

SimpleLibrary (2.0)

Kcits970

December 20, 2021

Sections

1. Description	3
2. Features	4
Account Setup	
Item Addition	
Item Checkout	
Status Saving	
3. Program Execution	7
Windows Terminal	
ConEmu	

Section 1. Description

The purpose of the program 'SimpleLibrary' is to simulate a small public library. It provides a menu-driven interface to create a library account, add/remove books, and borrow/return books.

The program is written in C language, and compiled using Microsoft Visual C++ 14.2. Attempts to compile the given source code with a different compiler may result in unexpected or undefined behavior.

Section 2. Features

As mentioned in the description, 'SimpleLibrary' provides a menu driven interface to simulate the works of a small public library. Its features can be divided into 4 main parts, which are account setup, item addition, item checkout, and status saving.

Account Setup

At program launch, the user is prompted with the following output.

```
>>LIBRARY SYSTEM: VERSION 2
```

```
    <ACCOUNT SETUP>
REGISTER ----- A
LOGIN  ----- B
>>
```

From here, the user can either register a new account, or log in if there's already an existing one. For the account registration to succeed, the ID must be between 4 and 16 characters, and the password must be between 8 and 16 characters. Once the account registration succeeds, a new file "account.dat" will be created. Note that the user cannot create multiple accounts. The account file can only contain the data of one account, which means that each registration will overwrite the existing account.

Item Addition

After the login succeeds, the user is then prompted with the following output.

```
    <LOGIN PROMPT>
ID: Kcits970
PASSWORD: *****
LOGIN SUCCESSFUL
```

```
    <MAIN OPTIONS>
VIEW AVAILABLE ITEMS ----- A
ADD ----- B
DELETE ----- C
VIEW MY ITEMS ----- D
CHECKOUT ----- E
RETURN ----- F
```

```
EXIT ----- Q
>>
```

The user can now start selecting the main options. For newly registered accounts, 4 books are automatically added by default. To add a custom item, select the `ADD` option. The user will be directed to an addition prompt. The title, author, and the number of pages will each be sequentially prompted for user input, and then item is added to the list.

```
<MAIN OPTIONS>
VIEW AVAILABLE ITEMS ----- A
ADD ----- B
DELETE ----- C
VIEW MY ITEMS ----- D
CHECKOUT ----- E
RETURN ----- F
EXIT ----- Q
>>B
```

```
<ITEM ADDITION PROMPT>
TITLE: Refactoring
AUTHOR: Martin Fowler
PAGES: 418
ITEM SUCCESSFULLY ADDED
```

It's important to note that inputs longer than 1023 characters are automatically truncated. This is to prevent the user from entering extremely long inputs, which may result in buffer overflow.

Item Checkout

Apart from adding new items, the user can also take existing items. This process is called a 'checkout'. In another words, the user can take items from the available list, and temporarily keep it. The taken items can be seen by selecting the `VIEW MY ITEMS` option.

```
<MAIN OPTIONS>
VIEW AVAILABLE ITEMS ----- A
ADD ----- B
DELETE ----- C
VIEW MY ITEMS ----- D
CHECKOUT ----- E
RETURN ----- F
EXIT ----- Q
>>E
```

```
<ITEM CHECKOUT PROMPT>
```

INDEX	TITLE	AUTHOR	PAGES
1	Head First C	David Griffiths	633
2	Head First Java	Kathy Sierra	722
3	Prime Obsession	John Derbyshire	447
4	Fermat's Last...	Simon Singh	368
5	Refactoring	Martin Fowler	418

INDEX: 3
ITEM SUCCESSFULLY RESERVED

<MAIN OPTIONS>

VIEW AVAILABLE ITEMS ----- A
ADD ----- B
DELETE ----- C
VIEW MY ITEMS ----- D
CHECKOUT ----- E
RETURN ----- F
EXIT ----- Q

>>D

<LIST OF MY ITEMS>

INDEX	TITLE	AUTHOR	PAGES
1	Prime Obsession	John Derbyshire	447

>>

To return the taken item, simply select the RETURN option, and follow the same process.

Status Saving

If the user selects the EXIT option, a short message will be prompted.

<MAIN OPTIONS>

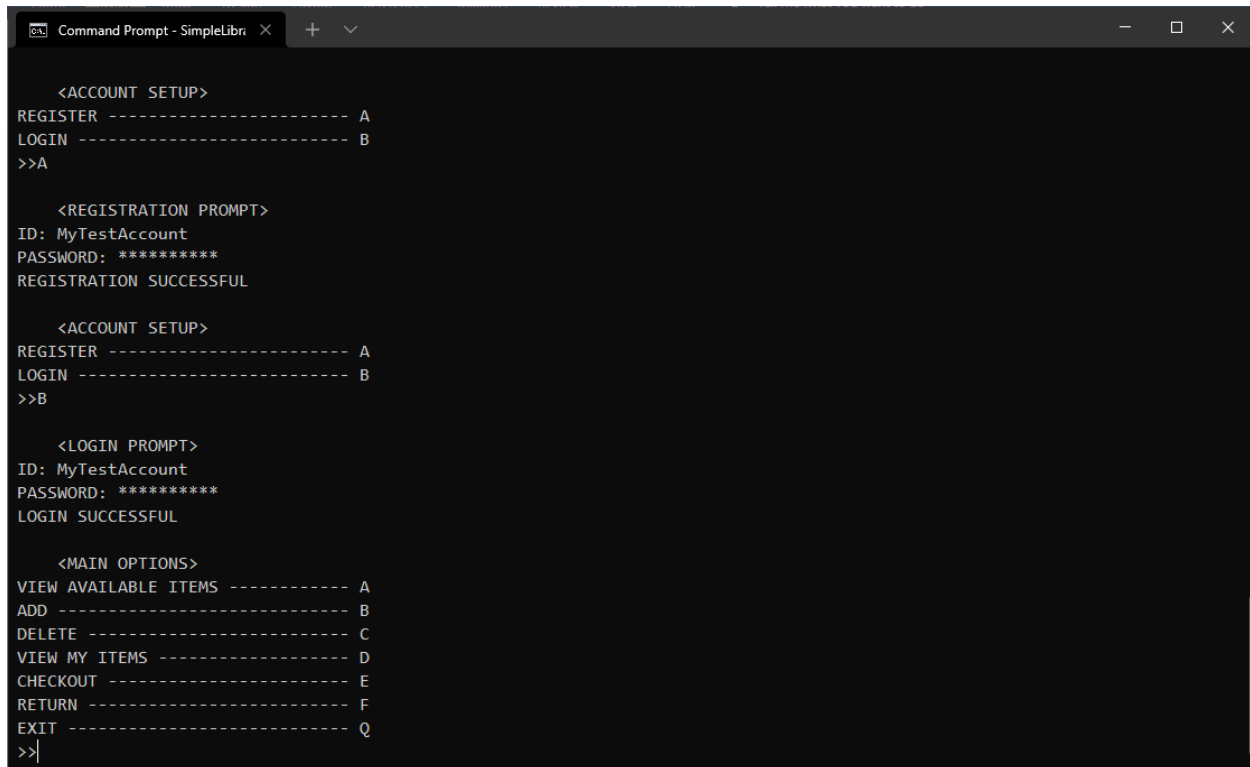
VIEW AVAILABLE ITEMS ----- A
ADD ----- B
DELETE ----- C
VIEW MY ITEMS ----- D
CHECKOUT ----- E
RETURN ----- F
EXIT ----- Q

>>Q
SAVE BEFORE EXIT? (Y/N):

If 'Y' is entered, 2 new files ("*publicitems.dat*" and "*myitems.dat*") will be created. These 2 files contain the data of publicly available items and the user's items, and are used to load the status of the last session during the next launch. If the files are corrupted or are not present, then the default status will be loaded.

Section 3. Program Execution

Execution in Windows Terminal:



```
Command Prompt - SimpleLibr. X + -
<ACCOUNT SETUP>
REGISTER ----- A
LOGIN ----- B
>>A

<REGISTRATION PROMPT>
ID: MyTestAccount
PASSWORD: *****
REGISTRATION SUCCESSFUL

<ACCOUNT SETUP>
REGISTER ----- A
LOGIN ----- B
>>B

<LOGIN PROMPT>
ID: MyTestAccount
PASSWORD: *****
LOGIN SUCCESSFUL

<MAIN OPTIONS>
VIEW AVAILABLE ITEMS ----- A
ADD ----- B
DELETE ----- C
VIEW MY ITEMS ----- D
CHECKOUT ----- E
RETURN ----- F
EXIT ----- Q
>>|
```

Execution in ConEmu:

```
cmd - SimpleLibrary.exe
<1> cmd - SimpleLib...
Search
EXIT ----- Q
>>F

<ITEM RETURN PROMPT>
INDEX | TITLE | AUTHOR | PAGES
1 | Prime Obsession | John Derbyshire | 447
INDEX: 1
ITEM SUCCESSFULLY RETURNED

<MAIN OPTIONS>
VIEW AVAILABLE ITEMS ----- A
ADD ----- B
DELETE ----- C
VIEW MY ITEMS ----- D
CHECKOUT ----- E
RETURN ----- F
EXIT ----- Q
>>A

<LIST OF AVAILABLE ITEMS>
INDEX | TITLE | AUTHOR | PAGES
1 | Head First C | David Griffiths | 633
2 | Head First Java | Kathy Sierra | 722
3 | Fermat's Last... | Simon Singh | 368
4 | Refactoring | Martin Fowler | 418
5 | Prime Obsession | John Derbyshire | 447
>>|
```

SimpleLibrary.exe[32]:15084 α 210912[64] 1/1 [+] NUM InpGrp W PRI: 84x27 (3,70) 25V