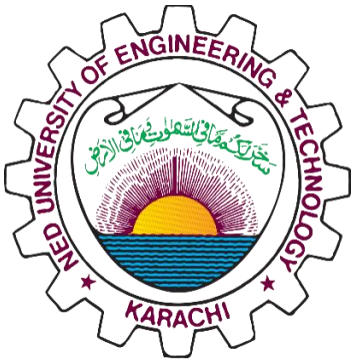
OEL



BATCH 2021-22

NAME	**Hunzala Mushtaq**
ROLL NO:	**CS - 20052**
CLASS/SEC:	**Section - C**
COURSE NAME	**Data Structure and Algorithm**
COURSE CODE	**CS-218**
SUBMITTED TO	**Miss Ibshar Ishrat, Sir Zaffar Qasim**

DEPARTMENT OF COMPUTER & INFO SYSTEM ENGINEERING

NED UNIVERSITY OF ENGINEERING & TECHNOLOGY

PROJECT DETAILS:

Tools and language used in project Library Management System are as follow

- Python 3
- SQL
- SQLite Database
- SQLite Browser

Concepts Covered:

- OOP based approach
- Binary Search
- Merge Sort
- Arrays
- CRUD in Database

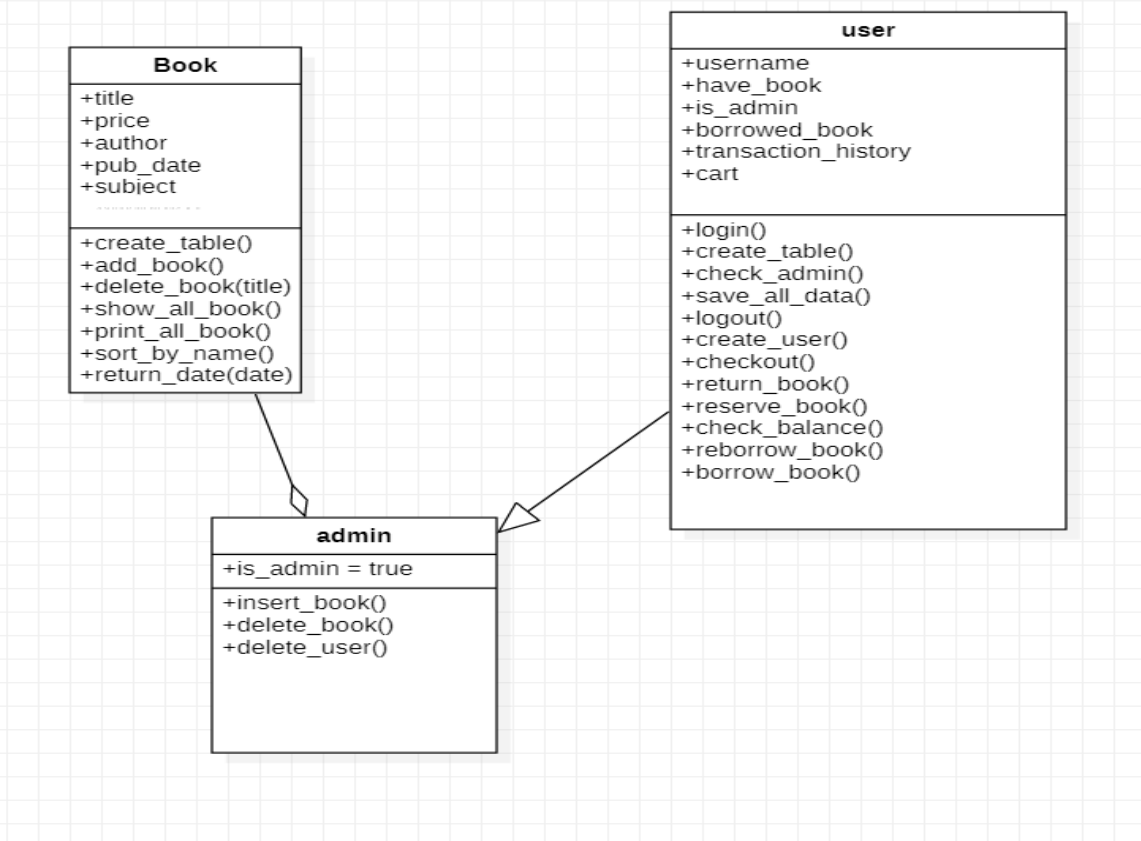
Admin:

- Binary check whether user is admin or not
- Admin can delete and add instances of book in Database
- Admin can cancel subscription of user
- Username is primary key for all operations

User:

- User can return all the books to library
- User can reserve and borrow a book

UML Diagram :



Merge Sort was utilized in method sort_by_name() in Book class

Binary search was Utilized in borrow_book() in user class

Features:

- All the Data will be stored in sqlite3 database
- Binary Search was used to find key of the book and store its instance to the db. and user account
- Merge sort was used for sorting all the records on the basis of Book name

Limitations:

- User have to insert integer in command line as it will exit the program and you have to rerun it,
- For creating new instance of database, you have to run make_table() operation in Book and user class for once only
- It's an CLI version

Merge Sort:

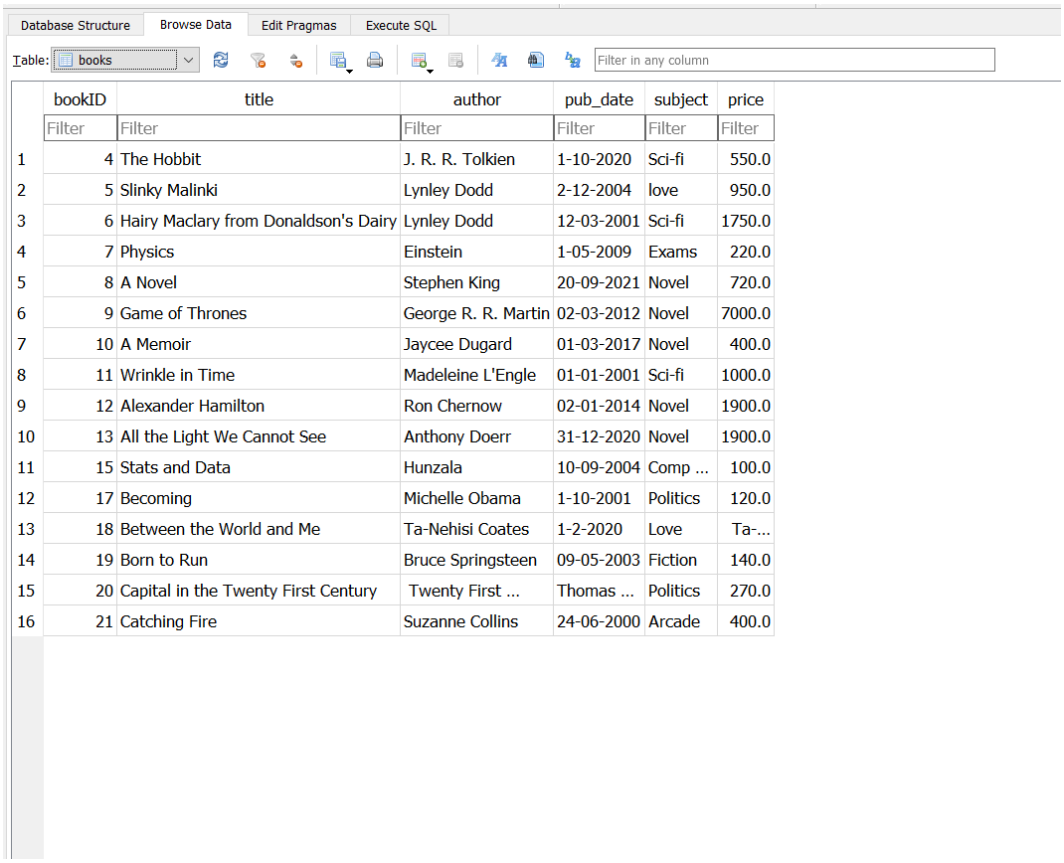
```
61      def sort_by_name(self):
62          cur = db.cursor()
63          cur.execute("SELECT * FROM books")
64          raw_result = cur.fetchall()
65          result = [data[1] for data in raw_result]
66          def mergeSort(arr):
67              if len(arr) > 1:
68                  mid = len(arr) // 2
69                  L = arr[:mid]
70                  R = arr[mid:]
71                  mergeSort(L)
72                  mergeSort(R)
73                  i = j = k = 0
74                  while i < len(L) and j < len(R):
75                      if L[i] < R[j]:
76                          arr[k] = L[i]
77                          i += 1
78                      else:
79                          arr[k] = R[j]
80                          j += 1
81                      k += 1
82                  while i < len(L):
83                      arr[k] = L[i]
84                      i += 1
85                      k += 1
86                  while j < len(R):
87                      arr[k] = R[j]
88                      j += 1
89                      k += 1
90          mergeSort(result)
```

Binary Search:

```
234     def borrow_book(self,id):
235         found = False
236         all_books= Book().show_all_book()
237         low = 0
238         high = len(all_books) - 1
239         mid = 0
240         while low <= high:
241             mid = (high + low) // 2
242
243             if all_books[mid][0] < id:
244                 low = mid + 1
245
246             elif all_books[mid][0] > id:
247                 high = mid - 1
248             else:
249                 print(all_books[mid])
250                 self.cart[all_books[mid][1]] = (all_books[mid][5])
251                 found = True
252                 self.have_book = True
253                 break
254         return found
```

Database:

Books Data in SQLite:



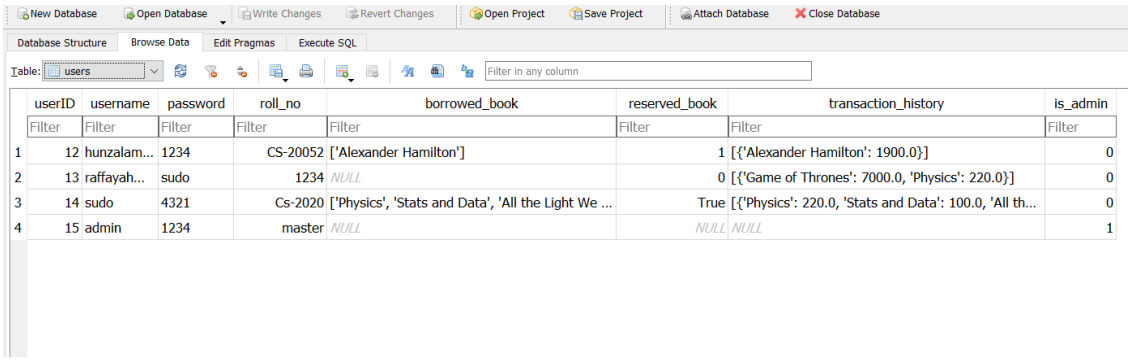
Database Structure | Browse Data | Edit Pragmas | Execute SQL

Table: books

Filter in any column

	bookID	title	author	pub_date	subject	price
	Filter	Filter	Filter	Filter	Filter	Filter
1	4	The Hobbit	J. R. R. Tolkien	1-10-2020	Sci-fi	550.0
2	5	Slinky Malinki	Lynley Dodd	2-12-2004	love	950.0
3	6	Hairy Maclary from Donaldson's Dairy	Lynley Dodd	12-03-2001	Sci-fi	1750.0
4	7	Physics	Einstein	1-05-2009	Exams	220.0
5	8	A Novel	Stephen King	20-09-2021	Novel	720.0
6	9	Game of Thrones	George R. R. Martin	02-03-2012	Novel	7000.0
7	10	A Memoir	Jaycee Dugard	01-03-2017	Novel	400.0
8	11	Wrinkle in Time	Madeleine L'Engle	01-01-2001	Sci-fi	1000.0
9	12	Alexander Hamilton	Ron Chernow	02-01-2014	Novel	1900.0
10	13	All the Light We Cannot See	Anthony Doerr	31-12-2020	Novel	1900.0
11	15	Stats and Data	Hunzala	10-09-2004	Comp ...	100.0
12	17	Becoming	Michelle Obama	1-10-2001	Politics	120.0
13	18	Between the World and Me	Ta-Nehisi Coates	1-2-2020	Love	Ta-...
14	19	Born to Run	Bruce Springsteen	09-05-2003	Fiction	140.0
15	20	Capital in the Twenty First Century	Twenty First ...	Thomas ...	Politics	270.0
16	21	Catching Fire	Suzanne Collins	24-06-2000	Arcade	400.0

User data in SQLite:



New Database | Open Database | Write Changes | Revert Changes | Open Project | Save Project | Attach Database | Close Database

Database Structure | Browse Data | Edit Pragmas | Execute SQL

Table: users

Filter in any column

	userID	username	password	roll_no	borrowed_book	reserved_book	transaction_history	is_admin
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	12	hunzalam...	1234	CS-20052	['Alexander Hamilton']	1	[{'Alexander Hamilton': 1900.0}]	0
2	13	raffayah...	sudo	1234	NULL	0	[{'Game of Thrones': 7000.0, 'Physics': 220.0}]	0
3	14	sudo	4321	Cs-2020	['Physics', 'Stats and Data', 'All the Light We ...	True	[{'Physics': 220.0, 'Stats and Data': 100.0, 'All th...	0
4	15	admin	1234	master	NULL	NULL	NULL	1

Sort By Names: