7. CODE LISTING

```
from tkinter import *
                                                             mycon=mysql.connector.connect(host=hostnm,
import tkinter.messagebox
                                                                                   user=user,
from datetime import datetime
                                                                                   passwd=passwd,
from datetime import timedelta
                                                                                   database=dbnm)
import webbrowser
                                                                except:
import time
                                                                   entall=Label(errorstreams, text='Invalid
import random
                                                            Credentials',fg='red')
import smtplib
                                                                   entall.pack()
import pickle
                                                                   return
import mysql.connector
                                                                detailsofpersonframe.pack_forget()
                                                                mainlbl.pack forget()
                                                                successmsg=Label(root1,text='Database Account
def createdbacc():
                                                            Verified', font= "timesnewroman 20")
  def Quitsys():
                                                                successmsg.pack()
    root1.destroy()
     quit()
  def submitsuggestion():
                                                                buttonsframe=Frame(root1)
                                                                buttonsframe.pack()
    def regandlog():
                                                                backbutton = Button(buttonsframe, text="Register
                                                            and Login", command=regandlog)
                                                                backbutton.pack()
databaseaccountfile=open('./STORAGE/databaseaccou
nts.txt','rb')
       try:
         listofallusers=pickle.load(databaseaccountfile)
       except:
         listofallusers=[]
       databaseaccountfile.close()
                                                              root1=Tk()
                                                              mainIbl=Label(root1,text='Database Error\n\nPlease
listofallusers.append((hostnm,user,passwd,dbnm))
                                                            Restart System or Register Database
                                                            account\n\n\n',font= "timesnewroman 20")
                                                              mainlbl.pack()
databaseaccountfile=open('./STORAGE/databaseaccou
                                                              detailsofpersonframe=Frame(root1)
nts.txt','wb')
                                                              detailsofpersonframe.pack()
listofallusers=pickle.dump(listofallusers,databaseaccoun
                                                              #name
tfile)
                                                              nameframe=Frame(detailsofpersonframe)
       databaseaccountfile.close()
                                                              nameframe.pack()
                                                              nameofuserlabel=Label(nameframe,text='Host Name
                                                              nameofuserentry=Entry(nameframe,width=20)
       root1.destroy()
                                                              nameofuserlabel.pack(side=LEFT)
                                                              nameofuserentry.pack(side=LEFT)
    global errorstreams
       errorstreams.pack forget()
                                                              #nric
                                                              nricofpersonframe=Frame(detailsofpersonframe)
    except:
                                                              nricofpersonframe.pack()
       pass
                                                              nricofuserlabel=Label(nricofpersonframe,text='User
    hostnm=nameofuserentry.get()
                                                           Name\t\t')
    user=nricofuserentry.get()
                                                              nricofuserentry=Entry(nricofpersonframe,width=20)
    passwd=emailofuserentry.get()
                                                              nricofuserlabel.pack(side=LEFT)
    dbnm=databaseentry.get()
                                                              nricofuserentry.pack(side=LEFT)
    errorstreams=Frame(root1)
                                                              #email
    errorstreams.pack()
                                                              emailofpersonframe=Frame(detailsofpersonframe)
    if len(hostnm)==0 or len(user)==0 or
                                                              emailofpersonframe.pack()
len(passwd)==0 or len(dbnm)==0:
       entall=Label(errorstreams, text='Please Enter all
                                                            emailofuserlabel=Label(emailofpersonframe,text='Pass
Details',fq='red')
                                                            word\t\t')
       entall.pack()
                                                            emailofuserentry=Entry(emailofpersonframe,width=20,s
       return
    try:
                                                           how='*')
```

emailofuserlabel.pack(side=LEFT)

global mycon

```
emailofuserentry.pack(side=LEFT)
                                                              langoptions. pack forget()
                                                              mainpage()
  #email
  databsenameframe=Frame(detailsofpersonframe)
  databsenameframe.pack()
                                                            def suggestionstomain():
                                                              titleoffeedbackpage. pack forget()
Databaselabel=Label(databsenameframe,text='Databas
                                                              detailsofpersonframe. pack forget()
e Name \t')
                                                              suggestiongivingframe. pack forget()
  databaseentry=Entry(databsenameframe,width=20)
                                                              mainpage()
  Databaselabel.pack(side=LEFT)
  databaseentry.pack(side=LEFT)
                                                            def loginpagetomain():
                                                              global invalidpasswordmessage
  buttonsframe=Frame(detailsofpersonframe)
                                                              global invalidcaptchacodeerror
  buttonsframe.pack(side=BOTTOM)
                                                              invalidpasswordmessage=0#this is for giving error
                                                            message in login page
                                                              invalidcaptchacodeerror=0#this is for giving error
  submitbutton= Button(buttonsframe, text="Register
                                                            message in login page
Database", command=submitsuggestion)
  submitbutton.pack()
                                                              frameloginbuttons. pack forget()
                                                              canvasloginouter. pack forget()
  backbutton = Button(buttonsframe, text="Quit",
                                                              captchaframe, pack forget()
command=Quitsvs)
                                                              try:
  backbutton.pack()
                                                                frameerrors. pack forget()
                                                              except:
  root1.mainloop()
                                                                 pass
                                                              mainpage()
                                                            def createaccountfromloginpage():
                                                              global invalidpasswordmessage
databaseaccountfile=open('./STORAGE/databaseaccou
                                                              global invalidcaptchacodeerror
nts.txt','rb')
                                                              frameloginbuttons. pack forget()
try:
  listofallusers=pickle.load(databaseaccountfile)
                                                              canvasloginouter. pack_forget()
                                                              captchaframe. pack forget()
except:
  listofallusers=[]
                                                              try:
databaseaccountfile.close()
                                                                frameerrors. pack_forget()
                                                              except:
for i in listofallusers:
                                                                 pass
  try:
                                                              createaccountpage()
     mycon=mysql.connector.connect(host=i[0],
                       user=i[1],
                                                            def searchtomain():
                       passwd=i[2],
                                                              titleofsearchbookpage. pack forget()
                       database=i[3])
                                                              keywordframe. pack forget()
                                                              resultframe. pack forget()
    break
                                                              backframe.pack forget()
  except:
                                                              if currentlocationforserachbook == 'LIBRARIAN':
     pass
                                                                 librarianspage()
else:
                                                              elif currentlocationforserachbook != 'GENERAL':
  createdbacc()
cursor=mycon.cursor()
                                                            userpage(currentlocationforserachbookwlcmmsgusefulfo
                                                            ruseraccount,
                                                                      currentlocationforserachbook,
#destorys everything in the page
#whenever a new frame or canvas is created, craete a
                                                            currentlocationforserachbookwlcmmsgusefulforuseracco
new function to delete everthing in it
                                                            untusername,
def destmainpageto():
                                                                      currentlocationforserachbookuserlibraryid)
  frame. pack forget()
                                                              try:
  canvas. pack_forget() #this is actually a frame
                                                                mainpage()
  spaceframe1. pack_forget()
                                                              except:
  loginframe. pack forget()
                                                                pass
  buttonsframe. pack forget()
                                                            def ebookpagetomain():
  aboutusframe. pack forget()
  spaceframe2. pack forget()
                                                              titleofebookpage.pack forget()
                                                              ebooksframetop, pack forget()
def aboutuspagetomain():
                                                              ebooksframebottom. pack forget()
  imageandback.pack_forget()
                                                              backbuttonframe. pack forget()
  contenttext. pack_forget()
                                                              mainpage()
```

```
ebooksframetop=Frame(root)
def ebookpage():
  destmainpageto()
                                                            ebooksframetop.pack()
  global titleofebookpage
                                                            ebooksframebottom=Frame(root)
  global ebooksframetop
                                                            ebooksframebottom.pack()
  global ebooksframebottom
  global backbuttonframe
  def openbook1():
                                                            book1=Frame(ebooksframetop)
                                                            book1.pack(side=LEFT)
blackbeautybook='https://manybooks.net/book/126828/r
ead#epubcfi(/6/2[titlepage]!/4/1:0)'
                                                          ebook1pic=PhotoImage(file='./IMAGES/ebookspictures/
    webbrowser.open(blackbeautybook, new=2)
                                                          ebook1.png')
  def openbook2():
                                                            openbutton1=Button(book1,text='Open', image =
                                                          ebook1pic,command=openbook1)
webbrowser.open('https://manybooks.net/book/127828/r
                                                            openbutton1.pack()
ead#epubcfi(/6/2[item3]!/4/2/1:0)', new=2)
                                                            ebooklabel1=Label(book1,text='Black Beauty',font=
  def openbook3():
                                                          "timesnewroman 20")
                                                            ebooklabel1.pack()
webbrowser.open('https://manybooks.net/book/135651/r
ead#epubcfi(/6/2[item3]!/4/2/2/2[pgheader]/2/2/1:0)',
                                                            spaceframe1=Frame(ebooksframetop)
                                                            spaceframe1.pack(side= LEFT)
new=2)
                                                            spacelabel1=Label(spaceframe1,text='
  def openbook4():
                                                                                                       ')
                                                            spacelabel1.pack()
webbrowser.open('https://manybooks.net/book/148396/r
ead#epubcfi(/6/2[coverpage-wrapper]!/4/1:0)', new=2)
                                                            book2=Frame(ebooksframetop)
  def openbook5():
                                                            book2.pack(side=LEFT)
webbrowser.open('https://manybooks.net/book/127190/r
                                                          ebook2pic=PhotoImage(file='./IMAGES/ebookspictures/
ead#epubcfi(/6/2[titlepage]!/4/1:0)', new=2)
                                                          ebook2.png')
  def openbook6():
                                                            openbutton2=Button(book2,text='Open', image =
                                                          ebook2pic,command=openbook2)
webbrowser.open('https://manybooks.net/book/127836/r
                                                            openbutton2.pack()
ead#epubcfi(/6/2[item4]!/4/2/1:0)', new=2)
                                                            ebooklabel2=Label(book2,text='The Invisible
  def openbook7():
                                                          Man',font= "timesnewroman 20")
                                                            ebooklabel2.pack()
webbrowser.open('https://manybooks.net/book/127829/r
ead#epubcfi(/6/2[titlepage]!/4/1:0)', new=2)
                                                            spaceframe2=Frame(ebooksframetop)
  def openbook8():
                                                            spaceframe2.pack(side= LEFT)
                                                            spacelabel2=Label(spaceframe2,text='
                                                                                                       ')
webbrowser.open('https://manybooks.net/book/139614/r
                                                            spacelabel2.pack()
ead#epubcfi(/6/2[coverpage-wrapper]!/4/1:0)', new=2)
  def openbook9():
                                                            book3=Frame(ebooksframetop)
                                                            book3.pack(side=LEFT)
webbrowser.open('https://manybooks.net/book/141234/r
ead#epubcfi(/6/2[item4]!/4/2/1:0)', new=2)
                                                          ebook3pic=PhotoImage(file='./IMAGES/ebookspictures/
  def openbook10():
                                                          ebook3.png')
                                                            openbutton3=Button(book3,text='Open', image =
webbrowser.open('https://manybooks.net/book/127534/r
                                                          ebook3pic,command=openbook3)
ead#epubcfi(/6/2[coverpage-wrapper]!/4/1:0)', new=2)
                                                            openbutton3.pack()
  def openbook11():
                                                            ebooklabel3=Label(book3,text='Beauty and the
                                                          Beast',font= "timesnewroman 20")
webbrowser.open('https://manybooks.net/book/131121/r
                                                            ebooklabel3.pack()
ead#epubcfi(/6/2[item4]!/4/2/1:0)', new=2)
  def openbook12():
                                                            spaceframe3=Frame(ebooksframetop)
                                                            spaceframe3.pack(side= LEFT)
webbrowser.open('https://manybooks.net/book/122925/r
                                                            spacelabel3=Label(spaceframe3,text='
                                                                                                       ')
ead#epubcfi(/6/2[id00000]!/4/2[id00000]/1:0)', new=2)
                                                            spacelabel3.pack()
                                                            book4=Frame(ebooksframetop)
                                                            book4.pack(side=LEFT)
  titleofebookpage=Frame(root)
  titleofebookpage.pack()
                                                          ebook4pic=PhotoImage(file='./IMAGES/ebookspictures/
  ebooktext = Label(titleofebookpage, text = 'E
                                                          ebook4.png')
```

ebooktext.pack(side= TOP)

Books\n\n', fg='blue',font= "applecherry 35")

```
openbutton4=Button(book4,text='Open', image =
                                                           book8.pack(side=LEFT)
ebook4pic,command=openbook4)
  openbutton4.pack()
                                                        ebook8pic=PhotoImage(file='./IMAGES/ebookspictures/
  ebooklabel4=Label(book4,text='A Pair of
                                                        ebook8.png')
Schoolgirls',font= "timesnewroman 20")
                                                           openbutton8=Button(book8,text='Open', image =
  ebooklabel4.pack()
                                                        ebook8pic,command=openbook8)
                                                           openbutton8.pack()
                                                           ebooklabel8=Label(book8,text='The Aliens\n',font=
  spaceframe4=Frame(ebooksframetop)
  spaceframe4.pack(side= LEFT)
                                                         "timesnewroman 20")
  spacelabel4=Label(spaceframe4,text='
                                            ')
                                                           ebooklabel8.pack()
  spacelabel4.pack()
                                                           spaceframe8=Frame(ebooksframebottom)
  book5=Frame(ebooksframetop)
                                                           spaceframe8.pack(side= LEFT)
  book5.pack(side=LEFT)
                                                           spacelabel8=Label(spaceframe8,text='
                                                                                                    ')
                                                           spacelabel8.pack()
ebook5pic=PhotoImage(file='./IMAGES/ebookspictures/
                                                           book9=Frame(ebooksframebottom)
ebook5.png')
  openbutton5=Button(book5,text='Open', image =
                                                           book9.pack(side=LEFT)
ebook5pic,command=openbook5)
  openbutton5.pack()
                                                        ebook9pic=PhotoImage(file='./IMAGES/ebookspictures/
  ebooklabel5=Label(book5,text='Treasure Island',font=
                                                        ebook9.png')
"timesnewroman 20")
                                                           openbutton9=Button(book9,text='Open', image =
  ebooklabel5.pack()
                                                        ebook9pic,command=openbook9)
                                                           openbutton9.pack()
  spaceframe5=Frame(ebooksframetop)
                                                           ebooklabel9=Label(book9,text='The Gods of
  spaceframe5.pack(side= LEFT)
                                                        Mars\n',font= "timesnewroman 20")
  spacelabel5=Label(spaceframe5,text='
                                            ')
                                                           ebooklabel9.pack()
  spacelabel5.pack()
                                                           spaceframe9=Frame(ebooksframebottom)
  book6=Frame(ebooksframetop)
                                                           spaceframe9.pack(side= LEFT)
  book6.pack(side=LEFT)
                                                           spacelabel9=Label(spaceframe9,text='
                                                                                                    ')
                                                           spacelabel9.pack()
ebook6pic=PhotoImage(file='./IMAGES/ebookspictures/
ebook6.png')
                                                           book10=Frame(ebooksframebottom)
  openbutton6=Button(book6,text='Open', image =
                                                           book10.pack(side=LEFT)
ebook6pic.command=openbook6)
  openbutton6.pack()
                                                        ebook10pic=PhotoImage(file='./IMAGES/ebookspictures
  ebooklabel6=Label(book6,text='The Time
                                                        /ebook10.png')
Machine', font= "timesnewroman 20")
                                                           openbutton10=Button(book10,text='Open', image =
  ebooklabel6.pack()
                                                        ebook10pic,command=openbook10)
                                                           openbutton10.pack()
  spaceframe6=Frame(ebooksframetop)
                                                           ebooklabel10=Label(book10.text='The Adventures of
  spaceframe6.pack(side= LEFT)
                                                        \nTom Sawyer',font= "timesnewroman 20")
  spacelabel6=Label(spaceframe6,text='
                                                           ebooklabel10.pack()
                                           ')
  spacelabel6.pack()
                                                           spaceframe10=Frame(ebooksframebottom)
                                                           spaceframe10.pack(side= LEFT)
                                                           spacelabel10=Label(spaceframe10,text='
  book7=Frame(ebooksframebottom)
                                                                                                      ')
  book7.pack(side=LEFT)
                                                           spacelabel10.pack()
ebook7pic=PhotoImage(file='./IMAGES/ebookspictures/
                                                           book11=Frame(ebooksframebottom)
ebook7.png')
                                                           book11.pack(side=LEFT)
  openbutton7=Button(book7,text='Open', image =
ebook7pic,command=openbook7)
                                                        ebook11pic=PhotoImage(file='./IMAGES/ebookspictures
                                                        /ebook11.png')
  openbutton7.pack()
                                                           openbutton11=Button(book11,text='Open', image =
  ebooklabel7=Label(book7,text='The Island of
\nDoctor Moreau',font= "timesnewroman 20")
                                                        ebook11pic,command=openbook11)
  ebooklabel7.pack()
                                                           openbutton11.pack()
                                                           ebooklabel11=Label(book11,text='A Journey to the
  spaceframe7=Frame(ebooksframebottom)
                                                        \nCentre of the Earth',font= "timesnewroman 20")
  spaceframe7.pack(side= LEFT)
                                                           ebooklabel11.pack()
  spacelabel7=Label(spaceframe7,text='
                                           ')
  spacelabel7.pack()
                                                           spaceframe11=Frame(ebooksframebottom)
                                                           spaceframe11.pack(side= LEFT)
                                                           spacelabel11=Label(spaceframe11,text='
  book8=Frame(ebooksframebottom)
                                                                                                      ')
```

```
spacelabel11.pack()
                                                                     #starts modify
                                                                     canvas = Canvas(resultframe, width=650,
  book12=Frame(ebooksframebottom)
                                                           height=400)
                                                                     scroll_y = Scrollbar(resultframe.
  book12.pack(side=LEFT)
                                                           orient="vertical", command=canvas.yview)
ebook12pic=PhotoImage(file='./IMAGES/ebookspictures
                                                                     frameforscroll = Frame(canvas)
/ebook12.png')
                                                                     ALLTEXTRESULTLABELTEXT="
  openbutton12=Button(book12,text='Open', image =
                                                                     for a in range(n):
ebook12pic,command=openbook12)
  openbutton12.pack()
                                                                       "'x='detailframe'+str(a)
  ebooklabel12=Label(book12,text='Robinson
                                                                       globals()[x]=Frame(frameforscroll)
Crusoe',font= "timesnewroman 20")
                                                                       globals()[x].pack()"
  ebooklabel12.pack()
                                                                       bookid=
                                                                                   BOOK ID
                                                           '+str(processlist[a][0])+'\n'
  spaceframe12=Frame(ebooksframebottom)
                                                                       bookname= 'BOOK TITLE :
  spaceframe12.pack(side= LEFT)
                                                           '+processlist[a][1].title()+'\n'
  spacelabel12=Label(spaceframe12,text='
                                                ')
                                                                       author=
                                                                                  'AUTHOR
  spacelabel12.pack()
                                                           '+processlist[a][2]+'\n'
                                                                       publisher= 'PUBLISHER
                                                           '+processlist[a][3]+'\n'
  backbuttonframe=Frame(root)
  backbuttonframe.pack()
                                                                       availability='AVAILABILITY:
                                                           '+processlist[a][4]+'\n'
backbutton=Button(backbuttonframe,text='Back',comma
nd=ebookpagetomain)
  backbutton.pack()
                                                           ALLTEXTRESULTLABELTEXT+=bookid+bookname+au
                                                           thor+publisher+availability
                                                                       ALLTEXTRESULTLABELTEXT+='\n------
                                                                -----\n\n'
  root.mainloop()
def searchbookfn():
  if currentlocationforserachbook == 'GENERAL':
     destmainpageto()
  global titleofsearchbookpage
                                                           ALLTEXTRESULTLABEL=Label(frameforscroll,text=ALL
  global keywordframe
                                                           TEXTRESULTLABELTEXT, justify=LEFT)
  global resultframe
                                                                     ALLTEXTRESULTLABEL.pack()
  global backframe
                                                                     canvas.create window(0, 0, anchor='nw',
                                                           window=frameforscroll)
  def search():
    alobal resultframe
    findnamevalue=findname.get()
                                                                # make sure everything is displayed before
    if len(findnamevalue) != 0:
                                                           configuring the scrollregion
       findnamevalue=findnamevalue.lower()
                                                                     canvas.update_idletasks()
cursor.execute("select * from books where BOOK_Title like '%{}%'".format(findnamevalue))
       I=cursor.fetchall()
                                                           canvas.configure(scrollregion=canvas.bbox('all'),
       processlist=list(I)
                                                           yscrollcommand=scroll y.set)
                                                                     canvas.pack(fill='both', expand=True,
       n=len(processlist)
                                                           side='left')
       resultframe. pack forget()
                                                                     scroll y.pack(fill='y', side='right')
       resultframe=Frame(root)
       resultframe.pack()
                                                                root.mainloop()
       if n == 0:
sorrydoesnotexistlabel=Label(resultframe,text='Sorry,
the book is not found', fg='red')
                                                              titleofsearchbookpage=Frame(root)
         sorrydoesnotexistlabel.pack()
                                                              titleofsearchbookpage.pack()
                                                              searchbooktext = Label(titleofsearchbookpage, text =
                                                           'Search a Book\n\n\n', fg='blue',font= "applecherry 35")
newlinelabel=Label(resultframe,text='\n\nRESULT\n\n\n'
                                                              searchbooktext.pack(side= TOP)
                                                              keywordframe=Frame(root)
, fg='green',font= "20")
                                                              keywordframe.pack()
         newlinelabel.pack()
```

```
lbltellingtext=Label(keywordframe,text='Type a
Keyword ')
                                                            backbtn=Button(frmthankyou,text='Back',command=suc
  lbltellingtext.pack()
                                                            cessfultomainpage)
  findname=Entry(keywordframe,width=30)
                                                                 backbtn.pack()
  findname.pack()
                                                                 root.mainloop()
findbutton=Button(keywordframe,text='Search',comman
d=search)
  findbutton.pack(side=BOTTOM)
                                                               def submitsuggestion():
  resultframe=Frame(root)
                                                                 global errormessagecount
  resultframe.pack()
                                                                 global errormessagecountforinvalidentry
                                                                 name = nameofuserentry.get()
  backframe=Frame(root)
  backframe.pack(side=BOTTOM)
                                                                 nric=nricofuserentry.get()
                                                                 email=emailofuserentry.get()
                                                                 noofstars= var.get()
backbutton=Button(backframe,text='Back',command=se
archtomain)
                                                                 othersuggestion=TextArea.get("1.0",END)
  backbutton.pack()
                                                                 if len(name)!=0 and len(nric)!=0 and len(email)>10
  root.mainloop()#find similarity between names and try
to capitaliese the first letter of entry before giving in the
                                                            and noofstars != 0:
                                                                   try:
  #create a for loop to make a table like structure in the
                                                                      m="aministrator.library@gmail.com"
                                                                      sub='Feedback from '+ name
tkinter page
                                                                      msg='Feedback reveived from ' + str(name) +
                                                            '\n NRIC/Passport number: ' + str(nric) +'\n given rating: '
                                                            + str(noofstars) +\n Email of user: ' + str(email) + \n
                                                            Other suggestions: ' + str(othersuggestion)
                                                                      with smtplib.SMTP('smtp.gmail.com', 587) as
def newpage():
                                                            smtp:
                                                                         smtp.ehlo()
  "framepic=Frame(root)
                                                                         smtp.starttls()
  framepic.pack()
                                                                         smtp.ehlo()
  photo2=PhotoImage(file='electrycity formula.png')
#REMOVE THIS
                                                            smtp.login('aministrator.library@gmail.com','scyjovpwhq
  labelframe=Label(framepic,image=photo2)
                                                            zfbrur')
  labelframe.pack()
                                                                        subject =sub
  root.mainloop()"
                                                                        body = msg
                                                                        msg = f'Subject: {subject}\n\n{body}'
def successfultomainpage():
  frmthankyou. pack forget()
                                                            smtp.sendmail('aministrator.library@gmail.com',m,msg)
  mainpage()
                                                                      #storing in txt file
def feedbackoage():
  destmainpageto()
                                                            savefile=open('./REPORTS/suggestions.txt','a')
  global titleoffeedbackpage
                                                                      msq+='\n\n'
  global detailsofpersonframe
                                                                      savefile.write(msg)
  global suggestiongivingframe
                                                                      savefile.close()
  global errormessagecount #just a variable
  global errormessagecountforinvalidentry#just a
                                                                      #thanking the user
variable
                                                                      m=email
                                                                      sub='Thank You '+ name
                                                                      msg='Dear'+str(name)+',\\ \  \  \, Thank\ You\ For
  errormessagecount=0 #variable for keeping track of
                                                            Your Feed Back\n\n Warm Regards \n Library Team'
error message
  errormessagecountforinvalidentry =0 #variable for
                                                                      with smtplib.SMTP('smtp.gmail.com', 587) as
keeping track of error message
                                                            smtp:
  def successful():
                                                                         smtp.ehlo()
     global frmthankyou
                                                                        smtp.starttls()
     titleoffeedbackpage.pack_forget()
                                                                        smtp.ehlo()
     detailsofpersonframe. pack forget()
     suggestiongivingframe. pack forget()
                                                            smtp.login('aministrator.library@gmail.com','scyjovpwhq
     frmthankyou=Frame(root)
                                                            zfbrur')
     frmthankyou.pack()
                                                                         subject =sub
     lblthankyou=Label(frmthankyou, text = '\n\n\nThank
                                                                        bodv = msa
You for your Feedback :)\n\n', fg='green',font=
                                                                        msg = f'Subject: {subject}\n\n{body}'
"applecherry 30 italic")
```

smtp.sendmail('aministrator.library@gmail.com',m,msg)

lblthankyou.pack()

except: if errormessagecount == 0: plscheck=Label(buttonsframe,text='Please Check Your Internet connection', fg='red') plscheck.pack() errormessagecount+=1 else: if errormessagecountforinvalidentry == 0: plscheckentry=Label(buttonsframe,text='Please Give all necessary informations', fg='red') plscheckentry.pack() errormessagecountforinvalidentry+=1	emailofuserentry=Entry(emailofpersonframe,width=30) emailofuserlabel.pack(side=LEFT) emailofuserentry.pack(side=LEFT) compemail=Label(emailofpersonframe,text=' *',fg='red') compemail.pack(side=LEFT) #rating stars ratingsframe=Frame(detailsofpersonframe) ratingsframe.pack() questionlabel=Label(ratingsframe,text='\nHow much do you rate our service?') questionlabel.pack(side=TOP) lowestlabel=Label(ratingsframe,text='Poor') lowestlabel.pack(side=LEFT)
,	var = IntVar()
titleoffeedbackpage=Frame(root) titleoffeedbackpage.pack() feedbacktitletext = Label(titleoffeedbackpage, text = 'Feedback', fg='blue',font= "applecherry 35") feedbacktitletext.pack(side= TOP) suggestiongivingframe=Frame(root) suggestiongivingframe.pack()	radiobtn1star = Radiobutton(ratingsframe) star1 = PhotoImage(file="./IMAGES/Icons/stars.png") radiobtn1star.config(image=star1, variable=var, value=1) radiobtn1star.pack(side=LEFT) radiobtn2star = Radiobutton(ratingsframe) star2 = PhotoImage(file="./IMAGES/Icons/stars.png") radiobtn2star.config(image=star2, variable=var,
leaveyoursuggestiontext=Label(suggestiongivingframe,	value=2) radiobtn2star.pack(side=LEFT)
text='\nLeave Your Feedback Here\n', font='verdana 15') leaveyoursuggestiontext.pack(side= LEFT) suggestiongivingframe.pack() detailsofpersonframe=Frame(root) detailsofpersonframe.pack()	radiobtn3star = Radiobutton(ratingsframe) star3 = PhotoImage(file="./IMAGES/Icons/stars.png") radiobtn3star.config(image=star3, variable=var, value=3) radiobtn3star.pack(side=LEFT)
#name nameframe=Frame(detailsofpersonframe) nameframe.pack() nameofuserlabel=Label(nameframe,text=' \t\t Your	radiobtn4star = Radiobutton(ratingsframe) star4 = PhotoImage(file="./IMAGES/Icons/stars.png") radiobtn4star.config(image=star4, variable=var,
Name ') nameofuserentry=Entry(nameframe,width=20) nameofuserlabel.pack(side=LEFT) nameofuserentry.pack(side=LEFT)	value=4) radiobtn4star.pack(side=LEFT)
compname=Label(nameframe,text=' *',fg='red') compname.pack(side=LEFT) #nric nricofpersonframe=Frame(detailsofpersonframe) nricofpersonframe.pack() nricofuserlabel=Label(nricofpersonframe,text='\n	radiobtn5star = Radiobutton(ratingsframe) star5 = PhotoImage(file="./IMAGES/Icons/stars.png") radiobtn5star.config(image=star5, variable=var, value=5) radiobtn5star.pack(side=LEFT)
Last 4 Characters Your NRIC/Passport') nricofuserentry=Entry(nricofpersonframe,width=20) nricofuserlabel.pack(side=LEFT) nricofuserentry.pack(side=LEFT) compnric=Label(nricofpersonframe,text=' *',fg='red') compnric.pack(side=LEFT)	highestlabel=Label(ratingsframe,text='Excellent') highestlabel.pack(side=LEFT) comperating=Label(ratingsframe,text=' *',fg='red') comperating.pack(side=LEFT)
#email emailofpersonframe=Frame(detailsofpersonframe) emailofpersonframe.pack()	#othersuggestions othersuggestionframe=Frame(detailsofpersonframe) othersuggestionframe.pack()
emailofuserlabel=Label(emailofpersonframe,text='\n\t\t\t Your Email Address \t')	othersuggestionslabel=Label(othersuggestionframe,text ='\nDo you have any other suggestions?') othersuggestionslabel.pack()

successful()

```
TextArea =
                                                          searchbookimage=PhotoImage(file='./IMAGES/Helppict
Text(othersuggestionframe, selectborderwidth=3)
                                                          ures/help-searchbook.png') ##INSERT THE IMAGE AS
  TextArea.pack(expand=YES, fill=BOTH)
                                                          imagename.extension
                                                          searchbooklabel=Label(frameforscroll,image=searchboo
                                                          kimage)
  buttonsframe=Frame(detailsofpersonframe)
                                                            searchbooklabel.pack()
  buttonsframe.pack(side=BOTTOM)
  compulsorylabel=Label(buttonsframe,text=' *
compulsory',fg='red')
  compulsorylabel.pack()
                                                          ebookimage=PhotoImage(file='./IMAGES/Helppictures/e
  submitbutton= Button(buttonsframe, text="Submit",
                                                          book(help).png')
command=submitsuggestion)
                                                             ebooklabel=Label(frameforscroll,image=ebookimage)
  submitbutton.pack()
                                                             ebooklabel.pack()
  backbutton = Button(buttonsframe, text="Back",
command=suggestionstomain)
                                                          ebookwimage=PhotoImage(file='./IMAGES/Helppictures
                                                          /ebookweb(help).png')
  backbutton.pack()
  root.mainloop()
                                                          ebookwlabel=Label(frameforscroll,image=ebookwimage)
                                                             ebookwlabel.pack()
def helpme():
  def back():
                                                          feedbackimage=PhotoImage(file='./IMAGES/Helppicture
    titleoffeedbackpage.pack forget()
                                                          s/feedback(help).png')
    frameforscroll.pack_forget()
                                                          feedbacklabel=Label(frameforscroll,image=feedbackima
    scroll y.pack forget()
    canvas.pack forget()
                                                          ge)
    mainpage()
                                                             feedbacklabel.pack()
                                                          aboutusimage=PhotoImage(file='./IMAGES/Helppictures
  destmainpageto()
                                                          /abous us(help).png')
  titleoffeedbackpage=Frame(root)
                                                          aboutuslabel=Label(frameforscroll,image=aboutusimage
  titleoffeedbackpage.pack()
                                                          )
  feedbacktitletext = Label(titleoffeedbackpage, text =
                                                             aboutuslabel.pack()
'Help\n', fg='blue',font= "applecherry 35")
  feedbacktitletext.pack(side= TOP)
                                                          createacimage=PhotoImage(file='./IMAGES/Helppicture
                                                          s/create page(help).png')
  canvas = Canvas(root, width=650, height=400)
                                                          createaclabel=Label(frameforscroll,image=createacimag
  scroll_y = Scrollbar(root, orient="vertical",
                                                          e)
command=canvas.yview)
                                                             createaclabel.pack()
  frameforscroll = Frame(root)
  canvas.create_window(0, 0, anchor='nw',
window=frameforscroll)
                                                          resetacimage=PhotoImage(file='./IMAGES/Helppictures/
                                                          reset password(help).png')
  #####START HERE####
                                                          resetaclabel=Label(frameforscroll,image=resetacimage)
                                                             resetaclabel.pack()
  #SAVE THE IMAGE IN folder -->
  #computer science project\computer science
project\help page
                                                          userintimage=PhotoImage(file='./IMAGES/Helppictures/
                                                          help-userinterface.png')
loginpageimage=PhotoImage(file='./IMAGES/Helppictur
                                                             userint=Label(frameforscroll,image=userintimage)
es/Login page(help).png') ##INSERT THE IMAGE AS
                                                             userint.pack()
```

borrowedbksimage=PhotoImage(file='./IMAGES/Helppic

tures/help-borrowedbooks.png')

imagename.extension

loginpagelabel.pack()

age)

loginpagelabel=Label(frameforscroll,image=loginpageim

borrowedbks=Label(frameforscroll,image=borrowedbksi mage)

borrowedbks.pack()

overduebksimage=PhotoImage(file='./IMAGES/Helppict ures/help-overduebks.png')

overduebks=Label(frameforscroll,image=overduebksima qe)

overduebks.pack()

reservedbksimage=PhotoImage(file='./IMAGES/Helppict ures/help-reservedbooks.png')

reservedbks=Label(frameforscroll,image=reservedbksim age)

reservedbks.pack()

composemailimage=PhotoImage(file='./IMAGES/Helppic tures/help-composemail.png')

composemail=Label(frameforscroll,image=composemaili mage)

composemail.pack()

inboxmailimage=PhotoImage(file='./IMAGES/Helppictures/help-inbox.png')

inboxmail=Label(frameforscroll,image=inboxmailimage)
inboxmail.pack()

#USE \n for going to next line

textinfo='\n\nFor any enquiries, please feel free to email Librarian\n\n'

finaltext=Label(frameforscroll,text=textinfo, font='verdana 20') finaltext.pack()

backbtn=Button(frameforscroll, text= 'Back', command=back) backbtn.pack(side=BOTTOM)

###END HERE####

make sure everything is displayed before configuring the scrollregion canvas.update_idletasks()

canvas.configure(scrollregion=canvas.bbox('all'), yscrollcommand=scroll_y.set)

canvas.pack(fill='both', expand=True, side='left') scroll y.pack(fill='y', side='right')

root.mainloop()

def aboutuspage():#LOT OF MISTAKES in tamiltext destmainpageto() global imageandback global contenttext global langoptions global englishtext #JUST variable name global tamiltext #JUST variable name

textenglish='\n\nHave you guys ever wondered how are some countries are suffering from poverty?\n It is quite sad to see children without a book or some of the children getting chased out from \n library because of their bad appearance. It is more disappointing to see children who have a \n capability to succeed in their life but they do not have source to develop their knowledge. We\n have opened our library in 1965 to give a hand to this type of children around the world. \n We have upgraded the library on 2020 and we have also made a website to make it reader friendly. \n This user friendly GUI interface supports all users and librarian to process transactions with\n computer using Python. This software can authenticate users and provide details accordingly. \n To make it secure features like OTP verification has also been added to increase overall\n productivity of the library and prevent data theft or mismatches. \n Readers can now enjoy faster process time for each of their requests... \n'

texttamil='\n\nசில நாடுகள் எவ்வாறு வறுமையால் பாதிக்கப்படுகின்றன என்பதை நீங்கள் எப்போதாவது \nயோசித்திருக்கிறீர்களா? குழந்தைகள் தோற்றத்தால் நூலகத்திலிருந்து துரத்தப்படுவதைப் \nபார்க்கும்போது மிகவும் வருத்தமாக இருக்கிறது. தங்கள் வாழ்க்கையில் வெற்றிபெறக்கூடிய ∖nதிறனைக் கொண்ட குழந்தைகளைப் பார்ப்பது மிகவும் ஏமாற்றமளிக்கிறது, ஆனால் \nஅவர்களின் அறிவை வளர்ப்பதற்கான ஆதாரங்கள் அவர்களிடம் இல்லை, உலகெங்கிலும் \nஉள்ள இந்த வகை குழந்தைகளுக்கு ஒரு கை கொடுக்க 1965 ஆம் ஆண்டில் நாங்கள் எங்கள் \n நூலகத்தைத் திறந்தோம். 2020 ஆம் ஆண்டில் நூலகத்தை மேம்படுத்தியுள்ளோம், மேலும் \n அதை வாசகருக்கு நட்பாக மாற்ற ஒரு வலைத்தளத்தையும் உருவாக்கியுள்ளோம். இந்த பயனர் ∖n நட்பு GUI இடைமுகம் பைத்தானைப் பயன்படுத்தி கணினியுடன் பரிவர்த்தனை செயலாக்க ∖n அனைத்து பயனர் மற்றும் நூலகரை ஆதரிக்கிறது. இந்த மென்பொருள் பயனரை \n அங்கீகரிக்கலாம் மற்றும் அதற்கேற்ப விவரங்களை வழங்க முடியும். அதைப் பாதுகாப்பாக \n வைக்க, நூலகத்தின் ஒட்டுமொத்த உற்பத்தித்திறனை அதிகரிக்கவும், தரவு திருட்டு அல்லது \n பொருந்தாத தன்மைகளைத் தடுக்கவும் OTP சரிபார்ப்பு போன்ற அம்சங்கள் \n சேர்க்கப்பட்டுள்ளன. வாசகர்கள் இப்போது தங்கள் கோரிக்கைகளின் விரைவான \n செயல்முறை நேரத்தை அனுபவிக்க முடியும்∖n'

telugutext='\n\nకొన్ని దేశాలు పేదరికంతో ఎలా బాధపడుతున్నాయో మీరు ఎప్పుడైనా ఆలోచించారా?\n\

పేద పిల్లలు వారి ప్రదర్భన వల్ల లైబ్రరీ నుండి బయటకు పంపించడం ಮಾడడం చాలా బాధగా ఉంటుంది.\n\ పేద పిల్లలు చాలా మంది ఉంటారు కానీ వాళ్ళకి తగిన వసత్తలు ఉండవు అందుకే వాళ్లకున్న తెలివితేటలు ప్రతిభలను ఉపయోగించలేరు.\n\ అందుకే 1965 లో ఎలాంటి పిల్లలు కోసం ఈ లైబ్రరీని ప్రారంభించాము. ఈ సంవత్సరం (2020) లో, ఈ లైబ్రరీని అప్గేడ్ చేయడం జరిగింది\n\ మరియు అంతర్జాలం లో కూడా అందరికి సులభంగా చదువుకునేవిధంగా, జరిగింది.\n\ ఈ యూజర్ ఫ్రెండ్డీ GUI ఇంటర్సేస్ చదువుకునే వాళ్ళకి, వాడుకునే వాళ్ళకి మరియు లైబ్రరీ లో పనిచేసే వారికీ కంప్పూటర్ విధానం లో పైథాన్సి \n\ ఉపయోగించి వారి-వారి లావాదేవీలను మరియు ప్రక్రియలను పూర్తీ చేస్తుంది. \n\ ఈ పైథాన్ సాఫ్ట్ఎేర్ ఉపయోగించి అన్స్తి వివరాలను ప్రమాణీకరించడానికి ఉపయోగ పడుతుంది.\n\ ఈ పెబ్స్ట్ టని సురక్షితంగా ఉంచడానికి మరియు సైబర్ నరకాలు పలుబడకుంగా OTP లాంటి సురక్షితమైన విధానాలు ఉపయోగించడం జరిగింది.\n\ పాఠకులు ఇప్పుడు వారి ప్రతి అభ్యర్థనల కోసం పేగంగా ప్రాసెస్ సమయాన్ని ఆస్వాదించవచ్చు ...'

hinditext='\n\nक्या आप लोगों ने कभी सोचा है कि क्छ देशों गरीबी से कैसे पीड़ित हैं?\n\ क्छ बच्चों उनके दिखाव के वजह से, या उनके गरीबी के कारन से लाइब्रेरी से बाहर निकलवाना बह्त द्खी बात है। \n\ ऐसे बच्चों को देखकर ज्यादा निराशा होती है जो अपने जीवन में सफल होने की क्षमता रखते हैं लेकिन उनके पास विकास करने के लिए कोई स्रोत नहीं है। \n\ हमने दुनिया भर में इस प्रकार के बच्चों को एक हाथ देने के लिए 1965 में ये लाइब्रेरी खोली है। हमने इस लाइब्रेरी को 2020 \n\ में अपग्रेड किया है और हमने इसे रीडर फ्रेंडली बनाने के लिए एक वेबसाइट भी बनाई है।\n\ यह उपयोगकर्ता के अनुकूल GUI इंटरफ़ेस पायथन का उपयोग करके सभी उपयोगकर्ताओं और लाइब्रेरियन को कंप्यूटर के साथ लेनदेन की प्रक्रिया का समर्थन करता है।\n\ यह सॉफ्टवेयर उपयोगकर्ताओं को प्रमाणित कर सकता है और तदन्सार विवरण प्रदान कर सकता है। इसे स्रक्षित बनाने के लिए प्स्तकालय की समग्र उत्पादकता बढ़ाने और डेटा \n\ की चोरी या बेमेल को रोकने के लिए ओटीपी सत्यापन भी जोड़ा गया है। पाठक अब उनके प्रत्येक अन्रोध के लिए तेज़ प्रक्रिया

समय का आनंद ले सकते हैं ...'

```
global contenttext
     global englishtext #JUST variable name
     global tamiltext #JUST variable name
     contenttext. pack_forget()
     lang=variable.get()
     if lang == 'English':
       contenttext=Frame(root)
       contenttext.pack()
englishtext=Label(contenttext,text=textenglish,font=
"timesnewroman 20")
       englishtext.pack()
     elif lang == 'தமிழ்':
       contenttext=Frame(root)
       contenttext.pack()
       tamiltext=Label(contenttext,text=texttamil,font=
"timesnewroman 20")
       tamiltext.pack()
     elif lang == 'ತಿಲುಗು':
       contenttext=Frame(root)
       contenttext.pack()
       tamiltext=Label(contenttext,text=telugutext,font=
"timesnewroman 20")
       tamiltext.pack()
     elif lang == "हिन्दी":
       contenttext=Frame(root)
       contenttext.pack()
       tamiltext=Label(contenttext,text=hinditext,font=
"timesnewroman 20")
       tamiltext.pack()
  imageandback=Frame(root)
  imageandback.pack(side= BOTTOM)
  #libraryphoto=PhotoImage(file='libraryphoto.png')
#libraryphotolabel=Label(imageandback,image=libraryp
  #libraryphotolabel.pack(side= TOP)
  backtomain = Button(imageandback, text="Back",
command=aboutuspagetomain)
  backtomain.pack()
  langoptions=Frame(root)
  langoptions.pack(side= TOP)
  feedbacktitletext = Label(langoptions, text = 'About
Us\n', fg='blue',font= "applecherry 35")
  feedbacktitletext.pack(side= TOP)
  langchangetext=Label(langoptions,text='Change
Language of this page')
  langchangetext.pack()
  OPTIONS = ["English","ಕ್ರೂமಿழ்","ತಿಲುಗು","हिन्दी"] #etc
  variable = StringVar(langoptions)
  variable.set(OPTIONS[0]) # default value
  contenttext=Frame(root)
  contenttext.pack()
  englishtext=Label(contenttext,text=textenglish,font=
"timesnewroman 20")
  englishtext.pack()
```

def langchange():

```
w = OptionMenu(langoptions, variable, *OPTIONS)
                                                                        while True:
                                                                          newlibid=random.randint(1000,9999)
  w.pack()
  langchangebutton = Button(langoptions,
                                                                          if newlibid not in libidlist:
text="Change Language", command=langchange)
                                                                             break
  langchangebutton.pack()
                                                            existingusernamesandpasswordlist=pickle.load(encrypte
  root.mainloop()
                                                            dfile)
def createaccountpagetomain():
  framehaveingheadingincreateaccount. pack forget()
                                                                        newexcrypteddata=(newlibid,uname,pwd)
  detailsofpersonframeforcreatingaccount.
                                                                        encryptedfile.close()
pack forget()
  createaccountframebuttons. pack_forget()
                                                            encryptedfile=open('./STORAGE/encrypteduserdetails.tx
  errorstreams. pack_forget()
                                                            t','wb')
  loginpage()
def createaccountpage():
                                                            existingusernamesandpasswordlist.append(newexcrypte
  global framehaveingheadingincreateaccount
                                                            ddata)
  global detailsofpersonframeforcreatingaccount
  global createaccountframebuttons
                                                            pickle.dump(existingusernamesandpasswordlist,encrypt
  global captchanumber #variable
                                                            edfile)
  global errorstreams
  framehaveingheadingincreateaccount =
                                                                        cursor.execute("insert into users values
Frame(root, width=1000, height= 1000)
                                                            ({},'{}','{}','{}','{}','{}')".format(newlibid,uname,fname,lna
  framehaveingheadingincreateaccount.pack()
                                                            me,phone,nric,email))
                                                                        mycon.commit()
lbl=Label(framehaveingheadingincreateaccount,text='Cr
                                                                        encryptedfile.close()
eate Account \n', fg='blue',font= "applecherry 35")
  lbl.pack(side= TOP)
                                                                        texttobedisp='\n\nYour Account has been
  def extractusers():
                                                            created successfully\n\nYour Library ID is %s \n\nTry to
    global unamelist
                                                            Login now'%newlibid
    global passlist
    global libidlist
                                                            succesfullycreated=Label(accountcreatedmessage,text=
    f=open('./STORAGE/encrypteduserdetails.txt','rb')
                                                            texttobedisp,fg='green')
    I=pickle.load(f)
                                                                        succesfullycreated.pack()
    libidlist=[]
                                                                        m=email
    unamelist=[]
                                                                        sub='Thank You '+ fname
                                                                        msg='Dear ' + str(fname) + ',\n Thank You
    passlist=[]
    for a in I:
                                                            For Creting an Account with us\nYour Library ID is
                                                            '+str(newlibid)+'\nYour Username is
       libidlist.append(a[0])
       unamelist.append(a[1])
                                                            '+str(uname)+'\n\nAll Details will be kept
       passlist.append(a[2])
                                                            confidential.\n\nWarm Regards \n Library Team'
                                                                        with smtplib.SMTP('smtp.gmail.com', 587)
  def validate():
                                                            as smtp:
    global countnowidontknowwhatelsetokeepnow
                                                                          smtp.ehlo()
#just a avriable
                                                                          smtp.starttls()
    countnowidontknowwhatelsetokeepnow = 0
                                                                          smtp.ehlo()
    def createthisaccount():
       def verifyotp():
                                                            smtp.login('aministrator.library@gmail.com','scyjovpwhq
         global errormessagecount
                                                            zfbrur')
          def loginnow():
                                                                          subject =sub
            accountcreatedmessage. pack_forget()
                                                                          body = msg
            loginpage()
                                                                          msg = f'Subject: {subject}\n\n{body}'
         errormessagecount = 0
         if str(valofotp)==labelaskingtheotp.get():
                                                            smtp.sendmail('aministrator.library@gmail.com',m,msg)
optconformationpageincreatingaccountheader.
                                                            loginnowbutton=Button(accountcreatedmessage,text='L
pack forget()
                                                            ogin now',command=loginnow)
            inotpconformotpframe. pack forget()
                                                                        loginnowbutton.pack()
            accountcreatedmessage=Frame(root)
                                                                     else:
            accountcreatedmessage.pack()
                                                                        global
                                                            countnowidontknowwhatelsetokeepnow
encryptedfile=open('./STORAGE/encrypteduserdetails.tx
                                                                        if countnowidontknowwhatelsetokeepnow >
t','rb')
                                                            0 and countnowidontknowwhatelsetokeepnow <5:
                                                            countnowidontknowwhatelsetokeepnow+=1
```

newlibid=0

inotpconformotpframe=Frame(root) >= 5: inotpconformotpframe.pack() labeltellingwrongtheotp=Label(inotpconformotpframe,tex t='Maximum Tries Reached',fg='red') labelgivinginformationforuser=Label(inotpconformotpfra labeltellingwrongtheotp.pack() me,text='An OTP has been sent to your Account\nPlease Confirm your Account\n') inotpconformotpframe. pack forget() inotpconformotpframe. pack forget() labelgivinginformationforuser.pack() optconformationpageincreatingaccountheader. labelaskingtheotp=Entry(inotpconformotpframe,width=20 pack_forget() ,show='*') loginpage() labelaskingtheotp.pack() else: checktheotp=Button(inotpconformotpframe,text='Submit' labeltellingwrongtheotp=Label(inotpconformotpframe,tex t='OTP does not match',fg='red') .command=verifyotp) checktheotp.pack() labeltellingwrongtheotp.pack() root.mainloop() countnowidontknowwhatelsetokeepnow+=1 root.mainloop() global invalidpasswordmessage global errorstreams framehaveingheadingincreateaccount. fname=nameofuserentry.get() Iname=Inameofuserentry.get() pack forget() detailsofpersonframeforcreatingaccount. nric=nricofuserentry.get() email=emailofuserentry.get() pack forget() createaccountframebuttons. pack forget() phone=phoneofuserentry.get() errorstreams. pack_forget() uname=usernameofuserentry.get() pwd=passwordchosenentry.get() optconformationpageincreatingaccountheader=Frame(ro pwdconf=cnpasswordchosenentry.get() captcha=captchaentry.get() optconformationpageincreatingaccountheader.pack() extractusers() errorstreams. pack_forget() lbl=Label(optconformationpageincreatingaccountheader, errorstreams=Frame(root) text='Confirm Your Account \n\n\n', fg='blue',font= errorstreams.pack(side=BOTTOM) "applecherry 35") lbl.pack() inbetweenthisvariableisusedforcountingpurpose=0 valofotp= random.randint(1000,9999) try: if uname in unamelist: m=email sub='OTP Verification' unameexistlabel=Label(errorstreams,text='Username msg='Dear '+str(fname)+',\nThe OTP for your Taken', fg='red')#EVALUATION FOR UNIQUE Libriary account is '+str(valofotp)+'\n\nWarm **USERNAME** Regards\nLibrary Team' unameexistlabel.pack() with smtplib.SMTP('smtp.gmail.com', 587) as inbetweenthisvariableisusedforcountingpurpose+=1 smtp: smtp.ehlo() smtp.starttls() smtp.ehlo() countforpasswordcaptletter, countforpasswordsmallletter, countforpasswordnumber=0,0,0 #PASSWORD **EVALUATION** smtp.login('aministrator.library@gmail.com','scyjovpwhq zfbrur') for varia in pwd: if ord(varia) in range(65,91): countforpasswordcaptletter+=1 subject = sub body = msqelif ord(varia) in range(97,123): countforpasswordsmallletter+=1 msg = f'Subject: {subject}\n\n{body}' elif ord(varia) in range(48,58): countforpasswordnumber+=1 smtp.sendmail('aministrator.library@gmail.com',m,msg) if countforpasswordcaptletter==0 or except: if errormessagecount == 0: countforpasswordsmallletter == 0 or countforpasswordnumber == 0 or len(pwd)<8: plscheck=Label(inotpconformotpframe.text='Please Check Your Internet connection', fg='red') passwordinvalidlabel=Label(errorstreams,text='Your Password is weak', fg='red')#EVALUATION FOR plscheck.pack()

elif countnowidontknowwhatelsetokeepnow

errormessagecount+=1

STRONG PASSWORD

```
passwordinvalidlabel.pack()
                                                            nricofuserlabel=Label(nricofpersonframe,text='\nLast
                                                          4 characters NRIC/Passport Number\t')
inbetweenthisvariableisusedforcountingpurpose+=1
                                                            nricofuserentry=Entry(nricofpersonframe,width=20)
    if pwd != pwdconf:#
                                                            nricofuserlabel.pack(side=LEFT)
#IF PASSWORD ENTERED MATCHEES WITH THE
                                                            nricofuserentry.pack(side=LEFT)
CONFIRMED ONE
                                                            compnric=Label(nricofpersonframe,text=' *',fg='red')
                                                            compnric.pack(side=LEFT)
passdoesnotmatchlabel=Label(errorstreams,text='Pass
words Entered does not match', fg='red')
                                                            #phonenumber
       passdoesnotmatchlabel.pack()
                                                          phoneofpersonframe=Frame(detailsofpersonframeforcre
inbetweenthisvariableisusedforcountingpurpose+=1
                                                          atingaccount)
                                                            phoneofpersonframe.pack()
listoffilesofcaptcha[captchanumber][1]!=captchaentry.get
                                                          phoneofuserlabel=Label(phoneofpersonframe,text='\n\tY
():
                                                          our Phone Number\t\t')
       lblcaptchastats=Label(errorstreams,text='Wrong
Captcha Please Retry', fg='red')
                                                          phoneofuserentry=Entry(phoneofpersonframe,width=20)
       lblcaptchastats.pack()
                                                            phoneofuserlabel.pack(side=LEFT)
inbetweenthisvariableisusedforcountingpurpose+=1
                                                            phoneofuserentry.pack(side=LEFT)
                                                            compphone=Label(phoneofpersonframe,text='
                                                          *',fg='red')
inbetweenthisvariableisusedforcountingpurpose==0 and
                                                            compphone.pack(side=LEFT)
len(fname)!=0 and len(lname)!=0 and len(phone)!=0 and
len(nric)!=0 and len(email)!=0 and len(uname)!=0 and
len(pwd)!=0 and len(pwdconf)!=0:
                                                            #email
       createthisaccount()
                                                          emailofpersonframe=Frame(detailsofpersonframeforcre
    else:
                                                          atingaccount)
enterallinfoerrormessagelabel=Label(errorstreams,text='
                                                            emailofpersonframe.pack()
Please Enter all Necessary Details', fg='red')
       enterallinfoerrormessagelabel.pack()
                                                          emailofuserlabel=Label(emailofpersonframe,text='\n\t\t\t
                                                          Your Email Address\t\t')
    root.mainloop()
                                                          emailofuserentry=Entry(emailofpersonframe,width=30)
  detailsofpersonframeforcreatingaccount=Frame(root)
                                                            emailofuserlabel.pack(side=LEFT)
                                                            emailofuserentry.pack(side=LEFT)
  detailsofpersonframeforcreatingaccount.pack()
  #name
                                                            compemail=Label(emailofpersonframe,text='
                                                          *\t\n\n',fg='red')
nameframe=Frame(detailsofpersonframeforcreatingacco
                                                            compemail.pack(side=LEFT)
  nameframe.pack()
                                                            #username
  nameofuserlabel=Label(nameframe,text='
                                           \t Your
                                                          usernameofpersonframe=Frame(detailsofpersonframefo
First Name
                 \t')
  nameofuserentry=Entry(nameframe,width=20)
                                                          rcreatingaccount)
  nameofuserlabel.pack(side=LEFT)
                                                            usernameofpersonframe.pack()
  nameofuserentry.pack(side=LEFT)
  compname=Label(nameframe,text=' *',fg='red')
                                                          usernameofuserlabel=Label(usernameofpersonframe,te
  compname.pack(side=LEFT)
                                                          xt='\n\tSelect an Username \t\t')
Inameframe=Frame(detailsofpersonframeforcreatingacc
                                                          usernameofuserentry=Entry(usernameofpersonframe,wi
  Inameframe.pack()
                                                            usernameofuserlabel.pack(side=LEFT)
  Inameofuserlabel=Label(Inameframe,text='
                                                            usernameofuserentry.pack(side=LEFT)
                                                            compusername=Label(usernameofpersonframe,text='
Last Name
                 \t')
  Inameofuserentry=Entry(Inameframe,width=20)
                                                          *',fq='red')
  Inameofuserlabel.pack(side=LEFT)
                                                            compusername.pack(side=LEFT)
  Inameofuserentry.pack(side=LEFT)
  complname=Label(Inameframe,text=' *',fg='red')
                                                            #password
  complname.pack(side=LEFT)
  #nric
                                                          passwordchosen=Frame(detailsofpersonframeforcreatin
                                                          gaccount)
nricofpersonframe=Frame(detailsofpersonframeforcreati
                                                            passwordchosen.pack()
ngaccount)
```

Password \t\t\t')

passwordchosenlabel=Label(passwordchosen,text='\n\t

nricofpersonframe.pack()

```
passwordchosenentry=Entry(passwordchosen,width=20,
show="*")
                                                          compulsorylabel=Label(createaccountframebuttons,text
  passwordchosenlabel.pack(side=LEFT)
                                                          ='\n\n * compulsory\n',fg='red')
  passwordchosenentry.pack(side=LEFT)
                                                             compulsorylabel.pack()
  compuserpassword=Label(passwordchosen,text='
                                                             Backtomain=Button(createaccountframebuttons, text
*',fg='red')
                                                          = 'Back', command= createaccountpagetomain)
  compuserpassword.pack(side=LEFT)
                                                             Backtomain.pack(side = BOTTOM)
                                                          sumbitbutton=Button(createaccountframebuttons,text='S
  #conformpassword
                                                          ubmit',command=validate)
                                                             sumbitbutton.pack(side=BOTTOM)
cnpasswordchosen=Frame(detailsofpersonframeforcrea
tingaccount)
                                                             submitcred=Button(frameloginbuttons, text = 'Submit',
                                                          command= validate)
  cnpasswordchosen.pack()
                                                             submitcred.pack(side = BOTTOM)
                                                             forgotpassbutton=Button(frameloginbuttons, text =
cnpasswordchosenlabel=Label(cnpasswordchosen,text=
'\n\tConfirm Password \t\t')
                                                          'Forgot Password', command= forgotpass)
                                                             forgotpassbutton.pack(side = BOTTOM)
                                                             createnewaccount=Button(frameloginbuttons, text =
cnpasswordchosenentry=Entry(cnpasswordchosen,widt
h=20. show="*")
                                                           'Create Account', command=
  cnpasswordchosenlabel.pack(side=LEFT)
                                                          createaccountfromloginpage)
                                                             createnewaccount.pack(side = BOTTOM)
  cnpasswordchosenentry.pack(side=LEFT)
                                                             captchaphoto.pack(side= LEFT)
cncompuserpassword=Label(cnpasswordchosen,text='
                                                             captchaentry.pack(side= BOTTOM)
*\n\n',fg='red')
  cncompuserpassword.pack(side=LEFT)
                                                             root.mainloop()
                                                          def librarianspage():
criteriaforpassword=Frame(detailsofpersonframeforcreat
                                                             def borrowbook():
ingaccount)
                                                               global statsframe
  criteriaforpassword.pack()
                                                               global bookid #JUST A VARIABLE
                                                               welcomeframe.pack_forget()
passwordcriteria=Label(criteriaforpassword,text='\nYour
                                                               librarianframetop.pack forget()
Password should contain\natleast 8 characters.\natleast
                                                               librarianframetop.pack forget()
1 digit and \natleast 1 uppercase character\n\n')
                                                               librarianframebottom.pack forget()
                                                               def back():
  passwordcriteria.pack(side=BOTTOM)
                                                                 detailstoprocessframe.pack forget()
                                                                 btnframe.pack forget()
  #captcha
                                                                 detailstoprocessframe.pack forget()
  captchaframe =
                                                                 statsframe.pack forget()
Frame(detailsofpersonframeforcreatingaccount,width=1
                                                                 lbl.pack forget()
000, height= 1000)
                                                                 librarianspage()
                                                               def viewborrowedbook():
  captchaframe.pack()
  captchanumber= random.randint(1,9)-1
                                                                 detailstoprocessframe.pack_forget()
                                                                 btnframe.pack forget()
photo=PhotoImage(file=listoffilesofcaptcha[captchanum
                                                                 detailstoprocessframe.pack forget()
ber][0])
                                                                 statsframe.pack_forget()
  captchaphoto=Label(captchaframe,image=photo)
                                                                 lbl.pack forget()
  captchaentry = Entry(captchaframe, width=20)
                                                                 def back():
  captchaentry.insert(0, "Enter Captcha Here")
                                                                    lblb.pack forget()
  captchaphoto.pack(side= LEFT)
                                                                    lbldetails.pack_forget()
                                                                    backbtn.pack forget()
                                                                    borrowbook()
compuserpassword=Label(captchaframe,text='\t',fg='red'
  compuserpassword.pack(side=LEFT)
                                                                 cursor.execute('select * from borrowedbooks')
  captchaentry.pack(side= LEFT)
                                                                 Is=cursor.fetchall()
  compuserpassword=Label(captchaframe,text='
*\n\n',fg='red')
                                                                 vblbks='Number of Books Borrowed:
  compuserpassword.pack(side=LEFT)
                                                          '+str(len(ls))+'\n\n'
  errorstreams=Frame(root)
                                                                 for a in range(len(ls)):
  errorstreams.pack(side=BOTTOM)
                                                                    vblbks+='\nUSER ID
                                                                                         : '+ str(ls[a][0]) +
                                                          '\nBOOK ID : '+ str(ls[a][1])+ '\nDUE DATE : '+
  createaccountframebuttons=Frame(root)
                                                          str(ls[a][2])+'\n\n'
```

createaccountframebuttons.pack(side=BOTTOM)

```
detailstoprocessframe.pack forget()
         vblbks+='\n-----
                                                                         btnframe.pack forget()
                                                                         detailstoprocessframe.pack forget()
    ----\n\n'
                                                                         statsframe.pack_forget()
                                                                         lbl.pack forget()
       Iblb=Label(root,text='Borrowed Books \n\n',
                                                                         librarianspage()
fg='blue',font= "applecherry 35") #THIS HAS TO BE
                                                                      except:
DETELED WHILE NAVIGATING
                                                                         if errormessagecount == 0:
       lblb.pack(side= TOP)
                                                                           plscheck=Label(statsframe,text='Please
                                                            Check Your Internet connection', fg='red')
       IbIdetails=Frame(root)
                                                                           plscheck.pack()
       Ibldetails.pack()
                                                                           errormessagecount+=1
                                                                    def nextbkfn():
                                                                      global statsframe
       canvas = Canvas(Ibldetails, width=650,
                                                                      global detailstoprocessframe
height=400)
       scroll y = Scrollbar(lbldetails, orient="vertical",
                                                                      global borrowedbooksvariable
command=canvas.yview)
                                                                      global bookid
       frameforscroll = Frame(canvas)
                                                                      statsframe.pack forget()
       canvas.create window(0, 0, anchor='nw',
                                                                      detailstoprocessframe.pack forget()
window=frameforscroll)
                                                                      detailstoprocessframe=Frame(root) #A
                                                            FRAME
       overduelbl=Label(frameforscroll, text= vblbks,
                                                                      detailstoprocessframe.pack()
                                                                      statsframe=Frame(root)
justify=LEFT)
       overduelbl.pack()
                                                                      statsframe.pack()
       # make sure everything is displayed before
configuring the scrollregion
                                                            bkidlbl=Label(detailstoprocessframe,text='Book ID')
       canvas.update idletasks()
                                                                      bkidlbl.pack()
                                                                      bookid=Entry(detailstoprocessframe,
       canvas.configure(scrollregion=canvas.bbox('all'),
                                                            width=20)
yscrollcommand=scroll y.set)
                                                                      bookid.pack(side= LEFT)
       canvas.pack(fill='both', expand=True, side='left')
       scroll y.pack(fill='y', side='right')
                                                                    global statsframe
                                                                    global borrowedbooksvariable
                                                                    global detailstoprocessframe
       backbtn=Button(root, text='Back', command =
                                                                    global bookid
back)
       backbtn.pack(side=BOTTOM)
       root.mainloop()
     def borrow():
                                                                      statsframe.pack forget()
       global errormessagecount
                                                                    except:
       errormessagecount=0 #just a variable
                                                                      pass
       def finishandsendamail():
                                                                    statsframe=Frame(root)
          global errormessagecount
                                                                    statsframe.pack()
          try:
            m=eml
            sub='Borrowed Books'
msg=ftxtborrowedbooksvariable+borrowedbooksvariable
                                                                    bid=bookid.get()
+'\n\nWarm Regards\nLibrary Team'
                                                                    libid=lid.get()
            with smtplib.SMTP('smtp.gmail.com', 587)
                                                                    if len(bid) == 0 or len(libid) == 0:
as smtp:
                                                                      notfoundIbl=Label(statsframe,text='\n\nPlease
               smtp.ehlo()
               smtp.starttls()
                                                            enter all details', fg='Red')
               smtp.ehlo()
                                                                      notfoundlbl.pack()
                                                                      return
smtp.login('aministrator.library@gmail.com','scyjovpwhq
                                                                    cursor.execute('select * from books where
zfbrur')
                                                            bookid= {}'.format(bid))
               subject = sub
                                                                    l=cursor.fetchone()
                                                                    if len(I) > 0:
               body = msg
               msg = f'Subject: {subject}\n\n{body}'
                                                                      req = list(I)
                                                                    else:
smtp.sendmail('aministrator.library@gmail.com',m,msg)
```

```
notfoundlbl=Label(statsframe,text='\n\nBook
not Found', fg='Red')
                                                                   ftxtborrowedbooksvariable='Dear '+regusr[2]+'
         notfoundlbl.pack()
                                                          '+regusr[3]+',\nYou have Borrowed The Following
                                                          Books\n\n
         return
       cursor.execute('select * from users where LIBID=
                                                                   borrowedbooksvariable+=BKNM+'\t'+ 'Due
                                                          Date: '+str(txtl)+'\n'
{}'.format(libid))
       m=cursor.fetchall()
       if len(m) > 0:
                                                                   eml=requsr[6]
         regusr = list(m[0])
                                                                   nextbk=Button(statsframe,text='Borrow
                                                          Another Book for Same User',command=nextbkfn)
         notfoundurlbl=Label(statsframe,text='\n\nUser
                                                                   nextbk.pack()
not Found', fg='Red')
                                                                   finshbtn=Button(statsframe,text='Finish and
         notfoundurlbl.pack()
                                                          Send a Mail', command= finishandsendamail)
                                                                   finshbtn.pack()
         return
       if req[4] == 'Borrowed':
         borrowedlbl=Label(statsframe,text='\n\nBook
                                                               global borrowedbooksvariable #JUST A VARIABLE
is already Borrowed', fg='Red')
                                                               global detailstoprocessframe
         borrowedlbl.pack()
                                                               Ibl=Label(root,text='Borrow Book \n\n',
       else:
                                                          fg='blue', font= "applecherry 35") #THIS HAS TO BE
                                                          DETELED WHILE NAVIGATING
         cursor.execute('select * from reservations
                                                               lbl.pack(side= TOP)
where BOOKID = \'{}\".format(bid))
                                                               detailstoprocessframe=Frame(root) #A FRAME
         resbkslst=cursor.fetchall()
                                                               detailstoprocessframe.pack()
         if len(resbkslst) > 0:
                                                               lidlbl=Label(detailstoprocessframe,text='Library ID')
                                                               lidlbl.pack(side= LEFT)
            if str(resbkslst[0][0]) == libid:
                                                               lid=Entry(detailstoprocessframe,width=20)
              cursor.execute("delete from reservations
                                                               lid.pack(side= LEFT)
where BOOKID = \'{}\'".format(bid))
                                                               bkidlbl=Label(detailstoprocessframe,text='Book ID')
              mycon.commit()
                                                               bkidlbl.pack(side= LEFT)
            else:
                                                               bookid=Entry(detailstoprocessframe, width=20)
                                                               bookid.pack(side= LEFT)
                                                               borrowedbooksvariable="
borrowedlbl=Label(statsframe,text='\n\nBook is
                                                               statsframe=Frame(root)
Reserved', fg='Red')
              borrowedlbl.pack()
                                                               statsframe.pack()
              return
                                                               btnframe=Frame(root)# A FRAME
                                                               btnframe.pack(side=BOTTOM)
                                                               borrowbtn=Button(btnframe, text= 'Borrow',
         reg[4] = 'Borrowed'
                                                          command= borrow)
         BKNM=reg[1]
                                                               borrowbtn.pack()
                                                               viewbrwbtn=Button(btnframe, text='View Borrowed
                                                          Books', command= viewborrowedbook)
         cursor.execute('update books set Availability =
                                                               viewbrwbtn.pack()
'{}' where bookid = '{}'.format(req[4],req[0]))
                                                               backbtn=Button(btnframe, text='Back', command=
         mycon.commit()
                                                          back)
         now=datetime.now() + timedelta(days=14)
                                                               backbtn.pack()
#take in function
                                                               root.mainloop()
         txtl=now.strftime("%d-%m-%y")
                                                            def returnbook():
          #SHOW MESSAGE THAT THE BOOK IS
                                                          BORROWED, ASK FOR NEXT BOOK TO BORROW,
                                                          brrbktxt=str(libid)+','+str(bid)+','+str(txtl)+'\n'
                                                               global statsframe
         cursor.execute("insert into borrowedbooks
                                                               global bookid #JUST A VARIABLE
values ({},{},DATE_ADD(curdate(), INTERVAL 14
                                                               welcomeframe.pack forget()
DAY))".format(libid,bid))
                                                               librarianframetop.pack_forget()
         mycon.commit()
                                                               librarianframetop.pack_forget()
                                                              librarianframebottom.pack forget()
                                                               def back():
                                                                 detailstoprocessframe.pack_forget()
         sucmsg='\n\nThe Book '+BKNM +' has been
Issued to '+regusr[2]+'\n Due Date: '+str(txtl)
                                                                 btnframe.pack forget()
         borrowedlbl=Label(statsframe,text=sucmsg,
                                                                 detailstoprocessframe.pack_forget()
                                                                 statsframe.pack forget()
fg='Green')
                                                                 lbl.pack_forget()
         borrowedlbl.pack()
```

librarianspage()	global bookid
def returnbk():	try:
global errormessagecount	statsframe.pack forget()
errormessagecount=0 #just a variable	except:
def finishandsendamail():	pass
global errormessagecount	statsframe=Frame(root)
try:	statsframe.pack()
m=eml	()
sub='Returned Books'	
	#######################################
msg=ftxtborrowedbooksvariable+returnedbooksvariable	bid=bookid.get()#
+'\n\nWarm Regards\nLibrary Team'	libid=lid.get()##
with smtplib.SMTP('smtp.gmail.com', 587)	#######################################
as smtp:	if $len(bid) == 0$ or $len(libid) == 0$:
smtp.ehlo()	notfoundlbl=Label(statsframe,text='\n\nPlease
smtp.starttls()	enter all details', fg='Red')
smtp.ehlo()	notfoundlbl.pack()
	return
smtp.login('aministrator.library@gmail.com','scyjovpwhq	
zfbrur')	cursor.execute('select * from books where
,	bookid= {}'.format(bid))
subject = sub	l=cursor.fetchone()
body = msg	if len(I) > 0:
msg = f'Subject: {subject}\n\n{body}'	req = list(I)
smtp.sendmail('aministrator.library@gmail.com',m,msg)	else:
detailstoprocessframe.pack_forget()	notfoundlbl=Label(statsframe,text='\n\nBook
btnframe.pack_forget()	not Found', fg='Red')
detailstoprocessframe.pack_forget()	notfoundlbl.pack()
statsframe.pack_forget()	return
lbl.pack_forget()	
librarianspage()	cursor.execute('select * from users where LIBID=
except:	{}'.format(libid))
if errormessagecount == 0:	m=cursor.fetchall()
plscheck=Label(statsframe,text='Please	if len(m) > 0:
Check Your Internet connection', fg='red')	requsr = list(m[0])
plscheck.pack()	else:
errormessagecount+=1	notfoundurlbl=Label(statsframe,text='\n\nUser
	not Found', fg='Red')
def nextbkfn():	notfoundurlbl.pack()
global statsframe	return
global detailstoprocessframe	
global returnedbooksvariable	
global bookid	
statsframe.pack_forget()	
detailstoprocessframe.pack_forget()	if req[4] == 'Available':
detailstoprocessframe=Frame(root) #A	borrowedlbl=Label(statsframe,text='\n\nThe
FRAME	book has not been borrowed yet', fg='Red')
detailstoprocessframe.pack()	borrowedlbl.pack()
statsframe=Frame(root)	else:
statsframe.pack()	cursor.execute('select * from borrowedbooks
	where bookid= {}'.format(bid))
	bbkslst=cursor.fetchall()
bkidlbl=Label(detailstoprocessframe,text='Book ID')	if len(bbkslst) != 0 and str(bbkslst[0][0]) !=
bkidlbl.pack()	libid:
bookid=Entry(detailstoprocessframe,	
width=20)	notfoundurlbl=Label(statsframe,text='\n\nThis User Did
bookid.pack(side= LEFT)	not borrow the mentioned book', fg='Red')
	notfoundurlbl.pack()
	return
global statsframe	######
global returnedbooksvariable	
#######################################	
global detailstoprocessframe	bookstatus8thsug="

cursor.execute('select * from reservations where BOOKID = \'{}\''.format(bid)) resbkslst=cursor.fetchall() if len(resbkslst) > 0: bookstatus8thsug = 'Reserved' else: bookstatus8thsug = 'Available'	btnframe=Frame(root)# A FRAME btnframe.pack(side=BOTTOM) borrowbtn=Button(btnframe, text= 'Return', command= returnbk) borrowbtn.pack() backbtn=Button(btnframe, text='Back', command=back) backbtn.pack() root.mainloop()
######################################	
cursor.execute('update books set Availability = \'{}\' where bookid = \'{}\".format(bookstatus8thsug,req[0]))	######################################
mycon.commit()	global bookid #JUST A VARIABLE welcomeframe.pack_forget() librarianframetop.pack_forget()
sucmsg='\n\nThe Book '+BKNM +' Borrowed by '+requsr[2]+'\n Has been returned' borrowedlbl=Label(statsframe,text=sucmsg, fg='Green') borrowedlbl.pack()	librarianframetop.pack_forget() librarianframebottom.pack_forget() def back(): detailstoprocessframe.pack_forget() btnframe.pack_forget() detailstoprocessframe.pack_forget()
ftxtborrowedbooksvariable='Dear '+requsr[2]+' '+requsr[3]+',\nYou have Returned The Following Books\n\n'	statsframe.pack_forget() lbl.pack_forget() librarianspage()
returnedbooksvariable+=BKNM+'\n' eml=requsr[6] nextbk=Button(statsframe,text='Return Another Book for Same User',command=nextbkfn) nextbk.pack() finshbtn=Button(statsframe,text='Finish and Send a Mail', command= finishandsendamail) finshbtn.pack()	def viewborrowedbook(): detailstoprocessframe.pack_forget() btnframe.pack_forget() detailstoprocessframe.pack_forget() statsframe.pack_forget() lbl.pack_forget() def back(): lblb.pack_forget() lbldetails.pack_forget() backbtn.pack_forget() reservebook()
global returnedbooksvariable #JUST A VARIABLE global detailstoprocessframe lbl=Label(root,text='Return Book \n\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING lbl.pack(side= TOP) detailstoprocessframe=Frame(root) #A FRAME detailstoprocessframe.pack() lidlbl=Label(detailstoprocessframe,text='Library ID') lidlbl.pack(side= LEFT) lid=Entry(detailstoprocessframe,width=20)	cursor.execute('select * from reservations') Is=cursor.fetchall() vblbks='Number of Books Reserved: '+str(len(ls))+'\n\n' for a in range(len(ls)): vblbks+='\nUSER ID : '+ str(ls[a][0]) + '\nBOOK ID : '+str(ls[a][1]) vblbks+='\n
lid.pack(side= LEFT) bkidlbl=Label(detailstoprocessframe,text='Book ID')	\n\n'
bkidlbl.pack(side= LEFT) bookid=Entry(detailstoprocessframe, width=20) bookid.pack(side= LEFT) returnedbooksvariable=" statsframe=Frame(root)	Iblb=Label(root,text='Reserved Books \n\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING

statsframe.pack()

lblb.pack(side= TOP)	if errormessagecount == 0:
IbIdetails=Frame(root) IbIdetails.pack()	plscheck=Label(statsframe,text='Please Check Your Internet connection', fg='red') plscheck.pack() errormessagecount+=1
canvas = Canvas(IbIdetails, width=650, height=400) scroll_y = Scrollbar(IbIdetails, orient="vertical", command=canvas.yview) frameforscroll = Frame(canvas) canvas.create_window(0, 0, anchor='nw', window=frameforscroll)	def nextbkfn(): global statsframe global detailstoprocessframe global reservedbooksvariable global bookid statsframe.pack_forget() detailstoprocessframe.pack_forget() detailstoprocessframe=Frame(root) #A
overduelbl=Label(frameforscroll, text= vblbks, justify=LEFT) overduelbl.pack()	FRAME detailstoprocessframe.pack() statsframe=Frame(root) statsframe.pack()
# make sure everything is displayed before configuring the scrollregion canvas.update_idletasks() canvas.configure(scrollregion=canvas.bbox('all'),	bkidlbl=Label(detailstoprocessframe,text='Book ID') bkidlbl.pack() bookid=Entry(detailstoprocessframe,
yscrollcommand=scroll_y.set) canvas.pack(fill='both', expand=True, side='left') scroll_y.pack(fill='y', side='right')	width=20) bookid.pack(side= LEFT)
backbtn=Button(root, text='Back', command = back) backbtn.pack(side=BOTTOM) root.mainloop()	global statsframe global reservedbooksvariable ####################################
def returnbk(): global errormessagecount errormessagecount=0 #just a variable def finishandsendamail(): global errormessagecount try: m=eml sub='Reserved Books'	try: statsframe.pack_forget() except: pass statsframe=Frame(root) statsframe.pack()
msg=ftxtborrowedbooksvariable+reservedbooksvariable +'\n\nWarm Regards\nLibrary Team' with smtplib.SMTP('smtp.gmail.com', 587) as smtp:	######################################
smtp.ehlo() smtp.starttls() smtp.ehlo() smtp.login('aministrator.library@gmail.com','scyjovpwhq zfbrur')	if len(bid) == 0 or len(libid) == 0:
subject = sub body = msg msg = f'Subject: {subject}\n\n{body}'	cursor.execute('select * from books where bookid= {}'.format(bid))
smtp.sendmail('aministrator.library@gmail.com',m,msg) detailstoprocessframe.pack_forget() btnframe.pack_forget() detailstoprocessframe.pack_forget() statsframe.pack_forget() lbl.pack_forget() librarianspage() except:	req = list(I) else: notfoundlbl=Label(statsframe,text='\n\nBook not Found', fg='Red') notfoundlbl.pack() finshbtn=Button(statsframe,text='Finish and Send a Mail', command= finishandsendamail)

finshbtn.pack()##	nextbk.pack()
return	finshbtn=Button(statsframe,text='Finish and
to the last of the second second libits	Send a Mail', command= finishandsendamail)
cursor.execute('select * from users where LIBID=	finshbtn.pack()
{}'.format(libid)) m=cursor.fetchall()	
if len(m) > 0:	
requsr = list(m[0])	global reservedbooksvariable #JUST A VARIABLE
requal = list(m[o])	global detailstoprocessframe
else:	Ibl=Label(root,text='Reserve Book \n\n',
notfoundurlbl=Label(statsframe,text='\n\nUser	fg='blue',font= "applecherry 35") #THIS HAS TO BE
not Found', fg='Red')	DETELED WHILE NAVIGATING
notfoundurlbl.pack()	lbl.pack(side= TOP)
return	detailstoprocessframe=Frame(root) #A FRAME
	detailstoprocessframe.pack()
	lidlbl=Label(detailstoprocessframe,text='Library ID')
	lidlbl.pack(side= LEFT)
	lid=Entry(detailstoprocessframe,width=20)
if req[4] == 'Available':	lid.pack(side= LEFT)
borrowedlbl=Label(statsframe,text='\n\nThe	bkidlbl=Label(detailstoprocessframe,text='Book ID')
book has not been borrowed yet', fg='Red')	bkidlbl.pack(side= LEFT)
borrowedlbl.pack()	bookid=Entry(detailstoprocessframe, width=20)
finshbtn=Button(btnframe,text='Finish and	bookid.pack(side= LEFT)
Send a Mail', command= finishandsendamail)	reservedbooksvariable="
finshbtn.pack() else:	statsframe=Frame(root) statsframe.pack()
cursor.execute('select * from reservations	StatStrattle.pack()
where BOOKID = \'{}\''.format(bid))	btnframe=Frame(root)# A FRAME
resbkslst=cursor.fetchall()	btnframe.pack(side=BOTTOM)
if len(resbkslst) > 0:	bumamo.paon(olao 2011 ola)
	borrowbtn=Button(btnframe, text= 'Reserve',
notfoundurlbl=Label(statsframe,text='\n\nThe Book is	command= returnbk)
Already Reserved', fg='Red')	borrowbtn.pack()
notfoundurlbl.pack()	viewbrwbtn=Button(btnframe, text='View Reserved
finshbtn=Button(statsframe,text='Finish and	Books', command= viewborrowedbook)
Send a Mail', command= finishandsendamail)	viewbrwbtn.pack()
finshbtn.pack()##	backbtn=Button(btnframe, text='Back', command=
return	back)
	backbtn.pack()
	root.mainloop()
cursor.execute('insert into reservations values	
({},{})'.format(libid,bid)) mycon.commit()	
mycon.commit()	
	def renewbook():
#######################################	global bookid
#######################################	global renewedbooksvariable
	global detailstoprocessframe
BKNM=req[1]	welcomeframe.pack_forget()
	librarianframetop.pack_forget()
sucmsg='\n\nThe Book '+BKNM +' has been	librarianframetop.pack_forget()
Reserved by '+requsr[2]	librarianframebottom.pack_forget()
borrowedlbl=Label(statsframe,text=sucmsg,	def back():
fg='Green')	global statsframe
borrowedlbl.pack()	lbl.pack_forget()
figures and the second	btnframe.pack_forget()
ftxtborrowedbooksvariable='Dear '+requsr[2]+'	detailstoprocessframe.pack_forget()
'+requsr[3]+',\nYou have Reserved the following Books\n\n'	statsframe.pack_forget()
DOOKS/II/II	librarianspage() def renew():
reservedbooksvariable+=BKNM+'\n'	global errormessagecount
16961 AGADOOV2AUIADIG 1-DI/IAIAIT /II	global errormessagecount global renewedbooksvariable
eml=requsr[6]	global detailstoprocessframe
nextbk=Button(statsframe,text='Reserve	errormessagecount=0 #just a variable
Another Book for Same User',command=nextbkfn)	def finishandsendamail():

```
global errormessagecount
                                                                      pass
         global bookid
                                                                    statsframe=Frame(root)
         global renewedbooksvariable
                                                                    statsframe.pack()
         try:
                                                                    yefrforat=0
            m=eml
            sub='Renewed Books'
                                                                    bid=bookid.get()
                                                                    libid=lid.get()
msg=ftxtborrowedbooksvariable+renewedbooksvariable
+'\n\nWarm Regards\nLibrary Team'
                                                            #[['\ufeffUser Lib ID', 'Book ID', 'Due Date\n'], ['4656',
                                                            '1', '06-07-20\n'], ['2', '2', '06-07-20\n'], ['4656', '3', '06-07-
            with smtplib.SMTP('smtp.gmail.com', 587)
as smtp:
               smtp.ehlo()
               smtp.starttls()
                                                                    if len(bid) == 0 or len(libid) == 0:
               smtp.ehlo()
                                                                      notfoundIbl=Label(statsframe,text='\n\nPlease
                                                            enter all details', fg='Red')
                                                                      notfoundlbl.pack()
smtp.login('aministrator.library@gmail.com','scyjovpwhq
                                                                      return
zfbrur')
               subject = sub
                                                                    cursor.execute('select * from borrowedbooks
                                                            where BOOKID = \'{}\".format(bid))
               body = msq
               msg = f'Subject: {subject}\n\n{body}'
                                                                    I=cursor.fetchall()
                                                                    if len(I) > 0:
smtp.sendmail('aministrator.library@gmail.com',m,msg)
                                                                      req = list(I[0])
            detailstoprocessframe.pack_forget()
                                                                    else:
            btnframe.pack forget()
                                                                      notfoundlbl=Label(statsframe,text='\n\nBook
            detailstoprocessframe.pack_forget()
                                                            not Found', fg='Red')
            statsframe.pack_forget()
                                                                      notfoundlbl.pack()
            lbl.pack forget()
                                                                      return
            librarianspage()
                                                                    if str(I[0][0]) != libid:
            if errormessagecount == 0:
                                                                      notfoundurlbl=Label(statsframe,text='\n\nThis
               plscheck=Label(statsframe,text='Please
                                                            user did not borrow the mentioned book', fg='Red')
Check Your Internet connection', fg='red')
                                                                      notfoundurlbl.pack()
               plscheck.pack()
                                                                      return
               errormessagecount+=1
       def nextbkfn():
          global statsframe
          global detailstoprocessframe
                                                                    now=datetime.now() + timedelta(days=18) #take
         global renewedbooksvariable
                                                            in function
          global bookid
                                                                    txtl=now.strftime("%d-%m-%v")
         statsframe.pack forget()
                                                                      #MODIFY HERE TO ADD 18 DAYS
          detailstoprocessframe.pack forget()
                                                                       #SHOW MESSAGE THAT THE BOOK IS
          detailstoprocessframe=Frame(root) #A
                                                            BORROWED, ASK FOR NEXT BOOK TO BORROW,
FRAME
          detailstoprocessframe.pack()
          statsframe=Frame(root)
                                                                    cursor.execute('update borrowedbooks set
                                                            duedate = DATE_ADD(duedate, INTERVAL 10 DAY)
          statsframe.pack()
                                                            where bookid = {}'.format(bid))
                                                                    mycon.commit()
bkidlbl=Label(detailstoprocessframe,text='Book ID')
          bkidlbl.pack()
                                                                    cursor.execute('select duedate from
          bookid=Entry(detailstoprocessframe,
                                                            borrowedbooks where BOOKID = \'{}\".format(bid))
                                                                    hts=cursor.fetchall()
width=20)
         bookid.pack(side= LEFT)
                                                                    txtl=hts[0][0]
       global statsframe
       global renewedbooksvariable
                                                                    sucmsg='\n\nThe Book has been Renewed \n
       global detailstoprocessframe
                                                            Due Date: '+str(txtl)
       global bookid
                                                                    borrowedlbl=Label(statsframe,text=sucmsg,
                                                            fg='Green')
                                                                    borrowedlbl.pack()
          statsframe.pack_forget()
       except:
```

cursor.execute('select * from users where LIBID= {}'.format(libid)) m=cursor.fetchall()	def back(): lbl.pack_forget() lbldetails.pack_forget()
requsr = list(m[0])	backbtn.pack_forget() librarianspage()
cursor.execute('select BOOK_Title from books where BOOKID = {}'.format(bid))	
Imn=cursor.fetchall()	odbks="
BKNM=lmn[0][0]	cursor.execute('select * from borrowedbooks where
	duedate < curdate()')
	Isl=cursor.fetchall()
######################################	if len(lsl)>0: for a in range(len(lsl)):
ftxtborrowedbooksvariable='Dear '+requsr[2]+'	odbks+='\nUser Id : '+ str(lsl[a][0]) + '\nBook
'+requsr[3]+',\nYou have Renewed the Following	Id : '+ str(lsl[a][1])+'\n\n'
Books\n\n'	odbks+='\n
renewedbooksvariable+=BKNM+'\t'+ 'Due Date:	\n\n'
'+str(txtl)+'\n'	 Ibl=Label(root,text='Overdue Books \n\n',
eml=requsr[6]	fg='blue',font= "applecherry 35") #THIS HAS TO BE
nextbk=Button(statsframe,text='Renew Another	DETELED WHILE NAVIGATING
Book for Same User',command=nextbkfn)	lbl.pack(side= TOP)
nextbk.pack() finshbtn=Button(statsframe,text='Finish and	lbldetails=Frame(root)
Send a Mail', command= finishandsendamail)	Ibidetails.pack()
finshbtn.pack()	
#######################################	
	canvas = Canvas(IbIdetails, width=650, height=400)
	scroll_y = Scrollbar(lbldetails, orient="vertical",
<pre>lbl=Label(root,text='Renew Books \n\n',</pre>	command=canvas.yview)
fg='blue',font= "applecherry 35") #THIS HAS TO BE	frameforscroll = Frame(canvas)
DETELED WHILE NAVIGATING	canvas.create_window(0, 0, anchor='nw',
lbl.pack(side= TOP) detailstoprocessframe=Frame(root) #A FRAME	window=frameforscroll)
detailstoprocessframe.pack()	overduelbl=Label(frameforscroll, text= odbks,
lidlbl=Label(detailstoprocessframe,text='Library ID')	justify=LEFT)
lidlbl.pack(side= LEFT)	overduelbl.pack()
lid=Entry(detailstoprocessframe,width=20)	
lid.pack(side= LEFT) bkidlbl=Label(detailstoprocessframe,text='Book ID')	# make sure everything is displayed before configuring the scrollregion
bkidlbl.pack(side= LEFT)	canvas.update idletasks()
bookid=Entry(detailstoprocessframe, width=20)	canvaciapaato_iaictache()
bookid.pack(side= LEFT)	canvas.configure(scrollregion=canvas.bbox('all'),
renewedbooksvariable="	yscrollcommand=scroll_y.set)
statsframe=Frame(root) statsframe.pack()	canvas.pack(fill='both', expand=True, side='left')
statstrattie.pack()	scroll y.pack(fill='y', side='right')
btnframe=Frame(root)# A FRAME	
btnframe.pack(side=BOTTOM)	backbtn=Button(root, text='Back', command =
borrowbtn=Button(btnframe, text= 'Renew',	back)
command= renew) borrowbtn.pack()	backbtn.pack(side=BOTTOM) root.mainloop()
backbtn=Button(btnframe, text='Back', command=	root.maiinoop()
back)	
backbtn.pack()	######################################
root.mainloop()	2020 ##################################
def overduebook():	def messaging():
welcomeframe.pack_forget()	
librarianframetop.pack_forget()	######################################
librarianframetop.pack_forget() librarianframebottom.pack_forget()	######################################
noranamiamooditti.paok_torget()	I The order to the sages

#even after clearing messages are displayed. they	lsl.pop(-1)
are to be rectified	except:
welcomeframe.pack_forget()	break
librarianframetop.pack_forget()	if len(lsl)>0:
librarianframetop.pack_forget()	for a in Isl:
librarianframebottom.pack_forget()	cursor.execute('insert into messaging
,	
def back():	values {}'.format(a))
global inboxframe	mycon.commit()
global sendframe	overduelbl.pack_forget()
lbldetails.pack_forget()	try:
lbl.pack_forget()	overduelbl12.pack_forget()
backbtn.pack_forget()	except:
try:	pass
inboxframe.pack_forget()	overduelbl12=Label(inboxframe, text= 'Last 5
except:	messages are Cleared\n', fg='green')
pass	overduelbl12.pack()
try:	odbks="
manucomframe.pack_forget()	if len(lsl)>0:
except:	for a in range(len(lsl)):
pass	odbks+='\n\n\nFROM: '+ str(lsl[a][1]) +
try:	'\tUSER Id: '+ str(Isl[a][0]) + '\nMESSAGE:\t\t\t\n\n'+
sendframe.pack_forget()	str(lsl[a][3])+'\n'
except:	odbks+='
pass	
·	\n'
librarianspage()	· · ·
	canvas = Canvas(inboxframe, width=1000,
def inbox():	height=400)
global overduelbl	scroll_y = Scrollbar(inboxframe,
global canvas	orient="vertical", command=canvas.yview)
global scroll_y	frameforscroll = Frame(canvas)
def clrmsgfn():	canvas.create_window(0, 0, anchor='nw',
global manucomframe	window=frameforscroll)
global overduelbl	,
global overduelbl12	overduelbl=Label(frameforscroll, text=
manucomframe.pack_forget()	odbks, justify=LEFT)
	1
scroll_y.pack_forget()	overduelbl.pack()
cursor.execute('delete from messaging where	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
toid = 2')	# make sure everything is displayed before
mycon.commit()	configuring the scrollregion
overduelbl.pack_forget()	canvas.update_idletasks()
canvas.pack forget()	. – "
try:	
overduelbl12.pack_forget()	canvas.configure(scrollregion=canvas.bbox('all'),
except:	yscrollcommand=scroll y.set)
	ysorolicommand_soroll_y.sor/
pass	annua nach/fill—lbathlasunand—Two
overduelbl=Label(inboxframe, text= 'All	canvas.pack(fill='both', expand=True,
Messages are Cleared', fg='green')	side='left')
overduelbl.pack()	scroll_y.pack(fill='y', side='right')
manucomframe.pack_forget()	
def clr10msgfn():	manucomframe=Frame(root)
global canvas	manucomframe.pack(side=BOTTOM)
global overduelbl	clrmsg=Button(manucomframe, text='Clear
global manucomframe	All Messages', command=clrmsgfn)
global scroll_y	clr10msg=Button(manucomframe,
global overduelbl12	text='Clear Last 5 Messages', command=clr10msgfn)
cursor.execute('delete from messaging where	clrmsg.pack()
toid = 2')	clr10msg.pack()
mycon.commit()	else:
manucomframe.pack_forget()	canvas.pack_forget()
canvas.pack_forget()	overduelbl=Label(inboxframe, text= 'There
scroll_y.pack_forget()	are no Messages')
overduelbl.pack_forget()	overduelbl.pack()
overdueibi.pack_ibiget()	overdueibi.pack()
for a large as (F)	
for a in range(5):	
trv:	global inboxframe

```
global sendframe
                                                                     overduelbl=Label(inboxframe, text= 'There are
       global manucomframe
                                                           no Messages')
                                                                     overduelbl.pack()
         inboxframe.pack forget()
         manucomframe.pack forget()
                                                                def send():
       except:
                                                                  global errorframe
         pass
       try:
         sendframe.pack forget()
                                                                  def submsg():
                                                                     global errorframe
         pass
                                                                     errorframe.pack forget()
       inboxframe=Frame(root)
                                                                     errorframe=Frame(sendframe)
       inboxframe.pack(side=TOP)
                                                                     errorframe.pack()
       Ibl=Label(inboxframe,text='INBOX \n',
fg='blue',font= "applecherry 25") #THIS HAS TO BE
                                                                     recid=nricofuserentry.get()
DETELED WHILE NAVIGATING
                                                                     messagesent=TextArea.get("1.0",END)
       lbl.pack(side= TOP)
                                                                     if len(recid)==0:
       odbks="
                                                           notfoundurlbl=Label(errorframe,text='\n\nPlease Enter
                                                           Recipient Library ID', fg='Red')
       cursor.execute('select * from messaging where
toid = 2')
                                                                       notfoundurlbl.pack()
       Isl=cursor.fetchall()
                                                                       return
       Isl=Isl[::-1]
                                                                     if len(messagesent)<=1:
       if len(lsl)>0:
         for a in range(len(lsl)):
                                                           notfoundurlbl=Label(errorframe,text='\n\nPlease Type
            odbks+=\n\ + str(\ [a][1]) +
                                                           your Message ', fg='Red')
"\tUSER Id: '+ str(IsI[a][0]) + '\n MESSAGE:\t\t\t\n\n '+
                                                                       notfoundurlbl.pack()
str(IsI[a][3])+'\n'
                                                                       return
                                                                     cursor.execute('select * from users where
           odbks+='-----
                                                           LIBID= {}'.format(recid))
                                                                     m=cursor.fetchall()
         canvas = Canvas(inboxframe, width=1000,
                                                                     if len(m) > 0:
height=400)
                                                                       pass
          scroll_y = Scrollbar(inboxframe,
orient="vertical", command=canvas.yview)
                                                                     else:
         frameforscroll = Frame(canvas)
         canvas.create window(0, 0, anchor='nw',
                                                           notfoundurlbl=Label(errorframe,text='\n\nUser not
window=frameforscroll)
                                                           Found', fg='Red')
                                                                       notfoundurlbl.pack()
         overduelbl=Label(frameforscroll, text= odbks,
                                                                       return
justify=LEFT)
         overduelbl.pack()
                                                                     if len(messagesent)>700:
         # make sure everything is displayed before
                                                           notfoundurlbl=Label(errorframe,text='\n\nMax
configuring the scrollregion
                                                           Characters Exceeded', fg='Red')
         canvas.update idletasks()
                                                                       notfoundurlbl.pack()
                                                                       return
canvas.configure(scrollregion=canvas.bbox('all'),
                                                                     sub='MESSAGE FROM LIBRARIAN'
yscrollcommand=scroll_y.set)
                                                                     msg='Dear '+str(m[0][2])+',\nYou have
                                                           Recived a message from Librarian \nThe message
         canvas.pack(fill='both', expand=True,
                                                           is:\n\n'+str(messagesent)+'\n\nWarm Regards\nLibrary
side='left')
                                                           Team'
          scroll y.pack(fill='y', side='right')
         manucomframe=Frame(root)
                                                                       with smtplib.SMTP('smtp.gmail.com', 587)
         manucomframe.pack(side=BOTTOM)
                                                           as smtp:
         clrmsg=Button(manucomframe, text='Clear All
                                                                          smtp.ehlo()
Messages', command=clrmsgfn)
                                                                          smtp.starttls()
         clr10msg=Button(manucomframe, text='Clear
                                                                          smtp.ehlo()
Last 5 Messages', command=clr10msgfn)
         clrmsg.pack()
                                                           smtp.login('aministrator.library@gmail.com','scyjovpwhq
         clr10msg.pack()
                                                           zfbrur')
                                                                          subject = sub
       else:
                                                                          body = msg
```

```
msg = f'Subject: {subject}\n\n{body}'
                                                         nricofuserlabel12=Label(nricofpersonframe,text='\t\t\t\t\t\t')
smtp.sendmail('aministrator.library@gmail.com',m[0][6],
                                                                nricofuserlabel12.pack(side=LEFT)
msg)
         except:
                                                                #message
           pass
                                                         usernameofpersonframe=Frame(detailsofpersonframefo
                                                         rcreatingaccount)
                                                                usernameofpersonframe.pack()
detailsofpersonframeforcreatingaccount.pack forget()
         notfoundurlbl=Label(errorframe,text='\n\nYour
                                                         usernameofuserlabel=Label(usernameofpersonframe,te
Message has been sent', fg='Green')
                                                         xt='\n\tYour Message \t')
         notfoundurlbl.pack()
                                                                TextArea =
                                                         Text(usernameofpersonframe, selectborderwidth=3)
                                                                usernameofuserlabel.pack(side=LEFT)
                                                                TextArea.pack(expand=YES, fill=BOTH)
         cursor.execute("insert into messaging values
(2,'LIBRARIAN',{},'{}')".format(recid,messagesent))
                                                         compusername=Label(usernameofpersonframe,text='M
                                                         ax 700 Characters\n\n\n')
         mycon.commit()
         root.mainloop()
                                                                compusername.pack(side=LEFT)
                                                         subbtn=Button(detailsofpersonframeforcreatingaccount,
                                                         text='Send', command=submsg)
                                                                subbtn.pack()
                                                                errorframe=Frame(sendframe)
                                                                errorframe.pack()
                                                                root.mainloop()
       global inboxframe
       global sendframe
       try:
         inboxframe.pack forget()
         manucomframe.pack forget()
                                                             lbl=Label(root,text='Messaging\n', fg='blue',font=
       except:
                                                         "applecherry 35") #THIS HAS TO BE DETELED WHILE
         pass
       try:
                                                         NAVIGATING
         sendframe.pack forget()
                                                             lbl.pack(side= TOP)
       except:
         pass
                                                             IbIdetails=Frame(root)
       sendframe=Frame(root)
                                                             Ibldetails.pack()
       sendframe.pack()
       Ibl=Label(sendframe,text='COMPOSE MAIL \n',
                                                             btninbox=Button(lbldetails, text='Inbox',
fg='blue',font= "applecherry 25") #THIS HAS TO BE
                                                         command=inbox)
DETELED WHILE NAVIGATING
                                                             btnsend=Button(IbIdetails, text='Compose Mail',
       Ibl.pack(side= TOP)
                                                         command=send)
                                                             btninbox.pack(side=BOTTOM)
                                                             btnsend.pack(side=BOTTOM)
detailsofpersonframeforcreatingaccount=Frame(sendfra
me)
       detailsofpersonframeforcreatingaccount.pack()
                                                              backbtn=Button(root, text='Back', command =
       #TO ID
                                                         back)
                                                              backbtn.pack(side=BOTTOM)
nricofpersonframe=Frame(detailsofpersonframeforcreati
                                                             root.mainloop()
ngaccount)
       nricofpersonframe.pack()
                                                           def addnewbook():
                                                              nricofuserlabel=Label(nricofpersonframe,text='\nRecipie
nt Library ID \t')
                                                             global errorframe
                                                             def back():
nricofuserentry=Entry(nricofpersonframe,width=10)
                                                                titleoffeedbackpage.pack forget()
                                                                detailsofpersonframe. pack forget()
       nricofuserlabel.pack(side=LEFT)
       nricofuserentry.pack(side=LEFT)
                                                                suggestiongivingframe. pack_forget()
```

```
buttonsframe.pack_forget()
                                                            fg='green', justify=LEFT)
       librarianspage()
                                                                     plscheckentry.pack()
    def submit():
       global errorframe
       errorframe.pack_forget()
       errorframe=Frame(root)
       errorframe.pack()
       cursor.execute('select * from books')
       existingbooklist=cursor.fetchall()
       noofbooks=len(existingbooklist)
       newbookid=noofbooks+1
                                                                welcomeframe.pack forget()
                                                                librarianframetop.pack forget()
                                                                librarianframetop.pack forget()
       bookname = nameofuserentry.get()
       author=nricofuserentry.get()
                                                                librarianframebottom.pack forget()
       publisher=emailofuserentry.get()
       availability='Available'
                                                                titleoffeedbackpage=Frame(root)
       if len(bookname)==0 or len(author)==0 or
                                                                titleoffeedbackpage.pack()
                                                                feedbacktitletext = Label(titleoffeedbackpage, text =
len(publisher)==0:
         plscheckentry=Label(errorframe,text='Please
                                                            'ADD NEW BOOK', fg='blue',font= "applecherry 35")
                                                                feedbacktitletext.pack(side= TOP)
Give all necessary informations', fg='red')
         plscheckentry.pack()
         return
                                                                 suggestiongivingframe=Frame(root)
       elif len(bookname)>40:
                                                                suggestiongivingframe.pack()
plscheckentry=Label(errorframe,text='Maximum
                                                            leaveyoursuggestiontext=Label(suggestiongivingframe,
Characters Exceeded for Book Title', fg='red')
                                                            text='\n\n\n\nEnter Information About New Book
         plscheckentry.pack()
                                                            Here\n\n\n')
         return
                                                                 leaveyoursuggestiontext.pack(side= LEFT)
       elif len(author)>40:
                                                                suggestiongivingframe.pack()
plscheckentry=Label(errorframe,text='Maximum
                                                                detailsofpersonframe=Frame(root)
Characters Exceeded for Author', fg='red')
                                                                detailsofpersonframe.pack()
         plscheckentry.pack()
         return
                                                                #name
       elif len(publisher)>100:
                                                                nameframe=Frame(detailsofpersonframe)
                                                                nameframe.pack()
plscheckentry=Label(errorframe,text='Maximum
                                                                nameofuserlabel=Label(nameframe,text='Book
Characters Exceeded for Publisher', fg='red')
                                                            Title ',justify=RIGHT)
         plscheckentry.pack()
                                                                 nameofuserentry=Entry(nameframe,width=30)
                                                                nameofuserlabel.pack(side=LEFT)
         return
                                                                nameofuserentry.pack(side=LEFT)
       else:
         cursor.execute("insert into books values
                                                                #nric
                                                                nricofpersonframe=Frame(detailsofpersonframe)
({},'{}','{}','{}')".format(newbookid,
                                                                nricofpersonframe.pack()
bookname,
                                                            nricofuserlabel=Label(nricofpersonframe,text='\nAuthor
author,
                                                            ,justify=RIGHT)
                                                                 nricofuserentry=Entry(nricofpersonframe,width=30)
                                                                nricofuserlabel.pack(side=LEFT)
publisher,
                                                                nricofuserentry.pack(side=LEFT)
availability))
         mycon.commit()
                                                                #email
         detailsofpersonframe. pack forget()
                                                                emailofpersonframe=Frame(detailsofpersonframe)
         suggestiongivingframe. pack forget()
                                                                emailofpersonframe.pack()
          submitbutton.pack forget()
                                                            emailofuserlabel=Label(emailofpersonframe,text='\nPubl
         plscheckentry=Label(errorframe,text='The
                                                            ication ',justify=RIGHT)
New Book Has Been Uploaded\n\n\
                                                            emailofuserentry=Entry(emailofpersonframe,width=30)
```

Book Id: {}'.format(newbookid),

errorframe.pack forget()

emailofuserlabel.pack(side=LEFT) emailofuserentry.pack(side=LEFT)	pass
	try:
errorframe=Frame(root)	updateuserframe.pack_forget()
errorframe.pack()	except:
1 (/	pass
buttonsframe=Frame(root)	try:
buttonsframe.pack(side=BOTTOM)	deleteuserframe.pack_forget()
battoristratric.pack(side-bo i row)	except:
submitbutton= Button(buttonsframe, text="Submit",	
	pass
command=submit)	try:
submitbutton.pack()	errorframe.pack_forget()
	errorframe=Frame(root)
backbutton = Button(buttonsframe, text="Back",	errorframe.pack()
command=back)	except:
backbutton.pack()	pass
	######################################
root.mainloop()	2020###################################
	def updatethatuser():
def updateuser():	def goverifyotp():
######################################	def verifyotp():
#######################################	global errorframe
global errorframe	try:
def back():	errorframe.pack_forget()
global errorframe	errorframe=Frame(root)
global detailsofpersonframeforcreatingaccount	errorframe.pack()
	except:
titleoffeedbackpage. pack_forget()	pass
errorframe.pack_forget()	otp = otpentry.get()
buttonsframe.pack_forget()	if len(otp)==0:
try:	
	notfoundurlbl=Label(errorframe,text='\n\nPlease Enter
detailsofpersonframeforcreatingaccount.pack_forget()	OTP', fg='Red')
except:	notfoundurlbl.pack()
pass	return
try:	
otpevalfr.pack_forget()	
except:	elif str(val)!= otp:
pass	om ou (var). Oct.
try:	notfoundurlbl=Label(errorframe,text='\n\nInvalid OTP',
updateuserframe.pack_forget()	fg='Red')
except:	notfoundurlbl.pack()
•	
pass	return
try:	else:
deleteuserframe.pack_forget()	
except:	
pass	cursor.execute("update users set
	First_Name = '{}',Last_Name = '{}', phone_Number = '{}'
librarianspage()	Identity= '{}', email_id = '{}' where LIBID =
	{}".format(fname,lname,phone,nric,email,lid))
def updateuserdet():	mycon.commit()
global deleteuserframe	
global otpevalfr	
global detailsofpersonframeforcreatingaccount	
global updateuserframe	
global errorframe	notfoundurlbl=Label(errorframe,text='\n\nAccount Has
global emailofpersonframe	Been Updated', fg='Green')
try:	notfoundurlbl.pack()
distribution of a section of	
detailsofpersonframeforcreatingaccount.pack_forget()	##############hkv######################
except:	global emailofpersonframe
pass	global
try:	detailsofpersonframeforcreatingaccount
otpevalfr.pack_forget()	global otpevalfr
except:	global errorframe

```
global detailsofpersonframeforcreatingaccount
            trv:
                                                                      global emailofpersonframe
               errorframe.pack forget()
               errorframe=Frame(root)
                                                                      alobal errorframe
               errorframe.pack()
                                                                      global emailofuserentry
            except:
                                                                      try:
                                                                         errorframe.pack forget()
              pass
                                                                         errorframe=Frame(root)
            fname=nameofuserentry.get()
            Iname=Inameofuserentry.get()
                                                                         errorframe.pack()
            nric=nricofuserentry.get()
                                                                      except:
            email=emailofuserentry.get()
                                                                         pass
            phone=phoneofuserentry.get()
                                                                      lid=emailofuserentry.get()
                                                                      if len(lid) == 0:
            if len(fname)==0 or len(lname)==0 or
                                                            notfoundurlbl=Label(errorframe,text='\n\nPlease Enter
len(phone)==0 or len(nric)==0 or len(email)==0:
                                                            Library ID of User', fg='Red')
enterallinfoerrormessagelabel=Label(errorframe,text='PI
                                                                         notfoundurlbl.pack()
ease Enter all Necessary Details', fg='red')
                                                                         return
              enterallinfoerrormessagelabel.pack()
                                                                      cursor.execute('select * from users where
               return
                                                            LIBID= {}'.format(lid))
                                                                      m=cursor.fetchall()
                                                                      if len(m) == 0:
                                                            notfoundurlbl=Label(errorframe,text='\n\nInvalid Library
                                                            ID', fg='Red')
            val=random.randint(1000,9999)
                                                                         notfoundurlbl.pack()
            sub='OTP Verification'
                                                                         return
            msg='Dear '+fname+',\nYour OTP for
                                                                      if int(lid) == 1 or int(lid) == 2:
Updating Your Libriary account is '+str(val)+'\n\nWarm
Regards\nLibrary Team'
                                                            notfoundurlbl=Label(errorframe,text='\n\nDefault
            with smtplib.SMTP('smtp.gmail.com', 587)
                                                            Account Cannt be Updated', fg='Red')
                                                                         notfoundurlbl.pack()
as smtp:
                                                                         return
               smtp.ehlo()
               smtp.starttls()
               smtp.ehlo()
                                                                      #NEW ENTRIES
                                                                      emailofpersonframe.pack forget()
smtp.login('aministrator.library@gmail.com','scyjovpwhq
zfbrur')
               subject = sub
                                                            detailsofpersonframeforcreatingaccount=Frame(root)
               body = msq
                                                                      detailsofpersonframeforcreatingaccount.pack()
               msg = f'Subject: {subject}\n\n{body}'
smtp.sendmail('aministrator.library@gmail.com',email,m
                                                            instrlbl=Label(detailsofpersonframeforcreatingaccount.te
                                                            xt='Update New User Details Here',font='verdana 15')
sg)
                                                                      instrlbl.pack()
detailsofpersonframeforcreatingaccount.pack forget()
                                                            nameframe=Frame(detailsofpersonframeforcreatingacco
            otpevalfr = Frame(updateuserframe)
                                                            unt)
            otpevalfr.pack()
                                                                      nameframe.pack()
            labelforspace = Label( otpevalfr, text ="
                                                                      nameofuserlabel=Label(nameframe,text='
\n\n\nPlease Enter OTP Received by User").pack()
                                                            First Name
                                                                      nameofuserentry=Entry(nameframe,width=20)
                                                                      nameofuserlabel.pack(side=LEFT)
            otpentry = Entry (otpevalfr, width=10)
            otpentry.pack()
                                                                      nameofuserentry.pack(side=LEFT)
submitbtn=Button(otpevalfr,text='Submit',command=veri
                                                            Inameframe=Frame(detailsofpersonframeforcreatingacc
fyotp)
            submitbtn.pack()
                                                            ount)
                                                                      Inameframe.pack()
                                                                      Inameofuserlabel=Label(Inameframe,text='
                                                            Last Name
                                                                               \t')
                                                            Inameofuserentry=Entry(Inameframe,width=20)
```

```
user through reset password in login
         Inameofuserentry.pack(side=LEFT)
                                                           page\n',font='verdana 20')
                                                                  instrlbl.pack()
         #nric
nricofpersonframe=Frame(detailsofpersonframeforcreati
                                                                  global emailofuserentry
ngaccount)
         nricofpersonframe.pack()
                                                           emailofuserlabel=Label(emailofpersonframe,text='LIBRA
                                                           RY ID')
nricofuserlabel=Label(nricofpersonframe,text='\nLast 4
characters NRIC/Passport\t')
                                                           emailofuserentry=Entry(emailofpersonframe,width=10)
                                                                  emailofuserlabel.pack(side=LEFT)
nricofuserentry=Entry(nricofpersonframe,width=20)
                                                                  emailofuserentry.pack(side=LEFT)
         nricofuserlabel.pack(side=LEFT)
                                                                  bufferIbl=Label(emailofpersonframe,text='\n\n')
         nricofuserentry.pack(side=LEFT)
                                                                  bufferlbl.pack()
         #phonenumber
                                                           btnsubgo=Button(emailofpersonframe,text='Submit',
phoneofpersonframe=Frame(detailsofpersonframeforcre
                                                           command=updatethatuser)
atingaccount)
                                                                  btnsubgo.pack()
         phoneofpersonframe.pack()
phoneofuserlabel=Label(phoneofpersonframe,text='\n\tP
hone Number\t')
                                                                def deleteusr():
                                                                  global deleteuserframe
phoneofuserentry=Entry(phoneofpersonframe,width=20)
                                                                  global emailofpersonframe
         phoneofuserlabel.pack(side=LEFT)
                                                                  global otpevalfr
         phoneofuserentry.pack(side=LEFT)
                                                                  global detailsofpersonframeforcreatingaccount
                                                                  global updateuserframe
                                                                  global errorframe
         #email
                                                                  try:
emailofperson1frame=Frame(detailsofpersonframeforcr
                                                           detailsofpersonframeforcreatingaccount.pack forget()
eatingaccount)
                                                                  except:
         emailofperson1frame.pack()
                                                                     pass
                                                                  try:
emailofuserlabel=Label(emailofperson1frame,text='\n\t\t
                                                                     otpevalfr.pack forget()
Email Address\t')
                                                                  except:
                                                                     pass
emailofuserentry=Entry(emailofperson1frame,width=30)
                                                                  try:
         emailofuserlabel.pack(side=LEFT)
                                                                     updateuserframe.pack_forget()
         emailofuserentry.pack(side=LEFT)
                                                                  except:
                                                                     pass
                                                                  try:
submtbtn=Button(detailsofpersonframeforcreatingaccou
                                                                     deleteuserframe.pack forget()
nt,text='Submit',command=goverifyotp)
                                                                  except:
                                                                    pass
         submtbtn.pack()
                                                                  try:
                                                                    errorframe.pack forget()
                                                                     errorframe=Frame(root)
                                                                     errorframe.pack()
                                                                  except:
                                                                     pass
       updateuserframe=Frame(root)
       updateuserframe.pack()
                                                                  def deletethatuser():
       global emailofpersonframe
                                                                     global emailofpersonframe
                                                                     def verifyotp():
       emailofpersonframe=Frame(updateuserframe)
                                                                       global errorframe
       emailofpersonframe.pack()
                                                                          errorframe.pack forget()
       instrlbl=Label(emailofpersonframe,text='Update
                                                                          errorframe=Frame(root)
User Details\nNOTE: Password can be only reseted by
                                                                          errorframe.pack()
                                                                       except:
```

Inameofuserlabel.pack(side=LEFT)

```
otp = otpentry.get()
            if len(otp)==0:
                                                                        global errorframe
notfoundurlbl=Label(errorframe,text='\n\nPlease Enter
                                                                        try:
OTP', fg='Red')
                                                                          errorframe.pack forget()
               notfoundurlbl.pack()
                                                                          errorframe=Frame(root)
                                                                          errorframe.pack()
                                                                        except:
            elif str(val)!= otp:
                                                                          pass
                                                                        lid=emailofuserentry.get()
notfoundurlbl=Label(errorframe,text='\n\nInvalid OTP',
                                                                        if len(lid) == 0:
fg='Red')
                                                              notfoundurlbl=Label(errorframe,text='\n\nPlease Enter
               notfoundurlbl.pack()
                                                             Library ID of User', fg='Red')
               return
                                                                          notfoundurlbl.pack()
            else:
                                                                          return
               cursor.execute("select * from
borrowedbooks where LIBID ={}".format(lid))
               checklst=cursor.fetchall()
                                                                        cursor.execute('select * from users where
                                                             LIBID= {}'.format(lid))
               if len(checklst)>0:
                                                                        m=cursor.fetchall()
notfoundurlbl=Label(errorframe.text='\n\nThe User has a
                                                                        if len(m) == 0:
Borrowed Book yet to be Returned\n Please Return All
Books Borrowed by User and Try Again', fg='Red')
                                                              notfoundurlbl=Label(errorframe,text='\n\nInvalid Library
                 notfoundurlbl.pack()
                                                              ID', fg='Red')
                 return
                                                                          notfoundurlbl.pack()
                                                                          return
               cursor.execute("delete from users where
                                                                        if int(lid) == 1 or int(lid) == 2:
LIBID ={}".format(lid))
               mycon.commit()
                                                              notfoundurlbl=Label(errorframe,text='\n\nDefault
               cursor.execute("delete from reservations
                                                              Account Cannt be Deleted', fg='Red')
where LIBID ={}".format(lid))
                                                                          notfoundurlbl.pack()
                                                                          return
               mycon.commit()
               cursor.execute("delete from messaging
where Fromid ={} or toid = {}".format(lid,lid))
                                                                        val=random.randint(1000,9999)
                                                                        sub='OTP Verification'
               mycon.commit()
                                                                        msg='Dear '+str(m[0][2])+',\nYour OTP for
                                                              Deleting Your Libriary account is '+str(val)+'\n\nWarm
                                                              Regards\nLibrary Team'
fhavpswd=open('./STORAGE/encrypteduserdetails.txt','r
                                                                        with smtplib.SMTP('smtp.gmail.com', 587) as
b')
                                                              smtp:
               lhavpswd=pickle.load(fhavpswd)
                                                                          smtp.ehlo()
               fhavpswd.close()
                                                                          smtp.starttls()
                                                                          smtp.ehlo()
               for a in range (len(lhavpswd)):
                                                              smtp.login('aministrator.library@gmail.com','scyjovpwhq
                                                              zfbrur')
                 if str(lhavpswd[a][0]) == str(lid):
                                                                          subject = sub
                    lhavpswd.pop(a)
                                                                          body = msg
                    break
                                                                          msg = f'Subject: {subject}\n\n{body}'
                                                              smtp.sendmail('aministrator.library@gmail.com',m[0][6],
fhavpswd=open('./STORAGE/encrypteduserdetails.txt','
                                                              msg)
wb')
               pickle.dump(lhavpswd,fhavpswd)
                                                                        emailofpersonframe.pack forget()
               fhavpswd.close()
                                                                        otpevalfr = Frame(deleteuserframe)
                                                                        otpevalfr.pack()
                                                                        labelforspace = Label( otpevalfr, text ="
                                                             \n\nPlease Enter OTP Received by User").pack()
                                                                        otpentry = Entry (otpevalfr, width=10)
notfoundurlbl=Label(errorframe,text='\n\nAccount Has
                                                                        otpentry.pack()
Been Deleted', fg='Green')
```

pass

notfoundurlbl.pack()

```
submitbtn=Button(otpevalfr,text='Submit',command=veri
                                                          text="Update User Details", command=updateuserdet)
                                                               addnewbannerbutton.pack()
fyotp)
         submitbtn.pack()
                                                               deletebannerbutton= Button(buttonsframe,
                                                          text="Delete User Account", command=deleteusr)
                                                               deletebannerbutton.pack()
                                                               backbutton = Button(buttonsframe, text="Back",
                                                          command=back)
       deleteuserframe=Frame(root)
                                                               backbutton.pack()
       deleteuserframe.pack()
                                                               root.mainloop()
       emailofpersonframe=Frame(deleteuserframe)
       emailofpersonframe.pack()
       instrlbl=Label(emailofpersonframe,text='Delete
User Account\n\n',font='verdana 20')
       instrlbl.pack()
emailofuserlabel=Label(emailofpersonframe,text='LIBRA
RY ID')
emailofuserentry=Entry(emailofpersonframe,width=10)
       emailofuserlabel.pack(side=LEFT)
       emailofuserentry.pack(side=LEFT)
       bufferlbl=Label(emailofpersonframe,text='\n\n')
                                                             def updateevents():
                                                               ############################ 10th Oct 2020
       bufferlbl.pack()
                                                          global errorframe
                                                               global updateeventpagefrm
btnsubgo=Button(emailofpersonframe,text='Submit',
                                                               def back():
command=deletethatuser)
                                                                 global deletebannerframe
       btnsubgo.pack()
                                                                 global errorframe
                                                                 global addbannerframe
                                                                 titleoffeedbackpage. pack forget()
                                                                 updateeventpagefrm.pack forget()
                                                                 errorframe.pack forget()
    welcomeframe.pack forget()
                                                                 buttonsframe.pack forget()
    librarianframetop.pack forget()
                                                                 try:
    librarianframetop.pack forget()
                                                                    deletebannerframe.pack forget()
    librarianframebottom.pack forget()
                                                                 except:
                                                                    pass
                                                                 try:
    titleoffeedbackpage=Frame(root)
                                                                    addbannerframe.pack forget()
    titleoffeedbackpage.pack()
                                                                 except:
    feedbacktitletext = Label(titleoffeedbackpage, text =
                                                                    pass
'UPDATE USER DETAILS / DELETE ACCOUNT',
                                                                 librarianspage()
fg='blue',font= "applecherry 35")
    feedbacktitletext.pack(side= TOP)
                                                               def addnewbanner():
                                                                 global deletebannerframe
                                                                 global addbannerframe
    buttonsframe=Frame(root)
    buttonsframe.pack(side=BOTTOM)
                                                                 global errorframe
                                                                 global updateeventpagefrm
    global errorframe
    try:
                                                                 try:
       errorframe.pack forget()
                                                                    deletebannerframe.pack forget()
    except:
                                                                    errorframe.pack forget()
                                                                 except:
       pass
                                                                    pass
    errorframe=Frame(root)
                                                                 try:
    errorframe.pack()
                                                                    addbannerframe.pack forget()
                                                                 except:
```

addnewbannerbutton= Button(buttonsframe.

```
if photobanner.width() > 1134:
                                                            notfoundurlbl=Label(errorframe,text='\nMaximum Width
       def submsg():
         global errorframe
                                                            Exceeded', fg='Red')
         global overduelbl
                                                                        notfoundurlbl.pack()
         global addbannerframe
                                                                        return
         global updateeventpagefrm
            errorframe.pack forget()
                                                                      lsl.append((frmtfilenamegn,bannerdesc))
            errorframe=Frame(root)
            errorframe.pack()
         except:
                                                            filehvingbannerimg=open('./STORAGE/bannerssource.t
                                                            xt','wb')
            pass
                                                                      pickle.dump(IsI,filehvingbannerimg)
                                                                      filehvingbannerimg.close()
filehvingbannerimg=open('./STORAGE/bannerssource.t
                                                                      plscheckentry=Label(errorframe,text='Banner
xt','rb')
         Isl=pickle.load(filehvingbannerimg)
                                                            Has been Added', fg='Green')
                                                                      plscheckentry.pack()
         filehvingbannerimg.close()
         bannerdesc=TextArea.get("1.0",END)
                                                                      addbannerframe.pack forget()
                                                                      updateeventpagefrm=Frame(root)
         filenamegn=nameofuserentry.get()
                                                                      updateeventpagefrm.pack()
frmtfilenamegn='./IMAGES/banners/'+filenamegn+'.png'
                                                                      odbks="
         ###checks
                                                                      for a in range(len(lsl)):
                                                                        odbks+='\n\n\nBanner Number: '+ str(a+1)
         if len(filenamegn)==0 or len(bannerdesc)==0:
                                                            + '\n\nFile: '+ str(IsI[a][0]) + '\n\nBanner
                                                            Description:\t\t\n\n '+ str(lsl[a][1])+'\n'
notfoundurlbl=Label(errorframe,text='\n\nPlease Enter
                                                                        odbks+='----
All Details ', fg='Red')
                                                            -----\n'
            notfoundurlbl.pack()
            return
                                                                      canvas = Canvas(updateeventpagefrm,
                                                            width=1000, height=400)
         if len(bannerdesc)<=1:
                                                                      scroll y = Scrollbar(updateeventpagefrm,
                                                            orient="vertical", command=canvas.yview)
notfoundurlbl=Label(errorframe,text=\\n\\nPlease Type
Banner Description ', fg='Red')
                                                                     frameforscroll = Frame(canvas)
            notfoundurlbl.pack()
                                                                      canvas.create window(0, 0, anchor='nw',
                                                            window=frameforscroll)
            return
         if len(bannerdesc)>=300:
                                                                      overduelbl=Label(frameforscroll, text= odbks,
notfoundurlbl=Label(errorframe,text='\n\nMaximum
                                                            justify=LEFT)
Characters Exceeded for Banner Description', fg='Red')
                                                                      overduelbl.pack()
            notfoundurlbl.pack()
            return
                                                                      # make sure everything is displayed before
                                                            configuring the scrollregion
            open(frmtfilenamegn)
                                                                     canvas.update_idletasks()
notfoundurlbl=Label(errorframe,text='\n\nFile Name is
                                                            canvas.configure(scrollregion=canvas.bbox('all'),
                                                            yscrollcommand=scroll y.set)
Incorrect', fg='Red')
            notfoundurlbl.pack()
            return
                                                                      canvas.pack(fill='both', expand=True,
                                                            side='left')
                                                                      scroll y.pack(fill='y', side='right')
         photobanner=PhotoImage(frmtfilenamegn)
                                                                      root.mainloop()
         if photobanner.height() > 397:
notfoundurlbl=Label(errorframe,text='\nMaximum Height
Exceeded', fg='Red')
```

return

pass

notfoundurlbl.pack()

```
def deletethatnow():
       updateeventpagefrm.pack forget()
                                                                    global errorframe
                                                                    global overduelbl
       addbannerframe=Frame(root)
                                                                    global updateeventpagefrm
       addbannerframe.pack()
                                                                      errorframe.pack forget()
       instrtxt='Please follow these instructions to add
                                                                      errorframe=Frame(root)
new banner\n\n\
                                                                      errorframe.pack()
--> The banner should be only a .png image \n\
                                                                    except:
--> Save the Image with an appropriate name\n\
                                                                      pass
--> The Size of Banner should be maximum \n\
                                                                    bannerno=emailofuserentry.get()
\tWdith = 40.01 cm and Height = 14.01 cm or\n\
\tWdith = 1134 pixels and Height = 397 pixels \n\
--> Image Should Have Resolution 72 pixels/inch\n\
                                                          filehvingbannerimg=open('./STORAGE/bannerssource.t
--> Save the Image in LIBRARY >> IMAGES >>
                                                          xt','rb')
                                                                    lsl=pickle.load(filehvingbannerimg)
banners\n\
--> Now Enter Just the name of the image without
                                                                    filehvingbannerimg.close()
directory names and extension\n\
--> In the next text box, give a small description about
                                                                    if len(bannerno)==0:
the banner'
                                                          plscheckentry=Label(errorframe,text='PLease Enter
                                                          Banner Number', fg='red')
instlbl=Label(addbannerframe,text=instrtxt,justify=LEFT)
                                                                      plscheckentry.pack()
       instlbl.pack()
                                                                      return
       nameframe=Frame(addbannerframe)
                                                                    try:
       nameframe.pack()
                                                                      int(bannerno)
       nameofuserlabel=Label(nameframe,text='File
                                                                    except:
Name
       nameofuserentry=Entry(nameframe,width=30)
                                                          plscheckentry=Label(errorframe,text='Invalid Entry for
       nameofuserlabel.pack(side=LEFT)
                                                          Banner Number', fg='red')
       nameofuserentry.pack(side=LEFT)
                                                                      plscheckentry.pack()
                                                                      return
usernameofpersonframe=Frame(addbannerframe)
                                                                    if 1<= int(bannerno) <= len(lsl)+1:
       usernameofpersonframe.pack()
                                                                      Isl.pop((int(bannerno)-1))
usernameofuserlabel=Label(usernameofpersonframe,te
                                                          filehvingbannerimg=open('./STORAGE/bannerssource.t
xt='\n\tBanner Description \t')
                                                          xt','wb')
       TextArea =
                                                                      pickle.dump(lsl,filehvingbannerimg)
Text(usernameofpersonframe, selectborderwidth=3)
                                                                      filehvingbannerimg.close()
       usernameofuserlabel.pack(side=LEFT)
       TextArea.pack(expand=YES, fill=BOTH)
                                                          plscheckentry=Label(errorframe,text='Banner Has been
                                                          Deleted', fg='Green')
                                                                      plscheckentry.pack()
compusername=Label(usernameofpersonframe,text='M
ax 300 Characters\n\n\n')
       compusername.pack(side=LEFT)
                                                                      deletebannerframe.pack_forget()
                                                                      updateeventpagefrm=Frame(root)
       subbtn=Button(usernameofpersonframe,
                                                                      updateeventpagefrm.pack()
text='Submit', command=submsg)
       subbtn.pack()
                                                                      odbks="
       root.mainloop()
                                                                      for a in range(len(lsl)):
                                                                        odbks+='\n\n\nBanner Number: '+
                                                          str(a+1) + \n\ + str(|s|[a][0]) + \n\
                                                          Description:\t\t\n\n '+ str(Isl[a][1])+'\n'
                                                                        odbks+='----
                                                                      canvas = Canvas(updateeventpagefrm,
                                                          width=1000, height=400)
                                                                      scroll y = Scrollbar(updateeventpagefrm,
```

orient="vertical", command=canvas.yview)

def deletebanner():

global updateeventpagefrm

```
frameforscroll = Frame(canvas)
            canvas.create window(0, 0, anchor='nw',
window=frameforscroll)
                                                                   emailofpersonframe=Frame(deletebannerframe)
            overduelbl=Label(frameforscroll, text=
                                                                   emailofpersonframe.pack()
odbks, justify=LEFT)
            overduelbl.pack()
                                                            instructionlabel=Label(emailofpersonframe,text='Enter
                                                            Details of Banner to Delete')
            # make sure everything is displayed before
                                                                   instructionlabel.pack()
configuring the scrollregion
            canvas.update idletasks()
                                                            emailofuserlabel=Label(emailofpersonframe,text='\nBan
                                                            ner Number ')
canvas.configure(scrollregion=canvas.bbox('all'),
                                                            emailofuserentry=Entry(emailofpersonframe,width=30)
yscrollcommand=scroll y.set)
                                                                   emailofuserlabel.pack(side=LEFT)
                                                                   emailofuserentry.pack(side=LEFT)
                                                                   bufferIbl=Label(emailofpersonframe,text='\n\n')
            canvas.pack(fill='both', expand=True,
                                                                   bufferlbl.pack()
side='left')
            scroll y.pack(fill='y', side='right')
                                                            btnsubgo=Button(emailofpersonframe,text='Submit',
         else:
                                                            command=deletethatnow)
plscheckentry=Label(errorframe,text='Invalid Banner
                                                                   btnsubgo.pack()
Number', fg='red')
            plscheckentry.pack()
            return
                                                                 welcomeframe.pack forget()
                                                                 librarianframetop.pack_forget()
                                                                 librarianframetop.pack forget()
                                                                 librarianframebottom.pack forget()
       global deletebannerframe
                                                                 titleoffeedbackpage=Frame(root)
       global addbannerframe
                                                                 titleoffeedbackpage.pack()
       global errorframe
                                                                 feedbacktitletext = Label(titleoffeedbackpage, text =
                                                            'UPDATE BANNER', fg='blue',font= "applecherry 35")
       try:
                                                                 feedbacktitletext.pack(side= TOP)
         errorframe.pack forget()
         errorframe=Frame(root)
         errorframe.pack()
                                                                 updateeventpagefrm=Frame(root)
       except:
                                                                 updateeventpagefrm.pack()
         pass
       try:
         deletebannerframe.pack forget()
                                                            filehvingbannerimg=open('./STORAGE/bannerssource.t
       except:
                                                            xt','rb')
         pass
                                                                 Isl=pickle.load(filehvingbannerimg)
       try:
                                                                 filehvingbannerimg.close()
         addbannerframe.pack forget()
       except:
                                                                 odbks="
         pass
                                                                 global overduelbl
       updateeventpagefrm.pack forget()
                                                                 global canvas
       deletebannerframe=Frame(root)
                                                                 global scroll_y
       deletebannerframe.pack()
                                                                 if len(lsl)>0:
                                                                      for a in range(len(lsl)):
filehvingbannerimg=open('./STORAGE/bannerssource.t
                                                                        odbks+='\n\n\nBanner Number: '+ str(a+1)
xt','rb')
                                                            + '\n\nFile: '+ str(IsI[a][0]) + '\n\nBanner
       lsl=pickle.load(filehvingbannerimg)
                                                            Description:\t\t\n\n '+ str(Isl[a][1])+'\n'
                                                                        odbks+='----
       filehvingbannerimg.close()
       if len(lsl)==1:
                                                                       -----\n'
                                                                     canvas = Canvas(updateeventpagefrm,
plscheckentry=Label(errorframe,text='Minimum Number
                                                            width=1000, height=400)
of Banner Reached', fg='red')
                                                                      scroll y = Scrollbar(updateeventpagefrm,
         plscheckentry.pack()
                                                            orient="vertical", command=canvas.yview)
                                                                      frameforscroll = Frame(canvas)
         return
```

```
canvas.create window(0, 0, anchor='nw',
                                                           welcomeframe=Frame(root) #
                                                                                                A FRAME
window=frameforscroll)
                                                         HERE
                                                           welcomeframe.pack(side=TOP)
         overduelbl=Label(frameforscroll, text= odbks,
                                                            x=time.localtime()
                                                            if 0 <= x[3] <= 6 or 17 <= x[3] <= 23:
justify=LEFT)
                                                              nom='Good Evening
         overduelbl.pack()
                                                            elif 7<=x[3]<=11:
         # make sure everything is displayed before
                                                              nom='Good Morning'
configuring the scrollregion
                                                            else:
         canvas.update idletasks()
                                                              nom='Good Afternoon'
                                                           wlcmmsg=(nom+'LIBRARIAN')
canvas.configure(scrollregion=canvas.bbox('all'),
                                                           welcomeusermsg=Label(welcomeframe,
yscrollcommand=scroll y.set)
                                                         text=wlcmmsg, fg= 'Blue', font=' verdana 25', anchor=W)
                                                           welcomeusermsg.pack(side= LEFT)
         canvas.pack(fill='both', expand=True,
                                                         side='left')
                                                         t\t\t\t\n\n\n\n\n\n\n\n\
         scroll y.pack(fill='y', side='right')
                                                           spacelabel1.pack(side=LEFT)
    buttonsframe=Frame(root)
    buttonsframe.pack(side=BOTTOM)
                                                            "spaceframe1=Frame(framedetails)
                                                            spaceframe1.pack(side= LEFT)
    errorframe=Frame(root)
                                                            spacelabel1=Label(spaceframe1,text='
                                                                                                      ')
    errorframe.pack()
                                                            spacelabel1.pack(side=LEFT)"
    addnewbannerbutton= Button(buttonsframe,
                                                            librarianframetop=Frame(root)
text="Add New Banner", command=addnewbanner)
    addnewbannerbutton.pack()
                                                            librarianframebottom=Frame(root)
    deletebannerbutton= Button(buttonsframe,
text="Delete Banner", command=deletebanner)
                                                            librarianframetop.pack()
    deletebannerbutton.pack()
                                                            librarianframebottom.pack()
                                                            borrowbk=Frame(librarianframetop)
                                                            borrowbk.pack(side=LEFT)
    backbutton = Button(buttonsframe, text="Back",
command=back)
                                                         borrowbkpic=PhotoImage(file='./IMAGES/Icons/borrow
    backbutton.pack()
                                                         book.png')
    root.mainloop()
                                                            borrowbkbtn=Button(borrowbk,text='Open', image =
                                                         borrowbkpic.command= borrowbook) # BORROW
                                                         BOOK
                                                            borrowbkbtn.pack()
                                                            borrowbklb=Label(borrowbk,text='Borrow Book',font=
                                                         "timesnewroman 20")
                                                            borrowbklb.pack()
                                                            spaceframe1=Frame(librarianframetop)
                                                            spaceframe1.pack(side= LEFT)
  def logout():
                                                            spacelabel1=Label(spaceframe1,text='
ans=tkinter.messagebox.askguestion('LIBRARY','Are
                                                            spacelabel1.pack()
you sure you want to log out?')
    if ans == 'yes':
                                                            returnbk=Frame(librarianframetop)
                                                            returnbk.pack(side=LEFT)
       welcomeframe.pack forget()
                                                            returnbkpic=PhotoImage(file='./IMAGES/Icons/return
       librarianframetop.pack forget()
       librarianframetop.pack forget()
                                                         book.png')
       librarianframebottom.pack forget()
                                                            returnbkbtn=Button(returnbk,text='Open', image =
       mainpage()
                                                         returnbkpic,command= returnbook) # return BOOK
  def searchbook():
                                                            returnbkbtn.pack()
    welcomeframe.pack forget()
                                                            returnbklbl=Label(returnbk,text='Return Book',font=
    librarianframetop.pack forget()
                                                         "timesnewroman 20")
    librarianframetop.pack forget()
                                                            returnbklbl.pack()
    librarianframebottom.pack forget()
                                                            spaceframe2=Frame(librarianframetop)
    searchbookfn()
                                                            spaceframe2.pack(side= LEFT)
```

```
spacelabel1=Label(spaceframe2,text='
                                           ')
                                                          overduebkbtn=Button(overduebk.text='Open', image
                                                        = overduebkpic.command= overduebook) # overdue
  spacelabel1.pack()
                                                        BOOK #CHANGE
  reservebk=Frame(librarianframetop)
                                                          overduebkbtn.pack()
  reservebk.pack(side=LEFT)
                                                          overduebklbl=Label(overduebk,text='Overdue
                                                        Books', font= "timesnewroman 20")
reservebkpic=PhotoImage(file='./IMAGES/Icons/reservat
                                                          overduebklbl.pack()
ion.png')
  reservebkbtn=Button(reservebk,text='Open', image =
                                                          spaceframe6=Frame(librarianframetop)
reservebkpic,command= reservebook) # reserve
                                                          spaceframe6.pack(side= LEFT)
BOOK
                                                          spacelabel1=Label(spaceframe6,text='
                                                                                                   ')
  reservebkbtn.pack()
                                                          spacelabel1.pack()
  reservebklbl=Label(reservebk,text='Reserve
Book',font= "timesnewroman 20")
                                                          reservebklbl.pack()
                                                          ############### 4th October 2020
                                                        spaceframe3=Frame(librarianframetop)
  spaceframe3.pack(side= LEFT)
                                                          #Space Frame 9
  spacelabel1=Label(spaceframe3,text='
                                                          addnewfrm=Frame(librarianframebottom)
                                           ')
  spacelabel1.pack()
                                                          addnewfrm.pack(side=LEFT)
  renewbk=Frame(librarianframetop)
                                                        addnewpic=PhotoImage(file='./IMAGES/Icons/newbook.
  renewbk.pack(side=LEFT)
                                                        png')
                                                          addnewbtn=Button(addnewfrm,text='Open', image =
renewbkpic=PhotoImage(file='./IMAGES/Icons/reservati
                                                        addnewpic,command= addnewbook)
                                                          addnewbtn.pack()
on.png')
  renewbkbtn=Button(renewbk,text='Open', image =
                                                          newbklbl=Label(addnewfrm,text='Add New
                                                        Books',font= "timesnewroman 20")
renewbkpic,command= renewbook) # reserve BOOK
  renewbkbtn.pack()
                                                          newbklbl.pack()
  renewbklbl=Label(renewbk,text='Renew Book',font=
"timesnewroman 20")
                                                          spaceframe9=Frame(librarianframebottom)
  renewbklbl.pack()
                                                          spaceframe9.pack(side= LEFT)
                                                          spacelabel1=Label(spaceframe9,text='
                                                                                                   ')
  spaceframe4=Frame(librarianframetop)
                                                          spacelabel1.pack()
  spaceframe4.pack(side= LEFT)
  spacelabel1=Label(spaceframe4,text='
                                           ')
                                                          updatevenfrm=Frame(librarianframebottom)
                                                          updatevenfrm.pack(side=LEFT)
  spacelabel1.pack()
  searchbk=Frame(librarianframetop)
                                                        updateventpic=PhotoImage(file='./IMAGES/Icons/eventu
  searchbk.pack(side=LEFT)
                                                        pdate.png')
                                                          updateeventbtn=Button(updatevenfrm,text='Open',
searchbkpic=PhotoImage(file='./IMAGES/Icons/searchb
                                                        image = updateventpic,command= updateevents)
ookimage.png')
                                                          updateeventbtn.pack()
  global currentlocationforserachbook
                                                          updatelbl=Label(updatevenfrm,text='Update
  currentlocationforserachbook='LIBRARIAN'
                                                        Banner',font= "timesnewroman 20")
  searchbkbtn=Button(searchbk,text='Open', image =
                                                          updatelbl.pack()
searchbkpic,command= searchbook) # search BOOK
                                                          spaceframe10=Frame(librarianframebottom)
#CHANGE
  searchbkbtn.pack()
                                                          spaceframe10.pack(side= LEFT)
  searchbklbl=Label(searchbk,text='Search Book',font=
                                                          spacelabel1=Label(spaceframe10,text='
                                                                                                    ')
"timesnewroman 20")
                                                          spacelabel1.pack()
  searchbklbl.pack()
                                                          updtuserfrm=Frame(librarianframebottom)
                                                          updtuserfrm.pack(side=LEFT)
  spaceframe5=Frame(librarianframetop)
  spaceframe5.pack(side= LEFT)
  spacelabel1=Label(spaceframe5,text='
                                           ')
                                                        usrupdateventpic=PhotoImage(file='./IMAGES/Icons/upd
  spacelabel1.pack()
                                                        ateuser.png')
                                                          usrupdateeventbtn=Button(updtuserfrm,text='Open',
  overduebk=Frame(librarianframetop)
                                                        image = usrupdateventpic,command= updateuser)
  overduebk.pack(side=LEFT)
                                                          usrupdateeventbtn.pack()
                                                          updateusrlbl=Label(updtuserfrm,text='Update User
overduebkpic=PhotoImage(file='./IMAGES/Icons/overdu
                                                        Account',font= "timesnewroman 20")
                                                          updateusrlbl.pack()
e books.png')
```

spaceframe11=Frame(librarianframebottom)

```
spacelabel1=Label(spaceframe11,text='
                                              ')
  spacelabel1.pack()
                                                               latestarrivalstxt = Label(root, text = 'Latest arrivals
                                                          \n\n', fg='magenta',font= "applecherry 35")
                                                               latestarrivalstxt.pack(side=TOP)
  latestarrivalsframetop = Frame(root)
                                                               latestarrivalsframetop.pack()
  notificationsfrm=Frame(librarianframebottom)
  notificationsfrm.pack(side=LEFT)
  notipic=PhotoImage(file='./IMAGES/Icons/inbox.png')
                                                               def openbook1():
  notibtn=Button(notificationsfrm,text='Open', image =
notipic,command= messaging) # overdue BOOK
                                                          webbrowser.open("https://manybooks.net/book/126571/r
#CHANGE
                                                          ead#epubcfi(/6/2[titlepage]!/4/1:0)")
  notibtn.pack()
                                                               book1=Frame(latestarrivalsframetop)
  notilbl=Label(notificationsfrm,text='Messaging',font=
                                                               book1.pack(side = LEFT)
"timesnewroman 20")
                                                          latestarrival1pic=PhotoImage(file='./IMAGES/latestarriva
  notilbl.pack()
                                                          Is/latestarrival1.png')
  spaceframe7=Frame(librarianframebottom)
                                                               openbutton1=Button(book1,text='Open',image =
  spaceframe7.pack(side= LEFT)
                                                          latestarrival1pic.command=openbook1)
  spacelabel1=Label(spaceframe7.text='
                                             ')
                                                               openbutton1.pack()
  spacelabel1.pack()
                                                               latestarrivallabel1=Label(book1,text='He fell in love
                                                          with his wife', font= "timesnewroman 20")
  logoutfrm=Frame(librarianframebottom)
                                                               latestarrivallabel1.pack()
  logoutfrm.pack(side=LEFT)
logoutpic=PhotoImage(file='./IMAGES/Icons/logout.png')
                                                               spaceframe1=Frame(latestarrivalsframetop)
  logoutbtn=Button(logoutfrm,text='Open', image =
                                                               spaceframe1.pack(side= LEFT)
logoutpic,command= logout) # overdue BOOK
                                                               spacelabel1=Label(spaceframe1,text='
                                                                                                         ')
#CHANGE
                                                               spacelabel1.pack(side= LEFT)
  logoutbtn.pack()
  logoutlbl=Label(logoutfrm,text='Log Out',font=
"timesnewroman 20")
                                                               def openbook2():
  logoutlbl.pack()
                                                          webbrowser.open("https://manybooks.net/book/123104/r
                                                          ead#epubcfi(/6/2[titlepage]!/4/1:0)")
  spaceframe6=Frame(librarianframebottom)
  spaceframe6.pack(side= LEFT)
  spacelabel1=Label(spaceframe6,text='
                                             ')
                                                               book2= Frame(latestarrivalsframetop)
  spacelabel1.pack()
                                                               book2.pack(side = LEFT)
  root.mainloop()
                                                          lasestarrival2pic=PhotoImage(file='./IMAGES/latestarriv
                                                          als/latestarrival2.png')
############################# 21st september 2020
openbutton2=Button(book2,text='Open',image=lasestarri
def userpage(wlcmmsg,uname,username,libid):
                                                          val2pic,command=openbook2)
  def latestarrivalspg():
                                                               openbutton2.pack()
                                                               latestarrivallabel2=Label(book2,text='the return of
    welcomeframe.pack forget()
                                                          sherlock holmes' font='timesnewroman 20')
    userframetop.pack_forget()
    userframebottom.pack forget()
                                                               latestarrivallabel2.pack()
    def back():
                                                               spaceframe2=Frame(latestarrivalsframetop)
       latestarrivalsframetop.pack_forget()
                                                               spaceframe2.pack(side= LEFT)
                                                               spacelabel2=Label(spaceframe2,text='
                                                                                                         ')
                                                               spacelabel2.pack(side= LEFT)
       backbtn.pack_forget()
       latestarrivalstxt.pack forget()
                                                               def openbook3():
userpage(currentlocationforserachbookwlcmmsgusefulfo
                                                          webbrowser.open("https://manybooks.net/book/127828/r
ruseraccount,
                                                          ead#epubcfi(/6/2[item3]!/4/2/1:0)")
          currentlocationforserachbook,
currentlocationforserachbookwlcmmsgusefulforuseracco
                                                               book3= Frame(latestarrivalsframetop)
                                                               book3.pack(side = LEFT)
untusername.
          currentlocationforserachbookuserlibraryid)
                                                          latestarrival3pic=PhotoImage(file='./IMAGES/latestarriva
    global latestarrivalsframetop
```

spaceframe11.pack(side= LEFT)

Is/latestarrival3.png')

```
openbutton3=Button(book3,text='Open',image =
latestarrival3pic,command=openbook3)
                                                            currentlocationforserachbookwlcmmsgusefulforuseracco
    openbutton3.pack()
                                                            untusername.
    latestarrivallabel3=Label(book3,text='The invisible
                                                                      currentlocationforserachbookuserlibraryid)
man',font= "timesnewroman 20")
    latestarrivallabel3.pack()
                                                                 def inbox():
                                                                   global overduelbl
    spaceframe3=Frame(latestarrivalsframetop)
                                                                   global canvas
    spaceframe3.pack(side= LEFT)
                                                                   global scroll y
    spacelabel3=Label(spaceframe3,text='
                                                 ')
                                                                   def clrmsgfn():
    spacelabel3.pack()
                                                                      global manucomframe
                                                                      global overduelbl
    def openbook4():
                                                                      global overduelbl12
                                                                      manucomframe.pack forget()
                                                                      scroll y.pack forget()
webbrowser.open("https://manybooks.net/book/127823/r
ead#epubcfi(/6/2[coverpage-wrapper]!/4/1:0)")
                                                                      cursor.execute('delete from messaging where
     book4= Frame(latestarrivalsframetop)
                                                            toid =
    book4.pack(side = LEFT)
                                                            {}'.format(currentlocationforserachbookuserlibraryid))
                                                                      mycon.commit()
latestarrival4pic=PhotoImage(file='./IMAGES/latestarriva
                                                                      overduelbl.pack forget()
Is/latestarrival4.png')
                                                                      canvas.pack forget()
     openbutton4=Button(book4,text='Open',image =
latestarrival4pic,command=openbook4)
                                                                        overduelbl12.pack forget()
    openbutton4.pack()
                                                                      except:
    latestarrivallabel4=Label(book4,text='The first men
                                                                        pass
on the moon',font= "timesnewroman 20")
                                                                      overduelbl=Label(inboxframe, text= 'All
    latestarrivallabel4.pack()
                                                            Messages are Cleared', fg='green')
                                                                      overduelbl.pack()
                                                                      manucomframe.pack forget()
    spaceframe4=Frame(latestarrivalsframetop)
                                                                   def clr10msgfn():
    spaceframe4.pack(side= LEFT)
                                                                      global canvas
    spacelabel4=Label(spaceframe4,text='
                                                 ')
                                                                      global overduelbl
    spacelabel4.pack(side = LEFT)
                                                                      global manucomframe
                                                                      global scroll_y
                                                                      global overduelbl12
    backbtn=Button(root, text='Back',command=back)
                                                                      cursor.execute('delete from messaging where
                                                            toid =
    backbtn.pack()
                                                            {}'.format(currentlocationforserachbookuserlibraryid))
    root.mainloop()
                                                                      mycon.commit()
  def messaging():
                                                                      manucomframe.pack forget()
    welcomeframe.pack forget()
                                                                      canvas.pack forget()
    userframetop.pack forget()
                                                                      scroll y.pack forget()
    userframebottom.pack_forget()
                                                                      overduelbl.pack_forget()
    def back():
       global inboxframe
                                                                      for a in range(5):
       global sendframe
                                                                        try:
       lbldetails.pack forget()
                                                                           Isl.pop(-1)
       lbl.pack_forget()
                                                                        except:
       backbtn.pack forget()
                                                                           break
                                                                      if len(lsl)>0:
         inboxframe.pack_forget()
                                                                        for a in Isl:
                                                                           cursor.execute('insert into messaging
       except:
         pass
                                                            values {}'.format(a))
                                                                           mycon.commit()
         manucomframe.pack forget()
                                                                      overduelbl.pack forget()
       except:
                                                                        overduelbl12.pack forget()
         pass
                                                                      except:
         sendframe.pack forget()
       except:
                                                                      overduelbl12=Label(inboxframe, text= 'Last 5
                                                            messages are Cleared\n', fg='green')
                                                                      overduelbl12.pack()
userpage(currentlocationforserachbookwlcmmsgusefulfo
                                                                      odbks="
ruseraccount,
                                                                      if len(lsl)>0:
          currentlocationforserachbook,
                                                                        for a in range(len(lsl)):
```

```
odbks+='\n\n\nFROM: '+ str(IsI[a][1]) +
                                                                 cursor.execute('select * from messaging where
'\nMESSAGE:\t\t\n\n'+ str(IsI[a][3])+'\n'
                                                          toid =
              odbks+='----
                                                          {}'.format(currentlocationforserachbookuserlibraryid))
                                                                 Isl=cursor.fetchall()
  -----\n'
                                                                 |sl=Isl[::-1]
           canvas = Canvas(inboxframe, width=1000,
                                                                 if len(lsl)>0:
height=400)
                                                                   for a in range(len(lsl)):
                                                                      odbks+=\n\nFROM: '+ str(IsI[a][1])
            scroll y = Scrollbar(inboxframe,
                                                          +'\nMESSAGE:\t\t\n'+ str(IsI[a][3])+'n'
orient="vertical", command=canvas.yview)
            frameforscroll = Frame(canvas)
                                                                      odbks+='----
            canvas.create_window(0, 0, anchor='nw',
window=frameforscroll)
                                                                   canvas = Canvas(inboxframe, width=1000,
            overduelbl=Label(frameforscroll, text=
                                                          height=400)
                                                                   scroll y = Scrollbar(inboxframe,
odbks, justify=LEFT)
                                                          orient="vertical", command=canvas.yview)
            overduelbl.pack(side=LEFT)
                                                                   frameforscroll = Frame(canvas)
            # make sure everything is displayed before
                                                                   canvas.create window(0, 0, anchor='nw',
                                                          window=frameforscroll)
configuring the scrollregion
            canvas.update idletasks()
                                                                   overduelbl=Label(frameforscroll, text= odbks,
                                                          justify=LEFT)
canvas.configure(scrollregion=canvas.bbox('all'),
                                                                   overduelbl.pack(side=LEFT)
yscrollcommand=scroll y.set)
                                                                   # make sure everything is displayed before
            canvas.pack(fill='both', expand=True,
                                                          configuring the scrollregion
side='left')
                                                                   canvas.update idletasks()
            scroll y.pack(fill='y', side='right')
            manucomframe=Frame(root)
                                                          canvas.configure(scrollregion=canvas.bbox('all'),
            manucomframe.pack(side=BOTTOM)
                                                          yscrollcommand=scroll y.set)
            clrmsg=Button(manucomframe, text='Clear
All Messages', command=clrmsqfn)
                                                                   canvas.pack(fill='both', expand=True,
            clr10msg=Button(manucomframe,
                                                          side='left')
text='Clear Last 5 Messages', command=clr10msgfn)
                                                                   scroll_y.pack(fill='y', side='right')
                                                                   manucomframe=Frame(root)
            clrmsg.pack()
            clr10msg.pack()
                                                                   manucomframe.pack(side=BOTTOM)
                                                                   clrmsg=Button(manucomframe, text='Clear All
         else:
                                                          Messages', command=clrmsqfn)
            canvas.pack forget()
            overduelbl=Label(inboxframe, text= 'There
                                                                   clr10msg=Button(manucomframe, text='Clear
are no Messages')
                                                          Last 5 Messages', command=clr10msgfn)
            overduelbl.pack()
                                                                   clrmsg.pack()
                                                                   clr10msg.pack()
       global inboxframe
       global sendframe
                                                                   overduelbl=Label(inboxframe, text= 'There are
       global manucomframe
                                                          no Messages')
                                                                   overduelbl.pack()
         inboxframe.pack forget()
         manucomframe.pack_forget()
                                                               def send():
       except:
                                                                 global errorframe
                                                                 ####################### 21st september 2020
         pass
                                                          sendframe.pack forget()
                                                                 def submsg():
       except:
                                                                   global errorframe
         pass
                                                                   errorframe.pack_forget()
       inboxframe=Frame(root)
                                                                   errorframe=Frame(sendframe)
       inboxframe.pack(side=TOP)
                                                                   errorframe.pack()
       Ibl=Label(inboxframe,text='INBOX \n',
fg='blue',font= "applecherry 25") #THIS HAS TO BE
                                                                   recid=var.get()
DETELED WHILE NAVIGATING
                                                                   messagesent=TextArea.get("1.0",END)
       lbl.pack(side= TOP)
                                                                   if recid==0:
```

odbks="

```
notfoundurlbl=Label(errorframe,text='\n\nPlease Select
                                                          nricofpersonframe=Frame(detailsofpersonframeforcreati
Recipient ', fg='Red')
                                                          ngaccount)
            notfoundurlbl.pack()
                                                                  nricofpersonframe.pack()
            return
                                                          nricofuserlabel=Label(nricofpersonframe,text='\nTo \t')
                                                                  nricofuserlabel.pack()
         if len(messagesent)<=1:
notfoundurlbl=Label(errorframe,text='\n\nPlease Type
your Message ', fg='Red')
            notfoundurlbl.pack()
            return
                                                                  radiobtn2star = Radiobutton(nricofpersonframe)
                                                                  radiobtn2star.config(text='Librarian',
         if len(messagesent)>700:
                                                          variable=var, value=2)
                                                                  radiobtn2star.pack(side=LEFT)
notfoundurlbl=Label(errorframe,text='\n\nMax
Characters Exceeded', fg='Red')
                                                                  #message
           notfoundurlbl.pack()
                                                          usernameofpersonframe=Frame(detailsofpersonframefo
            return
                                                          rcreatingaccount)
                                                                  usernameofpersonframe.pack()
detailsofpersonframeforcreatingaccount.pack forget()
         notfoundurlbl=Label(errorframe,text='\n\nYour
                                                          usernameofuserlabel=Label(usernameofpersonframe,te
Message has been sent', fg='Green')
                                                          xt='\n\tYour Message \t')
         notfoundurlbl.pack()
                                                                  TextArea =
                                                          Text(usernameofpersonframe, selectborderwidth=3)
         cursor.execute("insert into messaging values
                                                                  usernameofuserlabel.pack(side=LEFT)
({},'{}',{},)".format(currentlocationforserachbookuserlibr
                                                                  TextArea.pack(expand=YES, fill=BOTH)
aryid,
                                                          compusername=Label(usernameofpersonframe,text='M
currentlocationforserachbookwlcmmsgusefulforuseracco
                                                          ax 700 Characters\n\n\n')
                                                                  compusername.pack(side=LEFT)
untusername,
recid.
                                                          subbtn=Button(detailsofpersonframeforcreatingaccount,
                                                          text='Send', command=submsg)
messagesent))
         mycon.commit()
                                                                  subbtn.pack()
         root.mainloop()
                                                                  errorframe=Frame(sendframe)
       global inboxframe
                                                                  errorframe.pack()
       global sendframe
                                                                  root.mainloop()
                                                               lbl=Label(root,text='Messaging\n', fg='blue',font=
       try:
                                                          "applecherry 35") #THIS HAS TO BE DETELED WHILE
         inboxframe.pack forget()
                                                          NAVIGATING
         manucomframe.pack_forget()
                                                               lbl.pack(side= TOP)
       except:
         pass
                                                               IbIdetails=Frame(root)
         sendframe.pack forget()
                                                               Ibldetails.pack()
       except:
                                                               btninbox=Button(Ibldetails, text='Inbox',
         pass
       sendframe=Frame(root)
                                                          command=inbox)
       sendframe.pack()
                                                               btnsend=Button(IbIdetails, text='Compose Mail',
       Ibl=Label(sendframe,text='\nCOMPOSE MAIL
                                                          command=send)
\n', fg='blue',font= "applecherry 25") #THIS HAS TO BE
                                                               btninbox.pack(side=BOTTOM)
DETELED WHILE NAVIGATING
                                                               btnsend.pack(side=BOTTOM)
       Ibl.pack(side= TOP)
detailsofpersonframeforcreatingaccount=Frame(sendfra
                                                               backbtn=Button(root, text='Back', command =
                                                          back)
me)
       detailsofpersonframeforcreatingaccount.pack()
                                                               backbtn.pack(side=BOTTOM)
                                                               root.mainloop()
```

var = IntVar()

```
def borrowbook():
    welcomeframe.pack forget()
                                                                backbtn=Button(root, text='Back', command =
    userframetop.pack forget()
                                                            back)
    userframebottom.pack forget()
                                                                 backbtn.pack(side=BOTTOM)
    def back():
                                                                root.mainloop()
       lblb.pack_forget()
       lbldetails.pack forget()
                                                              def reservebook():
                                                                 welcomeframe.pack forget()
       backbtn.pack forget()
                                                                 userframetop.pack forget()
                                                                 userframebottom.pack forget()
                                                                 def back():
userpage(currentlocationforserachbookwlcmmsgusefulfo
                                                                    lbl.pack_forget()
ruseraccount,
                                                                   lbldetails.pack forget()
                                                                   backbtn.pack forget()
          currentlocationforserachbook,
                                                                   smallinfo.pack forget()
currentlocationforserachbookwlcmmsgusefulforuseracco
                                                            userpage(currentlocationforserachbookwlcmmsgusefulfo
untusername,
          currentlocationforserachbookuserlibraryid)
                                                            ruseraccount,
                                                                      currentlocationforserachbook,
    cursor.execute('select * from borrowedbooks where
LIBID = {}'.format(libid))
                                                            currentlocationforserachbookwlcmmsgusefulforuseracco
    Is=cursor.fetchall()
                                                            untusername,
                                                                      currentlocationforserachbookuserlibraryid)
                                                                 odbks="
    vblbks='Number of Books Borrowed:
'+str(len(ls))+'\n\n'
                                                                 cursor.execute('select * from reservations where
                                                            LIBID = {}'.format(libid))
                                                                 lsl=cursor.fetchall()
    for a in range(len(ls)):
       cursor.execute('select BOOK Title from books
                                                                 if len(lsl)>0:
where BOOKID = {}'.format(ls[a][1]))
                                                                   for a in range(len(lsl)):
       bknm=cursor.fetchone()
                                                                      bookidis=IsI[a][1]
       vblbks+='Book Id: '+ str(ls[a][1])+'\nBook Title: '+
                                                                      cursor.execute('select BOOK Title from
str(bknm[0])+ '\nDue Date: '+ str(ls[a][2])+'\n'
                                                            books where BOOKID = {}'.format(bookidis))
                                                                      msn=cursor.fetchone()
       vblbks+='----
                                                                      odbks+='\t Book Id: '+ str(IsI[a][1])+'\t Book
                                                            Title: '+ str(msn[0])+'\n'
\ln n'
                                                                      odbks+='----
    Iblb=Label(root,text='Borrowed Books \n\n',
fg='blue',font= "applecherry 35") #THIS HAS TO BE
                                                                -----\n'
DETELED WHILE NAVIGATING
    lblb.pack(side= TOP)
                                                                 lbl=Label(root,text='Reserved Books \n\n',
                                                            fg='blue',font= "applecherry 35") #THIS HAS TO BE
                                                            DETELED WHILE NAVIGATING
    IbIdetails=Frame(root)
    Ibldetails.pack()
                                                                 lbl.pack(side= TOP)
                                                                 IbIdetails=Frame(root)
                                                                 Ibldetails.pack()
    canvas = Canvas(Ibldetails, width=650,
height=400)
    scroll_y = Scrollbar(lbldetails, orient="vertical",
command=canvas.yview)
                                                                 canvas = Canvas(Ibldetails, width=650,
    frameforscroll = Frame(canvas)
                                                            height=400)
    canvas.create_window(0, 0, anchor='nw',
                                                                 scroll_y = Scrollbar(lbldetails, orient="vertical",
window=frameforscroll)
                                                            command=canvas.yview)
                                                                 frameforscroll = Frame(canvas)
    overduelbl=Label(frameforscroll, text= vblbks,
                                                                 canvas.create window(0, 0, anchor='nw',
justify=LEFT)
                                                            window=frameforscroll)
    overduelbl.pack()
                                                                 overduelbl=Label(frameforscroll, text= odbks,
    # make sure everything is displayed before
                                                            justify=LEFT)
configuring the scrollregion
                                                                 overduelbl.pack()
    canvas.update idletasks()
                                                                # make sure everything is displayed before
    canvas.configure(scrollregion=canvas.bbox('all'),
                                                            configuring the scrollregion
yscrollcommand=scroll y.set)
                                                                 canvas.update idletasks()
```

scroll y.pack(fill='y', side='right')

canvas.pack(fill='both', expand=True, side='left')

```
canvas.configure(scrollregion=canvas.bbox('all'),
                                                               canvas.create window(0, 0, anchor='nw',
yscrollcommand=scroll y.set)
                                                          window=frameforscroll)
     canvas.pack(fill='both', expand=True, side='left')
                                                               overduelbl=Label(frameforscroll, text= odbks,
     scroll y.pack(fill='y', side='right')
                                                          justify=LEFT)
                                                               overduelbl.pack(side=LEFT)
     smallinfo=Label(root,text='To Make Reservations,
Please Contact Librarian')
                                                              # make sure everything is displayed before
     smallinfo.pack()
                                                          configuring the scrollregion
     backbtn=Button(root, text='Back', command =
                                                               canvas.update idletasks()
back)
     backbtn.pack(side=BOTTOM)
                                                               canvas.configure(scrollregion=canvas.bbox('all'),
     root.mainloop()
                                                          yscrollcommand=scroll_y.set)
                                                               canvas.pack(fill='both', expand=True, side='left')
                                                               scroll y.pack(fill='y', side='right')
  def overduebook():
                                                               backbtn=Button(root, text='Back', command =
     welcomeframe.pack forget()
     userframetop.pack forget()
                                                          back)
     userframebottom.pack forget()
                                                               backbtn.pack(side=BOTTOM)
     def back():
                                                               root.mainloop()
       lbl.pack forget()
                                                            def logout():
       lbldetails.pack forget()
       backbtn.pack_forget()
                                                          ans=tkinter.messagebox.askquestion('LIBRARY','Are
                                                          you sure you want to log out?')
userpage(currentlocationforserachbookwlcmmsgusefulfo
                                                              if ans == 'yes':
                                                                 welcomeframe.pack_forget()
ruseraccount,
          currentlocationforserachbook,
                                                                 userframetop.pack forget()
currentlocationforserachbookwlcmmsgusefulforuseracco
                                                                 userframebottom.pack forget()
untusername,
                                                                 mainpage()
          currentlocationforserachbookuserlibraryid)
                                                            def searchbook():
                                                              welcomeframe.pack_forget()
                                                              userframetop.pack forget()
                                                              userframebottom.pack forget()
     odbks="
     cursor.execute('select * from borrowedbooks
                                                              searchbookfn()
where duedate < curdate() and LIBID = {}'.format(libid))
                                                            welcomeframe=Frame(root)#
                                                                                                 A FRAME
     Isl=cursor.fetchall()
                                                          HERE
                                                            welcomeframe.pack(side=TOP)
     if len(lsl)>0:
                                                            welcomeusermsg=Label(welcomeframe,
       for a in range(len(lsl)):
                                                          text=wlcmmsq, fg= 'Blue', font=' verdana 25', anchor=W)
          bookidis=IsI[a][1]
                                                            welcomeusermsg.pack(side= LEFT)
          cursor.execute('select BOOK Title from
books where BOOKID = {}'.format(bookidis))
                                                          msn=cursor.fetchone()
                                                          odbks+='\t Book Id: '+ str(IsI[a][1])+'\t Book
                                                            spacelabel1.pack(side=LEFT)
Title: '+ str(msn[0])+'\n'
         odbks+='----
                                                            userframetop=Frame(root)
        -----\n'
                                                            userframebottom=Frame(root)
     Ibl=Label(root,text='Overdue Books \n\n',
fg='blue',font= "applecherry 35") #THIS HAS TO BE
                                                            userframetop.pack()
DETELED WHILE NAVIGATING
                                                            userframebottom.pack()
     lbl.pack(side= TOP)
                                                            borrowbk=Frame(userframetop)
                                                            borrowbk.pack(side=LEFT)
     IbIdetails=Frame(root)
     Ibldetails.pack()
                                                          borrowbkpic=PhotoImage(file='./IMAGES/Icons/borrow
                                                          book.png')
     canvas = Canvas(Ibldetails, width=650,
                                                            borrowbkbtn=Button(borrowbk,text='Open', image =
height=400)
                                                          borrowbkpic,command= borrowbook) # BORROW
     scroll_y = Scrollbar(lbldetails, orient="vertical",
                                                          BOOK
command=canvas.yview)
                                                            borrowbkbtn.pack()
     frameforscroll = Frame(canvas)
                                                            borrowbklb=Label(borrowbk,text='View Borrowed
```

Books',font= "timesnewroman 20")

```
borrowbklb.pack()
                                                         currentlocationforserachbookwlcmmsgusefulforuseracco
  spaceframe1=Frame(userframetop)
                                                         unt=wlcmmsa
  spaceframe1.pack(side= LEFT)
                                                           searchbkbtn=Button(searchbk,text='Open', image =
  spacelabel1=Label(spaceframe1,text='
                                            ')
                                                         searchbkpic,command= searchbook) # search BOOK
  spacelabel1.pack()
                                                         #CHANGE
                                                           searchbkbtn.pack()
                                                           searchbklbl=Label(searchbk,text='Search Book',font=
  overduebk=Frame(userframetop)
  overduebk.pack(side=LEFT)
                                                         "timesnewroman 20")
                                                           searchbklbl.pack()
overduebkpic=PhotoImage(file='./IMAGES/Icons/overdu
e books.png')
                                                           spaceframe5=Frame(userframetop)
  overduebkbtn=Button(overduebk,text='Open', image
                                                           spaceframe5.pack(side= LEFT)
= overduebkpic,command= overduebook) # overdue
                                                           spacelabel1=Label(spaceframe5,text='
                                                                                                     ')
BOOK #CHANGE
                                                           spacelabel1.pack()
  overduebkbtn.pack()
  overduebklbl=Label(overduebk,text='Overdue
Books',font= "timesnewroman 20")
                                                           latestarrfrmfrm=Frame(userframebottom)
  overduebklbl.pack()
                                                           latestarrfrmfrm.pack(side=LEFT)
  spaceframe6=Frame(userframetop)
                                                         latstarrpic=PhotoImage(file='./IMAGES/Icons/latestarriva
  spaceframe6.pack(side= LEFT)
                                                         Is.png')
  spacelabel1=Label(spaceframe6,text='
                                                           latestarrbtn=Button(latestarrfrmfrm,text='Open', image
                                            ')
  spacelabel1.pack()
                                                         = latstarrpic,command= latestarrivalspg) # overdue
                                                         BOOK #CHANGE
  reservebk=Frame(userframetop)
                                                           latestarrbtn.pack()
  reservebk.pack(side=LEFT)
                                                         latestarrlbl=Label(latestarrfrmfrm,text='LatestArrivals',fon
reservebkpic=PhotoImage(file='./IMAGES/Icons/reservat
                                                         t= "timesnewroman 20")
                                                            latestarrlbl.pack()
  reservebkbtn=Button(reservebk,text='Open', image =
reservebkpic,command= reservebook) # reserve
                                                           spaceframe7=Frame(userframebottom)
                                                           spaceframe7.pack(side= LEFT)
BOOK
  reservebkbtn.pack()
                                                           spacelabel1=Label(spaceframe7,text='
                                                                                                     ')
  reservebklbl=Label(reservebk,text='Reserved
                                                           spacelabel1.pack()
Book',font= "timesnewroman 20")
  reservebklbl.pack()
                                                           spaceframe3=Frame(userframetop)
                                                         #################21st september
  spaceframe3.pack(side= LEFT)
                                                         2020###############
  spacelabel1=Label(spaceframe3,text='
                                            ')
  spacelabel1.pack()
                                                           notificationsfrm=Frame(userframebottom)
                                                           notificationsfrm.pack(side=LEFT)
                                                           notipic=PhotoImage(file='./IMAGES/Icons/inbox.png')
  searchbk=Frame(userframetop)
  searchbk.pack(side=LEFT)
                                                           notibtn=Button(notificationsfrm,text='Open', image =
                                                         notipic,command= messaging) # overdue BOOK
searchbkpic=PhotoImage(file='./IMAGES/Icons/searchb
                                                         #CHANGE
ookimage.png')
                                                           notibtn.pack()
  global currentlocationforserachbook
                                                           notilbl=Label(notificationsfrm,text='Messaging',font=
  currentlocationforserachbook=uname
                                                         "timesnewroman 20")
                                                           notilbl.pack()
  global currentlocationforserachbookuserlibraryid
  currentlocationforserachbookuserlibraryid=libid
                                                           spaceframe7=Frame(userframebottom)
                                                           spaceframe7.pack(side= LEFT)
  global
                                                           spacelabel1=Label(spaceframe7,text='
                                                                                                     ')
currentlocationforserachbookwlcmmsgusefulforuseracco
                                                           spacelabel1.pack()
                                                           logoutfrm=Frame(userframebottom)
  global
                                                           logoutfrm.pack(side=LEFT)
currentlocationforserachbookwlcmmsgusefulforuseracco
untusername
                                                         logoutpic=PhotoImage(file='./IMAGES/Icons/logout.png')
currentlocationforserachbookwlcmmsgusefulforuseracco
                                                           logoutbtn=Button(logoutfrm.text='Open', image =
                                                         logoutpic,command= logout) # overdue BOOK
untusername=username
                                                         #CHANGE
```

logoutbtn.pack()

logoutlbl=Label(logoutfrm,text='Log Out',font=	lid=libid.get()
"timesnewroman 20")	personinfo=extractusers()
logoutlbl.pack()	if lid in personinfo and uname == personinfo[lid][0]
	and pwd == personinfo[lid][1] and
spaceframe6=Frame(userframebottom)	listoffilesofcaptcha[captchanumber][1]==captchaentry.ge
spaceframe6.pack(side= LEFT)	t():
spacelabel1=Label(spaceframe6,text=' ')	frameerrors. pack_forget()
spacelabel1.pack()	frameloginbuttons. pack_forget()
	canvasloginouter. pack_forget()
root.mainloop()	captchaframe. pack_forget()
	global userlibid
#######################################	userlibid= lid
def firstpageafterlogin():	firstpageafterlogin()
lid = userlibid	catpval=0
usersname="	try:
theusername="	catpval=captchaentry.get()
cursor.execute('select * from users where LIBID=	except:
{}'.format(lid))	pass
m=cursor.fetchall()	if catpval == 0:
if len(m) > 0:	try:
requsr = list(m[0])	Hele and the set of a large 1/6 and a second of a CIDI and Endow
usersname = requsr[2] + ' '+ requsr[3]	lblcaptchastats=Label(frameerrors,text='Please Enter
theusername=requsr[1]	Captcha', fg='red')
x=time.localtime()	lblcaptchastats.pack()
if 0<=x[3]<=6 or 17<=x[3]<=23:	except:
nom='Good Evening '	pass if lietoffile of contabe leantaban umber [11] = cotto valu
elif 7<=x[3]<=11:	if listoffilesofcaptcha[captchanumber][1]!=catpval:
nom='Good Morning ' else:	try:
nom='Good Afternoon '	lblcaptchastats=Label(frameerrors,text='Wrong Captcha
Home Good Aitemoon	Please Retry', fg='red')
	Iblcaptchastats.pack()
if theusername == 'LIBRARIAN':	except:
wlcmmsg=(nom+'LIBRARIAN')	pass
librarianspage()	puoo
else:	if lid not in personinfo:
wlcmmsg=(nom+str(usersname))	try:
userpage(wlcmmsg,usersname,theusername,lid)	
def loginpage():	lblpassstats=Label(frameerrors,text='Library
destmainpageto()	ID does not exist', fg='red')
global frameloginbuttons	lblpassstats.pack()
global canvasloginouter	except:
global captchaframe	pass
global frameerrors	elif uname != personinfo[lid][0] or pwd !=
global captchanumber #this is just a variable dont	personinfo[lid][1]:
have to delete	try:
def extractusers():	lblpassstats=Label(frameerrors,text='Invalid
f=open('./STORAGE/encrypteduserdetails.txt','rb')	Credentials', fg='red')
l=pickle.load(f)	lblpassstats.pack()
unameandpass={}	except:
for a in I:	pass
currperson={str(a[0]):(a[1],a[2])}	root.mainloop()
unameandpass.update(currperson)	
return unameandpass	dof forgotpass(): #ADD ELINCTION HEDE
def validate():	def forgotpass(): #ADD FUNCTION HERE global invalidpasswordmessage
global frameerrors	global invalidpasswordmessage global invalidcaptchacodeerror
try:	invalidpasswordmessage=0#this is for giving error
frameerrors. pack_forget()	message in login page
except:	invalidcaptchacodeerror=0#this is for giving error
pass	message in login page
frameerrors=Frame(root)	frameloginbuttons. pack_forget()
frameerrors.pack()	canvasloginouter. pack_forget()
uname = username.get()	captchaframe. pack_forget()
pwd=password.get()	try:

```
frameerrors. pack_forget()
                                                                     if pwd != pwdconf:#
                                                           #IF PASSWORD ENTERED MATCHEES WITH THE
    except:
                                                           CONFIRMED ONE
       pass
    def back():
       global errorstreams
                                                           passdoesnotmatchlabel=Label(errorstreams,text='Pass
                                                           words Entered does not match', fg='red')
       global otpevalfr
       detailsframe.pack_forget()
                                                                       passdoesnotmatchlabel.pack()
       backbtn.pack forget()
       submitbtn.pack forget()
       try:
                                                                     #binary file
         errorstreams.pack forget()
                                                                     passdet =
                                                           open('./STORAGE/encrypteduserdetails.txt','rb')
       except:
         pass
                                                                     I=pickle.load(passdet)
                                                                     passdet.close()
       try:
                                                                     for a in range(len(I)):
         otpevalfr.pack forget()
                                                                       if str(l[a][0]) == lid:
       except:
         pass
                                                                         break
       loginpage()
                                                                     tup=list(I[a])
    def newpass():
                                                                     tup[2]=pwd
       global newpassentryconf
                                                                     tup=tuple(tup)
       global newpassentry1
                                                                     I[a]=tup
       global subbtn
       def evaluatepass():
         global newpassentryconf
                                                           passdet=open('./STORAGE/encrypteduserdetails.txt','wb
         global newpassentry1
         global subbtn
         def tryloginnow():
                                                                     pickle.dump(I,passdet)
            errorstreams.pack forget()
                                                                     passdet.close()
            passframe.pack_forget()
                                                                     subbtn.pack forget()
            loginpage()
                                      #CHANGE THIS
HERE
                                                                     passreset=Label(passframe, text='Your
                                                           Password has been reset. Try Loging in to your account
                                                           now').pack()
                                                           gobtn=Button(passframe,text='Go',command=tryloginno
         global errorstreams
                                                           w).pack()
         try:
            errorstreams.pack forget()
         except:
            pass
                                                                  passframe=Frame(root)
         errorstreams=Frame(root)
                                                                  passframe.pack()
         errorstreams.pack()
         pwd=newpassentry1.get()
                                                                  newpass1=Label(passframe, text='Enter Your
         pwdconf=newpassentryconf.get()
                                                           New Password').pack()
countforpasswordcaptletter, countforpasswordsmallletter,
                                                           newpassentry1=Entry(passframe,width=20,show='*')
countforpasswordnumber=0,0,0 #PASSWORD
                                                                  newpassentry1.pack()
                                                                  newpass2=Label(passframe, text='Confirm Your
EVALUATION
                                                           New Password').pack()
         for varia in pwd:
            if ord(varia) in range(65,91):
              countforpasswordcaptletter+=1
                                                           newpassentryconf=Entry(passframe,width=20,show='*')
            elif ord(varia) in range(97,123):
                                                                  newpassentryconf.pack()
              countforpasswordsmallletter+=1
            elif ord(varia) in range(48,58):
                                                                  subbtn=Button(passframe,text='Submit',
              countforpasswordnumber+=1
                                                           command=evaluatepass)
         if countforpasswordcaptletter==0 or
                                                                  subbtn.pack()
countforpasswordsmallletter == 0 or
countforpasswordnumber ==0 or len(pwd)<8:
passwordinvalidlabel=Label(errorstreams,text='Your
                                                                def submit():
Password is weak', fg='red')#EVALUATION FOR
                                                                  global lid
STRONG PASSWORD
                                                                  global otpentry
            passwordinvalidlabel.pack()
                                                                  def verifyotp():
                                                                     global errorstreams
            return
                                                                     try:
                                                                       errorstreams.pack_forget()
```

except:	smtp.ehlo()
pass errorstreams=Frame(root)	smtp.login('aministrator.library@gmail.com','scyjovpwho
errorstreams.pack()	zfbrur')
otp=otpentry.get()	subject = sub
if str(val) != otp:	body = msg msg = fSubject: (subject)\n\n(bedy)'
lber=Label(errorstreams,text='Incorrect OTP', fg='red')	msg = f'Subject: {subject}\n\n{body}'
lber.pack()	smtp.sendmail('aministrator.library@gmail.com',m,msg)
return	
detailsframe.pack_forget()	submitbtn.pack_forget()
backbtn.pack_forget()	global otpevalfr
submitbtn.pack_forget()	atnovalfr - Frama(root)
otpevalfr.pack_forget() try:	otpevalfr = Frame(root) otpevalfr.pack()
errorstreams.pack_forget()	labelforspace = Label(otpevalfr, text ="
except:	\n\n\nPlease Enter Your OTP which you recived").pack(
pass	
newpass()	otpentry = Entry (otpevalfr,width=10)
global errorstreams try:	otpentry.pack()
errorstreams.pack_forget()	submitbbtn=Button(otpevalfr,
except:	text='Submit',command=verifyotp).pack()
pass	
errorstreams=Frame(root)	
errorstreams.pack() m=emailentry.get()	
lid=lidentry.get()	detailsframe=Frame(root)
#lid=libraryid.get()	detailsframe.pack()
	#libid
	lbl=Label(detailsframe,text='Reset Password \n\n',
if len(m)==0 or len(lid)==0:	fg='blue',font= "applecherry 35")
lber=Label(errorstreams,text='Please enter all details', fg='red')	<pre>lbl.pack(side= TOP) textforemail=Label(detailsframe,text="please enter")</pre>
lber.pack()	your Email ID ").pack()
return	, , , ,
	emailentry=Entry(detailsframe, width=20)
<pre>cursor.execute('select * from users where LIBID = {}'.format(lid))</pre>	emailentry.pack() textfoflid=Label(detailsframe,text="please enter
m1=cursor.fetchall()	your Library ID ").pack()
if len(m1) > 0:	, , , , p ()
requsr = list(m1[0])	lidentry=Entry(detailsframe, width=20)
fname = requsr[2]	lidentry.pack()
else:	submitbtn=Button(root,text='submit',
lber=Label(errorstreams,text='Invalid Library	command=submit)
ID', fg='red')	submitbtn.pack()
lber.pack()	backbtn=Button(root,text='Back', command=back)
return if m1[0][6] != m:	backbtn.pack()
lber=Label(errorstreams,text='Incorrect Email	
ID\nPlease use the Registered Email ID', fg='red')	
lber.pack()	
return	
val=random.randint(1000,9999)	#Username and password canvasloginouter = Frame(root,width=1000, height=
sub='OTP Verification'	1000)
msg='Dear '+str(fname)+',\nYour OTP for you	canvasloginouter.pack()
Libriary account is '+str(val)+'\n\nWarm	lbl=Label(canvasloginouter,text='Login Page \n\n',
Regards\nLibrary Team'	fg='blue',font= "applecherry 35")
with smtplib.SMTP('smtp.gmail.com', 587) as	lbl.pack(side= TOP)
smtp: smtp.ehlo()	Iblid=Label(canvasloginouter,text=' Library ID') libid=Entry(canvasloginouter,width=20)
smtp.starttls()	

```
Iblusername=Label(canvasloginouter,text='
                                                             Ibl = Label(frame, text = 'Welcome To Our e-Library',
                                                          fg='blue',font= "applecherry 61 italic")
Username')
  username=Entry(canvasloginouter,width=20)
                                                             lbl.pack(side= TOP)
  lblpassword=Label(canvasloginouter,text='
                                                             #banner --> ' Change this entire place from gif to an
Password')
                                                           user controlled arrow mark
  password = Entry(canvasloginouter, show="*",
width=20)
                                                             def goleft():
  lblid.pack(side= LEFT)
                                                               global forbannerimagelistindex
  libid.pack(side= LEFT)
                                                               global forbannerimagelistlength
  Iblusername.pack(side= LEFT)
                                                               global framepicbanner
  username.pack(side= LEFT)
                                                               if forbannerimagelistindex == 0:
  lblpassword.pack(side= LEFT)
  password.pack(side= LEFT)
                                                          forbannerimagelistindex=forbannerimagelistlength-1
  frameloginbuttons = Frame(root, width=1000, height=
                                                               else:
                                                                  forbannerimagelistindex-=1
1000)
  frameloginbuttons.pack(side = BOTTOM)
                                                               framepicbanner.pack forget()
                                                               framepicbanner=Frame(encloseframeforbanner)
  #captcha
                                                               framepicbanner.pack()
  captchaframe = Frame(root,width=1000, height=
1000)
  captchaframe.pack()
                                                           photobanner=PhotoImage(file=imagelist[forbannerimage
  captchanumber= random.randint(1,9)-1
                                                           listindex][0])
photo=PhotoImage(file=listoffilesofcaptcha[captchanum
                                                          bannerphoto=Label(framepicbanner,image=photobanne
ber][0])
                                                           r)
                                                               bannerphoto.pack()
  captchaphoto=Label(captchaframe,image=photo)
                                                               root.mainloop()
  captchaentry = Entry(captchaframe, width=20)
  captchaentry.insert(0, "Enter Captcha Here")
                                                             def goright():
  #Sumbit buttons
                                                               global forbannerimagelistindex
  Backtomain=Button(frameloginbuttons, text = 'Back',
                                                               global forbannerimagelistlength
                                                               global framepicbanner
command= loginpagetomain)
  Backtomain.pack(side = BOTTOM)
                                                               if forbannerimagelistindex ==
  submitcred=Button(frameloginbuttons, text = 'Submit',
                                                          forbannerimagelistlength-1:
command= validate)
                                                                  forbannerimagelistindex=0
  submitcred.pack(side = BOTTOM)
                                                               else:
  forgotpassbutton=Button(frameloginbuttons, text =
                                                                  forbannerimagelistindex+=1
'Reset Password', command= forgotpass)
                                                               framepicbanner.pack forget()
  forgotpassbutton.pack(side = BOTTOM)
                                                               framepicbanner=Frame(encloseframeforbanner)
  createnewaccount=Button(frameloginbuttons, text =
                                                               framepicbanner.pack()
'Create Account', command=
createaccountfromloginpage)
  createnewaccount.pack(side = BOTTOM)
                                                           photobanner=PhotoImage(file=imagelist[forbannerimage
  captchaphoto.pack(side= LEFT)
                                                           listindex][0])
  captchaentry.pack(side= BOTTOM)
  root.mainloop()
                                                          bannerphoto=Label(framepicbanner,image=photobanne
                                                               bannerphoto.pack()
def mainpage(): #BE AWARE MANY VARIABLES ARE
CONVERTED TO LOCAL VARIABLE HERE
                                                               root.mainloop()
  global root
  global frame
  global canvas #actually a frame
                                                          filehavingbannerpics=open('./STORAGE/bannerssource.
  global spaceframe1
                                                          txt','rb')
  global loginframe
  global spaceframe2
                                                             imagelist = pickle.load(filehavingbannerpics)
  global aboutusframe
                                                             filehavingbannerpics.close()
                                                             canvas = Frame(root)
  global buttonsframe
  global forbannerimagelistindex
                                                             canvas.pack()
  global forbannerimagelistlength
  global framepicbanner
                                                           photoarrorleft=PhotoImage(file="./IMAGES/Icons/arrowl
  frame=Frame(root)
                                                           eft.png")
```

frame.pack()

```
leftbtn=Button(canvas, text = 'Click Me!', image =
                                                           loginlabel=Label(loginframe, text='Login and View
photoarrorleft, command= goleft) #CHANGE
                                                         Account')
COMMAND HERE #chaange frame name HERE TOO
                                                           loginlabel.pack(side= BOTTOM)
for next copy paste
  leftbtn.pack(side = LEFT)
                                                           #frame to leave space
                                                           spaceframe3=Frame(buttonsframe)
                                                           spaceframe3.pack(side= LEFT)
  forbannerimagelistlength=len(imagelist)
  forbannerimagelistindex=0
                                                           spacelabel3=Label(spaceframe3,text='
                                                                                                     ')
                                                           spacelabel3.pack()
  #update other files here
  encloseframeforbanner=Frame(canvas)
  encloseframeforbanner.pack(side = LEFT)
                                                           #SEARCH BOOK
  framepicbanner=Frame(encloseframeforbanner)
                                                           global currentlocationforserachbook
  framepicbanner.pack()
                                                           currentlocationforserachbook='GENERAL'
                                                           searchbook=Frame(buttonsframe)
                                                           searchbook.pack(side= LEFT)
                                                                                              #IMPORTANT
                                                         PLACE
photobanner=PhotoImage(file=imagelist[forbannerimage
listindex][0])
                                                           photosearchbook = PhotoImage(file =
                                                         "./IMAGES/Icons/searchbookimage.png") #Change the
bannerphoto=Label(framepicbanner,image=photobanne
                                                         command below to pass control to open new window
                                                           photosearchbutton=Button(searchbook, text = 'Click
  bannerphoto.pack()
                                                         Me!', image = photosearchbook, command=
                                                         searchbookfn) #CHANGE COMMAND HERE #chaange
                                                         frame name HERE TOO for next copy paste
                                                           photosearchbutton.pack(side = TOP)
photoarrorright=PhotoImage(file="./IMAGES/Icons/arrow
                                                           searchbooklabel=Label(searchbook, text='Search a
                                                         Book') #for next copypaste change this value too
right.png")
  rightbtn=Button(canvas, text = 'Click Me !', image =
                                                           searchbooklabel.pack(side= BOTTOM)
photoarrorright, command= goright) #CHANGE
COMMAND HERE #chaange frame name HERE TOO
                                                           #frame to leave space
                                                           spaceframe2=Frame(buttonsframe)
for next copy paste
  rightbtn.pack(side = LEFT)
                                                           spaceframe2.pack(side= LEFT)
                                                           spacelabel2=Label(spaceframe2,text='
                                                                                                     ')
                                                           spacelabel2.pack()
#photo=PhotoImage(file=listoffilesofcaptcha[captchanu
mber][0])
  #captchaphoto=Label(captchaframe,image=photo)
  "for k in range(0, 1): #can also change it to while true
                                                           #ebooks
    for gif in giflist:
                                                           ebookframe=Frame(buttonsframe)
       canvas.delete(ALL)
                                                           ebookframe.pack(side= LEFT)
                                                                                              #IMPORTANT
       canvas.create image(width/2, height/2,
                                                         PLACE
                                                           photoebook = PhotoImage(file =
image=gif)
                                                         "./IMAGES/Icons/ebook.png") #Change the command
       canvas.update()
       time.sleep(2)" #change this for adding a button
                                                         below to pass control to open new window
with moving images
                                                           ebookbutton=Button(ebookframe, text = 'Click Me!',
                                                         image = photoebook, command= ebookpage)
                                                         #CHANGE COMMAND HERE #chaange frame name
  #navigation buttons
  buttonsframe=Frame(root, width=1000)
                                                         HERE TOO for next copy paste
  buttonsframe.pack(side=BOTTOM)
                                                           ebookbutton.pack(side = TOP)
                                                           ebooklabel=Label(ebookframe, text='EBooks') #for
  #frame to leave space
  spaceframe1=Frame(buttonsframe)
                                                         next copypaste change this value too
  spaceframe1.pack(side= LEFT)
                                                           ebooklabel.pack(side= BOTTOM)
  spacelabel=Label(spaceframe1,text='
                                           ')
  spacelabel.pack()
                                                           #frame to leave space
                                                           spaceframe5=Frame(buttonsframe)
                                                           spaceframe5.pack(side= LEFT)
  #login to account
  loginframe=Frame(buttonsframe)
                                                           spacelabel5=Label(spaceframe5,text='
                                                                                                     ')
                                    #IMPORTANT
  loginframe.pack(side= LEFT)
                                                           spacelabel5.pack()
PLACE
  photo = PhotoImage(file =
                                                           #Suggestions
"./IMAGES/Icons/login.png") #Change the command
                                                           feedbackframe=Frame(buttonsframe)
below to pass control to open new window
                                                           feedbackframe.pack(side= LEFT)
  Button(loginframe, text = 'Click Me!', image = photo,
                                                         #IMPORTANT PLACE
command= loginpage).pack(side = TOP)
```

```
photofeedback = PhotoImage(file =
"./IMAGES/Icons/feedback.png") #Change the
command below to pass control to open new window
  feedbackbutton=Button(feedbackframe, text = 'Click
Me!', image = photofeedback, command=
feedbackoage) #CHANGE COMMAND HERE #chaange
frame name HERE TOO for next copy paste
  feedbackbutton.pack(side = TOP)
  feedbacklabel=Label(feedbackframe,
text='Feedback') #for next copypaste change this value
  feedbacklabel.pack(side= BOTTOM)
  #frame to leave space
  spaceframe4=Frame(buttonsframe)
  spaceframe4.pack(side= LEFT)
  spacelabel4=Label(spaceframe4,text='
                                           ')
  spacelabel4.pack()
    #About us
  aboutusframe=Frame(buttonsframe)
  aboutusframe.pack(side= LEFT)
#IMPORTANT PLACE
  photoaboutus = PhotoImage(file =
"./IMAGES/Icons/aboutus.png") #Change the command
below to pass control to open new window
  aboutusbutton=Button(aboutusframe, text = 'Click Me
!', image = photoaboutus, command= aboutuspage)
#CHANGE COMMAND HERE #chaange frame name
HERE TOO for next copy paste
  aboutusbutton.pack(side = TOP)
  aboutuslabel=Label(aboutusframe, text='About Us')
#for next copypaste change this value too
  aboutuslabel.pack(side= BOTTOM)
  #frame to leave space
  spaceframe5=Frame(buttonsframe)
  spaceframe5.pack(side= LEFT)
  spacelabel5=Label(spaceframe5,text='
                                           ')
  spacelabel5.pack()
  #About us
  helpframe=Frame(buttonsframe)
  helpframe.pack(side= LEFT)
                                   #IMPORTANT
PLACE
  photohelp = PhotoImage(file =
"./IMAGES/Icons/help.png") #Change the command
below to pass control to open new window
  helpbutton=Button(helpframe, text = 'Click Me!',
image = photohelp, command= helpme) #CHANGE
COMMAND HERE #chaange frame name HERE TOO
for next copy paste
  helpbutton.pack(side = TOP)
  helplabel=Label(helpframe, text='Help') #for next
copypaste change this value too
  helplabel.pack(side= BOTTOM) #HELP IN LINE 613
  spaceframe5=Frame(buttonsframe)
  spaceframe5.pack(side= LEFT)
  spacelabel5=Label(spaceframe5,text='
                                           ')
  spacelabel5.pack()
  #destmainpageto()
  root.mainloop()
```

```
#LIST OF LOCAL VARIABLES DECLARED GLOBAL
and stored outsude function
invalidpasswordmessage = 0 #in function to validate
password
invalidcaptchacodeerror = 0
listoffilesofcaptcha=[('./IMAGES/captchapictures/captcha
1.png','28ivw'),('./IMAGES/captchapictures/captcha
2.png','k4ez'),('./IMAGES/captchapictures/captcha
3.png','4D7YS'),('./IMAGES/captchapictures/captcha
4.png','6ne3'),('./IMAGES/captchapictures/captcha
5.png','e5hb'),('./IMAGES/captchapictures/captcha
6.png','FH2DE'),('./IMAGES/captchapictures/captcha
7.png','xmqki'),('./IMAGES/captchapictures/captcha
8.png','6H3T8'),('./IMAGES/captchapictures/captcha
9.png','RBSKW')]
#when adding new captcha, also hcange value which
will be used for generating random number in these
pages --> login page, create account, forgot pass
```

```
#tkinter window starts
root =Tk()
#root.configure(bg='skyblue') #used for coloring the
mainpage backgound
root.title("LIBRARY")
#root.geometry("%dx%d+%d+%d" % (1000, 1000, 2000,
150))
menu=Menu(root)
```

#menu settings

root.config(menu=menu) submenu= Menu(menu) menu.add_cascade(label='LIBRARY', menu=submenu) submenu.add_command(label='Quit', command=root.destroy) #try to quit tkinter file

mainpage()
root.mainloop()