DBMS-PROJECT

E- LIBRARY MANAGEMENT SYSTEM

Name- Sanidhya Varshney

Section:- C

Roll No./UID:- 65/191500714

E-library system makes the work of a person who is in charge of the library more convenient to search, arrange and make an inventory of the contents of the library.

Data Requirements:-

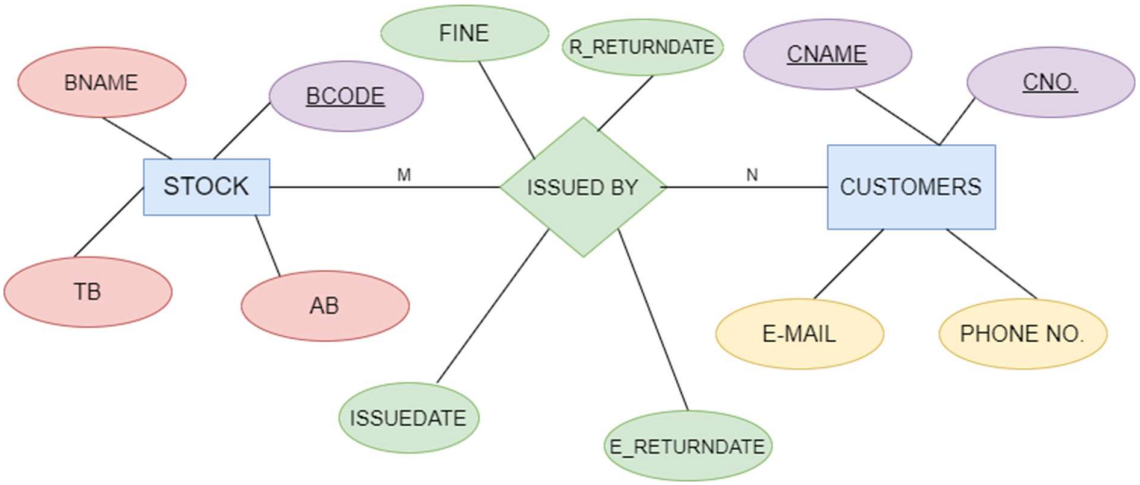
- Book Details:- Book name, Book code, total books, available books.
- Customer details:-name, address, email id ,phone number,etc.
- Each book will have a unique identification number
- There could be more than one copy of a book, and library members should be able to check-out and reserve any copy. We will call each copy of a book, a book item.
- The system should be able to retrieve information like who took a particular book or what are the books checked-out by a specific library member.
- There should be a maximum limit (5) on how many books a member can check-out.
- There should be a maximum limit (10) on how many days a member can keep a book.
- The system should be able to collect fines for books returned after the due date.
- Member: All members can search the catalog, as well as check-out, reserve, renew, and return a book.
- Here are the top use cases of the E-Library Management System.

- Register new account/cancel membership: To add a new member or cancel the membership of an existing member.

- Renew a book: To reborrow an already checked-out book.

- Return a book: To return a book to the library which was issued to a member

ER DIAGRAM OF E-LIBRARY MANAGEMENT SYSTEM



ER DIAGRAM TO TABLES:-

STOCK			
B CODE	**BNAME**	**T B**	**AB**

CUSTOMER			
CNO	**CNAME**	**E-MAIL**	**PHONE NUMBER**

MAINTAIN						
C_NO	**B_CODE**	**CNAME**	**ISSUEDATE**	**E_RDATE**	**A_RDATE**	**FINE**

These tables are working independently except maintain table because maintain table is executing at the runtime. There is no functional dependency between these three tables .Hence there is no need to normalize the tables.

-:IMPLEMENTATION:-

CREATING TABLE “STOCK”:-

```
import sqlite3
try:
   con=sqlite3.connect('p.db')
   create="create table stock(bcode integer primary key,bname text,tb integer,ab
integer);"
   con.execute(create)
   print("table created successfully")
   con.commit()
   con.close()
except sqlite3.Error as error:
   print("Error occurred is",error)
finally:
   if(con):
      con.close()
      print("connection closed")
```

INSERTING TABLE “STOCK”:-

```
import sqlite3
try:
   con=sqlite3.connect('p.db')
   cur=con.cursor()
   q1="""insert into stock values(001,'ENGLISH',10,10);"""
   q2="""insert into stock values(002,'HINDI',20,20);"""
   q3="""insert into stock values(003,'PHYSICS',15,15);"""
   q4="""insert into stock values(004,'BIOLOGY',25,25);"""
   q5="""insert into stock values(005,'CHEMISTRY',10,10);"""
   q6="""insert into stock values(006,'MATH',15,15);"""
   q7="""insert into stock values(007,'SCIENCE',20,20);"""
   q8="""insert into stock values(008,'SST',10,10);"""
   q9="""insert into stock values(009,'ART',50,50);"""
   q10="""insert into stock values(010,'GK',40,40);"""
   cur.execute(q1)
   cur.execute(q2)
   cur.execute(q3)
   cur.execute(q4)
   cur.execute(q5)
   cur.execute(q6)
   cur.execute(q7)
   cur.execute(q8)
   cur.execute(q9)
   cur.execute(q10)
   con.commit()
   cur.close()
```

```
except sqlite3.Error as error:
   print("Error occurred is",error)
finally:
   if(con):
      con.close()
      print("connection closed")
```

CREATING TABLE “CUSTOMERS”:-

```
import sqlite3

try:
   con=sqlite3.connect('p.db')
   create="create table customers(cno integer primary key,cname varchar2(20),phone integer not
null, email varchar2(20) not null);"
   con.execute(create)
   print("table created successfully")
   con.commit()
   con.close()
except sqlite3.Error as error:
   print("Error occurred is",error)
finally:
   if(con):
      con.close()
      print("connection closed")
```

INSERTING TABLE “CUSTOMERS”:-

```
import sqlite3
try:
   con=sqlite3.connect('p.db')
   cur=con.cursor()
   q1="""insert into customers values(1,'Akash',9874561230,'akash@gmail.com');"""
   q2="""insert into customers values(2,'Amit',9875642301,'amit@gmaiil.com');"""
   q3="""insert into customers values(3,'Anurag',9999999999,'anurag@gmail.com');"""
   q4="""insert into customers values(4,'Abhishek',9898989898,'abhishek@gmaiil.com');"""
   q5="""insert into customers values(5,'Sneha',9879879870,'sneha@gmaiil.com');"""
   q6="""insert into customers values(6,'Aman',9869869860,'aman@gmaiil.com');"""
   q7="""insert into customers values(7,'Ashutosh',9859859850,'ashutosh@gmaiil.com');"""
   q8="""insert into customers values(8,'Avinash',9849849840,'avinash@gmaiil.com');"""
   q9="""insert into customers values(9,'Mohini',9839839830,'mohini@gmaiil.com');"""
   q10="""insert into customers values(10,'Savita',8745895620,'savita@gmail.com');"""
   cur.execute(q1)
   cur.execute(q2)
   cur.execute(q3)
   cur.execute(q4)
   cur.execute(q5)
   cur.execute(q6)
   cur.execute(q7)
   cur.execute(q8)
   cur.execute(q9)
   cur.execute(q10)
   con.commit()
   cur.close()
except sqlite3.Error as error:
   print("Error occurred is",error)
finally:
   if(con):
```

```
    con.close()
    print("connection closed")
```

CREATING TABLE “MAINTAIN(USES AT RUNTIME)”:-

```
import sqlite3

try:
  con=sqlite3.connect('p.db')
  create="create table maintain(cno integer,bcode integer,cname text,issuedate date,E_returndate date,R_returndate date,fine integer,foreign key(cno) references customers,foreign key(bcode) references stock)"
  con.execute(create)
  print("table created successfully")
  con.commit()
  con.close()
except sqlite3.Error as error:
  print("Error occurred is",error)
finally:
  if(con):
    con.close()
    print("connection closed")
```

AFTER RUNNING ALL THE CODES SEPARATELY WE WILL GET OUR TABLES:-

1>TABLE STOCK LOOK LIKE THIS:-

bcode	bname	tb	ab
1	ENGLISH	10	10
2	HINDI	20	20
3	PHYSICS	15	15
4	BIOLOGY	25	25
5	CHEMISTRY	10	10
6	MATH	15	15
7	SCIENCE	20	20
8	SST	10	10
9	ART	50	50
10	GK	40	40

2>TABLE CUSTOMERS LOOK LIKE THIS:-

cno	cname	phone	email
1	Akash	9874561230	akash@gmail.com
2	Amit	9875642301	amit@gmaiil.com
3	Anuraq	9999999999	anuraq@gmail.com
4	Abhishek	9898989898	abhishek@gmaiil.com
5	Sneha	9879879870	sneha@gmaiil.com
6	Aman	9869869860	aman@gmaiil.com
7	Ashutosh	9859859850	ashutosh@gmaiil.com
8	Avinash	9849849840	avinash@gmaiil.com
9	Mohini	9839839830	mohini@gmaiil.com
10	Savita	8745895620	savita@gmail.com

MAIN CODE :-

```
from tkinter import *
import sqlite3
from tkinter import messagebox
from datetime import datetime,date,timedelta
root=Tk()
root.geometry("800x680")
root.title("Welcome To Online Library")
l1=Label(root,text="ONLINE LIBRARY SYSTEM",font=('times new roman',
20),fg="blue",bg="pink").pack()
Label(root,text=" ").pack()
l2=Label(root,text="CHOOSE MODE",font=('times new roman', 18),fg="green").pack()
Label(root,text=" ").pack()
conn=sqlite3.connect("p.db")
cur=conn.cursor()
def ddlc2():
    p=Toplevel()
    p.geometry("800x680")
    def Issue(event=None):
        global cname,issuedate,e_returndate
        l1 = []
        l2 = []
        q5 = cur.execute("select cno,cname from customers;")
        for i in q5:
            l1.append(i[0])
            l2.append(i[1])
        dct1 = {l1[i]: l2[i] for i in range(len(l1))}
        l3 = []
        l4 = []
        q7 = cur.execute("select bcode,bname from stock;")
        for i in q7:
            l3.append(i[0])
            l4.append(i[1])
        dct2 = {l3[i]: l4[i] for i in range(len(l3))}
        if number.get()=="" or code.get()=="":
            lbl_text.config(text="Please complete the required field!", fg="red")
        else:
```

```
        if number.get() in dct1 and code.get() in dct2:
            HomeWindow1()
            cname = dct1[number.get()]
            q1 = cur.execute("select ab from stock where bcode=?;", (code.get(),))
            q = """insert into maintain(cno,bcode,cname,issuedate,e_returndate)
values(?,?,?,?,?);"""
            for i in q1:
                if i[0] > 0:
                    cur.execute("update stock set ab=ab-1 where bcode=?;", (code.get(),))
                    issuedate = date.today()
                    e_returndate = date.today() + timedelta(12)
                    datatuple = (number.get(), code.get(), cname, issuedate, e_returndate)
                    cur.execute(q, datatuple)
                    conn.commit()
                else:
                    lbl_text.config(text="No such user",fg="blue")
                    number.set("")
                    code.set("")
    def HomeWindow1():
        global Home1
        root.withdraw()
        Home1 = Toplevel()
        Home1.title("ONLINE LIBRARY SYSTEM")
        width = 800
        height = 680
        screen_width = root.winfo_screenwidth()
        screen_height = root.winfo_screenheight()
        x = (screen_width / 2) - (width / 2)
        y = (screen_height / 2) - (height / 2)
        root.resizable(0, 0)
        Home1.geometry("%dx%d+%d+%d" % (width, height, x, y))
        lbl_home=Label(Home1,text="Book Issued Succesfully",font=('times new roman',
18),fg="green").pack()
        Label(Home1,text=" ").pack()
        Label(Home1,text=" ").pack()
        def show():
            hm=Tk()
            hm.geometry("400x500")
            r_set=cur.execute('''SELECT cno,bcode,cname,issuedate,e_returndate from maintain
where r_returndate is null''')
            i=0
            for student in r_set:
              for j in range(len(student)):
                e = Entry(hm, width=10, fg='blue')
                e.grid(row=i, column=j)
                e.insert(END, student[j])
              i=i+1
            hm.mainloop()
        bt=Button(Home1,text="Show Transaction",font=('times new roman',
12),height=1,width=20,bg="yellow",command=show).pack()

    def Back1():
        Home1.destroy()
        root.deiconify()
```

```
    number=IntVar()
    code=IntVar()
    Top = Frame(p, bd=2, relief=RIDGE)
    Top.pack(side=TOP, fill=X)
    Form = Frame(p, height=200)
    Form.pack(side=TOP, pady=20)

    lbl_title = Label(Top, text="ONLINE LIBRARY SYSTEM", font=('arial', 15))
    lbl_title.pack(fill=X)
    lbl_serial = Label(Form, text="USER NO.", font=('arial', 14), bd=15)
    lbl_serial.grid(row=0, sticky="e")
    lbl_code = Label(Form, text="BOOK CODE.", font=('arial', 14), bd=15)
    lbl_code.grid(row=1, sticky="e")
    lbl_text = Label(Form)
    lbl_text.grid(row=2, columnspan=2)
    n=Entry(Form, textvariable=number, font=(14))
    n.grid(row=0, column=1)
    c=Entry(Form, textvariable=code, font=(14))
    c.grid(row=1, column=1)
    btn_issue=Button(Form, text="Next", width=45, command=Issue)
    btn_issue.grid(pady=25, row=3, columnspan=2)
    btn_issue.bind('<Return>', Issue)
def ddlc3():
    y=Toplevel()
    y.geometry("800x680")
    def Return(event=None):
        l1 = []
        l2 = []
        q5 = cur.execute("select cno,cname from customers;")
        for i in q5:
            l1.append(i[0])
            l2.append(i[1])
        dct1 = {l1[i]: l2[i] for i in range(len(l1))}
        l3 = []
        l4 = []
        q7 = cur.execute("select bcode,bname from stock;")
        for i in q7:
            l3.append(i[0])
            l4.append(i[1])
        dct2 = {l3[i]: l4[i] for i in range(len(l3))}
        if number.get()=="" or code.get()=="" or r_returndate.get()=="":
            lbl_text.config(text="Please complete the required field!", fg="red")
        else:
            if number.get() in dct1 and code.get() in dct2:
                HomeWindow()
            cname = dct1[number.get()]
            q1 = cur.execute("select ab from stock where bcode=?;", (code.get(),))
            q="""insert into
maintain(cno,bcode,cname,issuedate,e_returndate,r_returndate,fine)
values(?,?,?,?,?,?,?);"""
            for i in q1:
                cur.execute("update stock set ab=ab+1 where bcode=?;", (code.get(),))
                q3 = cur.execute("select issuedate from maintain where cno=? and bcode=?;",
(number.get(), code.get()))
```

```
            issuedate = ""
            for i in q3:
                issuedate = i[0]

            q4 = cur.execute("select e_returndate from maintain where cno=? and bcode=?;",
(number.get(), code.get()))
            e_returndate = ""
            for i in q4:
                e_returndate = i[0]
            year1, month1, dt1 = map(int, r_returndate.get().split("-"))
            f_date = date(year1, month1, dt1)
            year2, month2, dt2 = map(int, e_returndate.split("-"))
            l_date = date(year2, month2, dt2)
            d = f_date - l_date
            if d.days <= 0:
                v=Tk()
                v.geometry("200x200")
                fine = 0
                messagebox.showinfo("Your Fine Amount",fine)
                v.mainloop()
                v,destroy()
            else:
                v=Tk()
                v.geometry("200x200")
                fine = d.days * 10
                messagebox.showinfo("Your Fine Amount",fine)
                v.mainloop()
                v.destroy()

        datatuple = (number.get(), code.get(), cname, issuedate, e_returndate, r_returndate.get(),
fine)
        cur.execute(q, datatuple)
        conn.commit()
    def HomeWindow():
        global Home
        root.withdraw()
        Home = Toplevel()
        Home.title("ONLINE LIBRARY SYSTEM")
        width = 600
        height = 500
        screen_width = root.winfo_screenwidth()
        screen_height = root.winfo_screenheight()
        x = (screen_width / 2) - (width / 2)
        y = (screen_height / 2) - (height / 2)
        root.resizable(0, 0)
        Home.geometry("%dx%d+%d+%d" % (width, height, x, y))
        lbl_home=Label(Home,text="Book Return Succesfully",font=('times new roman',
18),fg="green").pack()

    number = IntVar()
    code = IntVar()
    r_returndate=StringVar()
    Top = Frame(y, bd=2, relief=RIDGE)
    Top.pack(side=TOP, fill=X)
```

```
    Form = Frame(y, height=200)
    Form.pack(side=TOP, pady=20)

    lbl_title = Label(Top, text="ONLINE LIBRARY SYSTEM", font=('arial', 15))
    lbl_title.pack(fill=X)
    lbl_serial = Label(Form, text="USER NO.", font=('arial', 14), bd=15)
    lbl_serial.grid(row=0, sticky="e")
    lbl_code = Label(Form, text="BOOK CODE.", font=('arial', 14), bd=15)
    lbl_code.grid(row=1, sticky="e")
    lbl_date = Label(Form, text="RETURN DATE", font=('arial', 14), bd=15)
    lbl_date.grid(row=2, sticky="e")
    lbl_text = Label(Form)
    lbl_text.grid(row=3, columnspan=3)
    n = Entry(Form, textvariable=number, font=(14))
    n.grid(row=0, column=1)
    c = Entry(Form, textvariable=code, font=(14))
    c.grid(row=1, column=1)
    d=Entry(Form,textvariable=r_returndate,font=(14))
    d.grid(row=2,column=1)
    btn_issue = Button(Form, text="Next", width=45, command=Return)
    btn_issue.grid(pady=25, row=3, columnspan=2)
    btn_issue.bind('<Return>', Return)
def ddlc1():
  t=Toplevel()
  t.geometry("800x680")
  def Login(event=None):


    if email.get()=="" or cname.get()=="":
      lbl_text.config(text="Please complete the required field!", fg="red")

    else:
      cur.execute("select * from customers where email=? and
cname=?",(email.get(),cname.get()))
      if cur.fetchone() is not None:
        HomeWindow()
        email.set("")
        cname.set("")
        lbl_text.config(text="")
      else:
        lbl_text.config(text="Invalid username or password", fg="red")
        email.set("")
        cname.set("")
  def HomeWindow():
    global Home
    root.withdraw()
    Home = Toplevel()
    Home.title("ONLINE LIBRARY SYSTEM")
    width = 800
    height = 680
    screen_width = root.winfo_screenwidth()
    screen_height = root.winfo_screenheight()
    x = (screen_width / 2) - (width / 2)
    y = (screen_height / 2) - (height / 2)
```

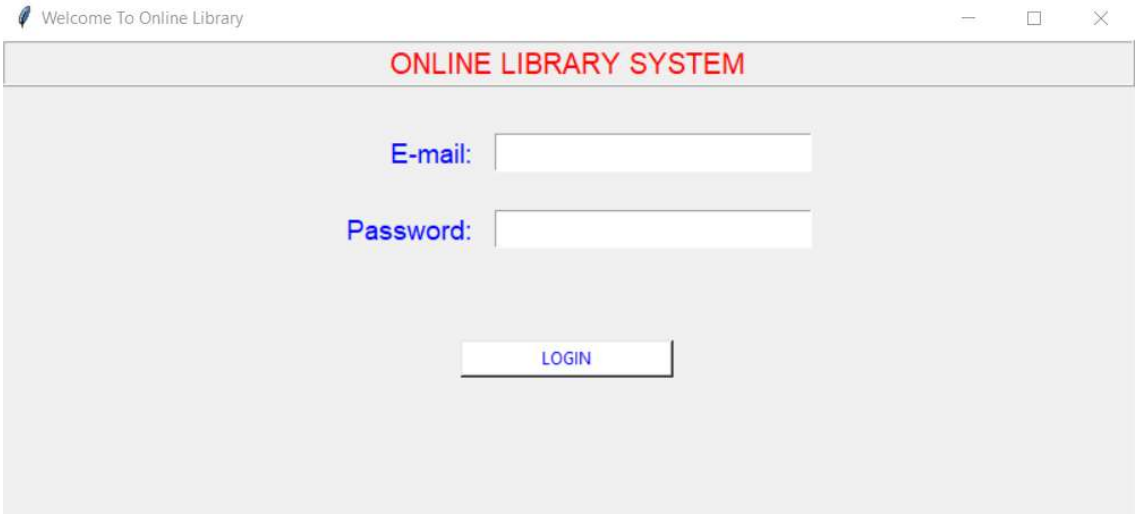
```
    root.resizable(0, 0)
    Home.geometry("%dx%d+%d+%d" % (width, height, x, y))
    Label(Home,text=" ").pack()
    def showcatalog():
        hm1=Tk()
        hm1.geometry("400x500")
        r_set=cur.execute("'SELECT * from stock'");
        i=0
        for student in r_set:
            for j in range(len(student)):
              e = Entry(hm1, width=10, fg='blue')
              e.grid(row=i, column=j)
              e.insert(END, student[j])
            i=i+1
        hm1.mainloop()

    b1=Button(Home,text="SHOW CATALOG",font=('times new roman',
12),height=2,width=20,bg="yellow",command=showcatalog).pack()
    Label(Home,text=" ").pack()
    b1=Button(Home,text="ISSUE",font=('times new roman',
12),height=2,width=20,bg="yellow",command=ddlc2).pack()
    Label(Home,text=" ").pack()
    b2=Button(Home,text="RETURN",font=('times new roman',
12),height=2,width=20,bg="orange",command=ddlc3).pack()



  email= StringVar()
  cname = StringVar()

  Top = Frame(t, bd=2, relief=RIDGE)
  Top.pack(side=TOP, fill=X)
  Form = Frame(t, height=200)
  Form.pack(side=TOP, pady=20)

  lbl_title = Label(Top, text="ONLINE LIBRARY SYSTEM", font=('arial', 15),fg="red")
  lbl_title.pack(fill=X)
  lbl_email = Label(Form, text="E-mail:", font=('arial', 14),fg="blue", bd=15)
  lbl_email.grid(row=0, sticky="e")
  lbl_cname = Label(Form, text="Password:", font=('arial', 14),fg="blue",bd=15)
  lbl_cname.grid(row=1, sticky="e")
  lbl_text = Label(Form)
  lbl_text.grid(row=2, columnspan=2)

  username = Entry(Form, textvariable=email, font=(14))
  username.grid(row=0, column=1)
  password = Entry(Form, textvariable=cname, show="*", font=(14))
  password.grid(row=1, column=1)

  btn_login = Button(Form, text="LOGIN", width=20,fg="blue",bg="white", command=Login)
  btn_login.grid(pady=30, row=3, columnspan=2)
  btn_login.bind('<Return>', Login)
def ddlc():
 f=Toplevel()
```

```
 f.geometry("800x680")
 def resgistration(event=None):
   if cname.get()=="" or email.get()=="":
     lbl_text.config(text="Please complete the required field!", fg="red")
   else:
     cur.execute("insert into customers(cno,cname,phone,email)
values(?,?,?,?)",(cno.get(),cname.get(),phone.get(),email.get()))
     homewindow()
     cno.set(0)
     cname.set("")
     phone.set(0)
     email.set("")
   conn.commit()
 def homewindow():
  global Home
  root.withdraw()
  Home = Toplevel()
  Home.title("ONLINE LIBRARY SYSTEM")
  width = 800
  height = 680
  screen_width = root.winfo_screenwidth()
  screen_height = root.winfo_screenheight()
  x = (screen_width / 2) - (width / 2)
  y = (screen_height / 2) - (height / 2)
  root.resizable(0, 0)
  Home.geometry("%dx%d+%d+%d" % (width, height, x, y))
  lbl_home = Label(Home, text="Successfully Registration!", font=('times new roman',
20)).pack()
  btn_back = Button(Home, text='Back', command=Back).pack(pady=20, fill=X)
 def Back():
  Home.destroy()
  root.deiconify()


 cno=IntVar()
 cname=StringVar()
 phone=IntVar()
 email=StringVar()

 Top = Frame(f, bd=2, relief=RIDGE)
 Top.pack(side=TOP, fill=X)
 Form = Frame(f, height=200)
 Form.pack(side=TOP, pady=20)

 lbl_title = Label(Top, text="ONLINE LIBRARY SYSTEM", font=('arial', 15))
 lbl_title.pack(fill=X)
 lbl_username = Label(Form, text="USER NO.:", font=('arial', 14),fg="blue", bd=15)
 lbl_username.grid(row=0, sticky="e")
 lbl_password = Label(Form, text="USERNAME:", font=('arial', 14),fg="blue", bd=15)
 lbl_password.grid(row=1, sticky="e")
 lbl_phone = Label(Form, text="PHONE.:", font=('arial', 14),fg="blue", bd=15)
 lbl_phone.grid(row=2, sticky="e")
 lbl_email = Label(Form, text="E-MAIL:", font=('arial', 14),fg="blue",bd=15)
 lbl_email.grid(row=3, sticky="e")
```

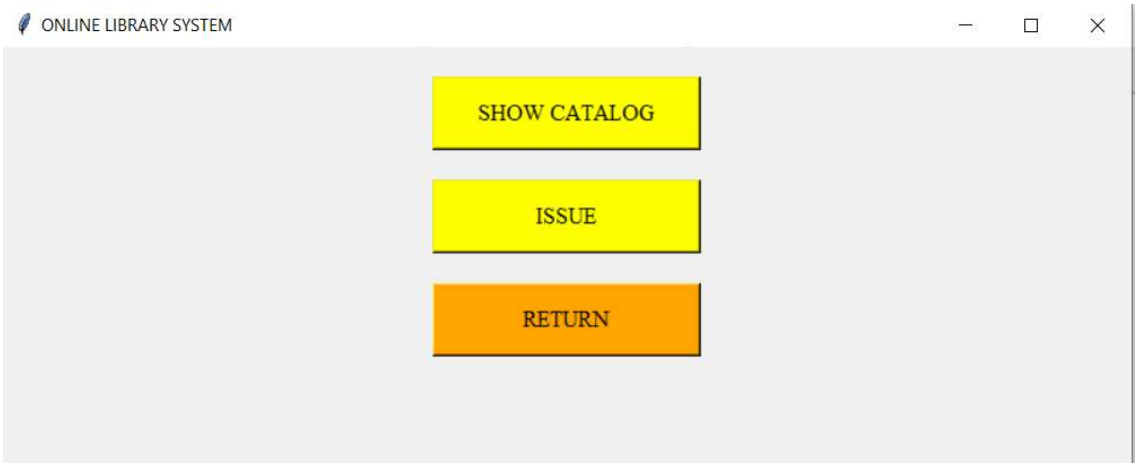
```
 lbl_text = Label(Form)
 lbl_text.grid(row=4, columnspan=4)

 serial=Entry(Form, textvariable=cno, font=(14))
 serial.grid(row=0,column=1)
 username = Entry(Form, textvariable=cname, font=(14))
 username.grid(row=1, column=1)
 mobile = Entry(Form, textvariable=phone, font=(14))
 mobile.grid(row=2, column=1)
 mail = Entry(Form, textvariable=email, font=(14))
 mail.grid(row=3, column=1)

 btn_login = Button(Form, text="REGISTER", width=25,fg="green", command=resgistration)
 btn_login.grid(pady=25, row=4, columnspan=3)
 btn_login.bind('<Return>', resgistration)
Label(root,text=" ").pack()
b1=Button(root,text="OLD USER",font=('times new
roman',12),height=1,width=20,fg="red",bg="yellow",command=ddlc1).pack()
Label(root,text=" ").pack()
b2=Button(root,text="NEW USER",font=('times new roman',
12),height=1,width=20,bg="yellow",command=ddlc).pack()
root.mainloop()
```

After running the main code we will get the interface like:-

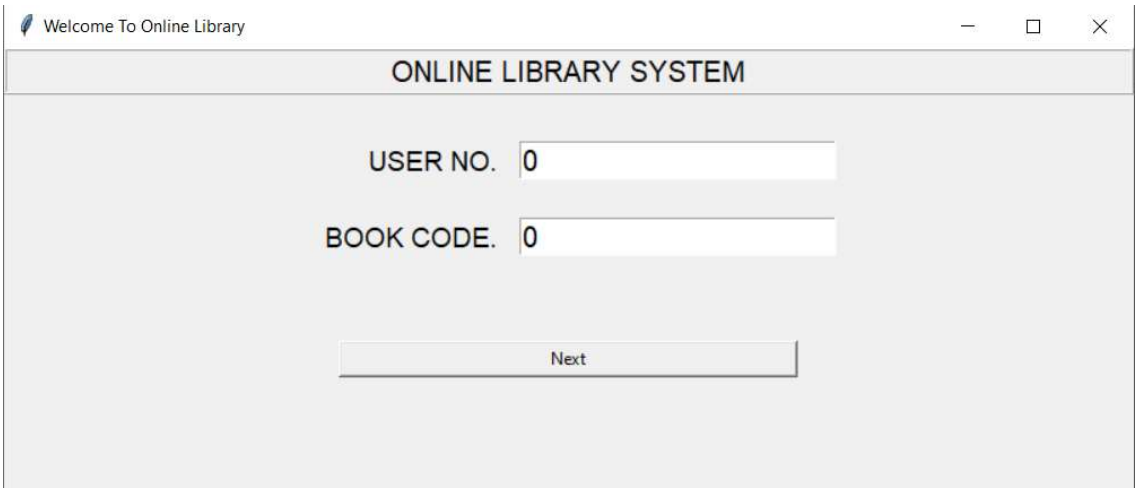
Choose one mode among the following two:
I am using “OLD USER” so just press the button and you will be redirected to the following page:-



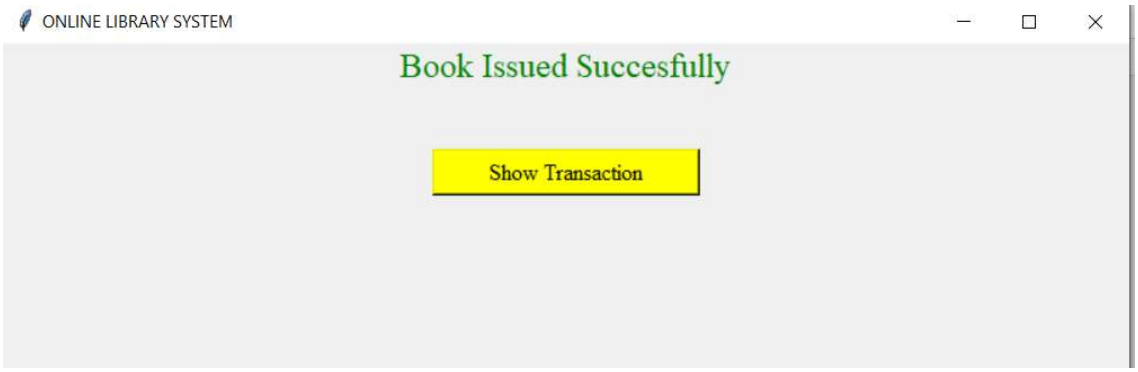
In this you have to fill your email and password. Remember that email and password using from customer table.Example:-
Email:-akash@gmail.com
Password:-Akash
Then:-



Three options given in the above window. Choose one at a time.
Show Catalog button:- works to show the table stock from this user fills the details in issue window after this. Press Issue button then:-

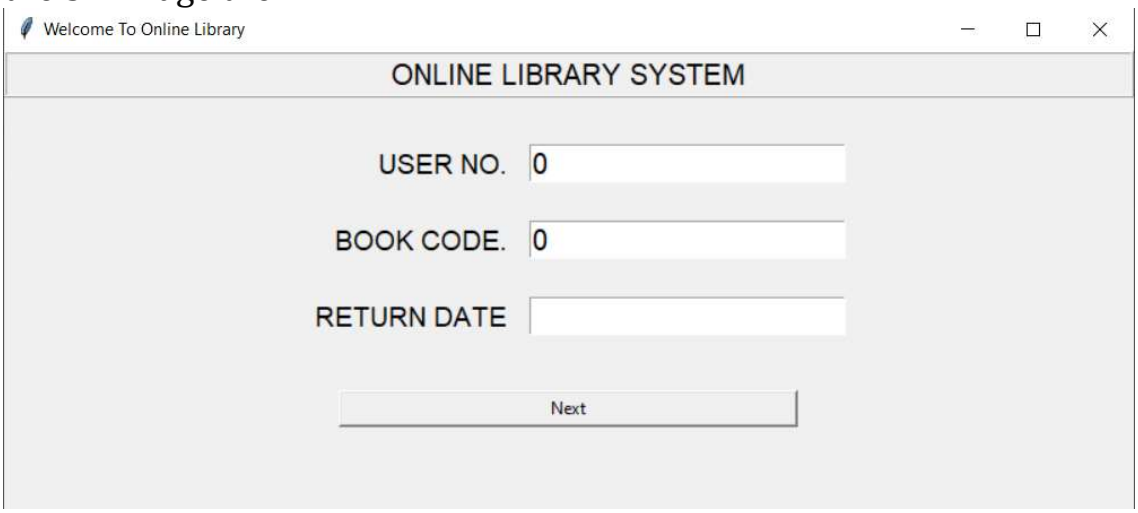


Now fill the user no in the login window we fill the email and name of ‘Akash’ so from customers table, user no. of akash is 1 and for book code Show catalog helps to show which book is available. I am selecting book code of English i.e. 1. After filling all the details press next button then message appears on new window:-

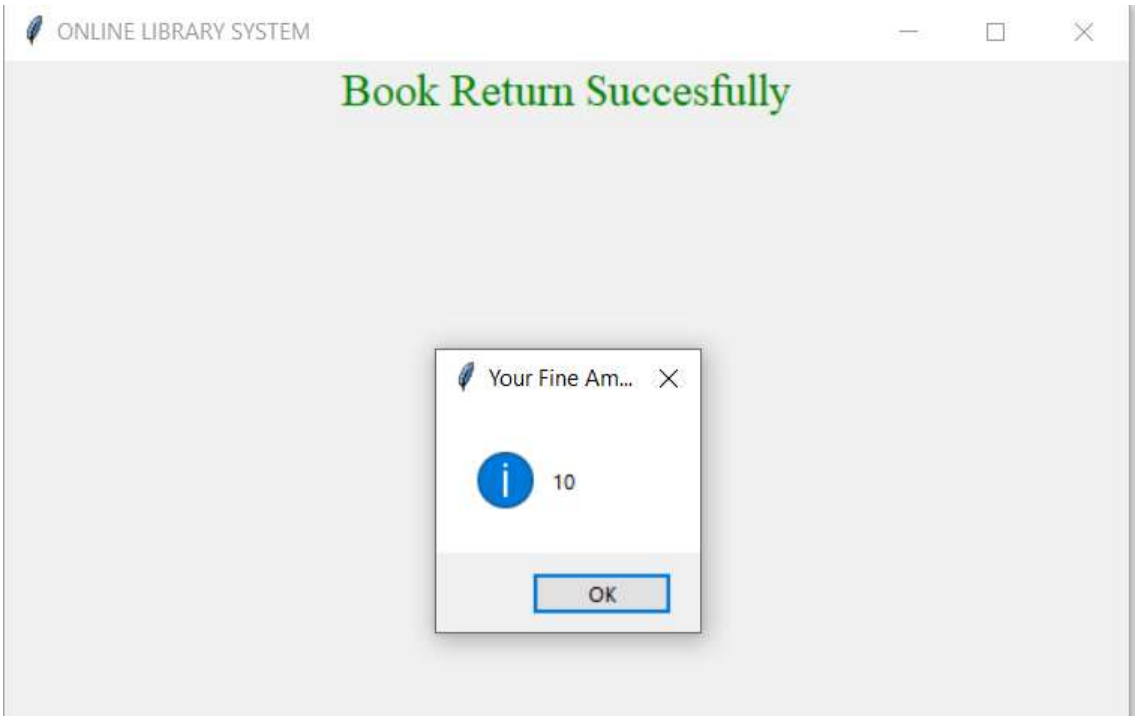


On Pressing Show Transaction button you will get your estimated return date in the fifth column.Memorize that date and return the book on time otherwise pay the fine amount:-

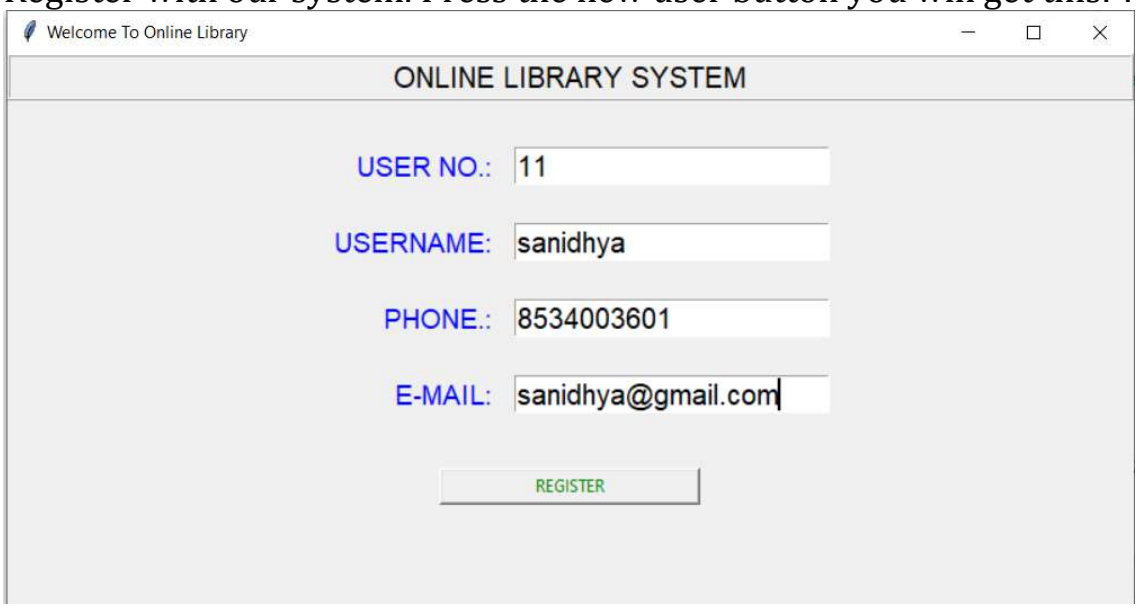
If you want to return then press the return button as you can see in the 3rd image then:-



Return date format:-2020-12-25.If you return the book on time then there is no fine amount otherwise a message box will appear to show your fine amount. I filled the returndate 2020-12-25 then message box show you have a fine amount of 10rs.:-



Now I want to tell you the procedure of new user. If someone wants to Register with our system. Press the new user button you will get this:-.



Then click on register button:-

Then press back button you will redirected to our main page and you will we our new member if you want to issue a book then press old user button and proceed the same from the above