

CA 1 Assignment



Student Number as per your student card: Ravindher Ashokan 20063604

Course Title: M.Sc. Cyber Security

Lecturer Name: Swati Dongre

Module/Subject Title: Advanced Programming Techniques

Assignment Title: CA_ONE_ (30%)

No of Words: 526 Words

GitHub Link - <https://github.com/Ravindher133/Programming-CA1.git>

TABEL OF CONTENT

1. Introduction

2. Program 1 – Contact Book Application (C#)

2.1 Contact list

3. Program 2 – File Extension Information System (C#)

3.1 Valid & Invalid Extension

4. Program 3 – College Admission Number Generator (Python)

4.1 Server details

4.2 Client details

5. Program 4

5.1 Scraping

6. Conclusion

7. References

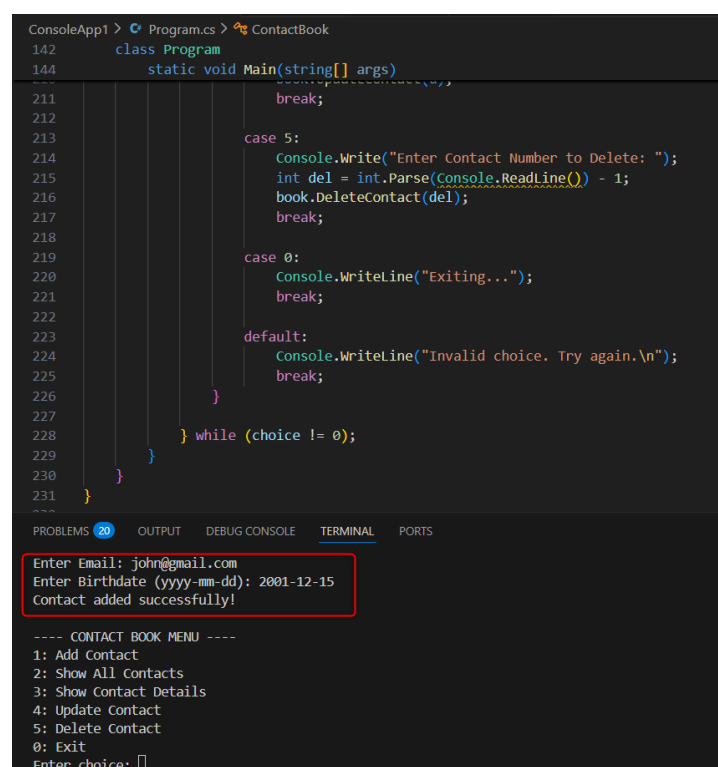
1. Introduction

This document will discuss the implementation and ensuing result outputs pertaining to four programming assignments done using C# and Python. Also, every task will detail the objective, functionality, and result outputs. The screenshots included within this document will be regarding result outputs.

2. Program 1 – Contact Book Application (C#)

The first program is to implement a console-based contact book system. The main objective is to permit the user to store, view, update, and delete the records of contacts. The application provides a menu-driven interface where users can input the details and perform operations on the stored contact list.

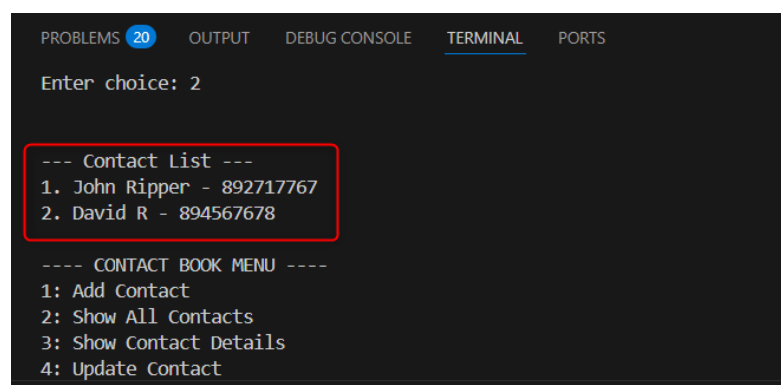
2.1 Contact list



```
ConsoleApp1 > Program.cs > ContactBook
142 class Program
143 {
144     static void Main(string[] args)
145     {
146         // ...
147         break;
148     }
149     case 5:
150         Console.Write("Enter Contact Number to Delete: ");
151         int del = int.Parse(Console.ReadLine()) - 1;
152         book.DeleteContact(del);
153         break;
154     case 0:
155         Console.WriteLine("Exiting...");
156         break;
157     default:
158         Console.WriteLine("Invalid choice. Try again.\n");
159         break;
160 }
161 } while (choice != 0);
162 }
```

Enter Email: john@gmail.com
Enter Birthdate (yyyy-mm-dd): 2001-12-15
Contact added successfully!

---- CONTACT BOOK MENU ----
1: Add Contact
2: Show All Contacts
3: Show Contact Details
4: Update Contact
5: Delete Contact
0: Exit
Enter choice: 1



```
PROBLEMS 20 OUTPUT DEBUG CONSOLE TERMINAL PORTS
Enter choice: 2
--- Contact List ---
1. John Ripper - 892717767
2. David R - 894567678
---- CONTACT BOOK MENU ----
1: Add Contact
2: Show All Contacts
3: Show Contact Details
4: Update Contact
5: Delete Contact
0: Exit
Enter choice: 1
```

Figure 2.1.1 Contact book list output

3. Program 2 – File Extension Information System (C#)

This program defines a meaning for various file extensions. The program is executed and based on a prompt; the user has to enter a file extension. In case the extension exists on the system, the description of that corresponding file will appear; otherwise, the program will notify the user that the extension is not recognized. Hint: typing 'exit' will terminate the program.

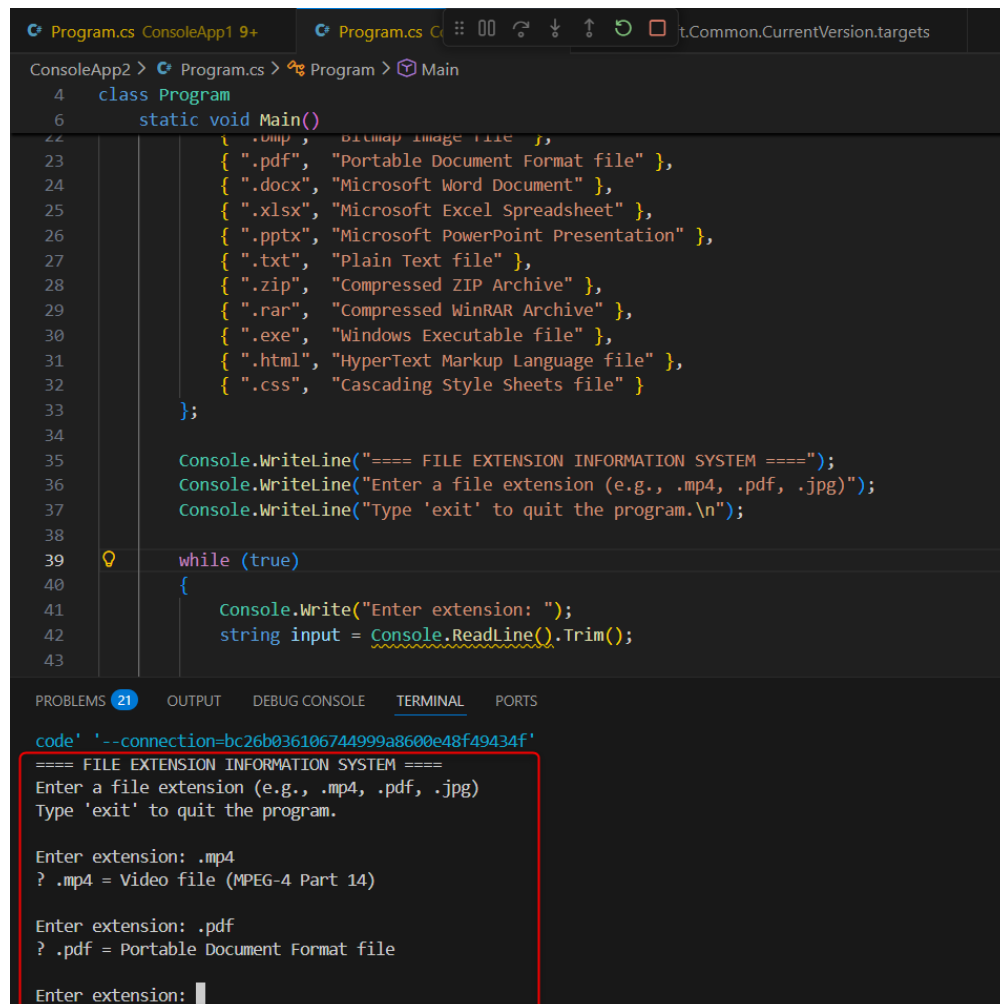
Ex: below

.mp4 -> Video File

.pdf -> Document

.jpg -> Image

3.1 Valid & Invalid Extension



```
Program.cs ConsoleApp1 9+ Program.cs C# :: 00 ↺ ↻ ↶ ↷ ↸ ↹ ↺ ↻ ↶ ↷ ↸ ↹ t.Common.CurrentVersion.targets
ConsoleApp2 > Program.cs > Program > Main
4 class Program
6 static void Main()
22 {
23     { ".mp4", "Video file (MPEG-4 Part 14)" },
24     { ".pdf", "Portable Document Format file" },
25     { ".docx", "Microsoft Word Document" },
26     { ".xlsx", "Microsoft Excel Spreadsheet" },
27     { ".pptx", "Microsoft PowerPoint Presentation" },
28     { ".txt", "Plain Text file" },
29     { ".zip", "Compressed ZIP Archive" },
30     { ".rar", "Compressed WinRAR Archive" },
31     { ".exe", "Windows Executable file" },
32     { ".html", "HyperText Markup Language file" },
33     { ".css", "Cascading Style Sheets file" }
34 };
35
36 Console.WriteLine("==== FILE EXTENSION INFORMATION SYSTEM ====");
37 Console.WriteLine("Enter a file extension (e.g., .mp4, .pdf, .jpg)");
38 Console.WriteLine("Type 'exit' to quit the program.\n");
39
40 while (true)
41 {
42     Console.Write("Enter extension: ");
43     string input = Console.ReadLine().Trim();
44 }
45
46 PROBLEMS 21 OUTPUT DEBUG CONSOLE TERMINAL PORTS
code' '--connection=bc26b036106744999a8600e48f49434f'
==== FILE EXTENSION INFORMATION SYSTEM ====
Enter a file extension (e.g., .mp4, .pdf, .jpg)
Type 'exit' to quit the program.

Enter extension: .mp4
? .mp4 = Video file (MPEG-4 Part 14)

Enter extension: .pdf
? .pdf = Portable Document Format file

Enter extension: 
```

Figure 3.1.1 Valid Extension output

4.2 Client details

```

server.py U client.py X
Q3> client.py> main
43 return v
44
45 def main():
46     print("DBS Application Client\nPlease enter applicant information.")
47     name = collect_input("Name: ")
48     address = collect_input("Address: ")
49     qualifications = collect_input("Educational qualifications: ")
50     # course should be one of three choices
51     courses = ["MSC in Cyber Security", "MSC Information Systems & computing", "MSC Data Analytics"]
52     print("Choose course:")
53     for i,c in enumerate(courses,1):
54         print(f"{i}. {c}")
55     while True:
56         sel = input("Course (1-3): ").strip()
57         if sel in ("1","2","3"):
58             course = courses[int(sel)-1]
59             break
60         else:
61             print("Invalid option, choose 1-3.")
62     while True:
63         start_year = input("Intended start year (e.g., 2025): ").strip()
64         start_month = input("Intended start month (1-12): ").strip()
65         try:
66             sy = int(start_year)

```

Figure 4.2.1 Client Output

5. Program 4 – Web Scraping and Comparative Analysis (Python)

Within this program, techniques for web scraping have been incorporated to analyze data obtained from the ‘Books to Scrape’ website. The functions of these scripts include acquiring data on books such as titles, prices, and reviews. It then proceeds to analyze these sets of data.

5.1 Scraping

```

Scrape.py x Program.cs 9+
Q4 > Scrape.py > scrape_site
5 def scrape_site(url):
6     books = []
7
8     response = requests.get(url)
9     soup = BeautifulSoup(response.text, "html.parser")
10
11     products = soup.select(".product_pod")
12
13     for item in products[:5]:
14         title = item.h3.a["title"]
15         price = item.select_one(".price_color").text
16
17         books.append([title, price])
18
19     return books
20
21 book1_url = "https://books.toscrape.com/catalogue/category/books/travel_2/index.html"
22 book2_url = "https://books.toscrape.com/catalogue/category/books/mystery_3/index.html"
23
24 book1_data = scrape_site(book1_url)
25 book2_data = scrape_site(book2_url)
26
27 csv_filename = "hotel_prices.csv"

```

PROBLEMS 20 OUTPUT DEBUG CONSOLE TERMINAL PORTS POSTMAN CONSOLE

```

PS C:\Users\aravi\OneDrive\Desktop\CA1 Program\Q4> python Scrape.py
['Book A', 'It's Only the Himalayas', '£45.17']
['Book A', 'Full Moon over Noah's Ark: An Odyssey to Mount Ararat and Beyond', '£49.43']
['Book A', 'See America: A Celebration of Our National Parks & Treasured Sites', '£48.87']
['Book A', 'Vagabonding: An Uncommon Guide to the Art of Long-Term World Travel', '£36.94']
['Book A', 'Under the Tuscan Sun', '£37.33']
['Book B', 'Sharp Objects', '£47.82']
['Book B', 'In a Dark, Dark Wood', '£19.63']
['Book B', 'The Past Never Ends', '£56.50']
['Book B', 'A Murder in Time', '£16.64']
['Book B', 'The Murder of Roger Ackroyd (Hercule Poirot #4)', '£44.10']
PS C:\Users\aravi\OneDrive\Desktop\CA1 Program\Q4>

```

Figure 5.1.1 Scrape Output

6. Conclusion

This CA really helped me to get a deeper understanding of both C# and Python. Working with C# significantly improved my problem-solving skills because the assignment needed thorough requirement analysis and detailed implementation of each task. This process also enhanced my ability to understand and interpret project requirements effectively. While working with Python, I furthered the practical knowledge of database concepts and integration.

7. References

Amos, D. (2024) 'A practical introduction to web scraping in Python'. Available at: <https://realpython.com/python-web-scraping-practical-introduction/>.

Fullstack Dev (2020) '*C# From scratch: Phonebook console application*'. Available at: <https://www.youtube.com/watch?v=D9xNawxQoTM>.

Manwani, H. (2021) '*How to create and use the Dictionary in C#*'. Available at: <https://www.loginradius.com/blog/engineering/how-to-create-and-use-dictionary-csharp>.

'3 Best ways to get a file extension in Python (With Examples)'. Available at: <https://www.index.dev/blog/get-file-extension-python>. (no date)

'Travel | Books to Scrape – Sandbox'. Available at: https://books.toscrape.com/catalogue/category/books/travel_2/index.html. (no date)