

Coursework-2: OOP in C++

Task_3



Submitted by

Name: Ajaj Ahmed

Student ID: 24000864

Cybersecurity and Digital Forensics

Kathmandu, Nepal

April 10, 2025

Task 3: Basics of File Handling

Write a program that reads the titles of 10 books (use an array of 150 characters) and writes them in a binary file selected by the user. The program should read a title and display a message to indicate if it is contained in the file or not.

Create a program that:

1. Reads student records (roll, name, marks) from a text file
2. Throws an exception if marks are not between 0 and 100
3. Allows adding new records with proper validation
4. Saves modified records back to file

```
5. #include <iostream>
6.
7. #include <fstream>
8.
9. #include <vector>
10.
11. #include <string>
12.
13. #include <stdexcept>
14.
15. using namespace std;
16.
17. class BookManager
18.
19. {
20.
21. public:
22.
23.     void addBooks()
24.
25.     {
26.
27.         ofstream out("books.dat", ios::binary | ios::app);
28.
29.         if (!out)
30.
31.             {
32.
33.                 cout << "Error opening books.dat for writing." << endl;
34.                 return;
35.             }
36.
```

```
37.         for (int i = 0; i < 10; ++i)
38.
39.         {
40.             string title;
41.
42.             cout << "Enter Title for book: " << i + 1 << ": ";
43.
44.             cin.ignore();
45.
46.             getline(cin, title);
47.
48.             out.write(title.c_str(), title.size() + 1);
49.         }
50.
51.         out.close();
52.
53.         cout << "-----" <<
    endl;
54.
55.         cout << "\n congrats!!! Books Details are saved successfully!" <<
    endl;
56.
57.         cout << "-----" <<
    endl;
58.     }
59.
60.     void searchBook()
61.
62.     {
63.
64.         string searchTitle;
65.
66.         cout << "Enter book title you want to search: ";
67.
68.         cin.ignore();
69.
70.         getline(cin, searchTitle);
71.
72.         ifstream in("books.dat", ios::binary);
73.         if (!in)
74.
75.         {
76.
77.             cout << "Error opening books.dat for reading." << endl;
78.
```

```
79.         return;
80.     }
81.
82.     string title;
83.
84.     bool found = false;
85.
86.     while (getline(in, title, '\0'))
87.     {
88.         if (title == searchTitle)
89.         {
90.             found = true;
91.             break;
92.         }
93.     }
94.
95.     in.close();
96.
97.     cout << (found ? "\n Book you searched is found !" : "\n Book You
98.     searched doesn't found.") << endl;
99.
100.    cout << "-----" << endl;
101. }
102.
103.
104. void showAllBooks()
105.
106. {
107.     ifstream in("books.dat", ios::binary);
108.
109.     if (!in)
110.     {
111.         cout << "Error opening books.dat for reading." <<
112.         endl;
113.         return;
114.     }
115.
116.     cout << "-----";
117.
118.     cout << "\n ----Book EXISTS in Library:---- \n";
119.
120.     cout << "-----" <<
121.         endl;
```

```
121.         string title;
122.
123.         int count = 0;
124.
125.         while (getline(in, title, '\0'))
126.
127.         {
128.
129.             cout << ++count << ". " << title << endl;
130.
131.         }
132.
133.         if (count == 0)
134.
135.             cout << "No Matching books found." << endl;
136.
137.             in.close();
138.             cout << "-----" << endl;
139.         }
140.     };
141.
142. class StudentManager
143.
144. {
145.
146. public:
147.     struct Student
148.
149.     {
150.         int roll;
151.
152.         string name;
153.
154.         float marks;
155.     };
156.
157. private:
158.
159.     vector<Student> students;
160.
161. public:
162.
163.     void loadStudents()
164.
165.     {
```

```
166.         students.clear();
167.
168.         ifstream in("students.txt");
169.
170.         Student s;
171.
172.         while (in >> s.roll >> ws && getline(in, s.name) && in >> s.marks)
173.
174.             {
175.
176.                 students.push_back(s);
177.             }
178.
179.             in.close();
180.         }
181.
182.     void addStudent()
183.
184.     {
185.         Student s;
186.
187.         cout << "Enter Student roll number: ";
188.
189.         cin >> s.roll;
190.         cin.ignore();
191.
192.         cout << "Enter student name: ";
193.         getline(cin, s.name);
194.
195.         cout << "Enter Student marks (0-100): ";
196.         cin >> s.marks;
197.
198.         if (s.marks < 0 || s.marks > 100)
199.
200.             {
201.
202.                 throw out_of_range(" Marks must be between 0 and 100.");
203.
204.             }
205.
206.         students.push_back(s);
207.
208.         saveStudents();
209.
210.         cout << "\n wow!! Student record saved successfully!" << endl;
```

```
211.         cout << "-----" << endl;
212.     }
213.
214. void saveStudents()
215.
216. {
217.     ofstream out("students.txt", ios::app);
218.
219.     for (const auto& s : students)
220.     {
221.         out << s.roll << endl
222.
223.         << s.name << endl
224.
225.         << s.marks << endl;
226.     }
227.
228.     out.close();
229. }
230.
231. void displayStudents()
232.
233. {
234.     if (students.empty())
235.
236.     {
237.
238.         cout << "!!! student Student Details found !!!." << endl;
239.
240.     }
241.     else
242.     {
243.
244.         cout << "\n !!! Student Records !!! :\n";
245.
246.         cout << "-----" << endl ;
247.
248.         for (const auto& s : students)
249.
250.         {
251.             cout << "Roll No: " << s.roll
252.
253.             << ", Name: " << s.name
254.
```

```
255.                                     << ", Marks: " << s.marks << endl;
256.                                 }
257.                             }
258.
259.                         cout << "-----" << endl;
260.                     }
261.     };
262.
263. int main()
264.
265. {
266.     BookManager bookManager;
267.
268.     StudentManager studentManager;
269.
270.     int choice;
271.     do
272.
273.     {
274.         cout << "\n===== !! LIBRARY MENU !! ====="
275.             << endl;
276.
277.         cout << "1. Add Book Titles" << endl;
278.
279.         cout << "2. Search for a Book" << endl;
280.
281.         cout << "3. Add New Student" << endl;
282.
283.         cout << "4. Show Students Details " << endl;
284.
285.         cout << "5. Show Existing Books" << endl;
286.
287.         cout << "0. Exit Program " << endl;
288.
289.         cout << "Enter your choice: ";
290.
291.         cin >> choice;
292.
293.         cout << "-----" << endl;
294.
295.         switch (choice)
296.         {
297.             case 1:
298.
299.                 bookManager.addBooks();
```

```
300.             break;
301.
302.         case 2:
303.
304.             bookManager.searchBook();
305.             break;
306.
307.         case 3:
308.
309.             try
310.
311.             {
312.
313.                 studentManager.loadStudents();
314.                 studentManager.addStudent();
315.
316.             }
317.
318.         catch (const exception& e)
319.
320.             {
321.                 cout << "\n Exception occurred: " << e.what() <<
322.                 endl;
323.
324.                 cout << "-----" <<
325.                 endl;
326.
327.             break;
328.         case 4:
329.
330.             studentManager.loadStudents();
331.             studentManager.displayStudents();
332.             break;
333.
334.         case 5:
335.
336.             bookManager.showAllBooks();
337.             break;
338.
339.         case 0:
340.
341.             cout << "-----";
342.             cout << "\n !! ThAnK YoU __ Visit Again !!"<<endl;
```

```

343.             cout << "-----" << endl;
344.             break;
345.
346.         default:
347.
348.             cout << " ! Invalid choice. Please try again." << endl;
349.             cout << "-----" << endl;
350.         }
351.
352.     }
353.
354.     while (choice != 0);
355.
356.     return 0;
357. }
358.

```

===== !! LIBRARY MENU !! =====

1. Add Book Titles
2. Search for a Book
3. Add New Student
4. Show Students Details
5. Show Existing Books
0. Exit Program

Enter your choice: 1

Enter Title for book: 1: Business Studies
 Enter Title for book: 2: C++ fundamentals
 Enter Title for book: 3: Python For Beginners
 Enter Title for book: 4: JAVASCRIPT/HTML/CSS
 Enter Title for book: 5: FUNDAMENTALS OF DATA SCIENCE
 Enter Title for book: 6: Digital Evidence
 Enter Title for book: 7: C++ Data Structures
 Enter Title for book: 8: Ethical Hacking Fundamentals
 Enter Title for book: 9: Cyber Laws
 Enter Title for book: 10: Machine Learning

congrats!!! Books Details are saved successfully!

===== !! LIBRARY MENU !! =====

1. Add Book Titles
2. Search for a Book
3. Add New Student
4. Show Students Details
5. Show Existing Books
0. Exit Program

Enter your choice: 3

Enter Student roll number: 100231
 Enter student name: Ram Kumar Sharma
 Enter Student marks (0-100): 78

wow!! Student record saved successfully!

```
C:\ "X:\2nd Year\Assignments\c+" X + | v  
===== !! LIBRARY MENU !! =====  
1. Add Book Titles  
2. Search for a Book  
3. Add New Student  
4. Show Students Details  
5. Show Existing Books  
0. Exit Program  
Enter your choice: 2
```

```
-----  
Enter book title you want to search: Cyber Laws  
  
Book you searched is found !
```

```
===== !! LIBRARY MENU !! =====  
1. Add Book Titles  
2. Search for a Book  
3. Add New Student  
4. Show Students Details  
5. Show Existing Books  
0. Exit Program  
Enter your choice: 4
```

```
-----  
!!! Student Records !!! :  
-----  
Roll No: 100231, Name: Ram Kumar Sharma, Marks: 78  
-----
```

- 3. Add New Student
- 4. Show Students Details
- 5. Show Existing Books
- 0. Exit Program

Enter your choice: 5

-----Book EXISTS in Library:-----

- 1. BBA Books
 - 2. ++
 - 3. ava Script
 - 4. atabases
 - 5. ython
 - 6. igital Evidence
 - 7. thical hacking
 - 8. ata Science
 - 9. achine Learning
 - 10. etworking Fundamentals
 - 11. Business Studies
 - 12. C++ fundaments
 - 13. Python For Beginners
 - 14. JAVASCRIPT/HTML/CSS
 - 15. FUndamentals of Data Science
 - 16. Digital Evidence
 - 17. C++ Data Structures
 - 18. Ethical Hacking Fundamentals
 - 19. Cyber Laws
 - 20. Machine Learning
-

===== !! LIBRARY MENU !! =====

1. Add Book Titles
2. Search for a Book
3. Add New Student
4. Show Students Details
5. Show Existing Books
0. Exit Program

Enter your choice: 0

!! ThAnK YoU __ Visit Again !!

Process returned 0 (0x0) execution time : 399.203 s

Press any key to continue.

|