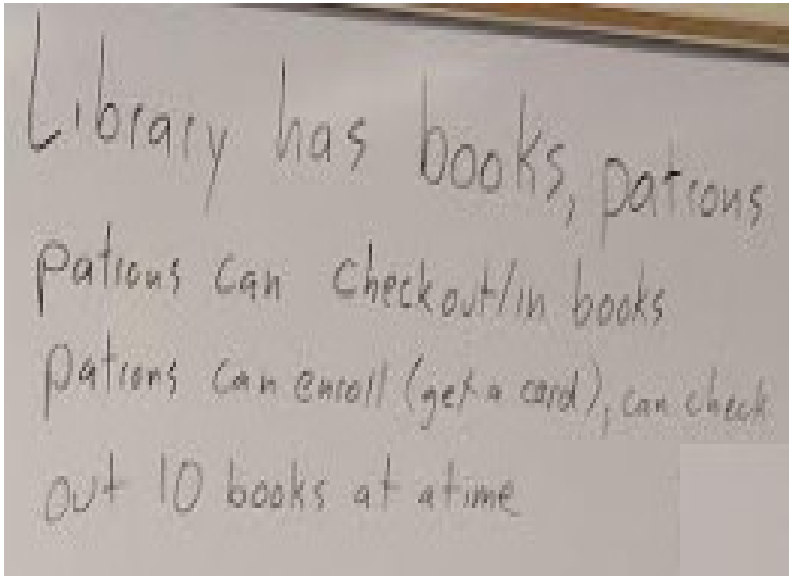
CPSC 327 – Model a library

A programming Assignment

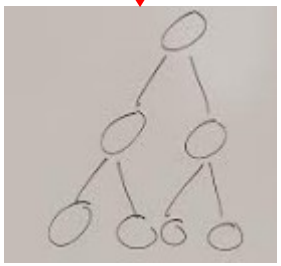
- First, a universal truth
  - *Many days of programming can save you hours of planning*

- *Model a library. It has books and patrons. Patrons can register for a card and check out books. Patrons can only checkout 10 books at a time.*

Problem

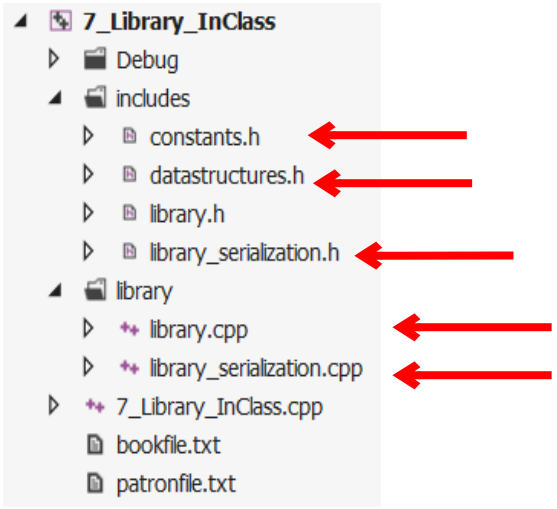


Top down design!!



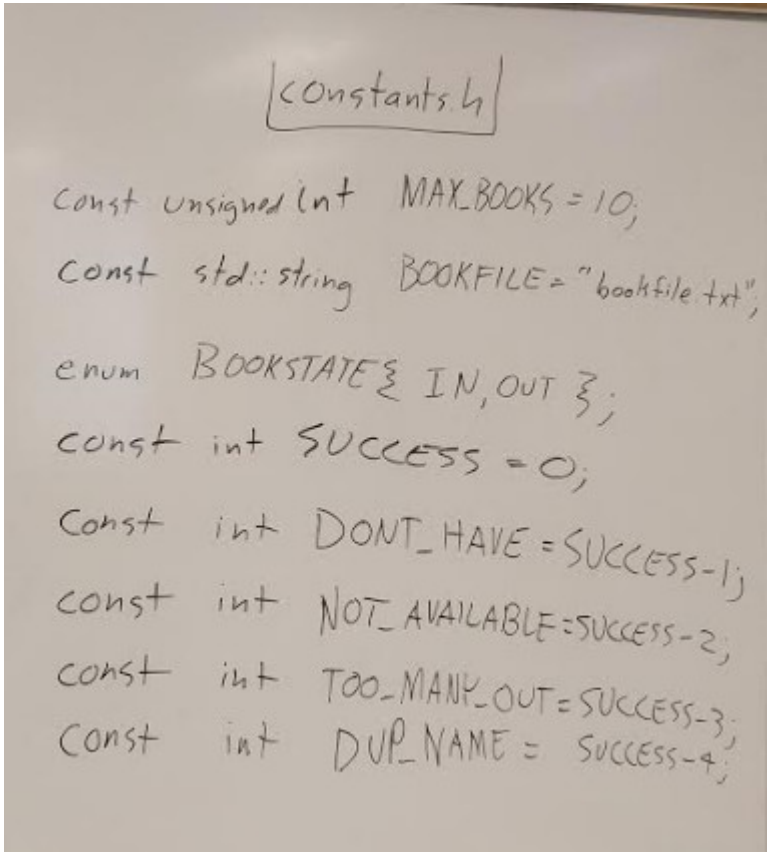
- Layout functions needed
- Generate constants file along the way
- put off defining datastructures until the last minute, hide them so user does not know what you are using (allows you to change them without affecting user)
- Make API easy to use and as simple as possible
- Hide what you can in cpp files (datastructures, containers, initialization).
- Anything you can automatically do for the user ... do

Directory structure



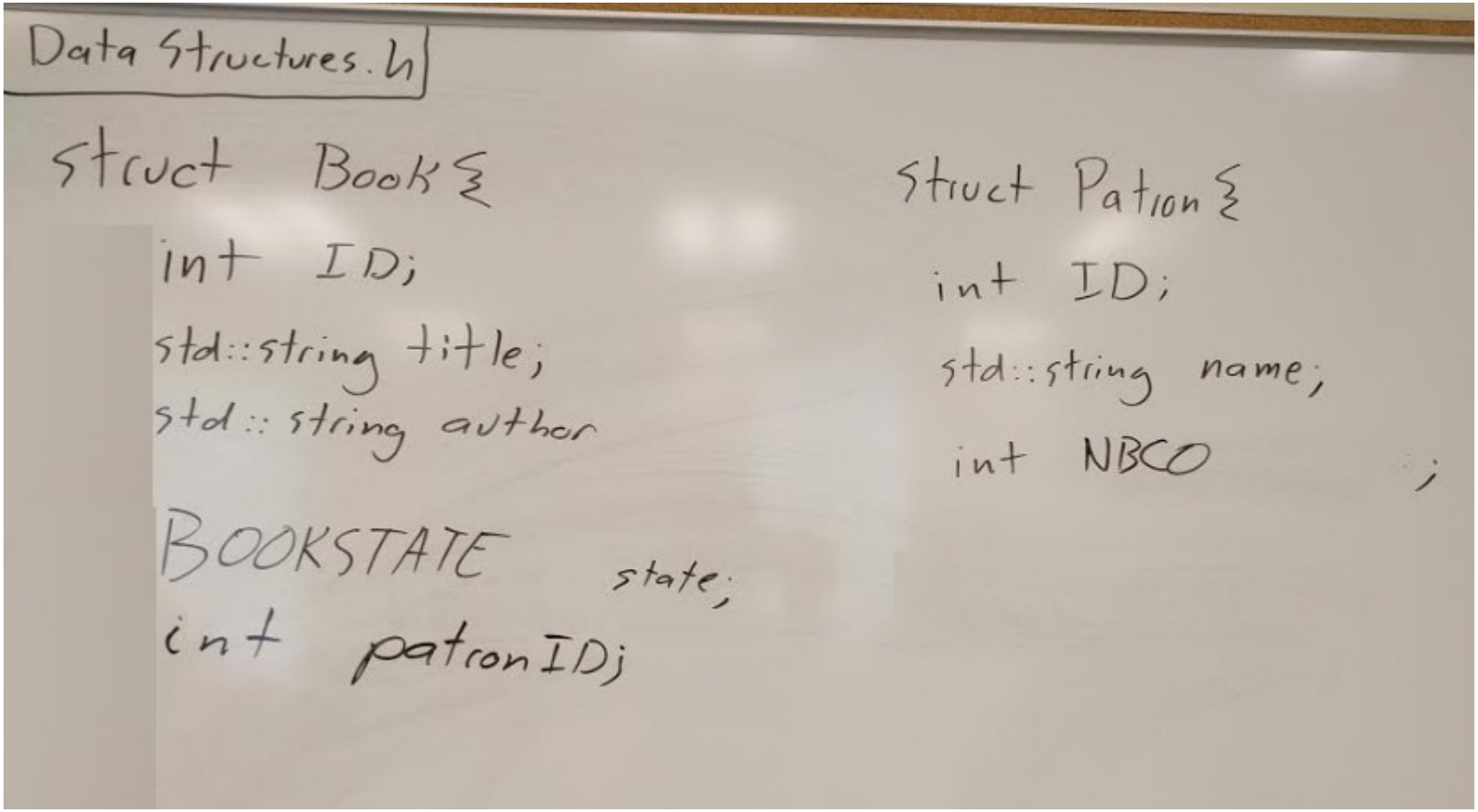
- Several files so lets put them in different directories.
- Helps organize code
- Essential for large projects

constants.h



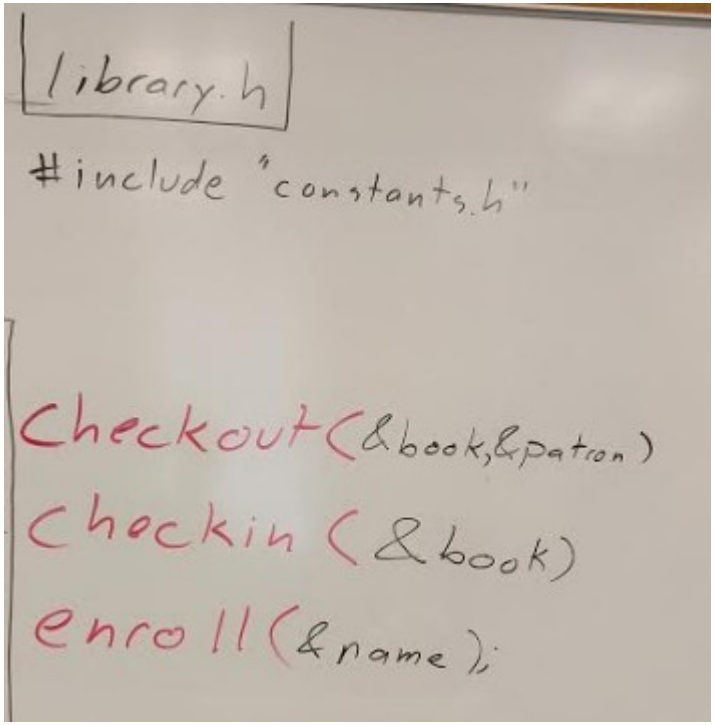
- Lets put this in its own folder, called constants
- Don’t forget include guards (#pragma once is the easiest)

datastructures.h



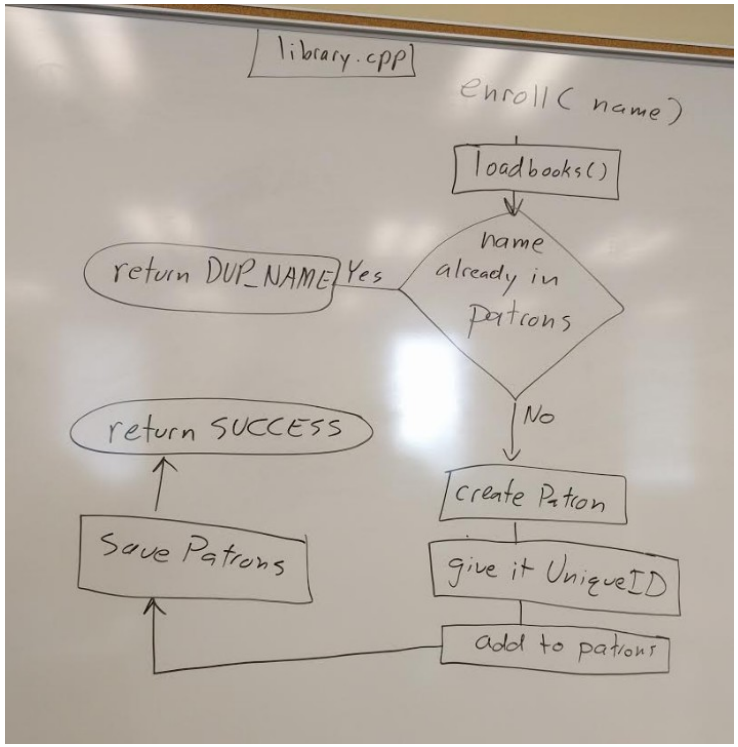
- Lets put this in includes folder

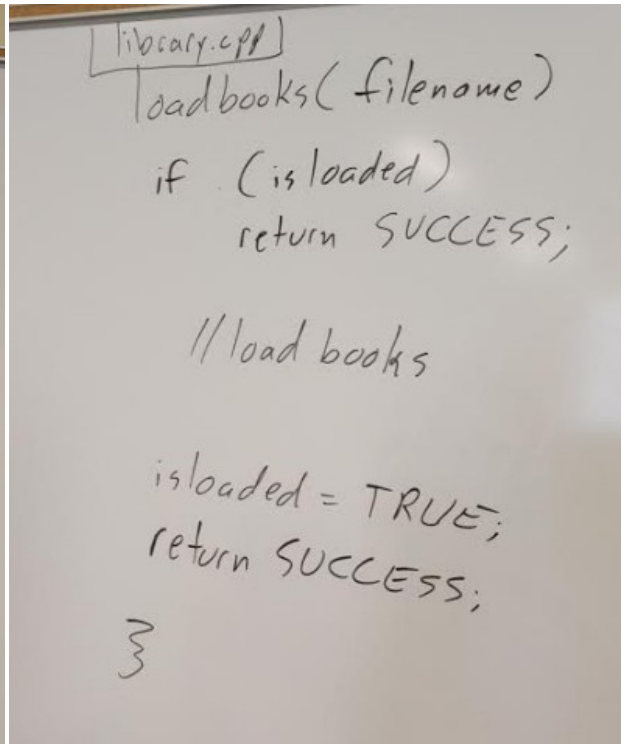
library.h



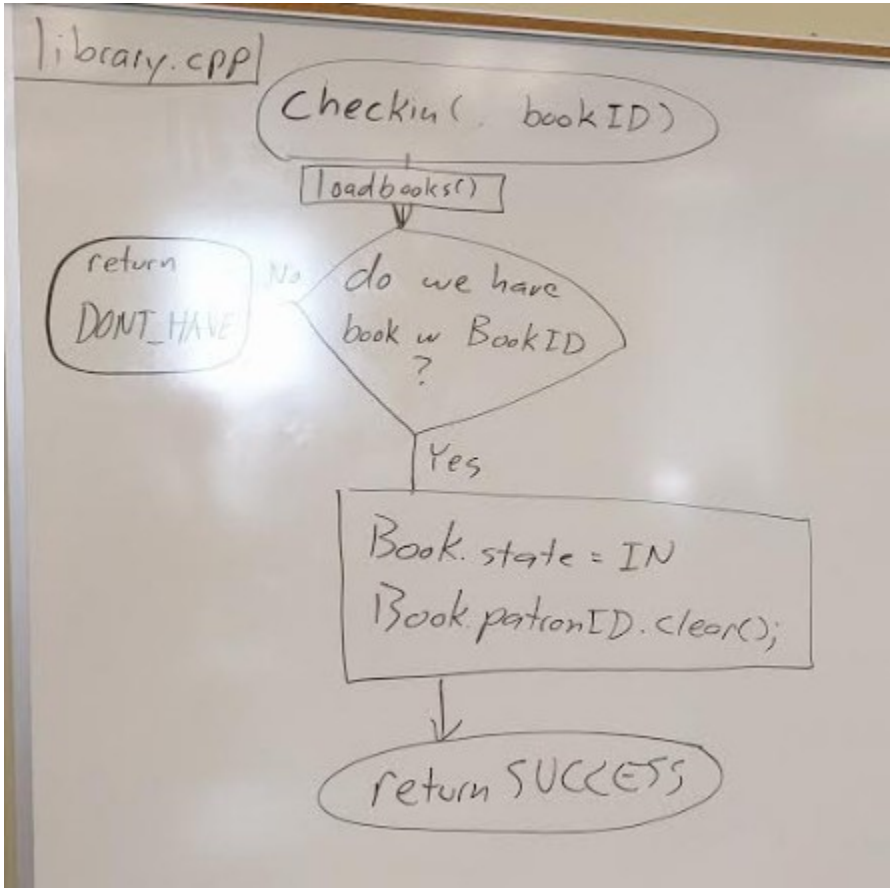
- Lets put this in includes folder

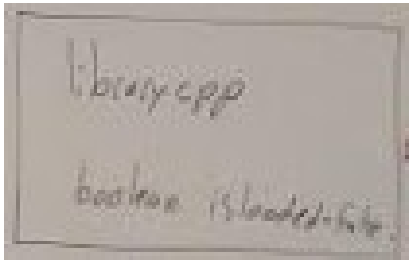
library.cpp


library.cpp
enroll(name)
loadbooks()
name already in patrons
Yes
return DUP_NAME
No
create Patron
give it UniqueID
add to patrons
Save Patrons
return SUCCESS

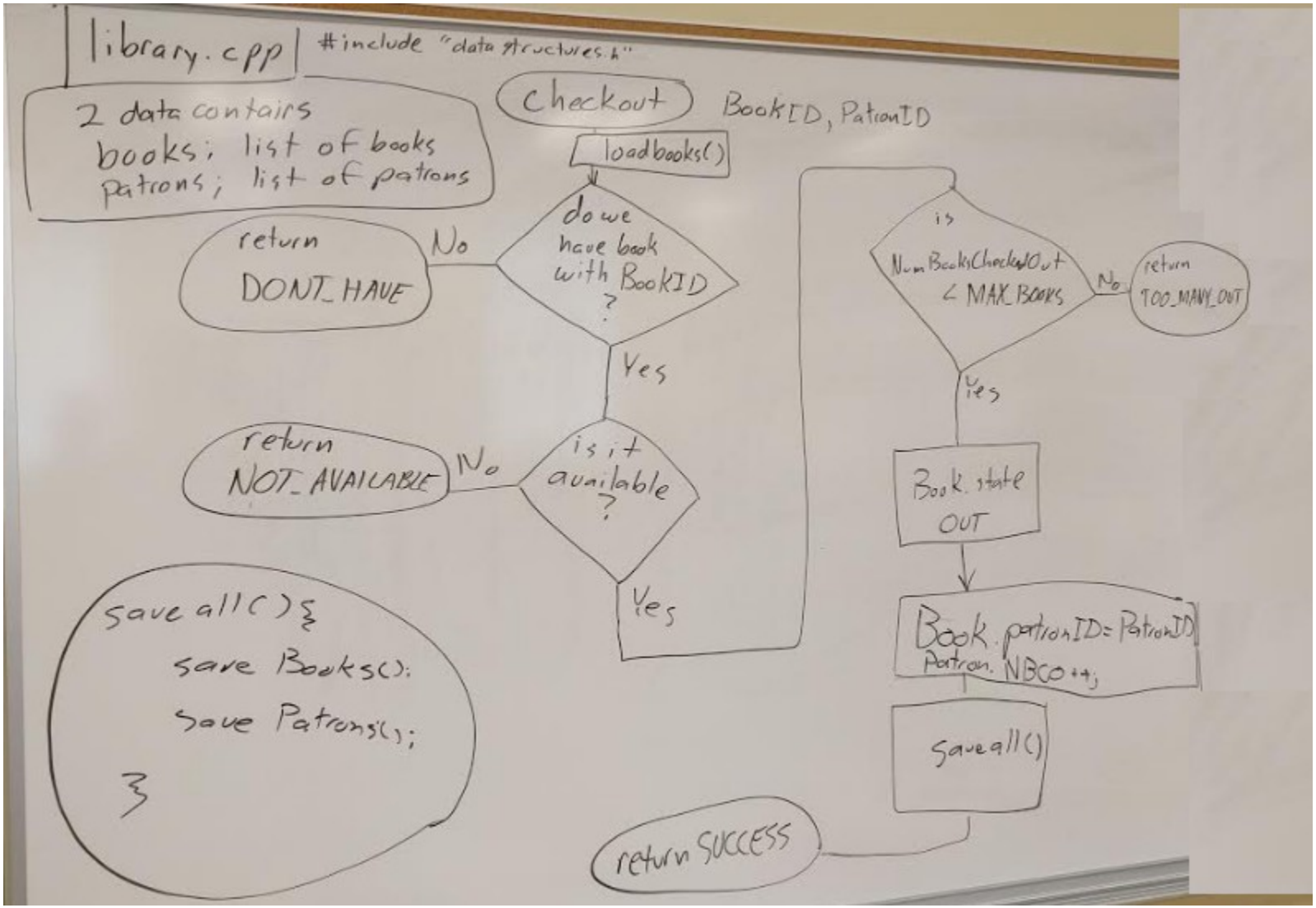

library.cpp
loadbooks(filename)
if (isloaded)
return SUCCESS;
// load books
isloaded = TRUE;
return SUCCESS;
}

library.cpp


library.cpp
checkin(bookID)
loadbooks()
do we have book w BookID ?
No
return DONT_HAVE
Yes
Book.state = IN
Book.patronID.clear();
return SUCCESS


library.cpp

library.cpp


library.cpp
#include "data structures.h"
2 data contains
books; list of books
patrons; list of patrons
Checkout
BookID, PatronID
loadbooks()
do we have book with BookID ?
No
return DONT_HAVE
Yes
is it available ?
No
return NOT_AVAILABLE
Yes
is NumBooksCheckedOut < MAX_BOOKS
No
return TOO_MANY_OUT
Yes
Book.state OUT
Book.patronID=PatronID
Patron.NBCO++;
saveall()
return SUCCESS
saveall(){
save Books();
save Patrons();
}