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 AddNewBookForm {

/\*\*

\* a new book with full information to add

\*/

public Book book = new Book();

/\*\*

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

\*/

void displayNewBookInformationForm(){

rendering form for filling out the information.

throws IOException

}

/\*\*

\*display result of validate of the form

\*/

void displayNewBookFormValidation(){

if form not validate

rendering error from which fields execute

}

/\*\*

\*add the information about the new book to add from the form to the instance

\*/

private void addCopy(){

add all the information form the form fields to the instance

}

}

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

class AddNewBookController{

/\*\*

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

\*/

Book booktoVerify = addNewBook.book;

/\*\*

\*an attribute to store bookNum was generated

\*/

String bookNum = "";

/\*\*

\*an attribute to store cpNum was generated

\*/

String cpNum = "";

/\*\*

\*an attribute to store value of error

\*/

int bookNumErr = 0;

/\*\*

\*generate Book Number of the book

\*@param instance of Book we created

\*@return a string

\*/

void generateBookNumber(Book booktoVerify){

generate bookNum;

return bookNum;

}

/\*\*

\*call method check bookNumber in database

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

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

\*/

int checkBookNumber(String bookNo){

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 == 1

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 bookNum and cpNum of the new book to to the instance after verify

\*@param instance of Book we created

\*/

void addNewBook(){

booktoVerify.bookNumber = this.bookNum;

booktoVerify.copyNumber = this.cpNum;

}

}

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

class BookInfo {

/\*\*

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

\*/

Book booktoAdd = AddNewBookController.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 new book 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

\*/

storeNewBookInfo(Book booktoAdd){

querrytoadd;

throws SQLException

}

}