_even():
    if EvenNumbers==[]:
        print("Underflow")
    else:
        print("Popped element:",EvenNumbers.pop())

def disp_even():
    if EvenNumbers == []:
        print("Empty Stack")
    else:
        print("Stack:", EvenNumbers)

while True:
    print("\nMenu:")
    print("1. PUSH\n2. POP\n3. DISPLAY\n4. Exit")
    choice = int(input("Enter choice: "))
    if choice == 1:
        push_even()
    elif choice == 2:
        pop_even()
    elif choice == 3:
        disp_even()
    elif choice == 4:
        break
    else:
        print("Invalid choice.")
```

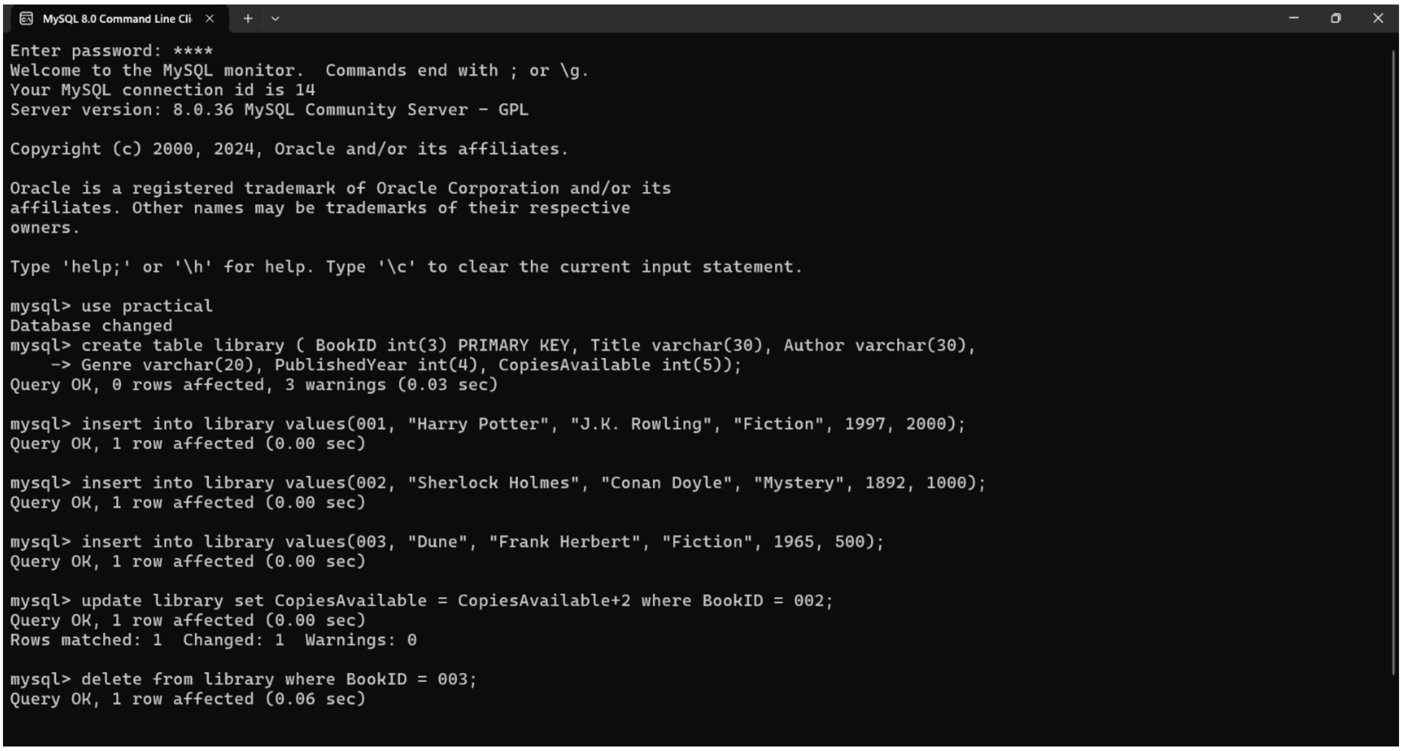
Output:-



```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: F:\Aman Mondal\Practicals\10.py

Menu:
1. PUSH
2. POP
3. DISPLAY
4. Exit
Enter choice: 1
Enter few numbers seperated by comma: 1,2,3,4,5,6,7,8,9,10

Menu:
1. PUSH
2. POP
3. DISPLAY
4. Exit
Enter choice: 3
Stack: [2, 4, 6, 8, 10]

Menu:
1. PUSH
2. POP
3. DISPLAY
4. Exit
Enter choice: 2
Popped element: 10

Menu:
1. PUSH
2. POP
3. DISPLAY
4. Exit
Enter choice: 3
Stack: [2, 4, 6, 8]

Menu:
1. PUSH
2. POP
3. DISPLAY
4. Exit
Enter choice: 4
>>> 
```

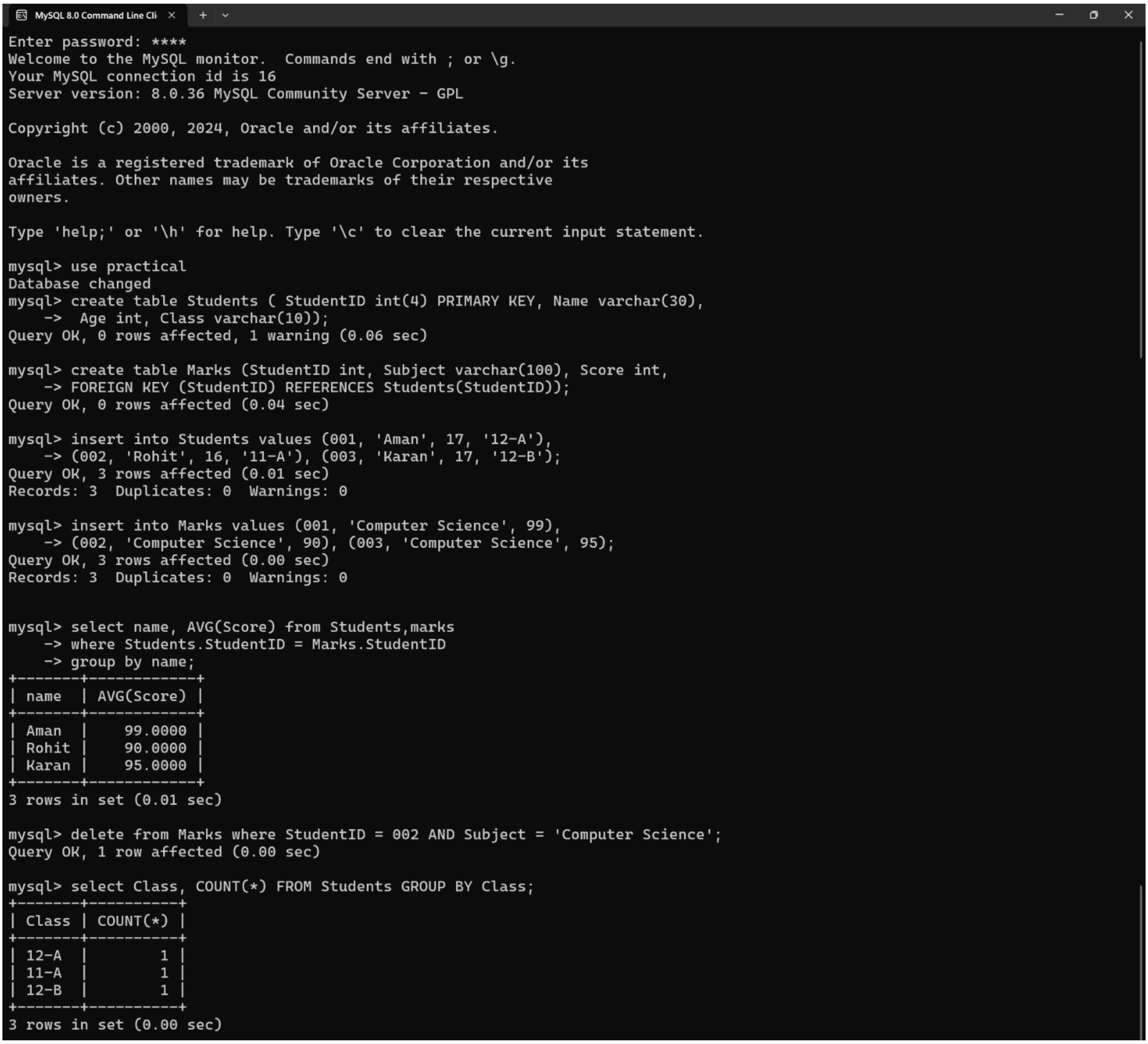
Q11.Perform the following SQL queries
i. Write a SQL query to create a table named Library with columns BookID, Title, Author, Genre, PublishedYear, and CopiesAvailable and. And perform the following questions
ii. Write SQL queries to insert at least five records into the Library table.
iii. Write a SQL query to update the number of available copies of a specific book (e.g., increase by 2).
iv. Write a SQL query to delete a book record from the Library table where the BookID is a specific value.

Source Code And Output :-



```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.36 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use practical
Database changed
mysql> create table library ( BookID int(3) PRIMARY KEY, Title varchar(30), Author varchar(30),
    -> Genre varchar(20), PublishedYear int(4), CopiesAvailable int(5));
Query OK, 0 rows affected, 3 warnings (0.03 sec)

mysql> insert into library values(001, "Harry Potter", "J.K. Rowling", "Fiction", 1997, 2000);
Query OK, 1 row affected (0.00 sec)

mysql> insert into library values(002, "Sherlock Holmes", "Conan Doyle", "Mystery", 1892, 1000);
Query OK, 1 row affected (0.00 sec)

mysql> insert into library values(003, "Dune", "Frank Herbert", "Fiction", 1965, 500);
Query OK, 1 row affected (0.00 sec)

mysql> update library set CopiesAvailable = CopiesAvailable+2 where BookID = 002;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> delete from library where BookID = 003;
Query OK, 1 row affected (0.06 sec)
```

Q12. Perform the following SQL queries
i. Create two tables: Students (with columns StudentID, Name, Age, Class) and Marks (with columns StudentID, Subject, Score). Ensure that StudentID in Marks references StudentID in Students.
ii. Write SQL queries to insert at least three records into both the Students and Marks tables.
iii. Write a SQL query to find the average score of each student across all subjects and display the results with student names.
iv. Write a SQL query to delete a record from the Marks table for a student who has withdrawn from a subject.
v. Write a SQL query to count the total number of students in each class.
Source Code and Output:-

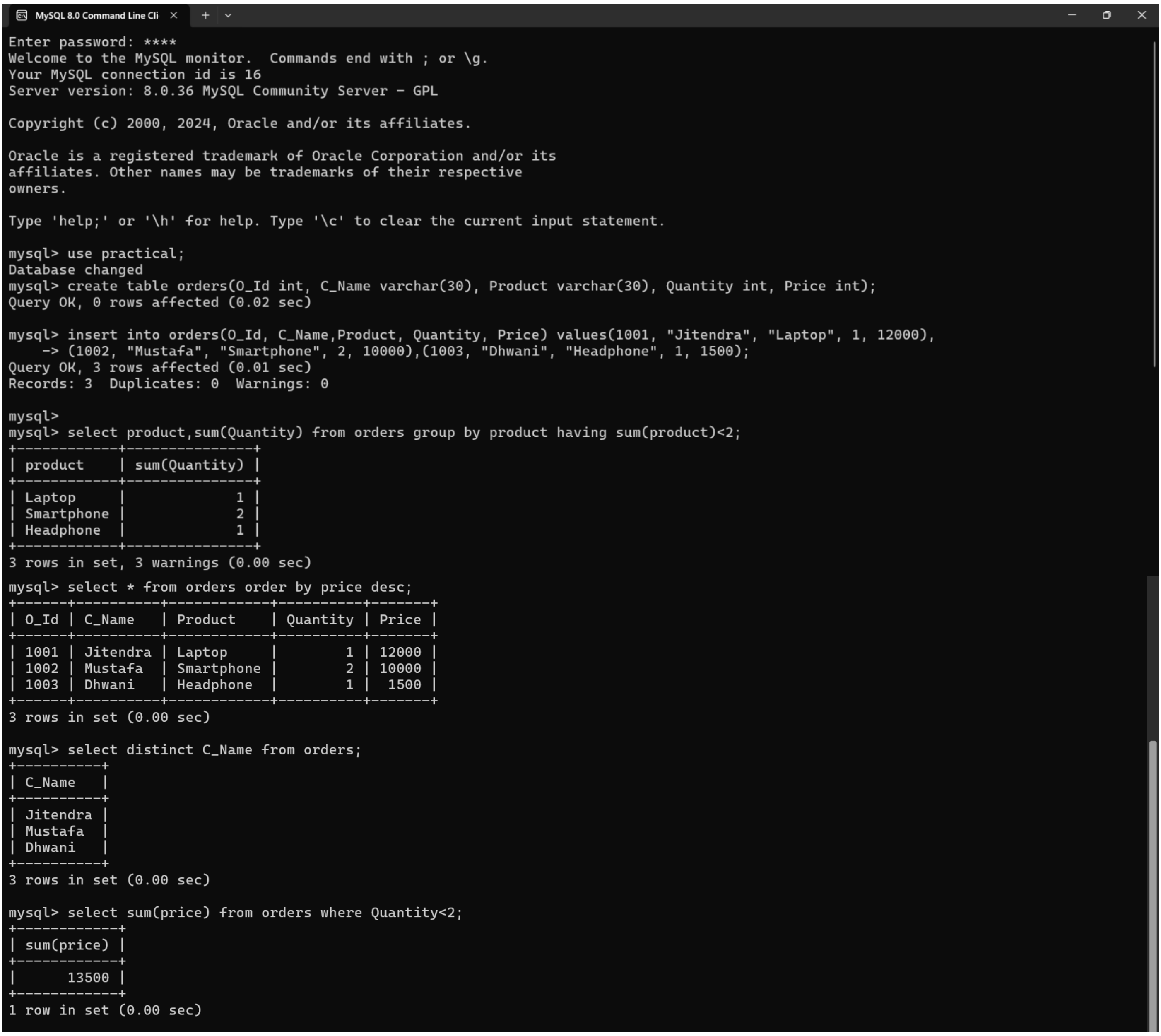


```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.36 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use practical
Database changed
mysql> create table Students ( StudentID int(4) PRIMARY KEY, Name varchar(30),
    ->  Age int, Class varchar(10));
Query OK, 0 rows affected, 1 warning (0.06 sec)

mysql> create table Marks (StudentID int, Subject varchar(100), Score int,
    -> FOREIGN KEY (StudentID) REFERENCES Students(StudentID));
Query OK, 0 rows affected (0.04 sec)

mysql> insert into Students values (001, 'Aman', 17, '12-A'),
    -> (002, 'Rohit', 16, '11-A'), (003, 'Karan', 17, '12-B');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> insert into Marks values (001, 'Computer Science', 99),
    -> (002, 'Computer Science', 90), (003, 'Computer Science', 95);
Query OK, 3 rows affected (0.00 sec)
Records: 3  Duplicates: 0  Warnings: 0


mysql> select name, AVG(Score) from Students,marks
    -> where Students.StudentID = Marks.StudentID
    -> group by name;
+-------+------------+
| name  | AVG(Score) |
+-------+------------+
| Aman  |    99.0000 |
| Rohit |    90.0000 |
| Karan |    95.0000 |
+-------+------------+
3 rows in set (0.01 sec)

mysql> delete from Marks where StudentID = 002 AND Subject = 'Computer Science';
Query OK, 1 row affected (0.00 sec)

mysql> select Class, COUNT(*) FROM Students GROUP BY Class;
+-------+----------+
| Class | COUNT(*) |
+-------+----------+
| 12-A  |        1 |
| 11-A  |        1 |
| 12-B  |        1 |
+-------+----------+
3 rows in set (0.00 sec)
```

Q13.Create the table ORDERS as given below and perform the following queries:

O_Id	C_Name	Product	Quantity	Price
1001	Jitendra	Laptop	1	12000
1002	Mustafa	Smartphone	2	10000
1003	Dhwani	Headphone	1	1500

(I) Display the total Quantity for each Product, excluding Products with total Quantity less than 2.
(II) Display the orders table sorted by total price in descending order.
(III) Display the distinct customer names from the Orders table.
(IV) Display the sum of Price of all the orders for which the quantity is less than 2.
Source Code and Output:-



```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.36 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use practical;
Database changed
mysql> create table orders(O_Id int, C_Name varchar(30), Product varchar(30), Quantity int, Price int);
Query OK, 0 rows affected (0.02 sec)

mysql> insert into orders(O_Id, C_Name,Product, Quantity, Price) values(1001, "Jitendra", "Laptop", 1, 12000),
    -> (1002, "Mustafa", "Smartphone", 2, 10000),(1003, "Dhwani", "Headphone", 1, 1500);
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql>
mysql> select product,sum(Quantity) from orders group by product having sum(product)<2;
+------------+---------------+
| product    | sum(Quantity) |
+------------+---------------+
| Laptop     |             1 |
| Smartphone |             2 |
| Headphone  |             1 |
+------------+---------------+
3 rows in set, 3 warnings (0.00 sec)

mysql> select * from orders order by price desc;
+------+----------+------------+----------+-------+
| O_Id | C_Name   | Product    | Quantity | Price |
+------+----------+------------+----------+-------+
| 1001 | Jitendra | Laptop     |        1 | 12000 |
| 1002 | Mustafa  | Smartphone |        2 | 10000 |
| 1003 | Dhwani   | Headphone  |        1 |  1500 |
+------+----------+------------+----------+-------+
3 rows in set (0.00 sec)

mysql> select distinct C_Name from orders;
+----------+
| C_Name   |
+----------+
| Jitendra |
| Mustafa  |
| Dhwani   |
+----------+
3 rows in set (0.00 sec)

mysql> select sum(price) from orders where Quantity<2;
+------------+
| sum(price) |
+------------+
|      13500 |
+------------+
1 row in set (0.00 sec)
```

Q14. Karan has been entrusted with the management of Law University Database. He needs to access some information from FACULTY and COURSES tables for a survey analysis. Create both the tables and help him extract the following information by performing the desired SQL queries as mentioned below.

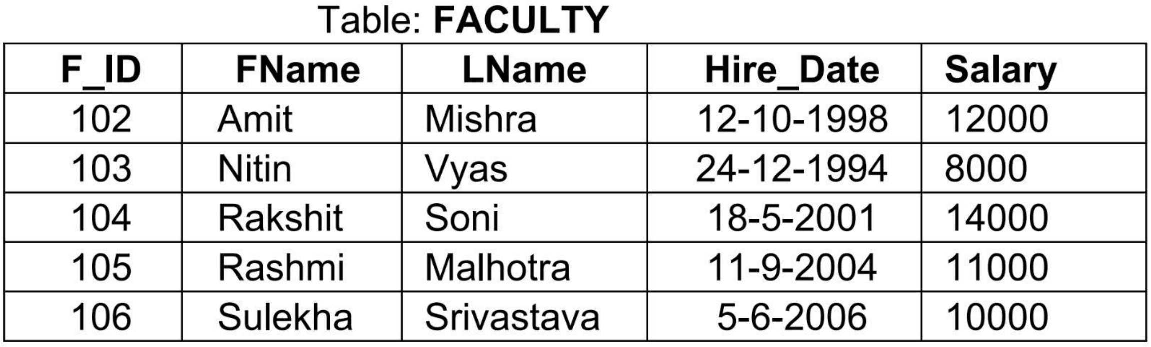


Table: **FACULTY**

F_ID	FName	LName	Hire_Date	Salary
102	Amit	Mishra	12-10-1998	12000
103	Nitin	Vyas	24-12-1994	8000
104	Rakshit	Soni	18-5-2001	14000
105	Rashmi	Malhotra	11-9-2004	11000
106	Sulekha	Srivastava	5-6-2006	10000

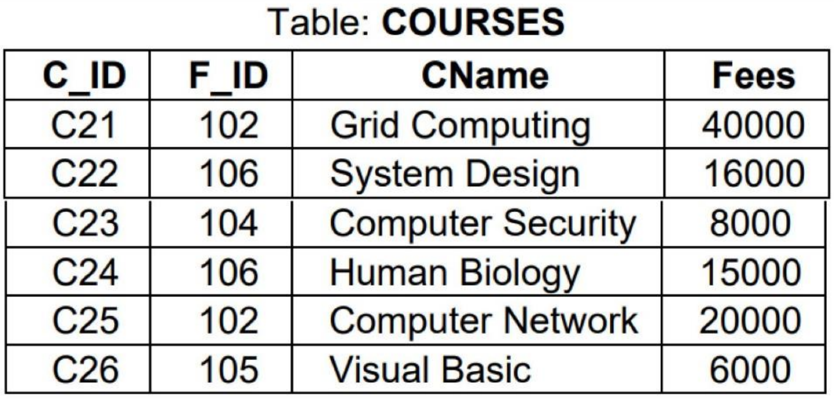


Table: **COURSES**

C_ID	F_ID	CName	Fees
C21	102	Grid Computing	40000
C22	106	System Design	16000
C23	104	Computer Security	8000
C24	106	Human Biology	15000
C25	102	Computer Network	20000
C26	105	Visual Basic	6000

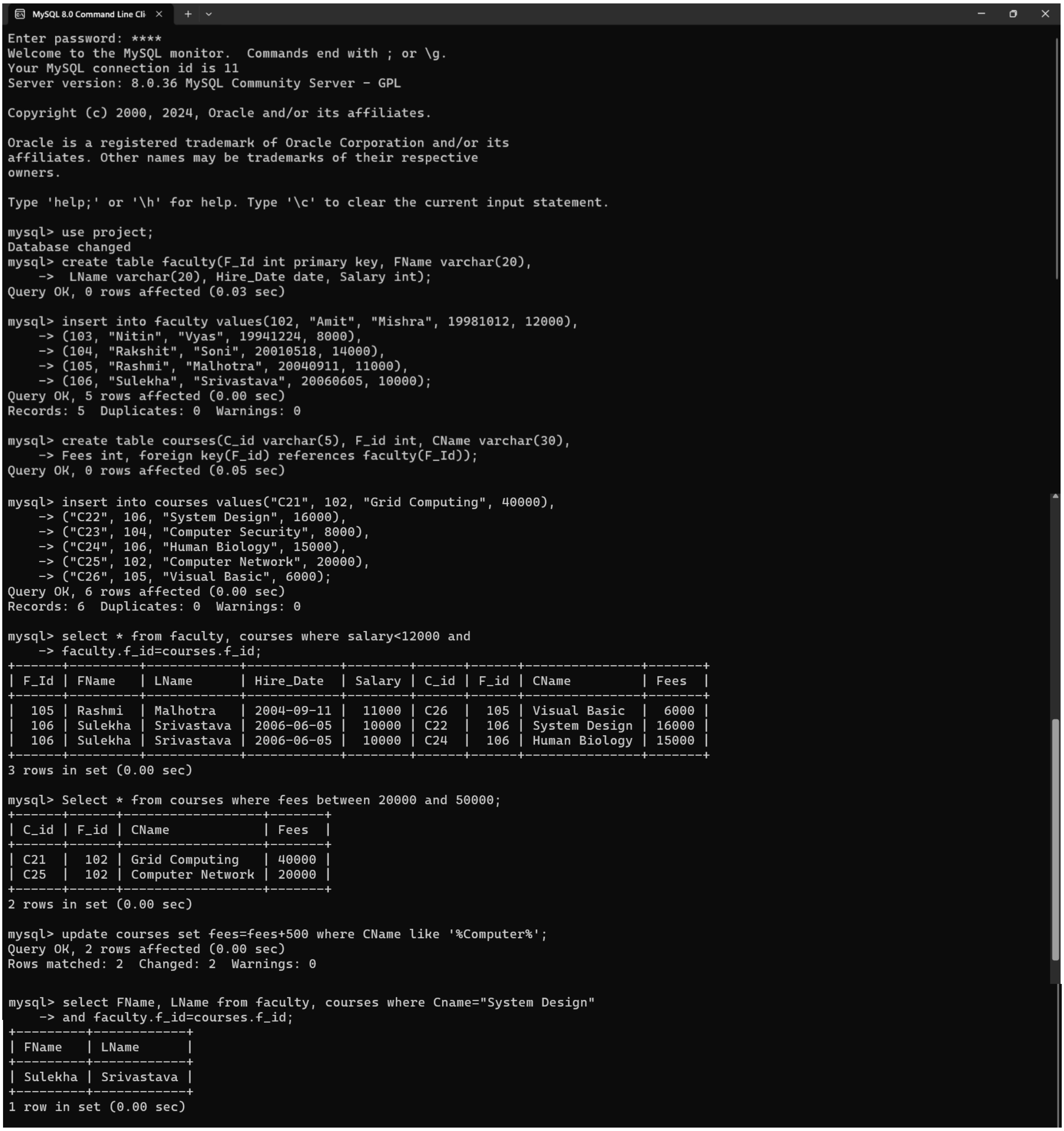
I) To display complete details (from both the tables) of those Faculties whose salary is less than 12000.

(II) To display the details of courses whose fees is in the range of 20000 to 50000 (both values included).

(III) To increase the fees of all courses by 500 which have "Computer" in their Course names.

(IV) To display names (FName and LName) of faculty taking System Design.

Source Code and Output:-



```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.36 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use project;
Database changed
mysql> create table faculty(F_Id int primary key, FName varchar(20),
    ->  LName varchar(20), Hire_Date date, Salary int);
Query OK, 0 rows affected (0.03 sec)

mysql> insert into faculty values(102, "Amit", "Mishra", 19981012, 12000),
    -> (103, "Nitin", "Vyas", 19941224, 8000),
    -> (104, "Rakshit", "Soni", 20010518, 14000),
    -> (105, "Rashmi", "Malhotra", 20040911, 11000),
    -> (106, "Sulekha", "Srivastava", 20060605, 10000);
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> create table courses(C_id varchar(5), F_id int, CName varchar(30),
    -> Fees int, foreign key(F_id) references faculty(F_Id));
Query OK, 0 rows affected (0.05 sec)

mysql> insert into courses values("C21", 102, "Grid Computing", 40000),
    -> ("C22", 106, "System Design", 16000),
    -> ("C23", 104, "Computer Security", 8000),
    -> ("C24", 106, "Human Biology", 15000),
    -> ("C25", 102, "Computer Network", 20000),
    -> ("C26", 105, "Visual Basic", 6000);
Query OK, 6 rows affected (0.00 sec)
Records: 6  Duplicates: 0  Warnings: 0

mysql> select * from faculty, courses where salary<12000 and
    -> faculty.f_id=courses.f_id;
+------+---------+------------+------------+--------+------+------+---------------+-------+
| F_Id | FName   | LName      | Hire_Date  | Salary | C_id | F_id | CName         | Fees  |
+------+---------+------------+------------+--------+------+------+---------------+-------+
|  105 | Rashmi  | Malhotra   | 2004-09-11 |  11000 | C26  |  105 | Visual Basic  |  6000 |
|  106 | Sulekha | Srivastava | 2006-06-05 |  10000 | C22  |  106 | System Design | 16000 |
|  106 | Sulekha | Srivastava | 2006-06-05 |  10000 | C24  |  106 | Human Biology | 15000 |
+------+---------+------------+------------+--------+------+------+---------------+-------+
3 rows in set (0.00 sec)

mysql> Select * from courses where fees between 20000 and 50000;
+------+------+------------------+-------+
| C_id | F_id | CName            | Fees  |
+------+------+------------------+-------+
| C21  |  102 | Grid Computing   | 40000 |
| C25  |  102 | Computer Network | 20000 |
+------+------+------------------+-------+
2 rows in set (0.00 sec)

mysql> update courses set fees=fees+500 where CName like '%Computer%';
Query OK, 2 rows affected (0.00 sec)
Rows matched: 2  Changed: 2  Warnings: 0


mysql> select FName, LName from faculty, courses where Cname="System Design"
    -> and faculty.f_id=courses.f_id;
+---------+------------+
| FName   | LName      |
+---------+------------+
| Sulekha | Srivastava |
+---------+------------+
1 row in set (0.00 sec)
```

Write a menu based program to perform the tasks given in the following questions(Q16-Q20).

Q16.Write a Python program to connect to a MySQL database named SchoolDB.

Q17. Write a SQL query to create a table named Students with the following columns:

StudentID, Name, Age, and Class. Execute this query using a Python script.

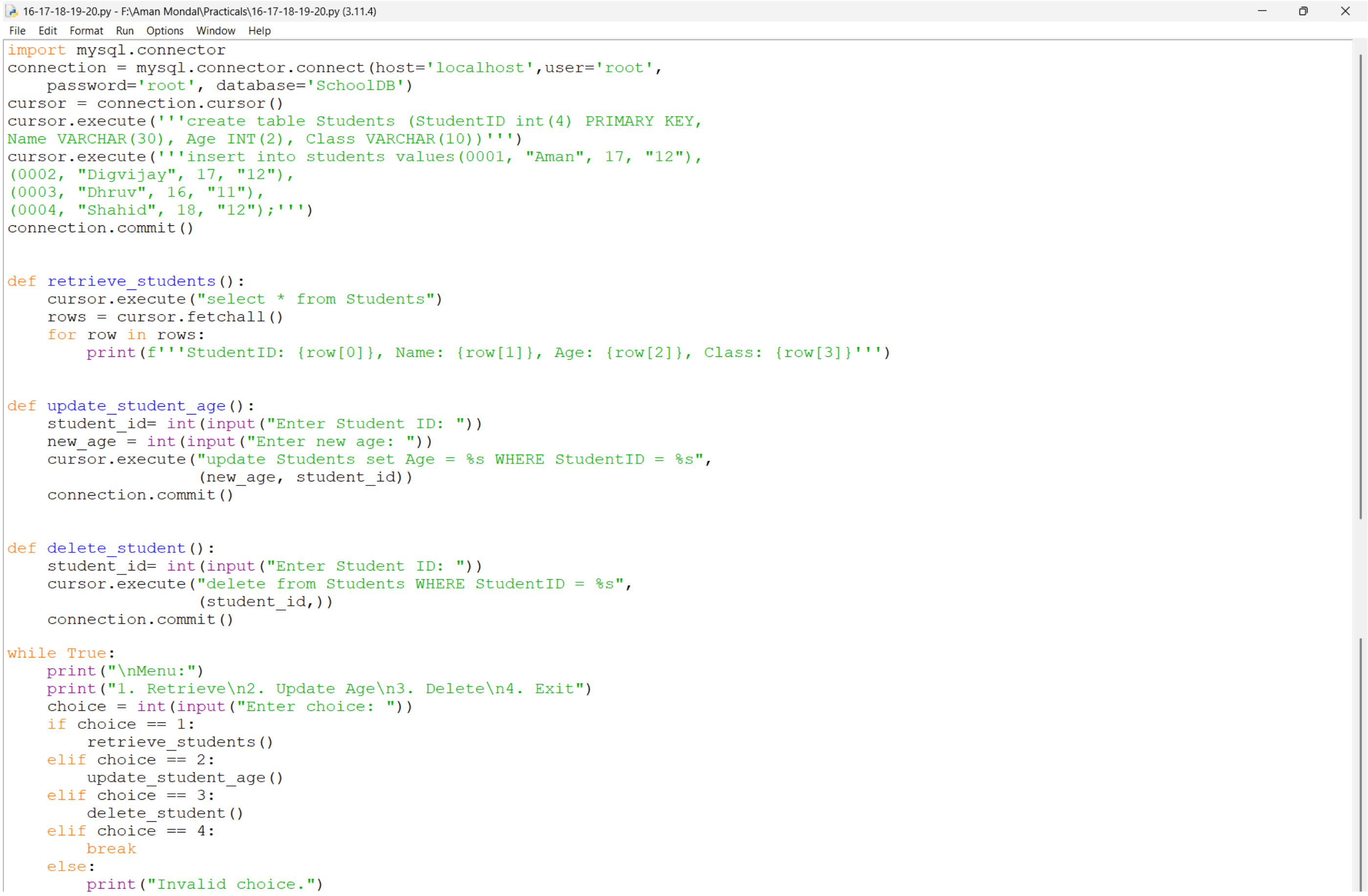
Q18. Create a Python program that retrieves all records from the Students table and

displays them in a formatted manner.

Q19. Write a Python program that updates the age of a specific student in the Students table based on StudentID.

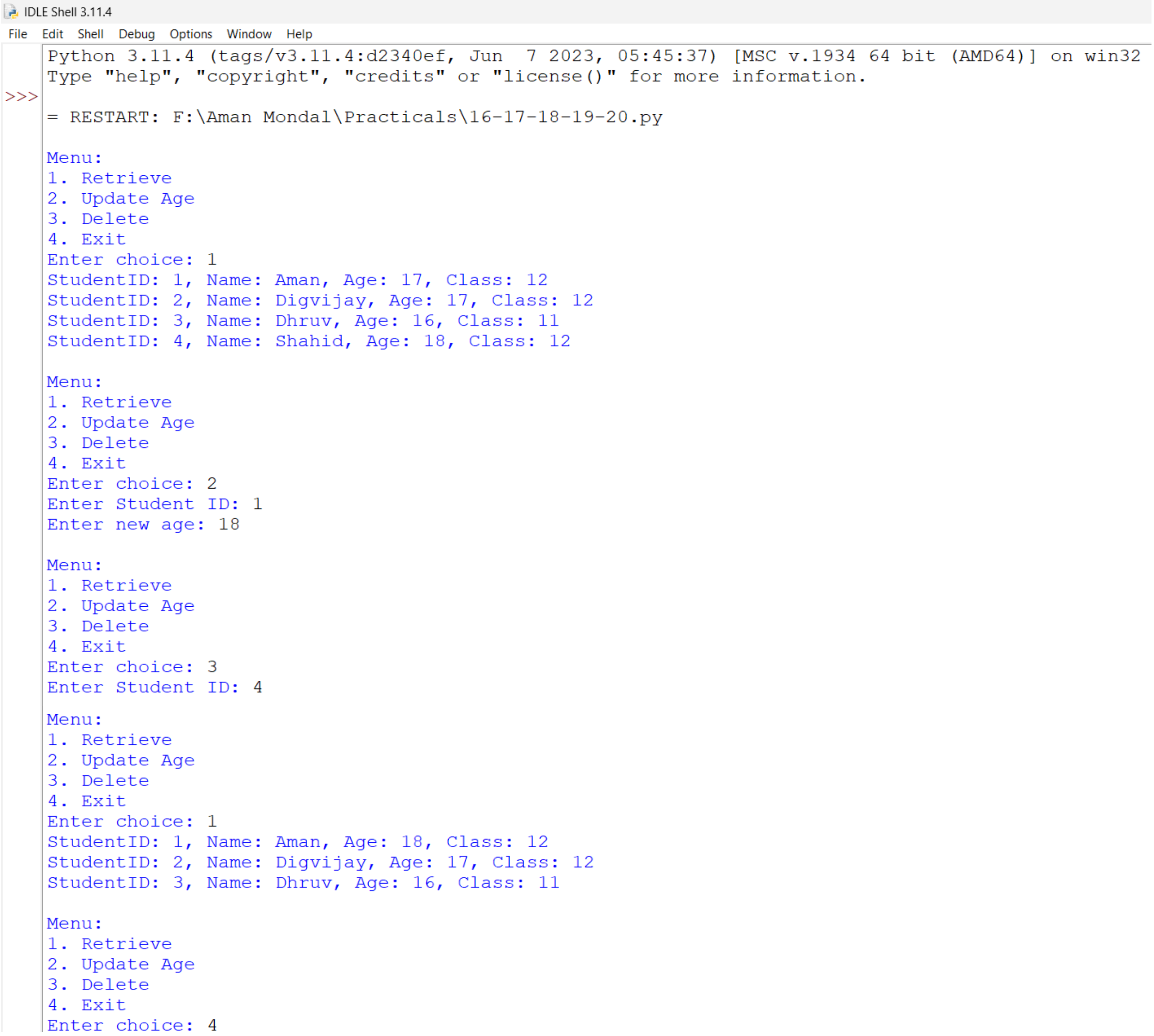
Q20. Write a Python program that deletes a record from the Students table where the StudentID is a specific value.

Source Code:-


16-17-18-19-20.py - F:\Aman Mondal\Practicals\16-17-18-19-20.py (3.11.4)
File Edit Format Run Options Window Help

```python
import mysql.connector
connection = mysql.connector.connect(host='localhost',user='root',
    password='root', database='SchoolDB')
cursor = connection.cursor()
cursor.execute('''create table Students (StudentID int(4) PRIMARY KEY,
Name VARCHAR(30), Age INT(2), Class VARCHAR(10))''')
cursor.execute('''insert into students values(0001, "Aman", 17, "12"),
(0002, "Digvijay", 17, "12"),
(0003, "Dhruv", 16, "11"),
(0004, "Shahid", 18, "12");''')
connection.commit()


def retrieve_students():
    cursor.execute("select * from Students")
    rows = cursor.fetchall()
    for row in rows:
        print(f'''StudentID: {row[0]}, Name: {row[1]}, Age: {row[2]}, Class: {row[3]}''')


def update_student_age():
    student_id= int(input("Enter Student ID: "))
    new_age = int(input("Enter new age: "))
    cursor.execute("update Students set Age = %s WHERE StudentID = %s",
                   (new_age, student_id))
    connection.commit()


def delete_student():
    student_id= int(input("Enter Student ID: "))
    cursor.execute("delete from Students WHERE StudentID = %s",
                   (student_id,))
    connection.commit()

while True:
    print("\nMenu:")
    print("1. Retrieve\n2. Update Age\n3. Delete\n4. Exit")
    choice = int(input("Enter choice: "))
    if choice == 1:
        retrieve_students()
    elif choice == 2:
        update_student_age()
    elif choice == 3:
        delete_student()
    elif choice == 4:
        break
    else:
        print("Invalid choice.")
```

Output:-



```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: F:\Aman Mondal\Practicals\16-17-18-19-20.py

Menu:
1. Retrieve
2. Update Age
3. Delete
4. Exit
Enter choice: 1
StudentID: 1, Name: Aman, Age: 17, Class: 12
StudentID: 2, Name: Digvijay, Age: 17, Class: 12
StudentID: 3, Name: Dhruv, Age: 16, Class: 11
StudentID: 4, Name: Shahid, Age: 18, Class: 12

Menu:
1. Retrieve
2. Update Age
3. Delete
4. Exit
Enter choice: 2
Enter Student ID: 1
Enter new age: 18

Menu:
1. Retrieve
2. Update Age
3. Delete
4. Exit
Enter choice: 3
Enter Student ID: 4

Menu:
1. Retrieve
2. Update Age
3. Delete
4. Exit
Enter choice: 1
StudentID: 1, Name: Aman, Age: 18, Class: 12
StudentID: 2, Name: Digvijay, Age: 17, Class: 12
StudentID: 3, Name: Dhruv, Age: 16, Class: 11

Menu:
1. Retrieve
2. Update Age
3. Delete
4. Exit
Enter choice: 4
```