


```

* if the specified Patron is not in the Library's members, return "patron not found"
* if the specified LibraryItem is not in the Library's holdings, return "item not found"
* if the specified LibraryItem is already checked out, return "item already checked out"
* if the specified LibraryItem is on hold by another Patron, return "item on hold by other patron"
* otherwise update the LibraryItem's checked_out_by, date_checked_out and location
* if the LibraryItem was on hold for this Patron, update requested_by
* update the Patron's checked_out_items
* return "check out successful"
* return_library_item
  * takes as its parameter a library item ID
  * if the specified LibraryItem is not in the Library's holdings, return "item not found"
  * if the LibraryItem is not checked out, return "item already in library"
  * update the Patron's checked_out_items
  * update the LibraryItem's location depending on whether another Patron has requested it (if so, it
should go on the hold shelf)
  * update the LibraryItem's checked_out_by
  * return "return successful"
* request_library_item
  * takes as parameters a patron ID and a library item ID, in that order
  * if the specified Patron is not in the Library's members, return "patron not found"
  * if the specified LibraryItem is not in the Library's holdings, return "item not found"
  * if the specified LibraryItem is already requested, return "item already on hold"
  * update the LibraryItem's requested_by
  * if the LibraryItem is on the shelf, update its location to on hold
  * return "request successful"
* pay_fine
  * takes as parameters a Patron ID and the amount (in dollars) being paid (in that order)
  * if the specified Patron is not in the Library's members, return "patron not found"
  * use amend_fine to update the Patron's fine; return "payment successful"
* increment_current_date
  * takes no parameters
  * increment current date
  * increase each Patron's fines by 10 cents for each overdue LibraryItem they have checked out (by
calling amend_fine)

```

Note - a LibraryItem can be on request without its location being the hold shelf (if another Patron has it checked out);

One limited example of how your classes might be used is:

```
b1 = Book("345", "Phantom Tollbooth", "Juster")
a1 = Album("456", "...And His Orchestra", "The Fastbacks")
m1 = Movie("567", "Laputa", "Miyazaki")
print(b1.get_author())
print(a1.get_artist())
print(m1.get_director())

p1 = Patron("abc", "Felicity")
p2 = Patron("bcd", "Waldo")

lib = Library()
lib.add_library_item(b1)
lib.add_library_item(a1)
lib.add_patron(p1)
lib.add_patron(p2)

lib.check_out_library_item("bcd", "456")
loc = a1.get_location()
lib.request_library_item("abc", "456")
for i in range(57):
    lib.increment_current_date()    # 57 days pass
p2_fine = p2.get_fine_amount()
lib.pay_fine("bcd", p2_fine)
```

```
    lib.return_library_item("456")  
...
```

You are responsible for testing all of the required functions to make sure they operate as specified.

Your file must be named: ****Library.py****

Just to think about: Since there are three possible locations for a `LibraryItem`, there are six hypothetical changes in location. Are all six possible according to these specifications?