



Student Name and Number : Sakshi Umesh Teli (20079303)

Course Title : M.Sc. Cyber Security

Lecturer Name: Swati Dongre

Module/Subject Title: Advanced Programming Techniques

Assignment Title: CA_ONE_(30%)

No of Words: 698

TABLE OF CONTENTS

1. Introduction.....	2
2. Part I: C# Programming.....	3-10
2.1 Contact Book.....	3-5
2.2 File Extension System.....	6-10
3. Part II: Python Programming.....	11-14
3.1 Client–Server Application.....	11-12
3.2 Web Scraping and CSV Handling.....	12-14
4. Conclusion.....	15
5. Reference.....	15
6. GitHub Repository Link.....	15

1. INTRODUCTION

In this assignment , I developed four applications in Python and C#. I developed a file extension information system and a contact book in C#. I wrote a hotel web-scraping application in Python which writes information to CSV file and a client-server application in Python to record student admission. These tasks helped me to practice programming, working with data, and solve real-life problems.

2. PART I: C# PROGRAMMING

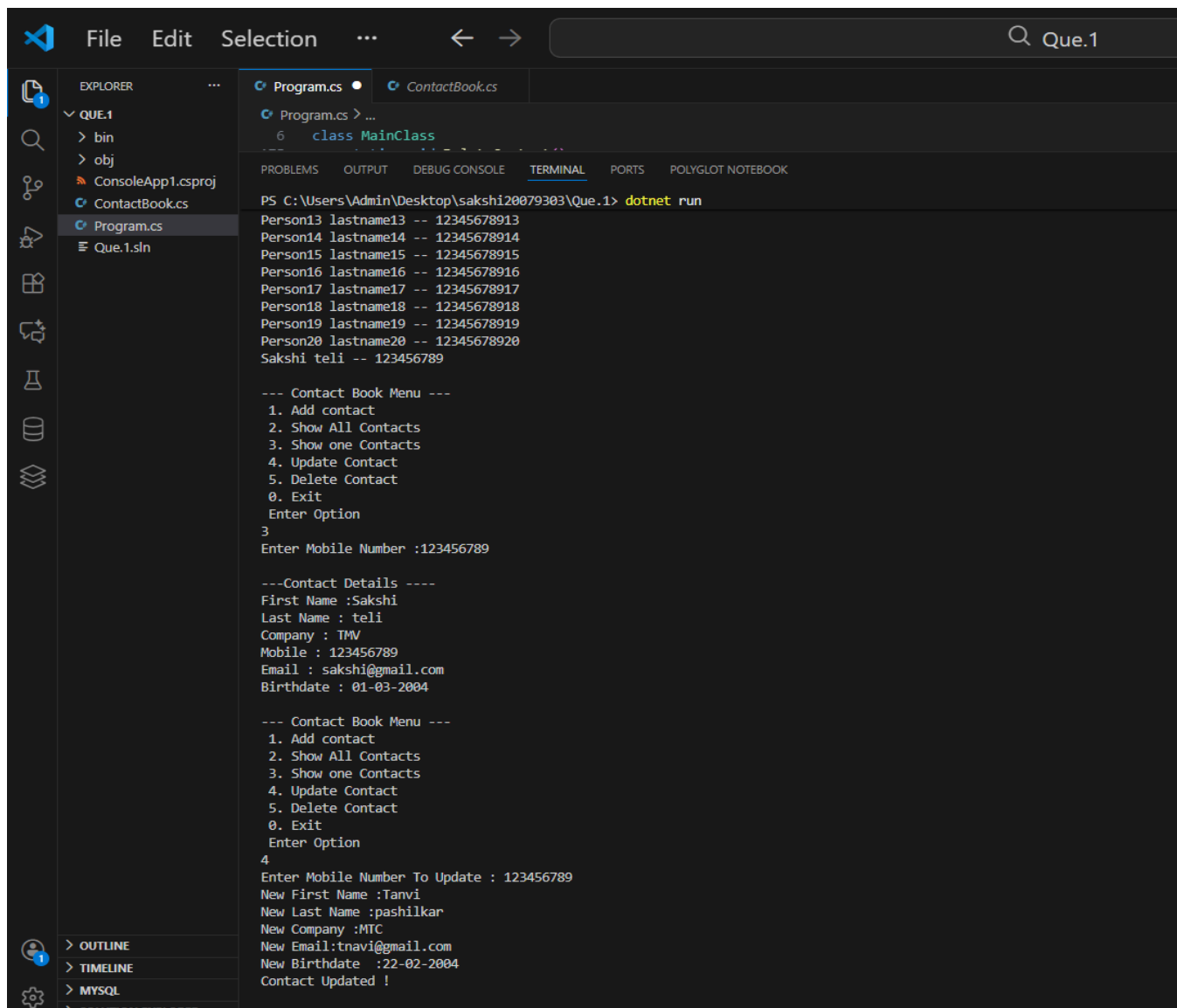
2.1 Contact Book

In this assignment, I developed a C# contact book program . This project consisted of two files :

a) **Program.cs** for the main menu and all functions.

b) **ContactBook.cs** for the **ContactBook** class.

I created a **ContactBook** class with a name, company , phone , email and birthdate of each contact . The program starts with 20 contacts in an array . The menu lets the user to **add , view , update , or delete** a contact , also it checks that the phone number is 9 digits. I used basic **OOP concepts** and tested the program using **dotnet run** , which keeps going until the user exits .



```
File Edit Selection ... < > Que.1
EXPLORER
  QUE.1
    > bin
    > obj
    ConsoleApp1.csproj
    ContactBook.cs
    Program.cs
    Que.1.sln
  SOLUTION EXPLORER
    > OUTLINE
    > TIMELINE
    > MYSQL
    > SOLUTION EXPLORER
  PROGRAM.CS
    class MainClass
  CONTACTBOOK.CS
  PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK
PS C:\Users\Admin\Desktop\sakshi20079303\Que.1> dotnet run
Person13 lastname13 -- 12345678913
Person14 lastname14 -- 12345678914
Person15 lastname15 -- 12345678915
Person16 lastname16 -- 12345678916
Person17 lastname17 -- 12345678917
Person18 lastname18 -- 12345678918
Person19 lastname19 -- 12345678919
Person20 lastname20 -- 12345678920
Sakshi teli -- 123456789

--- Contact Book Menu ---
1. Add contact
2. Show All Contacts
3. Show one Contacts
4. Update Contact
5. Delete Contact
0. Exit
Enter Option
3
Enter Mobile Number :123456789

---Contact Details ---
First Name :Sakshi
Last Name : teli
Company : TMV
Mobile : 123456789
Email : sakshi@gmail.com
Birthdate : 01-03-2004

--- Contact Book Menu ---
1. Add contact
2. Show All Contacts
3. Show one Contacts
4. Update Contact
5. Delete Contact
0. Exit
Enter Option
4
Enter Mobile Number To Update : 123456789
New First Name :Tanvi
New Last Name :pashilkar
New Company :MTC
New Email:tnavi@gmail.com
New Birthdate :22-02-2004
Contact Updated !

--- Contact Book Menu ---
```

The screenshot shows the Visual Studio Code interface with the following components:

- Explorer:** Displays the project structure for 'QUE.1', including 'bin', 'obj', 'ConsoleApp1.csproj', 'ContactBook.cs', 'Program.cs', and 'Que.1.sln'.
- Editor:** Shows the 'Program.cs' file with a 'MainClass' containing a menu and a list of 20 contacts.
- Terminal:** Shows the command 'dotnet run' being executed, resulting in the application's output.

Program.cs Code:

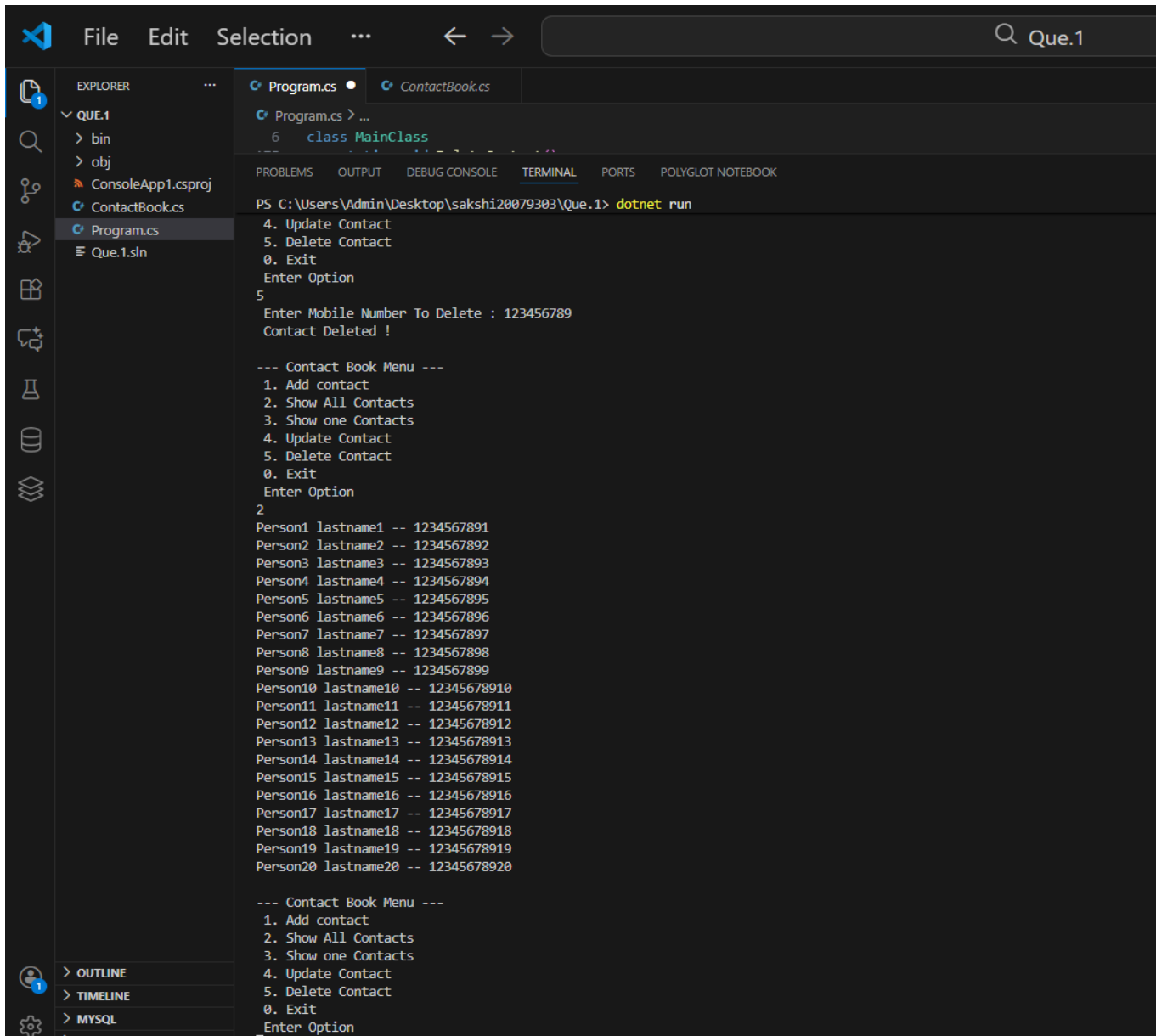
```
class MainClass
{
    static void Main()
    {
        1. Add contact
        2. Show All Contacts
        3. Show one Contacts
        4. Update Contact
        5. Delete Contact
        0. Exit
        Enter Option
        1
        First Name :Sakshi
        Last Name :teli
        Company : TMV
        Mobile Number ( 9 digits ) : 123456789
        Email : sakshi@gmail.com
        Birthdate :01-03-2004
        Contact Added !

        --- Contact Book Menu ---
        1. Add contact
        2. Show All Contacts
        3. Show one Contacts
        4. Update Contact
        5. Delete Contact
        0. Exit
        Enter Option
        2
        Person1 lastname1 -- 1234567891
        Person2 lastname2 -- 1234567892
        Person3 lastname3 -- 1234567893
        Person4 lastname4 -- 1234567894
        Person5 lastname5 -- 1234567895
        Person6 lastname6 -- 1234567896
        Person7 lastname7 -- 1234567897
        Person8 lastname8 -- 1234567898
        Person9 lastname9 -- 1234567899
        Person10 lastname10 -- 12345678910
        Person11 lastname11 -- 12345678911
        Person12 lastname12 -- 12345678912
        Person13 lastname13 -- 12345678913
        Person14 lastname14 -- 12345678914
        Person15 lastname15 -- 12345678915
        Person16 lastname16 -- 12345678916
        Person17 lastname17 -- 12345678917
        Person18 lastname18 -- 12345678918
        Person19 lastname19 -- 12345678919
        Person20 lastname20 -- 12345678920
        Sakshi teli -- 123456789
    }
}
```

Terminal Output:

```
PS C:\Users\Admin\Desktop\sakshi20079303\Que.1> dotnet run
1. Add contact
2. Show All Contacts
3. Show one Contacts
4. Update Contact
5. Delete Contact
0. Exit
Enter Option
1
First Name :Sakshi
Last Name :teli
Company : TMV
Mobile Number ( 9 digits ) : 123456789
Email : sakshi@gmail.com
Birthdate :01-03-2004
Contact Added !

--- Contact Book Menu ---
1. Add contact
2. Show All Contacts
3. Show one Contacts
4. Update Contact
5. Delete Contact
0. Exit
Enter Option
2
Person1 lastname1 -- 1234567891
Person2 lastname2 -- 1234567892
Person3 lastname3 -- 1234567893
Person4 lastname4 -- 1234567894
Person5 lastname5 -- 1234567895
Person6 lastname6 -- 1234567896
Person7 lastname7 -- 1234567897
Person8 lastname8 -- 1234567898
Person9 lastname9 -- 1234567899
Person10 lastname10 -- 12345678910
Person11 lastname11 -- 12345678911
Person12 lastname12 -- 12345678912
Person13 lastname13 -- 12345678913
Person14 lastname14 -- 12345678914
Person15 lastname15 -- 12345678915
Person16 lastname16 -- 12345678916
Person17 lastname17 -- 12345678917
Person18 lastname18 -- 12345678918
Person19 lastname19 -- 12345678919
Person20 lastname20 -- 12345678920
Sakshi teli -- 123456789
```



2.2 File Extension System

In response to this question, I created a C# console application that displays the information about different file extensions. In this project .

I was working with two files :

- a) **Program. cs** – The menu and all functions are contained here.
- b) **FileExtension. cs** – The class contains the FileExtension .

FileExtension class contains two aspects:

- a) The file extension (like .mp4,.pdf)
- b) Its description (like video file, document , audio file)

I did a program where 20 default file extensions were created by default with the help of the **AddDefaultExtension()** method. The menu will allow the user to **view , add, delete ,update** or search an extension . In case the user enters an extension but forgets to add a dot , the program will add it automatically . In case there is no extension , a message stating not found is displayed . I was able to apply simple OOP principles such as classes , objects , and methods and debugged all of them by dotnet run .

File Edit Selection ... ← → Que.2

EXPLORER

- QUE.2
 - ConsoleApp1
 - bin
 - obj
 - ConsoleApp1.csproj
 - FileExtension.cs
 - Program.cs
 - ConsoleApp1.sln

Program.cs

```

5 class MainClass

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK

```

PS C:\Users\Admin\Desktop\sakshi20079303\Que.2> cd "C:\Users\Admin\Desktop\sakshi20079303\Que.2\ConsoleApp1"
>>
PS C:\Users\Admin\Desktop\sakshi20079303\Que.2\ConsoleApp1> dotnet run

---File Extension ---
1.Show All Extensions
2.Show One Extensions
3.Add Extensions
4.Update Extensions
5.Delete Extensions
0.Exit
Enter option: 1

Supported Extensions :
.mp4 - Video File
.avi - Video File
.mov - Video File
.mkv - Video File
.webm - Video File
.mp3 - Audio File
.wav - Audio File
.flac - Audio File
.jpg - Image File
.jpeg - Image File
.png - Image File
.gif - Image File
.pdf - Document
.docx - Word Document
.xlsx - Excel Sheet
.pptx - PowerPoint
.txt - Text File
.zip - Compressed File
.rar - Compressed File
.7z - Compressed File

---File Extension ---
1.Show All Extensions
2.Show One Extensions
3.Add Extensions
4.Update Extensions
5.Delete Extensions
0.Exit
Enter option: 2
Enter Extension : .7z
Extension : .7z
Description: Compressed File

---File Extension ---
1.Show All Extensions

```

OUTLINE TIMELINE MYSQL SOLUTION EXPLORER

0 0 Projects: ConsoleApp1 Debug Any CPU Solution file opened: ConsoleApp1.sln

File Edit Selection ... ← → Que.2

EXPLORER

▼ QUE.2

▼ ConsoleApp1

- > bin
- > obj
- ConsoleApp1.csproj
- FileExtension.cs
- Program.cs

ConsoleApp1.sln

Program.cs

```

ConsoleApp1 > Program.cs > ...
5 class MainClass

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK

PS C:\Users\Admin\Desktop\sakshi20079303\Que.2\ConsoleApp1> dotnet run

```

---File Extension ---
1.Show All Extensions
2.Show One Extensions
3.Add Extensions
4.Update Extensions
5.Delete Extensions
0.Exit
Enter option: 3
Enter Extension : .png
Enter Description : Image file
Extension added !

---File Extension ---
1.Show All Extensions
2.Show One Extensions
3.Add Extensions
4.Update Extensions
5.Delete Extensions
0.Exit
Enter option: 1

Supported Extensions :
.mp4 - Video File
.avi - Video File
.mov - Video File
.mkv - Video File
.webm - Video File
.mp3 - Audio File
.wav - Audio File
.flac - Audio File
.jpg - Image File
.jpeg - Image File
.png - Image File
.gif - Image File
.pdf - Document
.docx - Word Document
.xlsx - Excel Sheet
.pptx - PowerPoint
.txt - Text File
.zip - Compressed File
.rar - Compressed File
.7z - Compressed File
.png - Image file

---File Extension ---
1.Show All Extensions
2.Show One Extensions
3.Add Extensions

```

> OUTLINE

> TIMELINE

> MYSQL

> SOLUTION EXPLORER

0 0 0 Projects: ConsoleApp1 Debug Any CPU Solution file opened: ConsoleApp1.sln

Visual Studio interface showing a C# console application named ConsoleApp1. The Explorer pane on the left shows the project structure: ConsoleApp1.csproj, FileExtension.cs, Program.cs, and ConsoleApp1.sln. The Solution Explorer pane at the bottom shows the project structure: ConsoleApp1 > bin, obj, ConsoleApp1.csproj, FileExtension.cs, Program.cs, and ConsoleApp1.sln.

The main editor area displays the Program.cs file, which contains the following code:

```
5 class MainClass
```

The TERMINAL pane shows the output of the application, which is a menu-driven program for managing file extensions. The output is as follows:

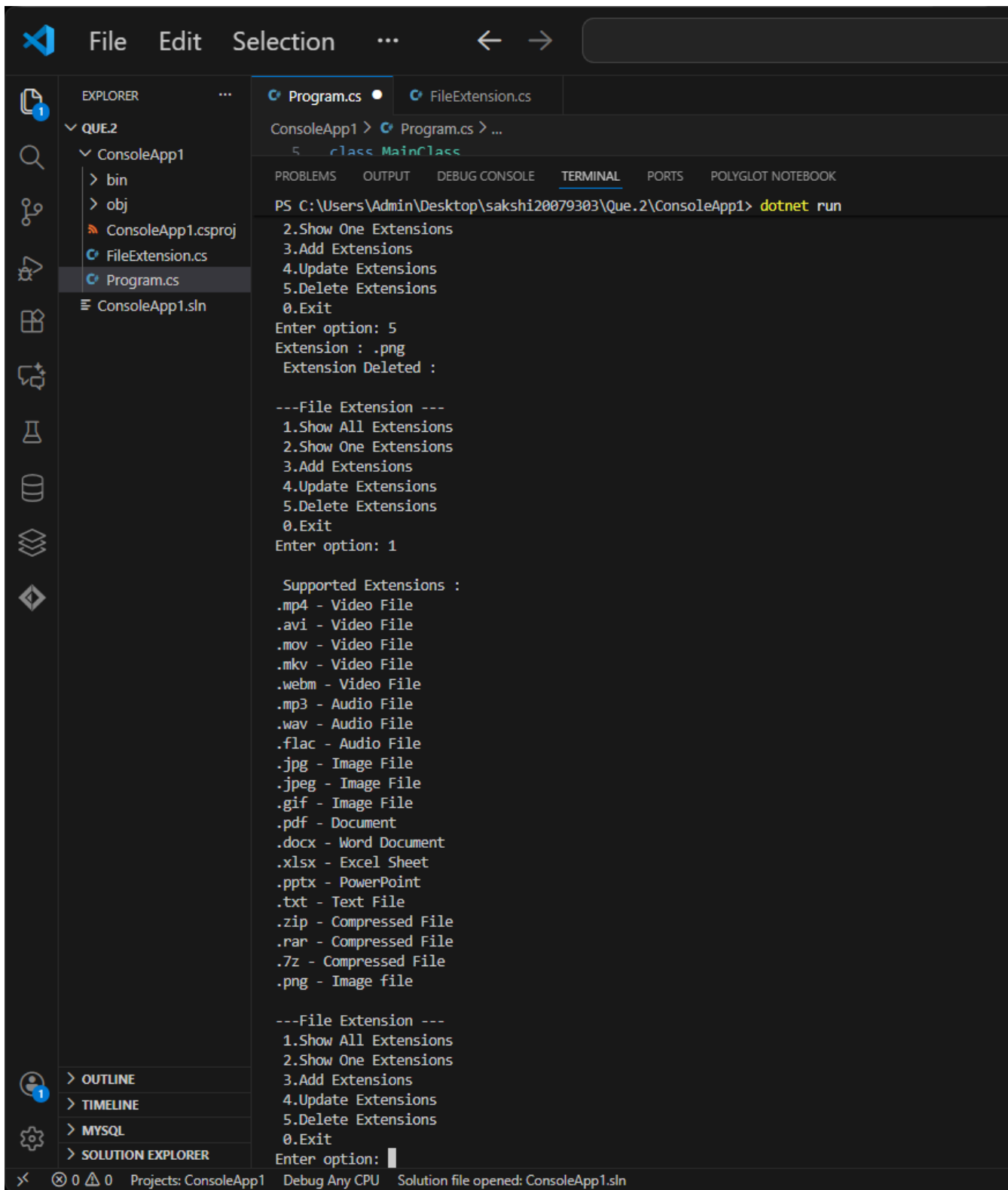
```
PS C:\Users\Admin\Desktop\sakshi20079303\Que.2\ConsoleApp1> dotnet run

2.Show One Extensions
3.Add Extensions
4.Update Extensions
5.Delete Extensions
0.Exit
Enter option: 4
Extension : .png
New Description : Picture
Extension Updated

---File Extension ---
1.Show All Extensions
2.Show One Extensions
3.Add Extensions
4.Update Extensions
5.Delete Extensions
0.Exit
Enter option: 1

Supported Extensions :
.mp4 - Video File
.avi - Video File
.mov - Video File
.mkv - Video File
.webm - Video File
.mp3 - Audio File
.wav - Audio File
.flac - Audio File
.jpg - Image File
.jpeg - Image File
.png - Picture
.gif - Image File
.pdf - Document
.docx - Word Document
.xlsx - Excel Sheet
.pptx - PowerPoint
.txt - Text File
.zip - Compressed File
.rar - Compressed File
.7z - Compressed File
.png - Image file

---File Extension ---
1.Show All Extensions
2.Show One Extensions
3.Add Extensions
4.Update Extensions
5.Delete Extensions
```

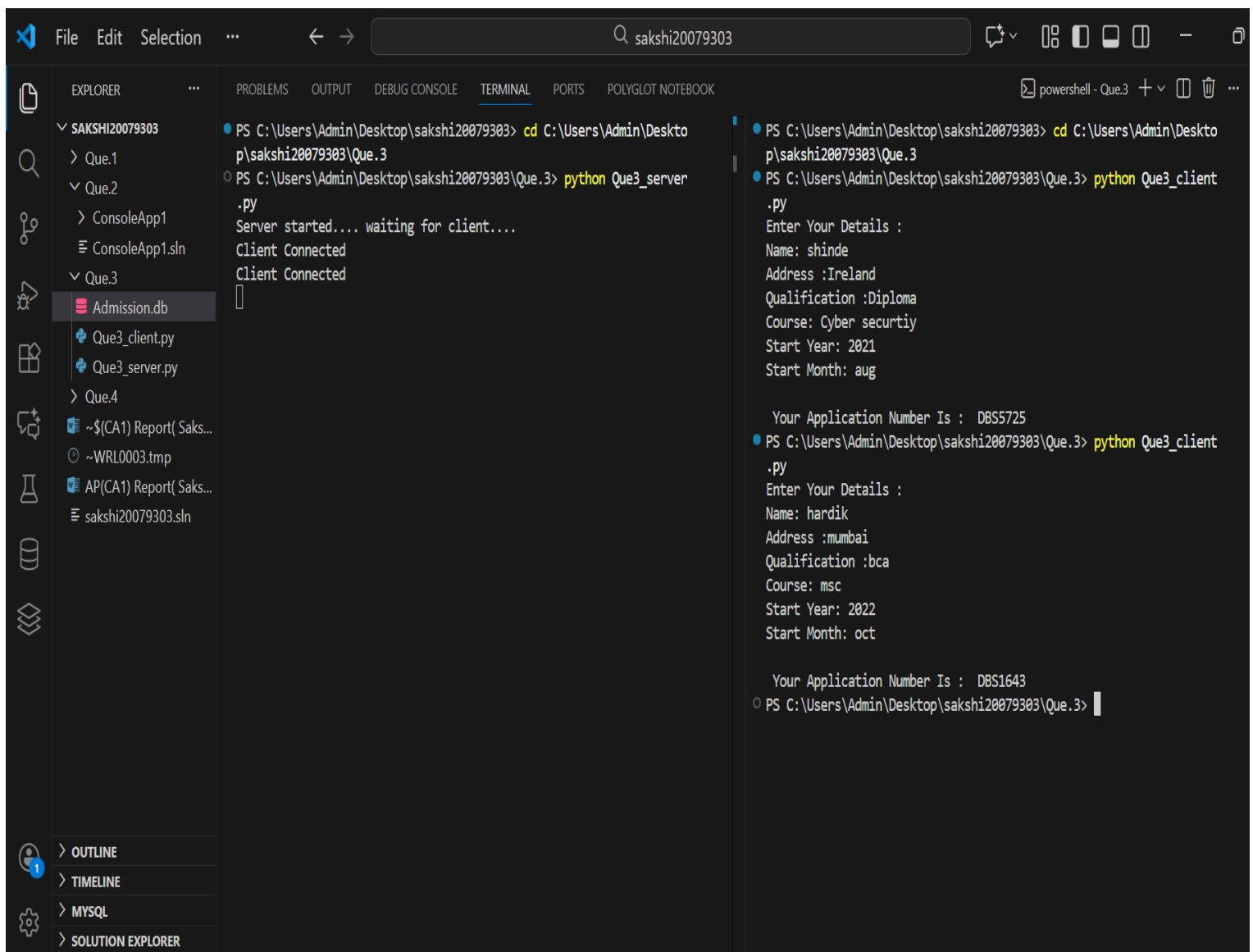


3. PART II: PYTHON PROGRAMMING

3.1 Client–Server Application

I created a server program based on **Python Sockets**, which waits to have a client . I created a client program that sends information about student admission to the server . I used **SQLite** to create a database file that contains all of the student data .

Once the data is received by the server , a unique application number is created and stored . The server sends the application number to the client . The client is sent back the application number by the server. The client shows the application number to the user . This work makes use of **databases**, **TCP communication** and **basic string manipulation**.



```
File Edit Selection ... sakshi20079303
```

EXPLORER

- SAKSHI20079303
 - Que.1
 - Que.2
 - ConsoleApp1
 - ConsoleApp1.sln
 - Que.3
 - Admission.db
 - Que3_client.py
 - Que3_server.py
 - Que.4
- ~\$(CA1) Report(Saks...
- ~WRL0003.tmp
- AP(CA1) Report(Saks...
- sakshi20079303.sln

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS POLYGLOT NOTEBOOK

PS C:\Users\Admin\Desktop\sakshi20079303> cd C:\Users\Admin\Desktop\sakshi20079303\Que.3

PS C:\Users\Admin\Desktop\sakshi20079303\Que.3> python Que3_server.py

Server started.... waiting for client....

Client Connected

Client Connected

PS C:\Users\Admin\Desktop\sakshi20079303\Que.3> python Que3_client.py

Enter Your Details :

Name: shinde

Address :Ireland

Qualification :Diploma

Course: Cyber securtiy

Start Year: 2021

Start Month: aug

Your Application Number Is : DBS5725

PS C:\Users\Admin\Desktop\sakshi20079303\Que.3> python Que3_client.py

Enter Your Details :

Name: hardik

Address :mumbai

Qualification :bca

Course: msc

Start Year: 2022

Start Month: oct

Your Application Number Is : DBS1643

PS C:\Users\Admin\Desktop\sakshi20079303\Que.3>

Admission.db X

Que.3 > Admission.db

Filter 1 t Rows: 5 Filter 5 rows...

	Applicati...	Name	Address	Qualificat...	Course	Year	Month
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	DBS6026	sakshi	pune	BCA	cyber security	2025	sep
2	DBS8901	tanvi	mumbai	bca	AI	2025	02
3	DBS8739	shinde	ireland	diploma	cyber	2021	03
4	DBS5725	shinde	Ireland	Diploma	Cyber securtiy	2021	aug
5	DBS1643	hardik	mumbai	bca	msc	2022	oct
6							

SQLITE VIEW... v0.10.6 Page 1 / 1

Try Hydejack, a Jekyll theme for hackers, nerds, and academics

3.2 Web Scraping and CSV Handling

I created two sample **Hotel Webpages (hotelA.html and hotelB.html)** containing the names of rooms and prices since scraping of actual hotel websites is not possible . Then I wrote Python program which scraped these two pages with **BeautifulSoup** and collected the room data which was then stored in a CSV file and finally read the CSV file to display the results . This also enabled me to show safe and legal web scraping .

Visual Studio Code interface showing a Python web scraping project.

EXPLORER

- SAKSHI20079303
 - Que.1
 - Que.2
 - Que.3
 - Admission.db
 - Que3_client.py
 - Que3_server.py
 - Que.4
 - hotelA.html
 - hotelB.html
 - HotelRooms.csv
 - webScraping.py

webScraping.py

```
1 import requests
```

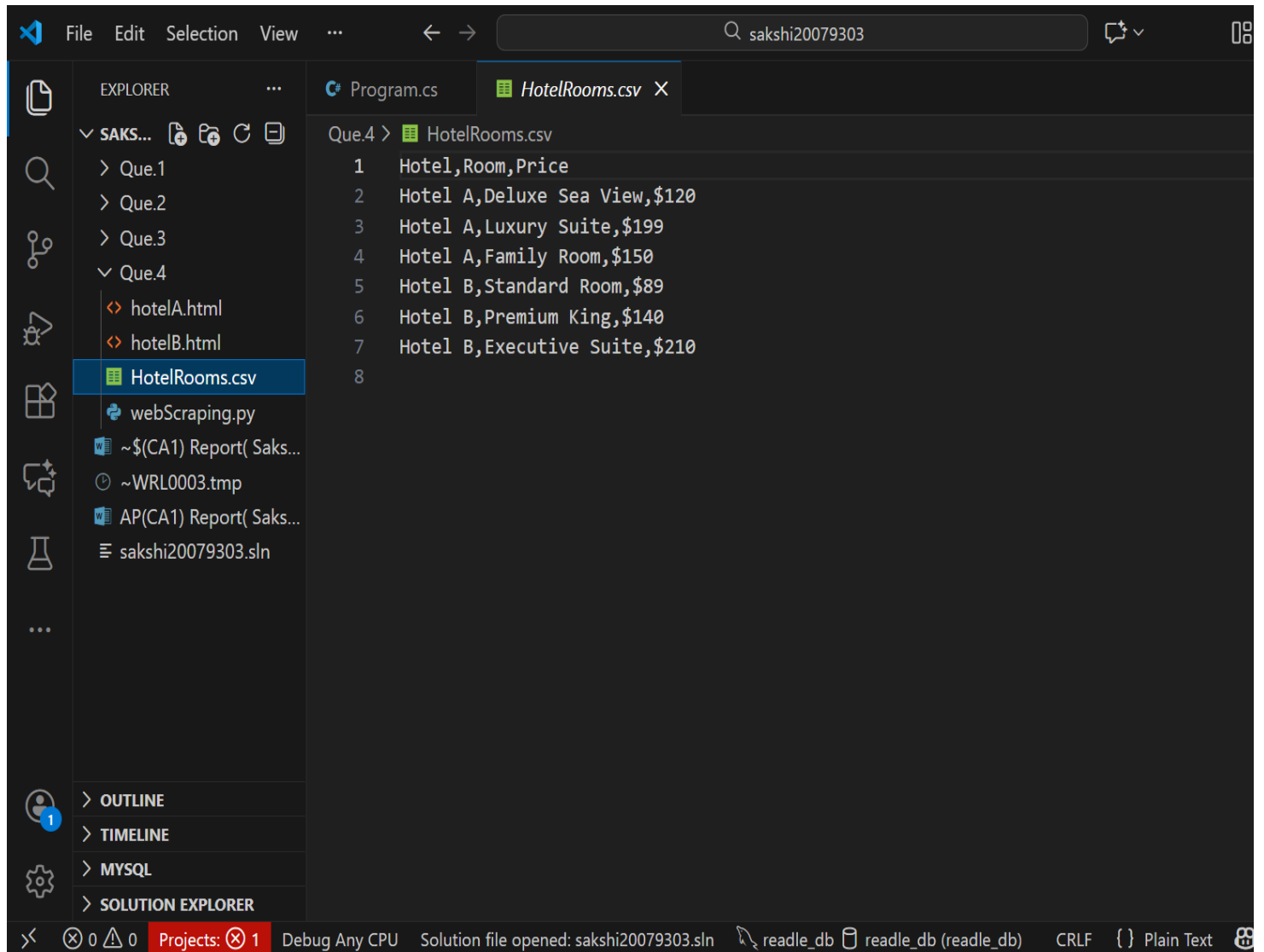
TERMINAL

```
PS C:\Users\Admin\Desktop\sakshi20079303> cd C:\Users\Admin\Desktop\sakshi20079303\Que.4
PS C:\Users\Admin\Desktop\sakshi20079303\Que.4> python webScraping.py
Scraping Hotel A...
Scraping Hotel B...
Saving to csv...
CSV saved as HotelRooms.csv

Reading csv data:

{'Hotel': 'Hotel A', 'Room': 'Deluxe Sea View', 'Price': '$120'}
{'Hotel': 'Hotel A', 'Room': 'Luxury Suite', 'Price': '$199'}
{'Hotel': 'Hotel A', 'Room': 'Family Room', 'Price': '$150'}
{'Hotel': 'Hotel B', 'Room': 'Standard Room', 'Price': '$89'}
{'Hotel': 'Hotel B', 'Room': 'Premium King', 'Price': '$140'}
{'Hotel': 'Hotel B', 'Room': 'Executive Suite', 'Price': '$210'}

Done!
PS C:\Users\Admin\Desktop\sakshi20079303\Que.4>
```



4. CONCLUSION

I was able to practice C# and Python by creating basic applications for this assignment. I learned how to use OOP , data management , database administration , and client-server communication.

5. REFERENCE

- a) **Microsoft. (2024)** *C# Documentation: Classes and Objects*. Available at:
<https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/object-oriented/classes>
- b) **Python Software Foundation. (2024)** *socket — Low-level networking interface*.
Available at: <https://docs.python.org/3/library/socket.html>
- c) **SQLite. (2024)** *SQLite Language Documentation*. Available at:
<https://www.sqlite.org/lang.html>

6. GITHUB REPOSITORY LINK

<https://github.com/sakshi220122/Sakshi20079303-AdvProgrammingCA1-.git>