Abstract

Library Automation system provides library functions. It shows basic functions like login, Book Reserve, Book Return, Administrator functions like create new reader and enter some data in existing system. Library Automation system is efficient and easy to use. It has a user friendly GUI.

Keywords: Automation

Library Automation System

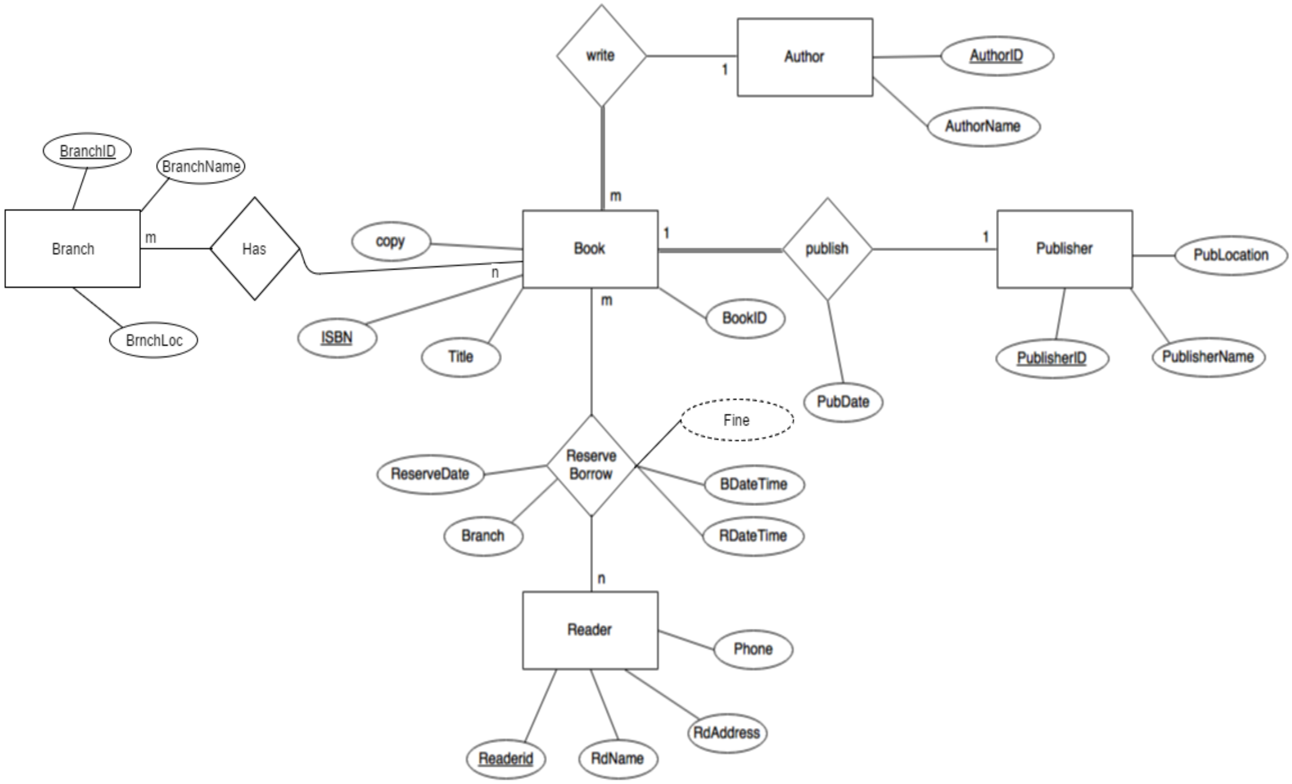
Introduction

Realizing the important role that the library management system will play in planning and implementing library automation projects. A library management system, also known as an automated library system is software that has been developed to handle basic housekeeping functions of a library. Here in Library Automation system I have used Java Server Pages (Jsp), Servlet and Hibernate for java class mapping. In database I have used MySQL database because it is more convenient and open source.

Library Automation System

ER data model design

ER Diagram.



Entities and their attributes.

Book: Book is an entity of library system.

BookId, title, author, ISBN, PublisherId, PublicationDate are the attributes of entity Book.

Reader: Reader is an entity of library system. Reader should be exist in the system before it can use any Functionality.

ReaderId,ReaderName, ReaderAddress, PhoneNumber, are the attributes of entity Reader.

Branch: Branch is an entity of library system. Library system have a different branch and branch has a book and detail about book.

BranchId, BranchName, BranchLoc, NoOfCopyOfBook

Publisher: Publisher is an entity of library system. Publisher publish a book.

PublisherId, PublisherAddress

Author: Author is an entity of library system. Author is write a book.

AuthorId, AuthorName

Borrow: Borrow is an entity of library system. Borrowed book is stored here.

BDateTime, RDateTime

Fine is derived attribute from the BDateTime and RDateTime

Adminlogin: Adminlogin is a relation of library system. Username and password are two attributes of it.

Relationships:

- Each book has a single publisher.
  Book-publisher (1:1) is publish by
  Publicationdate is the attribute for publish by
- Each book has a single Author. Author can have a more than 1 book.
  Book-Author (m:1) is Written by
- Reader can borrow maximum 10 books. Reader can borrow same copy of book. Same book with more than 1 copy can be borrowed by more than one Reader.
  Reader-Borrow (m:n) is borrowed by
  Borrow Date and time, Return date and time, Borrow branch are the attribute.
- Reader can reserve maximum 10 books. Reader can reserve same copy of book. Same book with more than 1 copy can be reserved by more than one Reader.
  Reader-Reserve (m:n) is reserved by
  Reserve date is the attribute.
- Each book has to be returned to the branch from which they are returned.

Logical Design of database.

- **Mapping Entity**

Schema

Primary Key cannot be NULL.

Author:

AuthorID	AuthorName

AuthorID is the primary key. Datatype is INT.

Datatype of AuthorName is CHAR.

Publisher:

PublisherID	PublisherName	PubLocation

PublisherID is the primary key. Datatype is INT.

Datatype of PublisherName is CHAR.

Datatype of PubLocation is CHAR.

Book:

BookID	BookTitle	ISBN	PublisherID	PubDate	AuthorID

BookID is the primary key. PublisherID is foreign key for the relation Publisher.

AuthorID is foreign key for the relation Author.

Datatype of BookID is INT.

Datatype of BookTitle is CHAR.

Datatype of ISBN is CHAR.

Datatype of PubDate is DATE.

Branch:

BranchID	BranchName	BranchLoc

BranchID is the primary key.
Datatype of BranchID is INT.
Datatype of BranchName is CHAR.
Datatype of BarnchLoc is CHAR.

Reader:

ReaderID	ReaderName	Address	Phone

ReaderID is the primary key.
Datatype of ReaderID is INT.
Datatype of ReaderName is CHAR.
Datatype of Address is CHAR.
Datatype of Phone is CHAR.

Borrow:

BookID	ReaderID	BranchID	BDateTime	RDateTime	Fine

BookID, ReaderID and BDateTime are the primary key. BookID is foreign key for the relation Book. ReaderID is foreign key for the relation Reader. BranchID is foreign key for the relation Branch.
Datatype of BDateTime is DATE.
Datatype of RDateTime is DATE.
Datatype of Fine is FLOAT.

Reserve:

BookID	ReaderID	ReserveDate	BranchID

BookID, ReaderID and ReserveDate are the primary key. BookID is foreign key for the relation Book. ReaderID is foreign key for the relation Reader. BranchID is foreign key for the relation Branch.
Datatype of ReserveDate is DATE.

NoOfCopy:

BranchID	BookID	Copy

BranchID and BookID are the primary key. BranchID is foreign key for the relation Branch. BookID is foreign key for the relation Book.
Datatype of Copy is INT.

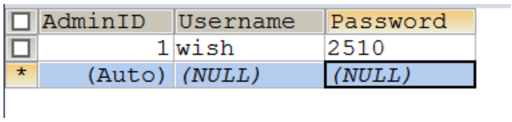
Relational Database Design

Here database is in 3NF. It preserve lossless join and dependency.

Database Implementation

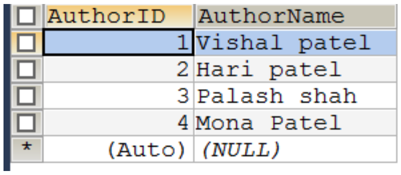
Here are some example of database.

Database for Administrator login



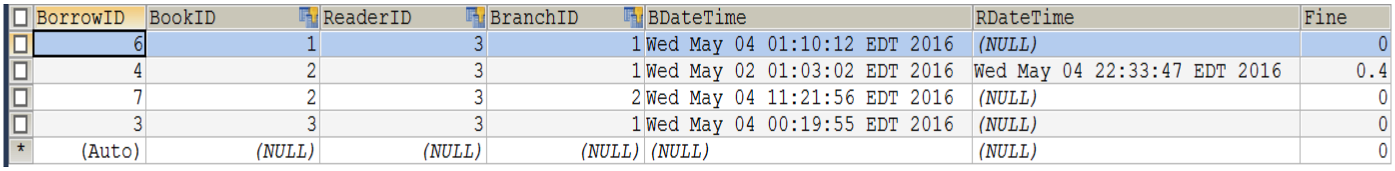
□	AdminID	Username	Password
□	1	wish	2510
*	(Auto)	*(NULL)*	*(NULL)*

Author Database:



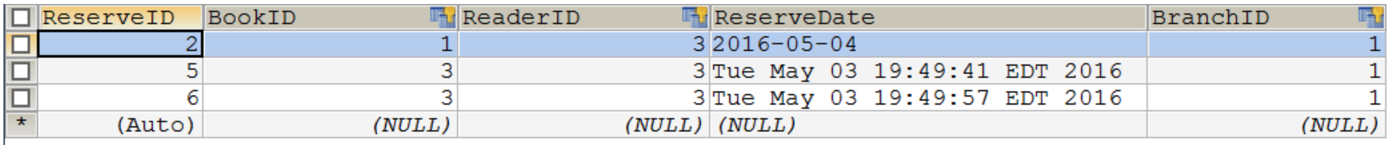
□	AuthorID	AuthorName
□	1	Vishal patel
□	2	Hari patel
□	3	Palash shah
□	4	Mona Patel
*	(Auto)	*(NULL)*

Borrow Database



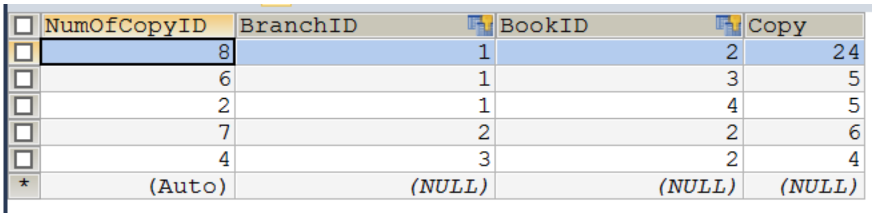
□	BorrowID	BookID	ReaderID	BranchID	BDateTime	RDateTime	Fine
□	6	1	3	1	Wed May 04 01:10:12 EDT 2016	*(NULL)*	0
□	4	2	3	1	Wed May 02 01:03:02 EDT 2016	Wed May 04 22:33:47 EDT 2016	0.4
□	7	2	3	2	Wed May 04 11:21:56 EDT 2016	*(NULL)*	0
□	3	3	3	1	Wed May 04 00:19:55 EDT 2016	*(NULL)*	0
*	(Auto)	*(NULL)*	*(NULL)*	*(NULL)*	*(NULL)*	*(NULL)*	0

Reserve Database



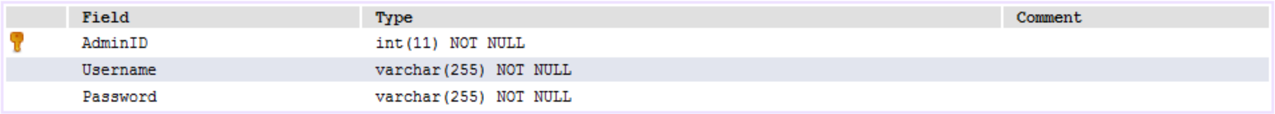
□	ReserveID	BookID	ReaderID	ReserveDate	BranchID
□	2	1	3	2016-05-04	1
□	5	3	3	Tue May 03 19:49:41 EDT 2016	1
□	6	3	3	Tue May 03 19:49:57 EDT 2016	1
*	(Auto)	*(NULL)*	*(NULL)*	*(NULL)*	*(NULL)*

Noofcopy Ddatabase



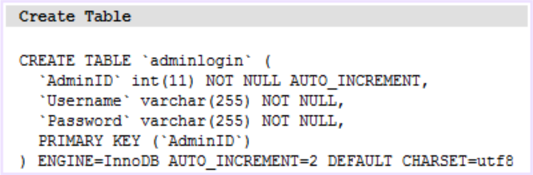
□	NumOfCopyID	BranchID	BookID	Copy
□	8	1	2	24
□	6	1	3	5
□	2	1	4	5
□	7	2	2	6
□	4	3	2	4
*	(Auto)	*(NULL)*	*(NULL)*	*(NULL)*

Database definition and schema



	Field	Type	Comment
🔑	AdminID	int(11) NOT NULL	
	Username	varchar(255) NOT NULL	
	Password	varchar(255) NOT NULL	

Schema and type of data base attribute. Here admin is Int datatype and password is of varchar.



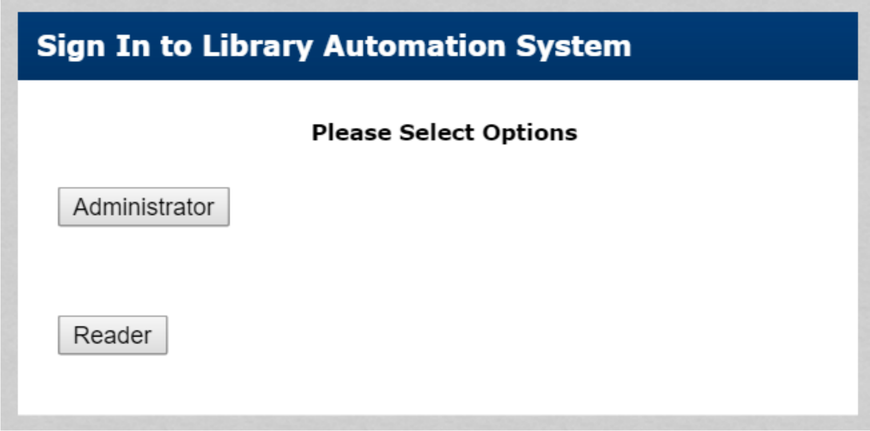
```
Create Table

CREATE TABLE `adminlogin` (
  `AdminID` int(11) NOT NULL AUTO_INCREMENT,
  `Username` varchar(255) NOT NULL,
  `Password` varchar(255) NOT NULL,
  PRIMARY KEY (`AdminID`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8
```

Create query for admin login to create database. It will create database of given detail.

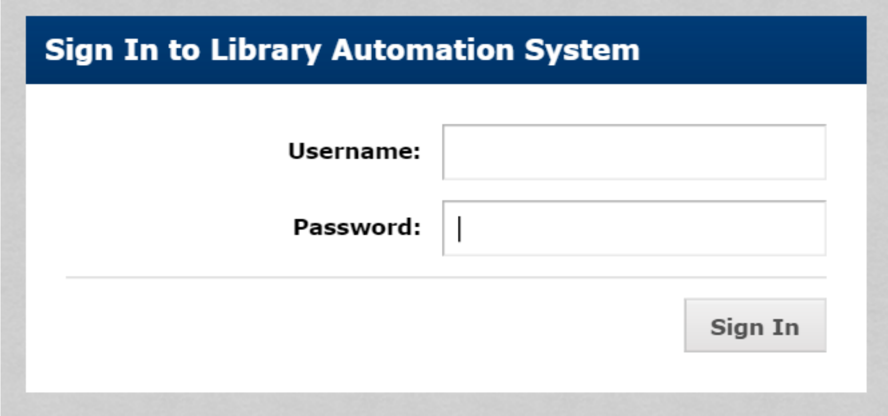
Application Design

Main Page



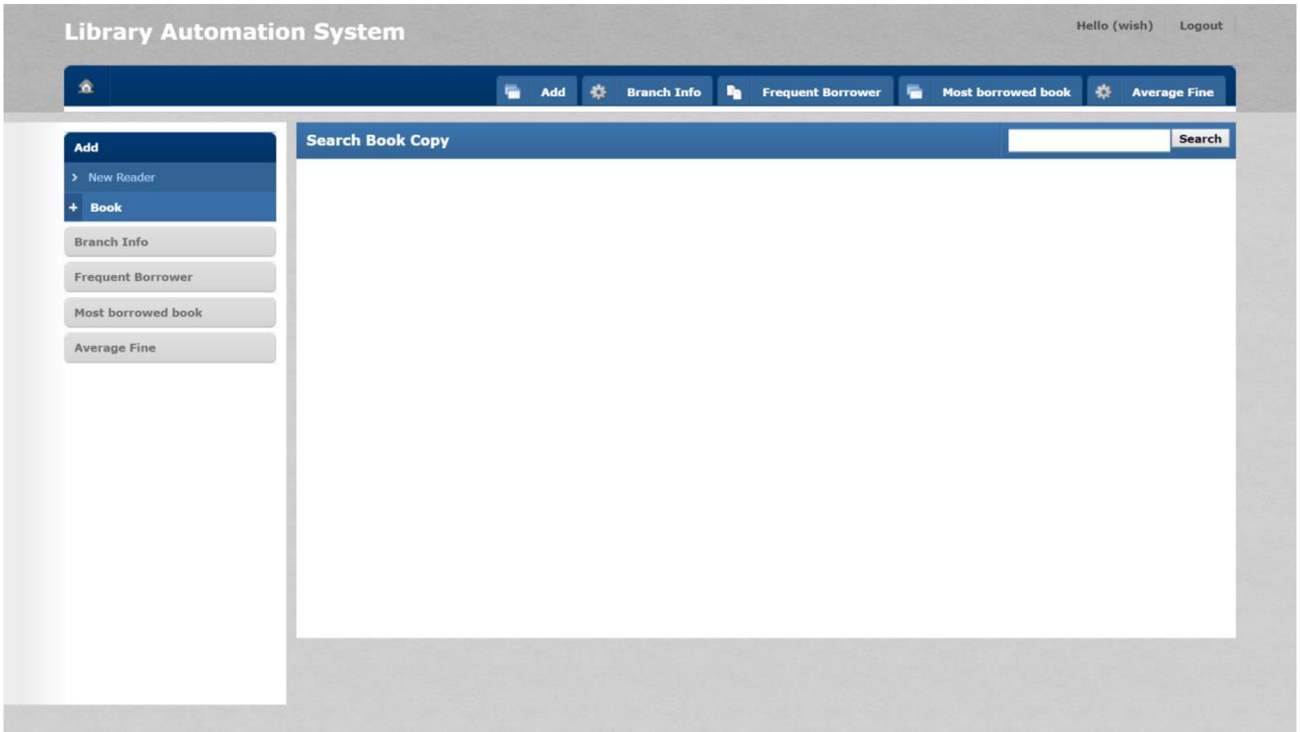
This is Main page of the system. From Here we can go for administrator functions and user login.

Admin Login



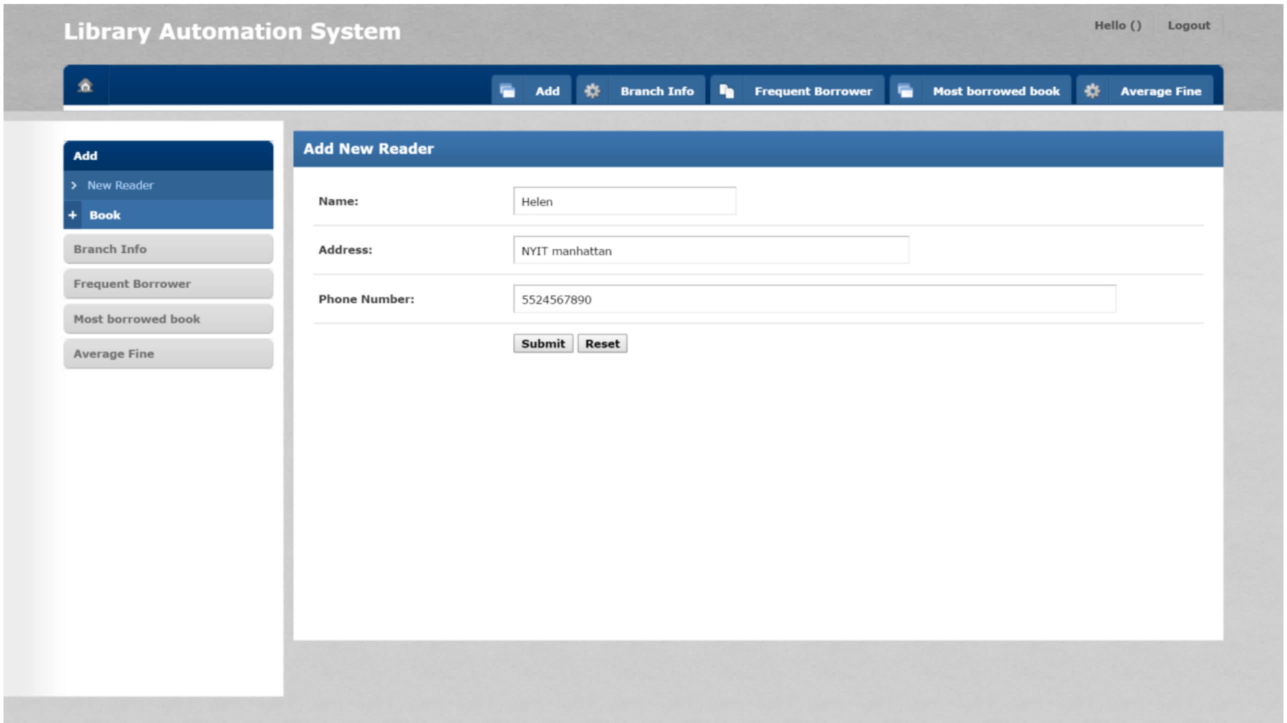
Here we need correct username and password to login and access administrator functions.

Admin Main Page



It is the home page for administrator. Here Add reader and add new book and branchinfo and Frequent borrower functions are available.

Add new user.



Add New Reader

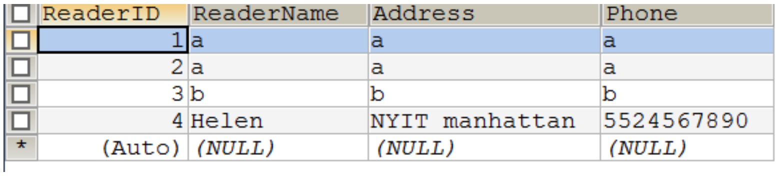
Name: Helen

Address: NYIT manhattan

Phone Number: 5524567890

Submit Reset

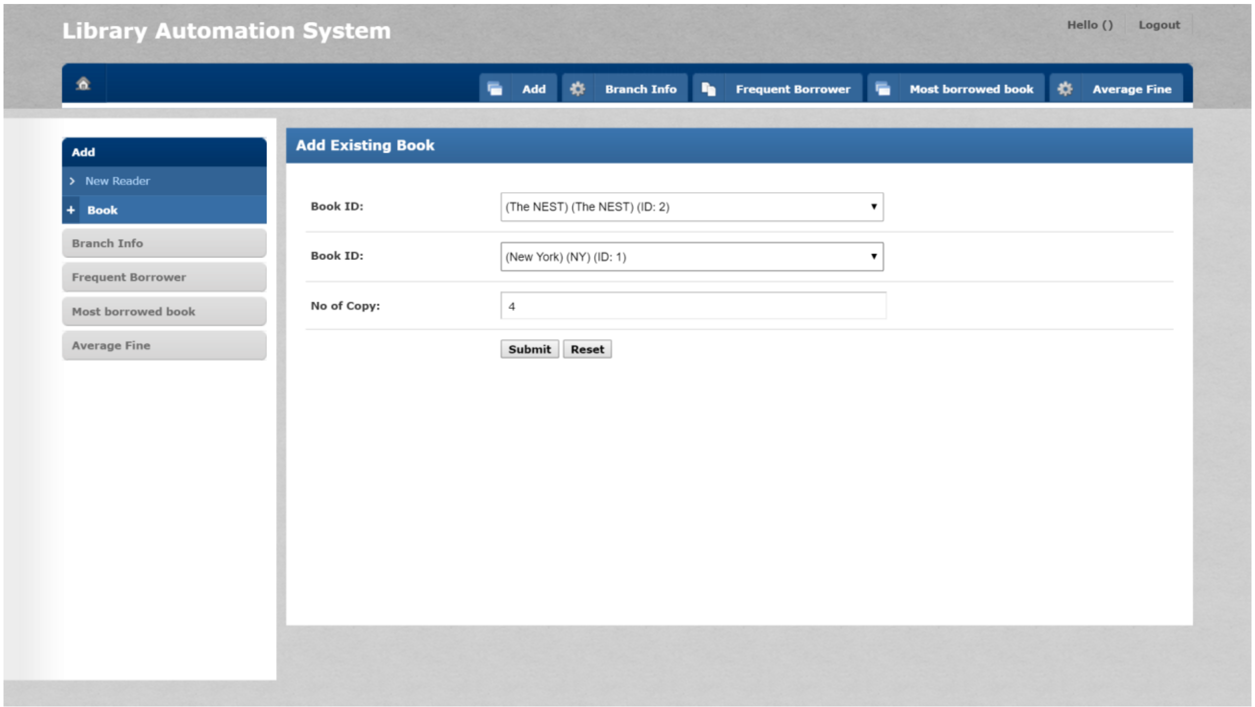
When we click on the Add new user it shows function like this and when we click submit button it stores data to database. Next figure shows database entry.



□	ReaderID	ReaderName	Address	Phone
□	1	a	a	a
□	2	a	a	a
□	3	b	b	b
□	4	Helen	NYIT manhattan	5524567890
*	(Auto)	*(NULL)*	*(NULL)*	*(NULL)*

New reader is created in the database.

Add Existing book



Add Existing Book

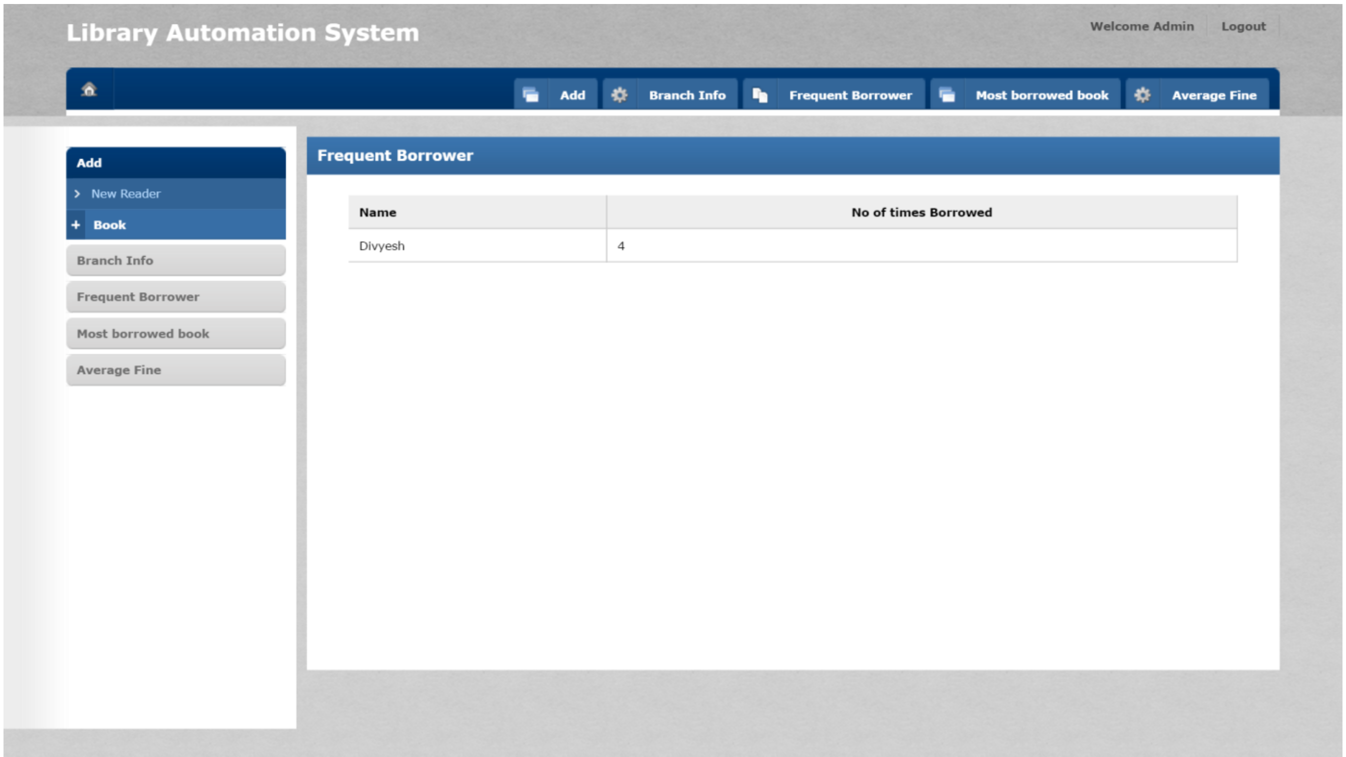
Book ID: (The NEST) (The NEST) (ID: 2)

Book ID: (New York) (NY) (ID: 1)

No of Copy: 4

This is the page for adding existing book. Here when we select branch id and book id to submit it will enter data into database.

Frequent Borrower

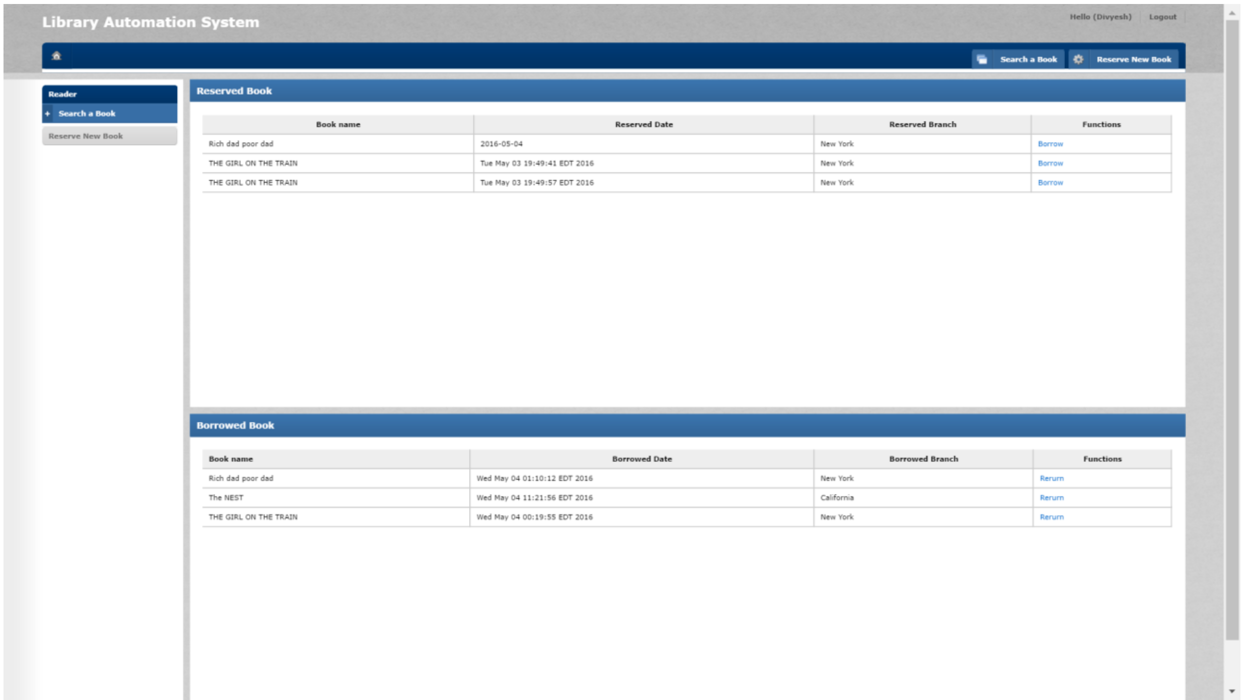


Frequent Borrower

Name	No of times Borrowed
Divyesh	4

This is the module to see frequent borrower from the database. It will so the most frequent borrower and how many times he borrowed a book from the library.

Reader Function



Reserved Book

Book name	Reserved Date	Reserved Branch	Functions
Rich dad poor dad	2016-05-04	New York	Borrow
THE GIRL ON THE TRAIN	Tue May 03 19:49:41 EDT 2016	New York	Borrow
THE GIRL ON THE TRAIN	Tue May 03 19:49:57 EDT 2016	New York	Borrow

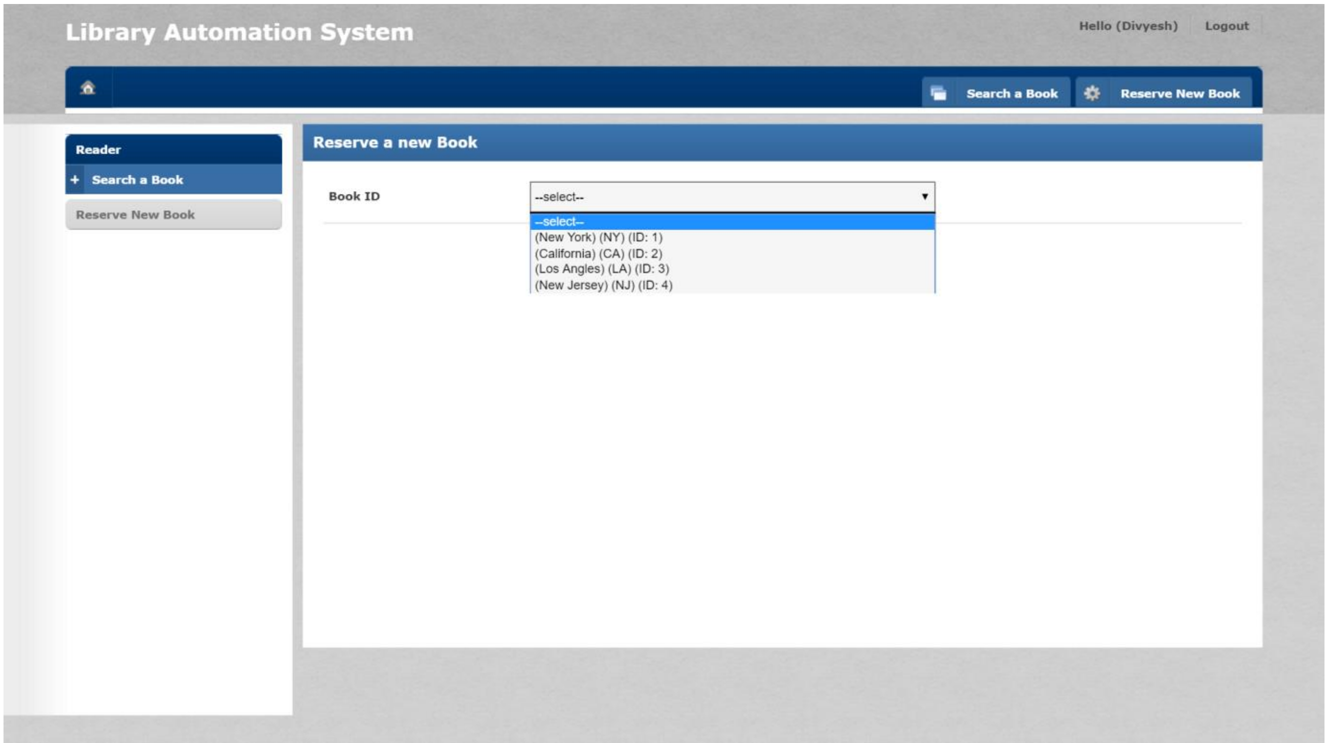
Borrowed Book

Book name	Borrowed Date	Borrowed Branch	Functions
Rich dad poor dad	Wed May 04 01:10:12 EDT 2016	New York	Rerurn
The NEST	Wed May 04 11:21:56 EDT 2016	California	Rerurn
THE GIRL ON THE TRAIN	Wed May 04 00:19:55 EDT 2016	New York	Rerurn

This is the homepage for the reader function it will show reserved and borrowed book of the reader. Then reader can borrow reserved book and can return borrowed book by directly link.

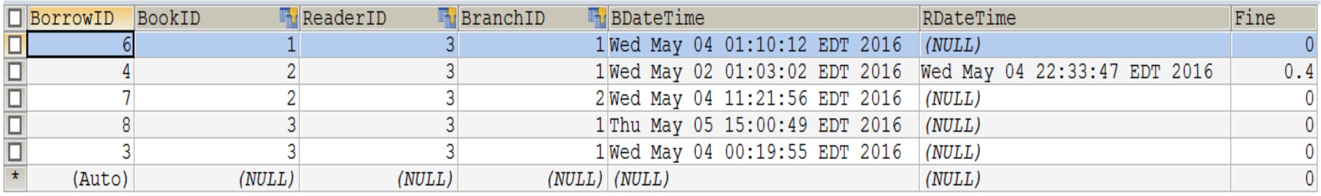
Here reader cannot borrow more than 10 books.
Reader will show only borrowed book in his own account.

Borrow a reserved book



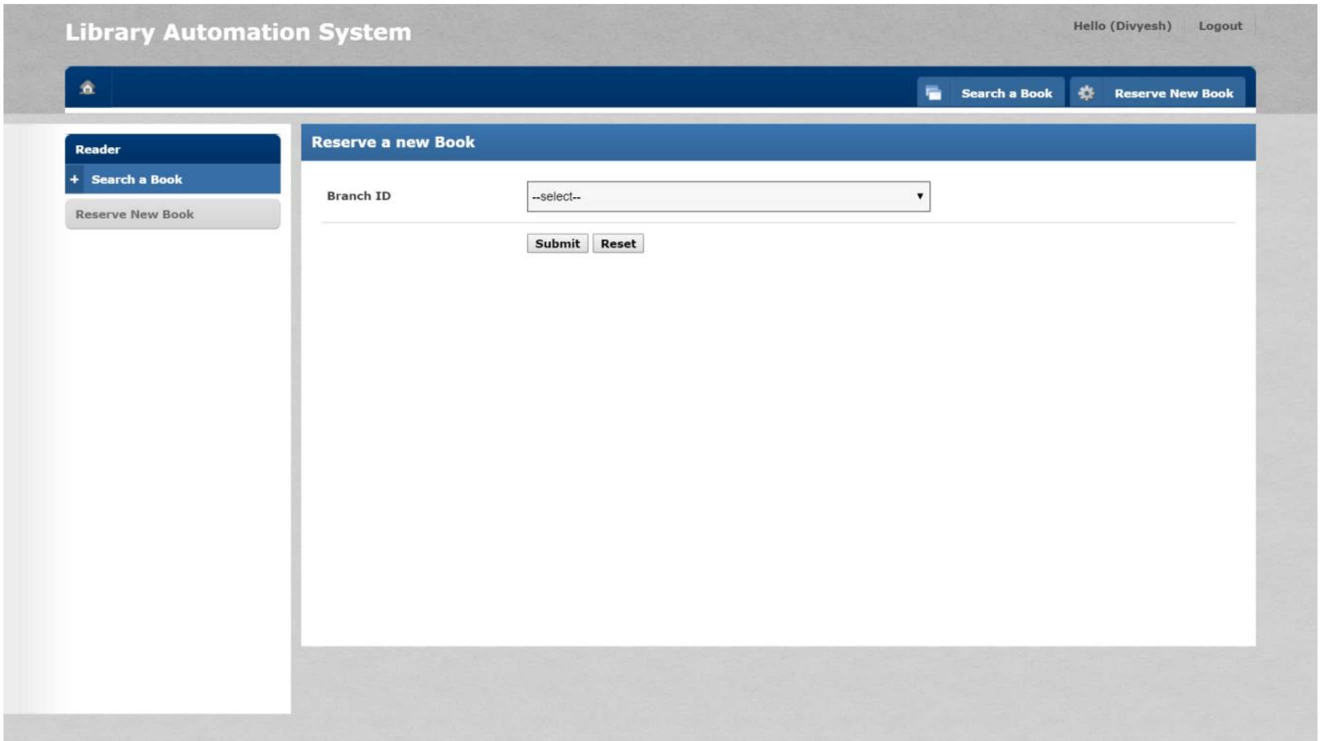
When you click on the book to borrow it will come to this page and reader has to select branch from where he/she reserved book and if it will be same branch then book will be borrowed for the reader. If the branch of the reserved is not same as selected then it will pop up that please select same branch.

When submit button is clicked it will enter database entry in the database



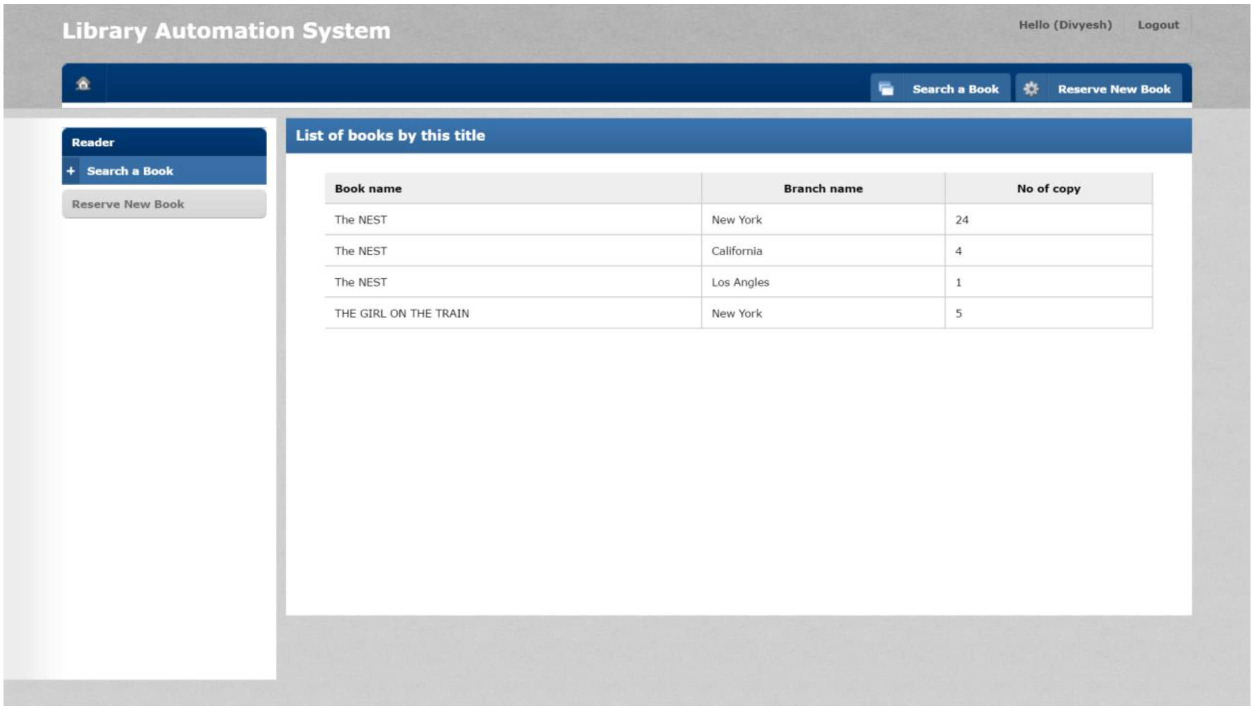
□	BorrowID	BookID	ReaderID	BranchID	BDateTime	RDateTime	Fine
□	6	1	3	1	Wed May 04 01:10:12 EDT 2016	*(NULL)*	0
□	4	2	3	1	Wed May 02 01:03:02 EDT 2016	Wed May 04 22:33:47 EDT 2016	0.4
□	7	2	3	2	Wed May 04 11:21:56 EDT 2016	*(NULL)*	0
□	8	3	3	1	Thu May 05 15:00:49 EDT 2016	*(NULL)*	0
□	3	3	3	1	Wed May 04 00:19:55 EDT 2016	*(NULL)*	0
*	(Auto)	*(NULL)*	*(NULL)*	*(NULL)*	*(NULL)*	*(NULL)*	0

Reserve a new book



This is the module for reserve a new book. It is for reserve a new book from branch. So reader has to select branch where he can reserve a book and it will enter a new entry to database.

Search By ID

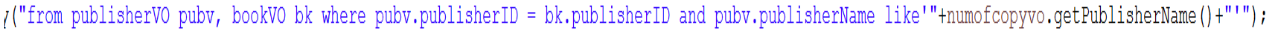


List of books by this title

Book name	Branch name	No of copy
The NEST	New York	24
The NEST	California	4
The NEST	Los Angles	1
THE GIRL ON THE TRAIN	New York	5

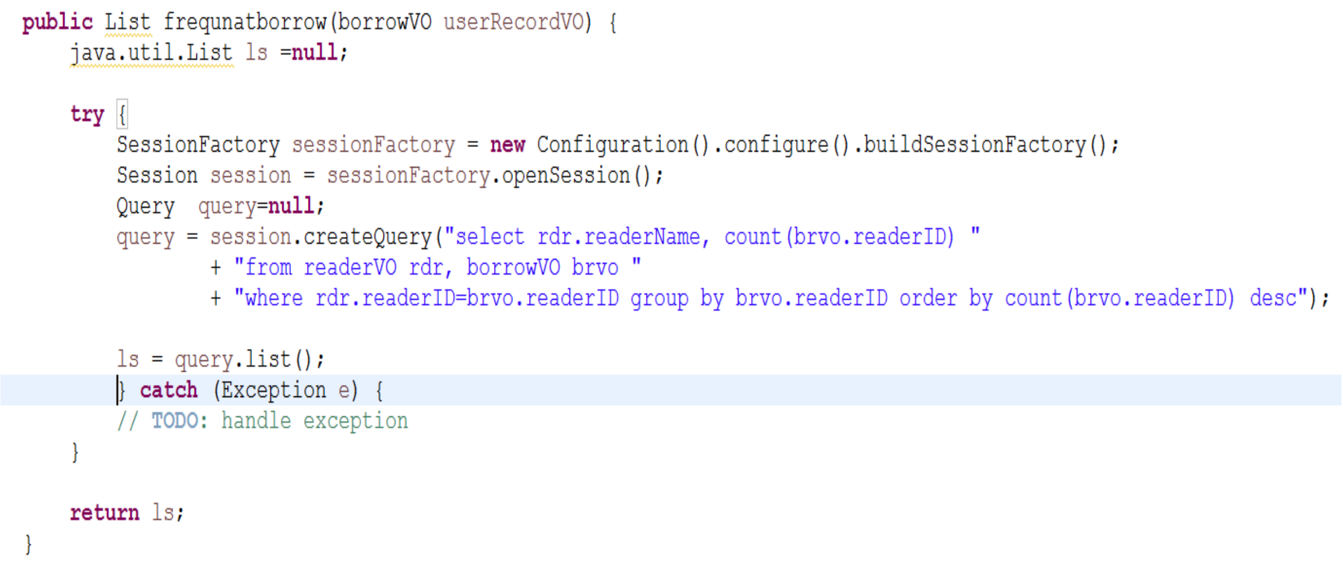
Search by ID shows all book as per the name of book from the database.

Here is some example of query to retrieve data from database.



```
("from publisherVO pubv, bookVO bk where pubv.publisherID = bk.publisherID and pubv.publisherName like'"+numofcopyvo.getPublisherName()+"'");
```

Some query from database



```
public List frequnatborrow(borrowVO userRecordVO) {
    java.util.List ls =null;

    try {
        SessionFactory sessionFactory = new Configuration().configure().buildSessionFactory();
        Session session = sessionFactory.openSession();
        Query  query=null;
        query = session.createQuery("select rdr.readerName, count(brvo.readerID) "
                + "from readerVO rdr, borrowVO brvo "
                + "where rdr.readerID=brvo.readerID group by brvo.readerID order by count(brvo.readerID) desc");

        ls = query.list();
        } catch (Exception e) {
        // TODO: handle exception
    }

    return ls;
}
```

It shows HQL (Hibernate query language) to use group by clauses.