Library Organization System

Lizbeth Sanchez

Introduction to Computer Science – CRN 80598

Dr. Deepa Devasenapathy

Content:

* The content within this project had many topics we have went through in class. For example, the def function, class function, print function, control loops, exceptions, list, and other data types are used as well. I tried to find a project with a lot of the contents talked about in class to a level where I was able to also add some things to the code.

Abstract:

* In the project I was able to add a fifth component to the code and this means adding a different section to the code where the person using the code can view the books “borrowed” from the system. Since this project is a library organizer, I added my own choice of books and organized the order to which you can see the list of choices.

Goal of Project:

* My goal for this project was to be able to find a way to add a new organization material/section to the original code and be able to combine my interest of reading and books with the new material I have learned in this class. I also wanted to combine this interest for other people that may also have the same hobbies and enjoy Python and literature.

Software Requirements:

* For this project, the software I used is PyCharm, but you can run this code with any system that has Python. I used my personal computer which is a Dell Laptop, and it uses Windows 11, so anything similar will also work.

Why I Chose This Project:

* I chose this project concept because reading is one of my favorite hobbies and wanted to combine what I have learned in class with my other interests. Also, I wanted to create a project that I know I could understand and was within my abilities and knowledge.

Existing technique and my improvisations to the existing concept:

* The existing code has four main choices in which the user could choose from to start their process in having to “borrow” the book. These four concepts are viewing the books available, borrowing a book, returning a book back to the system, and then exiting the code. What I just wanted to add was the choice to see the book in the user’s possession. So, that is the “view your books” choice on the flow chart. With this the added code was not so different from the previous steps. I just added another of the same function and added the new code where viewing your books was possible after the available book was chosen. So, that was it, I decided this was a project I could use my knowledge and was within my capabilities to complete.

Project Concept and procedure in detailed steps:

* Here I will add the flow chart I made for the project.
* When you run the code, you are given the list of choices you can make, which are the five in the flow chart. Then you can choose which option you want, if you want to see the books available type 1, and then the books “free” to borrow will show up. Then if you want to borrow a book you type 2 for that choice and type out the book title and then the code will print “‘Book Title’ marked as borrowed by ‘Student Name.’” Then the third choice is the “view your books” choice, but it will only show a book title if you have already borrowed on. Then when you want to return the book, you type 4, and then you enter the name of the book you are returning and then it will go right back into the system. The last choice is the exit and it will happen when you type 5 for that choice.

A diagram of a book

Description automatically generated

Screen Shots of Code:

A computer screen shot of a program

Description automatically generated

1. A computer screen shot of a program code

   Description automatically generated
2. A computer screen shot of a program

   Description automatically generated
3. A computer screen shot of text

   Description automatically generated
4. A computer screen shot of text

   Description automatically generated
5. A screen shot of a computer program

   Description automatically generated

Output of Code:

1. A screenshot of a computer program

   Description automatically generated
2. A screenshot of a computer program

   Description automatically generated
3. A computer screen shot of a program

   Description automatically generated
4. A computer screen shot of a menu

   Description automatically generated

Conclusion:

What do you conclude from the work you have done and is there any technique that you would consider to improvise your work?

* My answer to the question above is that I conclude from this project that I have learned a lot in this class and to the best of my abilities tried to show that with this project, but I know that there is so much more on which I can practice. A technique I would consider that could improve my work is just practicing my knowledge on Python and just doing any kind of activities to use everything I learned in class to create more complex and difficult projects that can challenge me and help me gain more knowledge and experience. Another, thing that I think could help me improve my work is understanding more about each topic we went over and continue to really understand what every function and data type does. Once I can do this I can use all these functions correctly and really put them to use together to create new complex projects.

Appendix:

* Code of Project:

class Library:  
  
 def \_\_init\_\_(self, books):  
 self.books = books  
  
 def show\_avail\_books(self):  
 print('Our Library Can Offer You The Following Books:')  
 print('================================================')  
 for book, borrower in self.books.items():  
 if borrower == 'Free':  
 print(book)  
  
 def lend\_book(self, requested\_book, name):  
 if self.books[requested\_book] == 'Free':  
 print(  
 f'{requested\_book} has been marked'  
 f' as \'Borrowed\' by: {name}')  
 self.books[requested\_book] = name  
 return True  
 else:  
 print(  
 f'Sorry, the {requested\_book} is currently'  
 f' on loan to: {self.books[requested\_book]}')  
 return False  
  
 def return\_book(self, returned\_book):  
 self.books[returned\_book] = 'Free'  
 print(f'Thanks for returning {returned\_book}')  
  
  
class Student:  
 def \_\_init\_\_(self, name, library):  
 self.name = name  
 self.books = []  
 self.library = library  
  
 def view\_borrowed(self):  
 if not self.books:  
 print('You haven\'t borrowed any books')  
 else:  
 for book in self.books:  
 print(book)  
  
 def request\_book(self):  
 book = input(  
 'Enter the name of the book you\'d like to borrow >> ')  
 if self.library.lend\_book(book, self.name):  
 self.books.append(book)  
  
 def return\_book(self):  
 book = input(  
 'Enter the name of the book you\'d like to return >> ')  
 if book in self.books:  
 self.library.return\_book(book)  
 else:  
 print('You haven\'t borrowed that book, try another...')  
  
  
def create\_lib():  
 books = {  
 '1989': 'Free',  
 'The Hunger Games': 'Free',  
 'Circe': 'Free',  
 'The Great Gatsby': 'Free'  
 }  
 library = Library(books)  
 student\_example = Student('Lizbeth S.', library)  
  
 while True:  
 print('''  
 ==========LIBRARY MENU===========  
 1. Display Available Books  
 2. Borrow a Book  
 3. View Your Books   
 4. Return a Book   
 5. Exit'''  
 )  
  
 choice = int(input('Enter Choice: '))  
 if choice == 1:  
 print()  
 library.show\_avail\_books()  
 elif choice == 2:  
 print()  
 student\_example.request\_book()  
 elif choice == 3:  
 print()  
 student\_example.view\_borrowed()  
 elif choice == 4:  
 print()  
 student\_example.return\_book()  
 elif choice == 5:  
 print('Goodbye')  
 exit()  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 create\_lib()

* References:
* <https://hackr.io/blog/python-projects>
* [(356) How to Create Your Own Library System in Python - YouTube](https://www.youtube.com/watch?v=-dUOCOR3YZk)

More Outputs I generated are in the PowerPoint.