class Book {

/\*\*

\* Store all attributes of a book

\*/

protected String title = "";

protected String publisher = "";

protected String author = "";

protected String ibsn = "";

protected int sequenceNumber = 0;

protected String typeofCopy = "";

protected int price = 0;

protected String bookNumber = "";

protected String copyNumber = "";

/\*\*

\*Contruct an object of new book to add

\*/

Book(){

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*/

class AddCopyForm{

/\*\*

\* a copy instance with full information to add

\*/

public Book book = new Book();

/\*\*

\* attribute to store edited information

\*/

String editedCpType = "";

int editedCpPrice = 0;

/\*\*

\*display form to fill out the information of the copy

\*@throws IOException if there was a problem reading from the string

\*/

void displayCopyInformationForm(){

rendering form for filling out the information.

throws IOException

}

/\*\*

\*display result of validate of the form

\*/

void displayCopyFormValidation(){

if form not validate

rendering error from which fields execute

}

/\*\*

\*add the information about the copy to add from the form to the instance

\*/

private void addCopy(){

add all the information form the form fields to the instance

}

/\*\*

\*display form to fill out the information of the copy to edit

\*@throws IOException if there was a problem reading from the string

\*/

void displayEditInformationForm(){

rendering form for filling out the information.

throws IOException

}

/\*\*

\*display result of validate of the form of edit information

\*/

void displayEditFormValidation(){

if form not validate

rendering error from which fields execute

}

/\*\*

\*add the edited information about the copy to add from the form to the attributes

\*/

private void addEdited(){

add all the information form the form fields to attributes to store.

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*/

class AddCopyController{

/\*\*

\*an instance of Book class which have all properties of the book from class addCopyForm

\*/

Book booktoVerify = AddCopyForm.book;

/\*\*

\*an attribute to store cpNum was generated

\*/

private String cpNum = "";

/\*\*

\*an attribute to store value of error

\*/

private int bookNumErr = 0;

/\*\*

\*an attribute to store TypeofCopy was changed

\*/

private String editedCpType = "";

/\*\*

\*an attribute to store PricceofCopy was changed

\*/

private int editedPriceofCopy = 0;

/\*\*

\*call method check bookNumber in database

\*@param bookNo a string represent bookNumber of the copy to add

\*@return an int in set of 0 and 1

\*/

int checkBookNumber(String booktoVerify.bookNumber){

call method from the other class to check

if exist

return 1

if not exist

return 0

}

/\*\*

\*display result after checking book number in database

\*@param bookNumber an int represents result of checking

\*/

void displayCheckBookNumberResult(int bookNumErr){

if bookNumErr == 0

display error

}

/\*\*

\*generate copy Number of the book

\*@param instance of Book we created

\*@return a string

\*/

void generateCopyNumber(Book booktoVerify){

generate cpNum;

return cpNum

}

/\*\*

\*add the information of cpNum of the new book to to the instance after verify

\*/

private void addCopy(){

booktoVerify.copyNumber = this.cpNum;

}

/\*\*

\*add the information of bookNum and cpNum of the new book to to the instance after verify

\*/

private void addEdited(){

this.editedCpType = AddCopyForm.editedCpType;

this.editedPriceofCopy = AddCopyForm.editedCpPrice;

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*/

class BookInfo {

/\*\*

\*an instance of Book class which have all properties of the book from class AddNewBookController

\*/

Book CptoAdd = AddCController.booktoVerify;

/\*\*

\*check book number exist in database of not

\*@param bookNo a string respresent book number of the book

\*@return boolean

\*@throws SQLException if the connection encounters an error during an interaction with a data source

\*/

public Boolean checkBookNumber(String bookNo){

querrytoCheck;

if exist

return true;

if not exist

return false;

throws SQLException

}

\*\*/

\*add the copy to database

\*@param instance of Book which store all verified attributes to add

\*@throws SQLException if the connection encounters an error during an interaction with a data source

\*/

storeCopyInfo(Book CptoAdd){

querrytoadd;

throws SQLException

}

\*\*/

\*add the edited information of the copy to database

\*@param instance of Book which store all verified attributes to add

\*@throws SQLException if the connection encounters an error during an interaction with a data source

\*/

storeCopyInfo(Book CptoAdd){

querrytoadd;

throws SQLException

}

}