all functions usage

void read(book arrbook[], book arrdlt[], sales arrsal[], brwd arrbrr[]); //read all data from all files

void balances(); //read all balances of all arrbookays

int lines\_count(string fileName);

void read\_library(book arrbook[]); //read books' data after opening program

void read\_out(brwd arrbrr[]); //read borrow transactions data after opening program

void read\_deleted(book arrdlt[]); //read deleted books' data after opening program

void read\_sales(sales arrsal[]); //read sales transactions data after opening program

void separating\_line(); //separate each action from following

int customer\_or\_employee(); //define if user is customer or employee

int password(); //check if password is right no

void customer\_list(); //display customer options list

void employee\_list(); //display employee options list

void customer(book arrbook[]); //access all customer functions

void employee(book arrbook[], book arrdlt[], sales arrsal[], brwd arrbrr[]); //access all employee functions

void mini\_outlook(book out[], int i); //display ISBN & name of one book

void outlook(book arrbook[], int i); //display details of ONE book

void show\_all(book arrbook[]); //display details of all books

void buy\_borrow\_both(book arrbook[]); //display details of books by type

void borrow\_book(book arrbook[], brwd arrbrr[]); //borrow a book

void borrow\_transaction(book arrbook[], brwd arrbrr[], int i); //input borrower data

void buy\_book(book arrbook[], book arrdlt[], sales arrsal[]); //buy a book

void buy\_transaction(book arrbook[], book arrdlt[], sales arrsal[], int i); //input transaction data

void search(book arrbook[], int look); //search for a book

void add\_choice(book arrbook[], book arrdlt[]); //choose way of addation

void manual\_add(book arrbook[]); //add new books

void add\_from\_delete(book arrbook[], book arrdlt[]); //add book from delete file

void increase\_amount(book arrbook[]); //change amount of book in library

void delete\_book(book arrbook[], book arrdlt[], string remove); //delete a book

void save\_delete(book arrbook[], book arrdlt[], int i); //save data of deleted book

void write(book arrbook[], book arrdlt[], sales arrsal[], brwd arrbrr[]); //write all data into all file

void write\_library(book arrbook[]); //save books' data before closing program

string getpassword(const string& prompt = "Enter password> "); //the function which gets the password from the user

void write\_out(brwd arrbrr[]); //save borrowers' data before closing program

void write\_deleted(book arrdlt[]); //save deleted books' data before closing program

void write\_sales(sales arrsal[]); //save sales' data before closing program

void autosave(book arrbook[], book arrdlt[], sales arrsal[], brwd arrbrr[]); //auto save data while program is running

void display\_sales\_transaction(sales arrsal[]); //displays the sales transaction

int mean\_title(book arrbook[], int counter, string srch); //search for the closest title which the user enters

int mean\_author(book arrbook[], int counter, string srch); //search for the closest author's name which the user enters

What Each Function Do

Main :  
it calls balances function which reads the amount of (books in library-borrowed books and the data of borrowers -deleted books-saled books and data of buyers)  
After that we create 6 arrays which holds the data of everything in files   
book \*arrbook = new book[bookNum + 1]; //books in library array  
book \*arrdlt = new book[deleteNum + 1]; //deleted books array  
book \*arrbrwd = new book[brwdNum + 1]; //borrowed books array  
sales \*arrsal = new sales[saleNum + 1]; //sales transactions array  
brwd \*arrbrr = new brwd[borrowNum + 1]; //borrow transactions array  
emply \*arremp = new emply[empNum + 1]; //employee data array  
then it calls read function which takes 4 parameters with special structure datatype  
book arrbook[], book arrdlt[], sales arrsal[], brwd arrbrr[]  
and the read function calls 4 more functions :

1. Read\_library “Reads all data of each book in the library”
2. Read\_out “Reads the data of borrowed books”
3. Read\_deleted “Reads all data of deleted books”
4. Read\_sales “Reads data of saled books”

It create a variable called “opt” of type int saves the return value of customer\_or\_employee function then check

* if “opt” == 1

calls customer function which takes an array as parameter which is “arrbook” –Contains all data of each book in library- and then calls Customer\_list function which display the customer list and options

1. List All books in library
2. list all books to buy or borrow
3. search for a book
4. exit program

if the user chose :   
“1” it will display all books in library

“2” it will display all books which can be bought or borrow depends on his needs

“3” will make him/her search for a book with many opntions like search with (ISBN/Title/Author Name/Category)

“4” to exit the running program

* if “opt” == 2

calls password function which asks for a password and checks if it’s the saved password or no for 3 times … after 3 tries the program exists . if the password is correct it access the employee function which calls employee\_list to display the employee list and options

1. List All books in library
2. List All books to buy or borrow
3. Borrow book
4. Buy book
5. Search for a book
6. Add a book
7. Delete a book
8. Display sales transactions
9. Exists program

If user chose :

“1” it will display all books in library

“2” it will display all books which can be bought or borrow depends on his needs

"3" gives him the ability of borrowing books and fill in the required data

"4" gives him the ability of buying book and fill in the required data

"5" will make him/her search for a book with many opntions like search with (ISBN/Title/Author Name/Category)

"6" add a book to the library which 3 options like (add manually-return a deleted book-increase amount of book in the library)

"7" delete a book from library and saves in deleted file

"8" shows the employee all sales transactions

"9" exists the running program

After the compiler excute the whole main function he write every change in the files again and delete the dynamic array allocation in memory

Employee :

It takes arrbook,arrdlt,arrsal,arrbrr as parameters just to make the employee able to

add a book to the array/search in the array/delete book from the array…etc

in otherwords manege everything in the program

Customer :

It takes arrbook as parameter to search/check which books can be borrowed or bought .. etc

Outlook :

Takes arrbooks & i as parameters … the data of book and its position in the file and display it

Mini\_outlook :

Takes arrbooks & i as parameters .. which refers to the data of books and its position in the file and only display the ISBN and Title of book

Buy\_Borrow\_Both :

Takes arrbooks and check if the book can be bought, borrowed or both and display the details of book

Borrow\_book :

It takes arrbook[] as parameter and check every book in it if it can be borrowed or both and saves its ISBN and number in an array , then it displays the ISBN and Name of books it found to make the user choose which book he wants to borrow as soon as the user enters the ISBN of the book which he wants and it is matched with a book which already in the library … it calls borrow\_transaction function else it prints this book is not for borrow . if there are no book for borrow it will print no books for borrow

Borrow\_Transaction :

It displays book details and asks the user to enter the amount he wants to borrow , then asks the user to enter his information like ID,Name and Period and saves whole data with date and transaction number in a text file

Buy\_Book :

It takes arrbook[] as parameter and check ever book in it if it can be buy or both and saves its ISBN and number in an array , then it displays ISBN and Name of books it found and asks the user to enter the ISBN of the book he wants to buy , if the ISBN matches with any ISBN of the books which can be bought or both it calls buy\_transaction function

Buy\_Transaction :

It displays book details and asks the user to enter the amount he wants to buy then asks for Customer details like ID,name and payment method and if the book amount reached 0 it automatically deletes the book