## 7. CODE LISTING

from tkinter import \*

import tkinter.messagebox

from datetime import datetime

from datetime import timedelta

import webbrowser

import time

import random

import smtplib

import pickle

import mysql.connector

def createdbacc():

def Quitsys():

root1.destroy()

quit()

def submitsuggestion():

def regandlog():

databaseaccountfile=open('./STORAGE/databaseaccounts.txt','rb')

try:

listofallusers=pickle.load(databaseaccountfile)

except:

listofallusers=[]

databaseaccountfile.close()

listofallusers.append((hostnm,user,passwd,dbnm))

databaseaccountfile=open('./STORAGE/databaseaccounts.txt','wb')

listofallusers=pickle.dump(listofallusers,databaseaccountfile)

databaseaccountfile.close()

root1.destroy()

global errorstreams

try:

errorstreams.pack\_forget()

except:

pass

hostnm=nameofuserentry.get()

user=nricofuserentry.get()

passwd=emailofuserentry.get()

dbnm=databaseentry.get()

errorstreams=Frame(root1)

errorstreams.pack()

if len(hostnm)==0 or len(user)==0 or len(passwd)==0 or len(dbnm)==0:

entall=Label(errorstreams, text='Please Enter all Details',fg='red')

entall.pack()

return

try:

global mycon

mycon=mysql.connector.connect(host=hostnm,

user=user,

passwd=passwd,

database=dbnm)

except:

entall=Label(errorstreams, text='Invalid Credentials',fg='red')

entall.pack()

return

detailsofpersonframe.pack\_forget()

mainlbl.pack\_forget()

successmsg=Label(root1,text='Database Account Verified',font= "timesnewroman 20")

successmsg.pack()

buttonsframe=Frame(root1)

buttonsframe.pack()

backbutton = Button(buttonsframe, text="Register and Login", command=regandlog)

backbutton.pack()

root1=Tk()

mainlbl=Label(root1,text='Database Error\n\nPlease Restart System or Register Database account\n\n\n',font= "timesnewroman 20")

mainlbl.pack()

detailsofpersonframe=Frame(root1)

detailsofpersonframe.pack()

#name

nameframe=Frame(detailsofpersonframe)

nameframe.pack()

nameofuserlabel=Label(nameframe,text='Host Name ')

nameofuserentry=Entry(nameframe,width=20)

nameofuserlabel.pack(side=LEFT)

nameofuserentry.pack(side=LEFT)

#nric

nricofpersonframe=Frame(detailsofpersonframe)

nricofpersonframe.pack()

nricofuserlabel=Label(nricofpersonframe,text='User Name\t\t')

nricofuserentry=Entry(nricofpersonframe,width=20)

nricofuserlabel.pack(side=LEFT)

nricofuserentry.pack(side=LEFT)

#email

emailofpersonframe=Frame(detailsofpersonframe)

emailofpersonframe.pack()

emailofuserlabel=Label(emailofpersonframe,text='Password\t\t')

emailofuserentry=Entry(emailofpersonframe,width=20,show='\*')

emailofuserlabel.pack(side=LEFT)

emailofuserentry.pack(side=LEFT)

#email

databsenameframe=Frame(detailsofpersonframe)

databsenameframe.pack()

Databaselabel=Label(databsenameframe,text='Database Name \t')

databaseentry=Entry(databsenameframe,width=20)

Databaselabel.pack(side=LEFT)

databaseentry.pack(side=LEFT)

buttonsframe=Frame(detailsofpersonframe)

buttonsframe.pack(side=BOTTOM)

submitbutton= Button(buttonsframe, text="Register Database", command=submitsuggestion)

submitbutton.pack()

backbutton = Button(buttonsframe, text="Quit", command=Quitsys)

backbutton.pack()

root1.mainloop()

databaseaccountfile=open('./STORAGE/databaseaccounts.txt','rb')

try:

listofallusers=pickle.load(databaseaccountfile)

except:

listofallusers=[]

databaseaccountfile.close()

for i in listofallusers:

try:

mycon=mysql.connector.connect(host=i[0],

user=i[1],

passwd=i[2],

database=i[3])

break

except:

pass

else:

createdbacc()

cursor=mycon.cursor()

#destorys everything in the page

#whenever a new frame or canvas is created, craete a new function to delete everthing in it

def destmainpageto():

frame. pack\_forget()

canvas. pack\_forget() #this is actually a frame

spaceframe1. pack\_forget()

loginframe. pack\_forget()

buttonsframe. pack\_forget()

aboutusframe. pack\_forget()

spaceframe2. pack\_forget()

def aboutuspagetomain():

imageandback. pack\_forget()

contenttext. pack\_forget()

langoptions. pack\_forget()

mainpage()

def suggestionstomain():

titleoffeedbackpage. pack\_forget()

detailsofpersonframe. pack\_forget()

suggestiongivingframe. pack\_forget()

mainpage()

def loginpagetomain():

global invalidpasswordmessage

global invalidcaptchacodeerror

invalidpasswordmessage=0#this is for giving error message in login page

invalidcaptchacodeerror=0#this is for giving error message in login page

frameloginbuttons. pack\_forget()

canvasloginouter. pack\_forget()

captchaframe. pack\_forget()

try:

frameerrors. pack\_forget()

except:

pass

mainpage()

def createaccountfromloginpage():

global invalidpasswordmessage

global invalidcaptchacodeerror

frameloginbuttons. pack\_forget()

canvasloginouter. pack\_forget()

captchaframe. pack\_forget()

try:

frameerrors. pack\_forget()

except:

pass

createaccountpage()

def searchtomain():

titleofsearchbookpage. pack\_forget()

keywordframe. pack\_forget()

resultframe. pack\_forget()

backframe. pack\_forget()

if currentlocationforserachbook == 'LIBRARIAN':

librarianspage()

elif currentlocationforserachbook != 'GENERAL':

userpage(currentlocationforserachbookwlcmmsgusefulforuseraccount,

currentlocationforserachbook,

currentlocationforserachbookwlcmmsgusefulforuseraccountusername,

currentlocationforserachbookuserlibraryid)

try:

mainpage()

except:

pass

def ebookpagetomain():

titleofebookpage. pack\_forget()

ebooksframetop. pack\_forget()

ebooksframebottom. pack\_forget()

backbuttonframe. pack\_forget()

mainpage()

def ebookpage():

destmainpageto()

global titleofebookpage

global ebooksframetop

global ebooksframebottom

global backbuttonframe

def openbook1():

blackbeautybook='https://manybooks.net/book/126828/read#epubcfi(/6/2[titlepage]!/4/1:0)'

webbrowser.open(blackbeautybook, new=2)

def openbook2():

webbrowser.open('https://manybooks.net/book/127828/read#epubcfi(/6/2[item3]!/4/2/1:0)', new=2)

def openbook3():

webbrowser.open('https://manybooks.net/book/135651/read#epubcfi(/6/2[item3]!/4/2/2/2/2[pgheader]/2/2/1:0)', new=2)

def openbook4():

webbrowser.open('https://manybooks.net/book/148396/read#epubcfi(/6/2[coverpage-wrapper]!/4/1:0)', new=2)

def openbook5():

webbrowser.open('https://manybooks.net/book/127190/read#epubcfi(/6/2[titlepage]!/4/1:0)', new=2)

def openbook6():

webbrowser.open('https://manybooks.net/book/127836/read#epubcfi(/6/2[item4]!/4/2/1:0)', new=2)

def openbook7():

webbrowser.open('https://manybooks.net/book/127829/read#epubcfi(/6/2[titlepage]!/4/1:0)', new=2)

def openbook8():

webbrowser.open('https://manybooks.net/book/139614/read#epubcfi(/6/2[coverpage-wrapper]!/4/1:0)', new=2)

def openbook9():

webbrowser.open('https://manybooks.net/book/141234/read#epubcfi(/6/2[item4]!/4/2/1:0)', new=2)

def openbook10():

webbrowser.open('https://manybooks.net/book/127534/read#epubcfi(/6/2[coverpage-wrapper]!/4/1:0)', new=2)

def openbook11():

webbrowser.open('https://manybooks.net/book/131121/read#epubcfi(/6/2[item4]!/4/2/1:0)', new=2)

def openbook12():

webbrowser.open('https://manybooks.net/book/122925/read#epubcfi(/6/2[id00000]!/4/2[id00000]/1:0)', new=2)

titleofebookpage=Frame(root)

titleofebookpage.pack()

ebooktext = Label(titleofebookpage, text = 'E Books\n\n', fg='blue',font= "applecherry 35")

ebooktext.pack(side= TOP)

ebooksframetop=Frame(root)

ebooksframetop.pack()

ebooksframebottom=Frame(root)

ebooksframebottom.pack()

book1=Frame(ebooksframetop)

book1.pack(side=LEFT)

ebook1pic=PhotoImage(file='./IMAGES/ebookspictures/ebook1.png')

openbutton1=Button(book1,text='Open', image = ebook1pic,command=openbook1)

openbutton1.pack()

ebooklabel1=Label(book1,text='Black Beauty',font= "timesnewroman 20")

ebooklabel1.pack()

spaceframe1=Frame(ebooksframetop)

spaceframe1.pack(side= LEFT)

spacelabel1=Label(spaceframe1,text=' ')

spacelabel1.pack()

book2=Frame(ebooksframetop)

book2.pack(side=LEFT)

ebook2pic=PhotoImage(file='./IMAGES/ebookspictures/ebook2.png')

openbutton2=Button(book2,text='Open', image = ebook2pic,command=openbook2)

openbutton2.pack()

ebooklabel2=Label(book2,text='The Invisible Man',font= "timesnewroman 20")

ebooklabel2.pack()

spaceframe2=Frame(ebooksframetop)

spaceframe2.pack(side= LEFT)

spacelabel2=Label(spaceframe2,text=' ')

spacelabel2.pack()

book3=Frame(ebooksframetop)

book3.pack(side=LEFT)

ebook3pic=PhotoImage(file='./IMAGES/ebookspictures/ebook3.png')

openbutton3=Button(book3,text='Open', image = ebook3pic,command=openbook3)

openbutton3.pack()

ebooklabel3=Label(book3,text='Beauty and the Beast',font= "timesnewroman 20")

ebooklabel3.pack()

spaceframe3=Frame(ebooksframetop)

spaceframe3.pack(side= LEFT)

spacelabel3=Label(spaceframe3,text=' ')

spacelabel3.pack()

book4=Frame(ebooksframetop)

book4.pack(side=LEFT)

ebook4pic=PhotoImage(file='./IMAGES/ebookspictures/ebook4.png')

openbutton4=Button(book4,text='Open', image = ebook4pic,command=openbook4)

openbutton4.pack()

ebooklabel4=Label(book4,text='A Pair of Schoolgirls',font= "timesnewroman 20")

ebooklabel4.pack()

spaceframe4=Frame(ebooksframetop)

spaceframe4.pack(side= LEFT)

spacelabel4=Label(spaceframe4,text=' ')

spacelabel4.pack()

book5=Frame(ebooksframetop)

book5.pack(side=LEFT)

ebook5pic=PhotoImage(file='./IMAGES/ebookspictures/ebook5.png')

openbutton5=Button(book5,text='Open', image = ebook5pic,command=openbook5)

openbutton5.pack()

ebooklabel5=Label(book5,text='Treasure Island',font= "timesnewroman 20")

ebooklabel5.pack()

spaceframe5=Frame(ebooksframetop)

spaceframe5.pack(side= LEFT)

spacelabel5=Label(spaceframe5,text=' ')

spacelabel5.pack()

book6=Frame(ebooksframetop)

book6.pack(side=LEFT)

ebook6pic=PhotoImage(file='./IMAGES/ebookspictures/ebook6.png')

openbutton6=Button(book6,text='Open', image = ebook6pic,command=openbook6)

openbutton6.pack()

ebooklabel6=Label(book6,text='The Time Machine',font= "timesnewroman 20")

ebooklabel6.pack()

spaceframe6=Frame(ebooksframetop)

spaceframe6.pack(side= LEFT)

spacelabel6=Label(spaceframe6,text=' ')

spacelabel6.pack()

book7=Frame(ebooksframebottom)

book7.pack(side=LEFT)

ebook7pic=PhotoImage(file='./IMAGES/ebookspictures/ebook7.png')

openbutton7=Button(book7,text='Open', image = ebook7pic,command=openbook7)

openbutton7.pack()

ebooklabel7=Label(book7,text='The Island of \nDoctor Moreau',font= "timesnewroman 20")

ebooklabel7.pack()

spaceframe7=Frame(ebooksframebottom)

spaceframe7.pack(side= LEFT)

spacelabel7=Label(spaceframe7,text=' ')

spacelabel7.pack()

book8=Frame(ebooksframebottom)

book8.pack(side=LEFT)

ebook8pic=PhotoImage(file='./IMAGES/ebookspictures/ebook8.png')

openbutton8=Button(book8,text='Open', image = ebook8pic,command=openbook8)

openbutton8.pack()

ebooklabel8=Label(book8,text='The Aliens\n',font= "timesnewroman 20")

ebooklabel8.pack()

spaceframe8=Frame(ebooksframebottom)

spaceframe8.pack(side= LEFT)

spacelabel8=Label(spaceframe8,text=' ')

spacelabel8.pack()

book9=Frame(ebooksframebottom)

book9.pack(side=LEFT)

ebook9pic=PhotoImage(file='./IMAGES/ebookspictures/ebook9.png')

openbutton9=Button(book9,text='Open', image = ebook9pic,command=openbook9)

openbutton9.pack()

ebooklabel9=Label(book9,text='The Gods of Mars\n',font= "timesnewroman 20")

ebooklabel9.pack()

spaceframe9=Frame(ebooksframebottom)

spaceframe9.pack(side= LEFT)

spacelabel9=Label(spaceframe9,text=' ')

spacelabel9.pack()

book10=Frame(ebooksframebottom)

book10.pack(side=LEFT)

ebook10pic=PhotoImage(file='./IMAGES/ebookspictures/ebook10.png')

openbutton10=Button(book10,text='Open', image = ebook10pic,command=openbook10)

openbutton10.pack()

ebooklabel10=Label(book10,text='The Adventures of \nTom Sawyer',font= "timesnewroman 20")

ebooklabel10.pack()

spaceframe10=Frame(ebooksframebottom)

spaceframe10.pack(side= LEFT)

spacelabel10=Label(spaceframe10,text=' ')

spacelabel10.pack()

book11=Frame(ebooksframebottom)

book11.pack(side=LEFT)

ebook11pic=PhotoImage(file='./IMAGES/ebookspictures/ebook11.png')

openbutton11=Button(book11,text='Open', image = ebook11pic,command=openbook11)

openbutton11.pack()

ebooklabel11=Label(book11,text='A Journey to the \nCentre of the Earth',font= "timesnewroman 20")

ebooklabel11.pack()

spaceframe11=Frame(ebooksframebottom)

spaceframe11.pack(side= LEFT)

spacelabel11=Label(spaceframe11,text=' ')

spacelabel11.pack()

book12=Frame(ebooksframebottom)

book12.pack(side=LEFT)

ebook12pic=PhotoImage(file='./IMAGES/ebookspictures/ebook12.png')

openbutton12=Button(book12,text='Open', image = ebook12pic,command=openbook12)

openbutton12.pack()

ebooklabel12=Label(book12,text='Robinson Crusoe',font= "timesnewroman 20")

ebooklabel12.pack()

spaceframe12=Frame(ebooksframebottom)

spaceframe12.pack(side= LEFT)

spacelabel12=Label(spaceframe12,text=' ')

spacelabel12.pack()

backbuttonframe=Frame(root)

backbuttonframe.pack()

backbutton=Button(backbuttonframe,text='Back',command=ebookpagetomain)

backbutton.pack()

root.mainloop()

def searchbookfn():

if currentlocationforserachbook == 'GENERAL':

destmainpageto()

global titleofsearchbookpage

global keywordframe

global resultframe

global backframe

def search():

global resultframe

findnamevalue=findname.get()

if len(findnamevalue) != 0:

findnamevalue=findnamevalue.lower()

cursor.execute("select \* from books where BOOK\_Title like '%{}%'".format(findnamevalue))

l=cursor.fetchall()

processlist=list(l)

n=len(processlist)

resultframe. pack\_forget()

resultframe=Frame(root)

resultframe.pack()

if n == 0:

sorrydoesnotexistlabel=Label(resultframe,text='Sorry, the book is not found', fg='red')

sorrydoesnotexistlabel.pack()

else:

newlinelabel=Label(resultframe,text='\n\nRESULT\n\n\n', fg='green',font= "20")

newlinelabel.pack()

#starts modify

canvas = Canvas(resultframe, width=650, height=400)

scroll\_y = Scrollbar(resultframe, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

ALLTEXTRESULTLABELTEXT=''

for a in range(n):

'''x='detailframe'+str(a)

globals()[x]=Frame(frameforscroll)

globals()[x].pack()'''

bookid= 'BOOK ID : '+str(processlist[a][0])+'\n'

bookname= 'BOOK TITLE : '+processlist[a][1].title()+'\n'

author= 'AUTHOR : '+processlist[a][2]+'\n'

publisher= 'PUBLISHER : '+processlist[a][3]+'\n'

availability='AVAILABILITY : '+processlist[a][4]+'\n'

ALLTEXTRESULTLABELTEXT+=bookid+bookname+author+publisher+availability

ALLTEXTRESULTLABELTEXT+='\n---------------------------------------------------------------------------------------------------------------------------------------\n\n'

#

ALLTEXTRESULTLABEL=Label(frameforscroll,text=ALLTEXTRESULTLABELTEXT, justify=LEFT)

ALLTEXTRESULTLABEL.pack()

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

# 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()

titleofsearchbookpage=Frame(root)

titleofsearchbookpage.pack()

searchbooktext = Label(titleofsearchbookpage, text = 'Search a Book\n\n\n', fg='blue',font= "applecherry 35")

searchbooktext.pack(side= TOP)

keywordframe=Frame(root)

keywordframe.pack()

lbltellingtext=Label(keywordframe,text='Type a Keyword ')

lbltellingtext.pack()

findname=Entry(keywordframe,width=30)

findname.pack()

findbutton=Button(keywordframe,text='Search',command=search)

findbutton.pack(side=BOTTOM)

resultframe=Frame(root)

resultframe.pack()

backframe=Frame(root)

backframe.pack(side=BOTTOM)

backbutton=Button(backframe,text='Back',command=searchtomain)

backbutton.pack()

root.mainloop()#find similarity between names and try to capitaliese the first letter of entry before giving in the search variable

#create a for loop to make a table like structure in the tkinter page

def newpage():

'''framepic=Frame(root)

framepic.pack()

photo2=PhotoImage(file='electrycity formula.png') #REMOVE THIS

labelframe=Label(framepic,image=photo2)

labelframe.pack()

root.mainloop()'''

def successfultomainpage():

frmthankyou. pack\_forget()

mainpage()

def feedbackoage():

destmainpageto()

global titleoffeedbackpage

global detailsofpersonframe

global suggestiongivingframe

global errormessagecount #just a variable

global errormessagecountforinvalidentry#just a variable

errormessagecount=0 #variable for keeping track of error message

errormessagecountforinvalidentry =0 #variable for keeping track of error message

def successful():

global frmthankyou

titleoffeedbackpage. pack\_forget()

detailsofpersonframe. pack\_forget()

suggestiongivingframe. pack\_forget()

frmthankyou=Frame(root)

frmthankyou.pack()

lblthankyou=Label(frmthankyou, text = '\n\n\nThank You for your Feedback :)\n\n', fg='green',font= "applecherry 30 italic")

lblthankyou.pack()

backbtn=Button(frmthankyou,text='Back',command=successfultomainpage)

backbtn.pack()

root.mainloop()

def submitsuggestion():

global errormessagecount

global errormessagecountforinvalidentry

name = nameofuserentry.get()

nric=nricofuserentry.get()

email=emailofuserentry.get()

noofstars= var.get()

othersuggestion=TextArea.get("1.0",END)

if len(name)!=0 and len(nric)!=0 and len(email)>10 and noofstars != 0:

try:

m="aministrator.library@gmail.com"

sub='Feedback from '+ name

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 smtp:

smtp.ehlo()

smtp.starttls()

smtp.ehlo()

smtp.login('aministrator.library@gmail.com','scyjovpwhqzfbrur')

subject =sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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

#storing in txt file

savefile=open('./REPORTS/suggestions.txt','a')

msg+='\n\n\n'

savefile.write(msg)

savefile.close()

#thanking the user

m=email

sub='Thank You '+ name

msg='Dear ' + str(name) + ',\n Thank You For Your Feed Back\n\n Warm Regards \n Library Team'

with smtplib.SMTP('smtp.gmail.com', 587) as smtp:

smtp.ehlo()

smtp.starttls()

smtp.ehlo()

smtp.login('aministrator.library@gmail.com','scyjovpwhqzfbrur')

subject =sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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

successful()

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

titleoffeedbackpage=Frame(root)

titleoffeedbackpage.pack()

feedbacktitletext = Label(titleoffeedbackpage, text = 'Feedback', fg='blue',font= "applecherry 35")

feedbacktitletext.pack(side= TOP)

suggestiongivingframe=Frame(root)

suggestiongivingframe.pack()

leaveyoursuggestiontext=Label(suggestiongivingframe, text='\nLeave Your Feedback Here\n', font='verdana 15')

leaveyoursuggestiontext.pack(side= LEFT)

suggestiongivingframe.pack()

detailsofpersonframe=Frame(root)

detailsofpersonframe.pack()

#name

nameframe=Frame(detailsofpersonframe)

nameframe.pack()

nameofuserlabel=Label(nameframe,text=' \t\t Your Name ')

nameofuserentry=Entry(nameframe,width=20)

nameofuserlabel.pack(side=LEFT)

nameofuserentry.pack(side=LEFT)

compname=Label(nameframe,text=' \*',fg='red')

compname.pack(side=LEFT)

#nric

nricofpersonframe=Frame(detailsofpersonframe)

nricofpersonframe.pack()

nricofuserlabel=Label(nricofpersonframe,text='\n 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)

#email

emailofpersonframe=Frame(detailsofpersonframe)

emailofpersonframe.pack()

emailofuserlabel=Label(emailofpersonframe,text='\n\t\t\t Your Email Address \t')

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()

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, value=2)

radiobtn2star.pack(side=LEFT)

radiobtn3star = Radiobutton(ratingsframe)

star3 = PhotoImage(file="./IMAGES/Icons/stars.png")

radiobtn3star.config(image=star3, variable=var, value=3)

radiobtn3star.pack(side=LEFT)

radiobtn4star = Radiobutton(ratingsframe)

star4 = PhotoImage(file="./IMAGES/Icons/stars.png")

radiobtn4star.config(image=star4, variable=var, value=4)

radiobtn4star.pack(side=LEFT)

radiobtn5star = Radiobutton(ratingsframe)

star5 = PhotoImage(file="./IMAGES/Icons/stars.png")

radiobtn5star.config(image=star5, variable=var, value=5)

radiobtn5star.pack(side=LEFT)

highestlabel=Label(ratingsframe,text='Excellent')

highestlabel.pack(side=LEFT)

comperating=Label(ratingsframe,text=' \*',fg='red')

comperating.pack(side=LEFT)

#othersuggestions

othersuggestionframe=Frame(detailsofpersonframe)

othersuggestionframe.pack()

othersuggestionslabel=Label(othersuggestionframe,text='\nDo you have any other suggestions?')

othersuggestionslabel.pack()

TextArea = Text(othersuggestionframe,selectborderwidth=3)

TextArea.pack(expand=YES, fill=BOTH)

buttonsframe=Frame(detailsofpersonframe)

buttonsframe.pack(side=BOTTOM)

compulsorylabel=Label(buttonsframe,text=' \* compulsory',fg='red')

compulsorylabel.pack()

submitbutton= Button(buttonsframe, text="Submit", command=submitsuggestion)

submitbutton.pack()

backbutton = Button(buttonsframe, text="Back", command=suggestionstomain)

backbutton.pack()

root.mainloop()

def helpme():

def back():

titleoffeedbackpage.pack\_forget()

frameforscroll.pack\_forget()

scroll\_y.pack\_forget()

canvas.pack\_forget()

mainpage()

destmainpageto()

titleoffeedbackpage=Frame(root)

titleoffeedbackpage.pack()

feedbacktitletext = Label(titleoffeedbackpage, text = 'Help\n', fg='blue',font= "applecherry 35")

feedbacktitletext.pack(side= TOP)

canvas = Canvas(root, width=650, height=400)

scroll\_y = Scrollbar(root, orient="vertical", command=canvas.yview)

frameforscroll = Frame(root)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

#####START HERE#####

#SAVE THE IMAGE IN folder -->

#computer science project\computer science project\help page

loginpageimage=PhotoImage(file='./IMAGES/Helppictures/Login page(help).png') ##INSERT THE IMAGE AS imagename.extension

loginpagelabel=Label(frameforscroll,image=loginpageimage)

loginpagelabel.pack()

searchbookimage=PhotoImage(file='./IMAGES/Helppictures/help-searchbook.png') ##INSERT THE IMAGE AS imagename.extension

searchbooklabel=Label(frameforscroll,image=searchbookimage)

searchbooklabel.pack()

ebookimage=PhotoImage(file='./IMAGES/Helppictures/ebook(help).png')

ebooklabel=Label(frameforscroll,image=ebookimage)

ebooklabel.pack()

ebookwimage=PhotoImage(file='./IMAGES/Helppictures/ebookweb(help).png')

ebookwlabel=Label(frameforscroll,image=ebookwimage)

ebookwlabel.pack()

feedbackimage=PhotoImage(file='./IMAGES/Helppictures/feedback(help).png')

feedbacklabel=Label(frameforscroll,image=feedbackimage)

feedbacklabel.pack()

aboutusimage=PhotoImage(file='./IMAGES/Helppictures/abous us(help).png')

aboutuslabel=Label(frameforscroll,image=aboutusimage)

aboutuslabel.pack()

createacimage=PhotoImage(file='./IMAGES/Helppictures/create page(help).png')

createaclabel=Label(frameforscroll,image=createacimage)

createaclabel.pack()

resetacimage=PhotoImage(file='./IMAGES/Helppictures/reset password(help).png')

resetaclabel=Label(frameforscroll,image=resetacimage)

resetaclabel.pack()

userintimage=PhotoImage(file='./IMAGES/Helppictures/help-userinterface.png')

userint=Label(frameforscroll,image=userintimage)

userint.pack()

borrowedbksimage=PhotoImage(file='./IMAGES/Helppictures/help-borrowedbooks.png')

borrowedbks=Label(frameforscroll,image=borrowedbksimage)

borrowedbks.pack()

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

overduebks=Label(frameforscroll,image=overduebksimage)

overduebks.pack()

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

reservedbks=Label(frameforscroll,image=reservedbksimage)

reservedbks.pack()

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

composemail=Label(frameforscroll,image=composemailimage)

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 ప్రదర్శన వల్ల లైబ్రరీ నుండి బయటకు పంపించడం చూడడం 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 అప్గ్రేడ్ చేయడం జరిగింది\n\

మరియు అంతర్జాలం లో కూడా అందరికి సులభంగా చదువుకునేవిధంగా, జరిగింది.\n\

ఈ యూజర్ ఫ్రెండ్లీ GUI ఇంటర్ఫేస్ చదువుకునే వాళ్ళకి, వాడుకునే వాళ్ళకి మరియు లైబ్రరీ లో పనిచేసే వారికీ కంప్యూటర్‌ విధానం లో పైథాన్ని \n\

ఉపయోగించి వారి-వారి లావాదేవీలను మరియు ప్రక్రియలను పూర్తీ చేస్తుంది. \n\

ఈ పైథాన్ సాఫ్ట్‌వేర్ ఉపయోగించి అన్ని వివరాలను ప్రమాణీకరించడానికి ఉపయోగ పడుతుంది.\n\

ఈ వెబ్‌సైటని సురక్షితంగా ఉంచడానికి మరియు సైబర్ నరకాలు పలుబడకుంగా OTP లాంటి సురక్షితమైన విధానాలు ఉపయోగించడం జరిగింది.\n\

పాఠకులు ఇప్పుడు వారి ప్రతి అభ్యర్థనల కోసం వేగంగా ప్రాసెస్ సమయాన్ని ఆస్వాదించవచ్చు ...'

hinditext='\n\nक्या आप लोगों ने कभी सोचा है कि कुछ देशों गरीबी से कैसे पीड़ित हैं?\n\

कुछ बच्चों उनके दिखाव के वजह से, या उनके गरीबी के कारन से लाइब्रेरी से बाहर निकलवाना बहुत दुखी 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 तदनुसार विवरण प्रदान कर सकता है। इसे सुरक्षित बनाने के लिए पुस्तकालय 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 अनुरोध के लिए तेज़ प्रक्रिया समय का आनंद ले सकते हैं ...'

def langchange():

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 == 'తెలుగు':

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=libraryphoto)

#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","தமிழ்","తెలుగు","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()

w = OptionMenu(langoptions, variable, \*OPTIONS)

w.pack()

langchangebutton = Button(langoptions, text="Change Language", command=langchange)

langchangebutton.pack()

root.mainloop()

def createaccountpagetomain():

framehaveingheadingincreateaccount. pack\_forget()

detailsofpersonframeforcreatingaccount. pack\_forget()

createaccountframebuttons. pack\_forget()

errorstreams. pack\_forget()

loginpage()

def createaccountpage():

global framehaveingheadingincreateaccount

global detailsofpersonframeforcreatingaccount

global createaccountframebuttons

global captchanumber #variable

global errorstreams

framehaveingheadingincreateaccount = Frame(root,width=1000, height= 1000)

framehaveingheadingincreateaccount.pack()

lbl=Label(framehaveingheadingincreateaccount,text='Create Account \n', fg='blue',font= "applecherry 35")

lbl.pack(side= TOP)

def extractusers():

global unamelist

global passlist

global libidlist

f=open('./STORAGE/encrypteduserdetails.txt','rb')

l=pickle.load(f)

libidlist=[]

unamelist=[]

passlist=[]

for a in l:

libidlist.append(a[0])

unamelist.append(a[1])

passlist.append(a[2])

def validate():

global countnowidontknowwhatelsetokeepnow #just a avriable

countnowidontknowwhatelsetokeepnow = 0

def createthisaccount():

def verifyotp():

global errormessagecount

def loginnow():

accountcreatedmessage. pack\_forget()

loginpage()

errormessagecount = 0

if str(valofotp)==labelaskingtheotp.get():

optconformationpageincreatingaccountheader. pack\_forget()

inotpconformotpframe. pack\_forget()

accountcreatedmessage=Frame(root)

accountcreatedmessage.pack()

encryptedfile=open('./STORAGE/encrypteduserdetails.txt','rb')

newlibid=0

while True:

newlibid=random.randint(1000,9999)

if newlibid not in libidlist:

break

existingusernamesandpasswordlist=pickle.load(encryptedfile)

newexcrypteddata=(newlibid,uname,pwd)

encryptedfile.close()

encryptedfile=open('./STORAGE/encrypteduserdetails.txt','wb')

existingusernamesandpasswordlist.append(newexcrypteddata)

pickle.dump(existingusernamesandpasswordlist,encryptedfile)

cursor.execute("insert into users values ({},'{}','{}','{}','{}','{}','{}')".format(newlibid,uname,fname,lname,phone,nric,email))

mycon.commit()

encryptedfile.close()

texttobedisp='\n\n\nYour Account has been created successfully\n\nYour Library ID is %s \n\nTry to Login now'%newlibid

succesfullycreated=Label(accountcreatedmessage,text=texttobedisp,fg='green')

succesfullycreated.pack()

m=email

sub='Thank You '+ fname

msg='Dear ' + str(fname) + ',\n Thank You For Creting an Account with us\nYour Library ID is '+str(newlibid)+'\nYour Username is '+str(uname)+'\n\nAll Details will be kept confidential.\n\nWarm Regards \n Library Team'

with smtplib.SMTP('smtp.gmail.com', 587) as smtp:

smtp.ehlo()

smtp.starttls()

smtp.ehlo()

smtp.login('aministrator.library@gmail.com','scyjovpwhqzfbrur')

subject =sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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

loginnowbutton=Button(accountcreatedmessage,text='Login now',command=loginnow)

loginnowbutton.pack()

else:

global countnowidontknowwhatelsetokeepnow

if countnowidontknowwhatelsetokeepnow > 0 and countnowidontknowwhatelsetokeepnow <5:

countnowidontknowwhatelsetokeepnow+=1

elif countnowidontknowwhatelsetokeepnow >= 5:

labeltellingwrongtheotp=Label(inotpconformotpframe,text='Maximum Tries Reached',fg='red')

labeltellingwrongtheotp.pack()

inotpconformotpframe. pack\_forget()

inotpconformotpframe. pack\_forget()

optconformationpageincreatingaccountheader. pack\_forget()

loginpage()

else:

labeltellingwrongtheotp=Label(inotpconformotpframe,text='OTP does not match',fg='red')

labeltellingwrongtheotp.pack()

countnowidontknowwhatelsetokeepnow+=1

root.mainloop()

framehaveingheadingincreateaccount. pack\_forget()

detailsofpersonframeforcreatingaccount. pack\_forget()

createaccountframebuttons. pack\_forget()

errorstreams. pack\_forget()

optconformationpageincreatingaccountheader=Frame(root)

optconformationpageincreatingaccountheader.pack()

lbl=Label(optconformationpageincreatingaccountheader,text='Confirm Your Account \n\n\n', fg='blue',font= "applecherry 35")

lbl.pack()

valofotp= random.randint(1000,9999)

try:

m=email

sub='OTP Verification'

msg='Dear '+str(fname)+',\nThe OTP for your Libriary account is '+str(valofotp)+'\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','scyjovpwhqzfbrur')

subject = sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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

except:

if errormessagecount == 0:

plscheck=Label(inotpconformotpframe,text='Please Check Your Internet connection', fg='red')

plscheck.pack()

errormessagecount+=1

inotpconformotpframe=Frame(root)

inotpconformotpframe.pack()

labelgivinginformationforuser=Label(inotpconformotpframe,text='An OTP has been sent to your Account\nPlease Confirm your Account\n')

labelgivinginformationforuser.pack()

labelaskingtheotp=Entry(inotpconformotpframe,width=20,show='\*')

labelaskingtheotp.pack()

checktheotp=Button(inotpconformotpframe,text='Submit',command=verifyotp)

checktheotp.pack()

root.mainloop()

global invalidpasswordmessage

global errorstreams

fname=nameofuserentry.get()

lname=lnameofuserentry.get()

nric=nricofuserentry.get()

email=emailofuserentry.get()

phone=phoneofuserentry.get()

uname=usernameofuserentry.get()

pwd=passwordchosenentry.get()

pwdconf=cnpasswordchosenentry.get()

captcha=captchaentry.get()

extractusers()

errorstreams. pack\_forget()

errorstreams=Frame(root)

errorstreams.pack(side=BOTTOM)

inbetweenthisvariableisusedforcountingpurpose=0

if uname in unamelist:

unameexistlabel=Label(errorstreams,text='Username Taken', fg='red')#EVALUATION FOR UNIQUE USERNAME

unameexistlabel.pack()

inbetweenthisvariableisusedforcountingpurpose+=1

countforpasswordcaptletter,countforpasswordsmallletter, countforpasswordnumber=0,0,0 #PASSWORD EVALUATION

for varia in pwd:

if ord(varia) in range(65,91):

countforpasswordcaptletter+=1

elif ord(varia) in range(97,123):

countforpasswordsmallletter+=1

elif ord(varia) in range(48,58):

countforpasswordnumber+=1

if countforpasswordcaptletter==0 or countforpasswordsmallletter == 0 or countforpasswordnumber ==0 or len(pwd)<8:

passwordinvalidlabel=Label(errorstreams,text='Your Password is weak', fg='red')#EVALUATION FOR STRONG PASSWORD

passwordinvalidlabel.pack()

inbetweenthisvariableisusedforcountingpurpose+=1

if pwd != pwdconf:# #IF PASSWORD ENTERED MATCHEES WITH THE CONFIRMED ONE

passdoesnotmatchlabel=Label(errorstreams,text='Passwords Entered does not match', fg='red')

passdoesnotmatchlabel.pack()

inbetweenthisvariableisusedforcountingpurpose+=1

if listoffilesofcaptcha[captchanumber][1]!=captchaentry.get():

lblcaptchastats=Label(errorstreams,text='Wrong Captcha Please Retry', fg='red')

lblcaptchastats.pack()

inbetweenthisvariableisusedforcountingpurpose+=1

if inbetweenthisvariableisusedforcountingpurpose==0 and 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:

createthisaccount()

else:

enterallinfoerrormessagelabel=Label(errorstreams,text='Please Enter all Necessary Details', fg='red')

enterallinfoerrormessagelabel.pack()

root.mainloop()

detailsofpersonframeforcreatingaccount=Frame(root)

detailsofpersonframeforcreatingaccount.pack()

#name

nameframe=Frame(detailsofpersonframeforcreatingaccount)

nameframe.pack()

nameofuserlabel=Label(nameframe,text=' \t Your First Name \t')

nameofuserentry=Entry(nameframe,width=20)

nameofuserlabel.pack(side=LEFT)

nameofuserentry.pack(side=LEFT)

compname=Label(nameframe,text=' \*',fg='red')

compname.pack(side=LEFT)

lnameframe=Frame(detailsofpersonframeforcreatingaccount)

lnameframe.pack()

lnameofuserlabel=Label(lnameframe,text=' \t Your Last Name \t')

lnameofuserentry=Entry(lnameframe,width=20)

lnameofuserlabel.pack(side=LEFT)

lnameofuserentry.pack(side=LEFT)

complname=Label(lnameframe,text=' \*',fg='red')

complname.pack(side=LEFT)

#nric

nricofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

nricofpersonframe.pack()

nricofuserlabel=Label(nricofpersonframe,text='\nLast 4 characters NRIC/Passport Number\t')

nricofuserentry=Entry(nricofpersonframe,width=20)

nricofuserlabel.pack(side=LEFT)

nricofuserentry.pack(side=LEFT)

compnric=Label(nricofpersonframe,text=' \*',fg='red')

compnric.pack(side=LEFT)

#phonenumber

phoneofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

phoneofpersonframe.pack()

phoneofuserlabel=Label(phoneofpersonframe,text='\n\tYour Phone Number\t\t')

phoneofuserentry=Entry(phoneofpersonframe,width=20)

phoneofuserlabel.pack(side=LEFT)

phoneofuserentry.pack(side=LEFT)

compphone=Label(phoneofpersonframe,text=' \*',fg='red')

compphone.pack(side=LEFT)

#email

emailofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

emailofpersonframe.pack()

emailofuserlabel=Label(emailofpersonframe,text='\n\t\t\tYour Email Address\t\t')

emailofuserentry=Entry(emailofpersonframe,width=30)

emailofuserlabel.pack(side=LEFT)

emailofuserentry.pack(side=LEFT)

compemail=Label(emailofpersonframe,text=' \*\t\n\n',fg='red')

compemail.pack(side=LEFT)

#username

usernameofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

usernameofpersonframe.pack()

usernameofuserlabel=Label(usernameofpersonframe,text='\n\tSelect an Username \t\t')

usernameofuserentry=Entry(usernameofpersonframe,width=20)

usernameofuserlabel.pack(side=LEFT)

usernameofuserentry.pack(side=LEFT)

compusername=Label(usernameofpersonframe,text=' \*',fg='red')

compusername.pack(side=LEFT)

#password

passwordchosen=Frame(detailsofpersonframeforcreatingaccount)

passwordchosen.pack()

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

passwordchosenentry=Entry(passwordchosen,width=20, show="\*")

passwordchosenlabel.pack(side=LEFT)

passwordchosenentry.pack(side=LEFT)

compuserpassword=Label(passwordchosen,text=' \*',fg='red')

compuserpassword.pack(side=LEFT)

#conformpassword

cnpasswordchosen=Frame(detailsofpersonframeforcreatingaccount)

cnpasswordchosen.pack()

cnpasswordchosenlabel=Label(cnpasswordchosen,text='\n\tConfirm Password \t\t')

cnpasswordchosenentry=Entry(cnpasswordchosen,width=20, show="\*")

cnpasswordchosenlabel.pack(side=LEFT)

cnpasswordchosenentry.pack(side=LEFT)

cncompuserpassword=Label(cnpasswordchosen,text=' \*\n\n',fg='red')

cncompuserpassword.pack(side=LEFT)

criteriaforpassword=Frame(detailsofpersonframeforcreatingaccount)

criteriaforpassword.pack()

passwordcriteria=Label(criteriaforpassword,text='\nYour Password should contain\natleast 8 characters,\natleast 1 digit and \natleast 1 uppercase character\n\n')

passwordcriteria.pack(side=BOTTOM)

#captcha

captchaframe = Frame(detailsofpersonframeforcreatingaccount,width=1000, height= 1000)

captchaframe.pack()

captchanumber= random.randint(1,9)-1

photo=PhotoImage(file=listoffilesofcaptcha[captchanumber][0])

captchaphoto=Label(captchaframe,image=photo)

captchaentry = Entry(captchaframe, width=20)

captchaentry.insert(0, "Enter Captcha Here")

captchaphoto.pack(side= LEFT)

compuserpassword=Label(captchaframe,text='\t',fg='red')

compuserpassword.pack(side=LEFT)

captchaentry.pack(side= LEFT)

compuserpassword=Label(captchaframe,text=' \*\n\n',fg='red')

compuserpassword.pack(side=LEFT)

errorstreams=Frame(root)

errorstreams.pack(side=BOTTOM)

createaccountframebuttons=Frame(root)

createaccountframebuttons.pack(side=BOTTOM)

compulsorylabel=Label(createaccountframebuttons,text='\n\n \* compulsory\n',fg='red')

compulsorylabel.pack()

Backtomain=Button(createaccountframebuttons, text = 'Back', command= createaccountpagetomain)

Backtomain.pack(side = BOTTOM)

sumbitbutton=Button(createaccountframebuttons,text='Submit',command=validate)

sumbitbutton.pack(side=BOTTOM)

'''

submitcred=Button(frameloginbuttons, text = 'Submit', command= validate)

submitcred.pack(side = BOTTOM)

forgotpassbutton=Button(frameloginbuttons, text = 'Forgot Password', command= forgotpass)

forgotpassbutton.pack(side = BOTTOM)

createnewaccount=Button(frameloginbuttons, text = 'Create Account', command= createaccountfromloginpage)

createnewaccount.pack(side = BOTTOM)

captchaphoto.pack(side= LEFT)

captchaentry.pack(side= BOTTOM)

'''

root.mainloop()

def librarianspage():

def borrowbook():

global statsframe

global bookid #JUST A VARIABLE

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

def back():

detailstoprocessframe.pack\_forget()

btnframe.pack\_forget()

detailstoprocessframe.pack\_forget()

statsframe.pack\_forget()

lbl.pack\_forget()

librarianspage()

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()

borrowbook()

cursor.execute('select \* from borrowedbooks')

ls=cursor.fetchall()

vblbks='Number of Books Borrowed: '+str(len(ls))+'\n\n'

for a in range(len(ls)):

vblbks+='\nUSER ID : '+ str(ls[a][0]) + '\nBOOK ID : '+ str(ls[a][1])+ '\nDUE DATE : '+ str(ls[a][2])+'\n\n'

vblbks+='\n---------------------------------------------------------------------------------------------------------------------------------------\n\n'

lblb=Label(root,text='Borrowed Books \n\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING

lblb.pack(side= TOP)

lbldetails=Frame(root)

lbldetails.pack()

canvas = Canvas(lbldetails, width=650, height=400)

scroll\_y = Scrollbar(lbldetails, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= vblbks, justify=LEFT)

overduelbl.pack()

# 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')

backbtn=Button(root, text='Back', command = back)

backbtn.pack(side=BOTTOM)

root.mainloop()

def borrow():

global errormessagecount

errormessagecount=0 #just a variable

def finishandsendamail():

global errormessagecount

try:

m=eml

sub='Borrowed Books'

msg=ftxtborrowedbooksvariable+borrowedbooksvariable+'\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','scyjovpwhqzfbrur')

subject = sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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:

if errormessagecount == 0:

plscheck=Label(statsframe,text='Please Check Your Internet connection', fg='red')

plscheck.pack()

errormessagecount+=1

def nextbkfn():

global statsframe

global detailstoprocessframe

global borrowedbooksvariable

global bookid

statsframe.pack\_forget()

detailstoprocessframe.pack\_forget()

detailstoprocessframe=Frame(root) #A FRAME

detailstoprocessframe.pack()

statsframe=Frame(root)

statsframe.pack()

bkidlbl=Label(detailstoprocessframe,text='Book ID')

bkidlbl.pack()

bookid=Entry(detailstoprocessframe, width=20)

bookid.pack(side= LEFT)

global statsframe

global borrowedbooksvariable

global detailstoprocessframe

global bookid

try:

statsframe.pack\_forget()

except:

pass

statsframe=Frame(root)

statsframe.pack()

bid=bookid.get()

libid=lid.get()

if len(bid) == 0 or len(libid) == 0:

notfoundlbl=Label(statsframe,text='\n\nPlease enter all details', fg='Red')

notfoundlbl.pack()

return

cursor.execute('select \* from books where bookid= {}'.format(bid))

l=cursor.fetchone()

if len(l) > 0:

req = list(l)

else:

notfoundlbl=Label(statsframe,text='\n\nBook not Found', fg='Red')

notfoundlbl.pack()

return

cursor.execute('select \* from users where LIBID= {}'.format(libid))

m=cursor.fetchall()

if len(m) > 0:

requsr = list(m[0])

else:

notfoundurlbl=Label(statsframe,text='\n\nUser not Found', fg='Red')

notfoundurlbl.pack()

return

if req[4] == 'Borrowed':

borrowedlbl=Label(statsframe,text='\n\nBook is already Borrowed', fg='Red')

borrowedlbl.pack()

else:

cursor.execute('select \* from reservations where BOOKID = \'{}\''.format(bid))

resbkslst=cursor.fetchall()

if len(resbkslst) > 0:

if str(resbkslst[0][0]) == libid:

cursor.execute("delete from reservations where BOOKID = \'{}\'".format(bid))

mycon.commit()

else:

borrowedlbl=Label(statsframe,text='\n\nBook is Reserved', fg='Red')

borrowedlbl.pack()

return

req[4] = 'Borrowed'

BKNM=req[1]

cursor.execute('update books set Availability = \'{}\' where bookid = \'{}\''.format(req[4],req[0]))

mycon.commit()

now=datetime.now() + timedelta(days=14) #take in function

txtl=now.strftime("%d-%m-%y")

#SHOW MESSAGE THAT THE BOOK IS BORROWED, ASK FOR NEXT BOOK TO BORROW,

brrbktxt=str(libid)+','+str(bid)+','+str(txtl)+'\n'

cursor.execute("insert into borrowedbooks values ({},{},DATE\_ADD(curdate(), INTERVAL 14 DAY))".format(libid,bid))

mycon.commit()

sucmsg='\n\nThe Book '+BKNM +' has been Issued to '+requsr[2]+'\n Due Date: '+str(txtl)

borrowedlbl=Label(statsframe,text=sucmsg, fg='Green')

borrowedlbl.pack()

ftxtborrowedbooksvariable='Dear '+requsr[2]+' '+requsr[3]+',\nYou have Borrowed The Following Books\n\n'

borrowedbooksvariable+=BKNM+'\t'+ 'Due Date: '+str(txtl)+'\n'

eml=requsr[6]

nextbk=Button(statsframe,text='Borrow Another Book for Same User',command=nextbkfn)

nextbk.pack()

finshbtn=Button(statsframe,text='Finish and Send a Mail', command= finishandsendamail)

finshbtn.pack()

global borrowedbooksvariable #JUST A VARIABLE

global detailstoprocessframe

lbl=Label(root,text='Borrow 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)

lid.pack(side= LEFT)

bkidlbl=Label(detailstoprocessframe,text='Book ID')

bkidlbl.pack(side= LEFT)

bookid=Entry(detailstoprocessframe, width=20)

bookid.pack(side= LEFT)

borrowedbooksvariable=''

statsframe=Frame(root)

statsframe.pack()

btnframe=Frame(root)# A FRAME

btnframe.pack(side=BOTTOM)

borrowbtn=Button(btnframe, text= 'Borrow', command= borrow)

borrowbtn.pack()

viewbrwbtn=Button(btnframe, text='View Borrowed Books', command= viewborrowedbook)

viewbrwbtn.pack()

backbtn=Button(btnframe, text='Back', command= back)

backbtn.pack()

root.mainloop()

def returnbook():

###############################29th june 2020###########################

global statsframe

global bookid #JUST A VARIABLE

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

def back():

detailstoprocessframe.pack\_forget()

btnframe.pack\_forget()

detailstoprocessframe.pack\_forget()

statsframe.pack\_forget()

lbl.pack\_forget()

librarianspage()

def returnbk():

global errormessagecount

errormessagecount=0 #just a variable

def finishandsendamail():

global errormessagecount

try:

m=eml

sub='Returned Books'

msg=ftxtborrowedbooksvariable+returnedbooksvariable+'\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','scyjovpwhqzfbrur')

subject = sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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:

if errormessagecount == 0:

plscheck=Label(statsframe,text='Please Check Your Internet connection', fg='red')

plscheck.pack()

errormessagecount+=1

def nextbkfn():

global statsframe

global detailstoprocessframe

global returnedbooksvariable

global bookid

statsframe.pack\_forget()

detailstoprocessframe.pack\_forget()

detailstoprocessframe=Frame(root) #A FRAME

detailstoprocessframe.pack()

statsframe=Frame(root)

statsframe.pack()

bkidlbl=Label(detailstoprocessframe,text='Book ID')

bkidlbl.pack()

bookid=Entry(detailstoprocessframe, width=20)

bookid.pack(side= LEFT)

global statsframe

global returnedbooksvariable ##################

global detailstoprocessframe

global bookid

try:

statsframe.pack\_forget()

except:

pass

statsframe=Frame(root)

statsframe.pack()

#################

bid=bookid.get()#

libid=lid.get()##

#################

if len(bid) == 0 or len(libid) == 0:

notfoundlbl=Label(statsframe,text='\n\nPlease enter all details', fg='Red')

notfoundlbl.pack()

return

cursor.execute('select \* from books where bookid= {}'.format(bid))

l=cursor.fetchone()

if len(l) > 0:

req = list(l)

else:

notfoundlbl=Label(statsframe,text='\n\nBook not Found', fg='Red')

notfoundlbl.pack()

return

cursor.execute('select \* from users where LIBID= {}'.format(libid))

m=cursor.fetchall()

if len(m) > 0:

requsr = list(m[0])

else:

notfoundurlbl=Label(statsframe,text='\n\nUser not Found', fg='Red')

notfoundurlbl.pack()

return

if req[4] == 'Available':

borrowedlbl=Label(statsframe,text='\n\nThe book has not been borrowed yet', fg='Red')

borrowedlbl.pack()

else:

cursor.execute('select \* from borrowedbooks where bookid= {}'.format(bid))

bbkslst=cursor.fetchall()

if len(bbkslst) != 0 and str(bbkslst[0][0]) != libid:

notfoundurlbl=Label(statsframe,text='\n\nThis User Did not borrow the mentioned book', fg='Red')

notfoundurlbl.pack()

return

######

bookstatus8thsug=''

cursor.execute('select \* from reservations where BOOKID = \'{}\''.format(bid))

resbkslst=cursor.fetchall()

if len(resbkslst) > 0:

bookstatus8thsug = 'Reserved'

else:

bookstatus8thsug = 'Available'

###### ############## CONVERT SQL ################

BKNM=req[1]

cursor.execute('update books set Availability = \'{}\' where bookid = \'{}\''.format(bookstatus8thsug,req[0]))

mycon.commit()

cursor.execute("delete from borrowedbooks where BOOKID = \'{}\'".format(bid))

mycon.commit()

sucmsg='\n\nThe Book '+BKNM +' Borrowed by '+requsr[2]+'\n Has been returned'

borrowedlbl=Label(statsframe,text=sucmsg, fg='Green')

borrowedlbl.pack()

ftxtborrowedbooksvariable='Dear '+requsr[2]+' '+requsr[3]+',\nYou have Returned The Following Books\n\n'

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()

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)

lid.pack(side= LEFT)

bkidlbl=Label(detailstoprocessframe,text='Book ID')

bkidlbl.pack(side= LEFT)

bookid=Entry(detailstoprocessframe, width=20)

bookid.pack(side= LEFT)

returnedbooksvariable=''

statsframe=Frame(root)

statsframe.pack()

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()

################

########################### 29th June 2020#####################################################

def reservebook():

global statsframe

global bookid #JUST A VARIABLE

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

def back():

detailstoprocessframe.pack\_forget()

btnframe.pack\_forget()

detailstoprocessframe.pack\_forget()

statsframe.pack\_forget()

lbl.pack\_forget()

librarianspage()

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()

cursor.execute('select \* from reservations')

ls=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---------------------------------------------------------------------------------------------------------------------------------------\n\n'

lblb=Label(root,text='Reserved Books \n\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING

lblb.pack(side= TOP)

lbldetails=Frame(root)

lbldetails.pack()

canvas = Canvas(lbldetails, width=650, height=400)

scroll\_y = Scrollbar(lbldetails, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= vblbks, justify=LEFT)

overduelbl.pack()

# 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')

backbtn=Button(root, text='Back', command = back)

backbtn.pack(side=BOTTOM)

root.mainloop()

def returnbk():

global errormessagecount

errormessagecount=0 #just a variable

def finishandsendamail():

global errormessagecount

try:

m=eml

sub='Reserved Books'

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','scyjovpwhqzfbrur')

subject = sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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:

if errormessagecount == 0:

plscheck=Label(statsframe,text='Please Check Your Internet connection', fg='red')

plscheck.pack()

errormessagecount+=1

def nextbkfn():

global statsframe

global detailstoprocessframe

global reservedbooksvariable

global bookid

statsframe.pack\_forget()

detailstoprocessframe.pack\_forget()

detailstoprocessframe=Frame(root) #A FRAME

detailstoprocessframe.pack()

statsframe=Frame(root)

statsframe.pack()

bkidlbl=Label(detailstoprocessframe,text='Book ID')

bkidlbl.pack()

bookid=Entry(detailstoprocessframe, width=20)

bookid.pack(side= LEFT)

global statsframe

global reservedbooksvariable ##################

global detailstoprocessframe

global bookid

try:

statsframe.pack\_forget()

except:

pass

statsframe=Frame(root)

statsframe.pack()

#################

bid=bookid.get()#

libid=lid.get()##

#################

if len(bid) == 0 or len(libid) == 0:

notfoundlbl=Label(statsframe,text='\n\nPlease enter all details', fg='Red')

notfoundlbl.pack()

return

cursor.execute('select \* from books where bookid= {}'.format(bid))

l=cursor.fetchone()

if len(l) > 0:

req = list(l)

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()##

return

cursor.execute('select \* from users where LIBID= {}'.format(libid))

m=cursor.fetchall()

if len(m) > 0:

requsr = list(m[0])

else:

notfoundurlbl=Label(statsframe,text='\n\nUser not Found', fg='Red')

notfoundurlbl.pack()

return

if req[4] == 'Available':

borrowedlbl=Label(statsframe,text='\n\nThe book has not been borrowed yet', fg='Red')

borrowedlbl.pack()

finshbtn=Button(btnframe,text='Finish and Send a Mail', command= finishandsendamail)

finshbtn.pack()

else:

cursor.execute('select \* from reservations where BOOKID = \'{}\''.format(bid))

resbkslst=cursor.fetchall()

if len(resbkslst) > 0:

notfoundurlbl=Label(statsframe,text='\n\nThe Book is Already Reserved', fg='Red')

notfoundurlbl.pack()

finshbtn=Button(statsframe,text='Finish and Send a Mail', command= finishandsendamail)

finshbtn.pack()##

return

cursor.execute('insert into reservations values ({},{})'.format(libid,bid))

mycon.commit()

########################################################################################

BKNM=req[1]

sucmsg='\n\nThe Book '+BKNM +' has been Reserved by '+requsr[2]

borrowedlbl=Label(statsframe,text=sucmsg, fg='Green')

borrowedlbl.pack()

ftxtborrowedbooksvariable='Dear '+requsr[2]+' '+requsr[3]+',\nYou have Reserved the following Books\n\n'

reservedbooksvariable+=BKNM+'\n'

eml=requsr[6]

nextbk=Button(statsframe,text='Reserve Another Book for Same User',command=nextbkfn)

nextbk.pack()

finshbtn=Button(statsframe,text='Finish and Send a Mail', command= finishandsendamail)

finshbtn.pack()

global reservedbooksvariable #JUST A VARIABLE

global detailstoprocessframe

lbl=Label(root,text='Reserve 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)

lid.pack(side= LEFT)

bkidlbl=Label(detailstoprocessframe,text='Book ID')

bkidlbl.pack(side= LEFT)

bookid=Entry(detailstoprocessframe, width=20)

bookid.pack(side= LEFT)

reservedbooksvariable=''

statsframe=Frame(root)

statsframe.pack()

btnframe=Frame(root)# A FRAME

btnframe.pack(side=BOTTOM)

borrowbtn=Button(btnframe, text= 'Reserve', command= returnbk)

borrowbtn.pack()

viewbrwbtn=Button(btnframe, text='View Reserved Books', command= viewborrowedbook)

viewbrwbtn.pack()

backbtn=Button(btnframe, text='Back', command= back)

backbtn.pack()

root.mainloop()

def renewbook():

global bookid

global renewedbooksvariable

global detailstoprocessframe

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

def back():

global statsframe

lbl.pack\_forget()

btnframe.pack\_forget()

detailstoprocessframe.pack\_forget()

statsframe.pack\_forget()

librarianspage()

def renew():

global errormessagecount

global renewedbooksvariable

global detailstoprocessframe

errormessagecount=0 #just a variable

def finishandsendamail():

global errormessagecount

global bookid

global renewedbooksvariable

try:

m=eml

sub='Renewed Books'

msg=ftxtborrowedbooksvariable+renewedbooksvariable+'\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','scyjovpwhqzfbrur')

subject = sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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:

if errormessagecount == 0:

plscheck=Label(statsframe,text='Please Check Your Internet connection', fg='red')

plscheck.pack()

errormessagecount+=1

def nextbkfn():

global statsframe

global detailstoprocessframe

global renewedbooksvariable

global bookid

statsframe.pack\_forget()

detailstoprocessframe.pack\_forget()

detailstoprocessframe=Frame(root) #A FRAME

detailstoprocessframe.pack()

statsframe=Frame(root)

statsframe.pack()

bkidlbl=Label(detailstoprocessframe,text='Book ID')

bkidlbl.pack()

bookid=Entry(detailstoprocessframe, width=20)

bookid.pack(side= LEFT)

global statsframe

global renewedbooksvariable

global detailstoprocessframe

global bookid

try:

statsframe.pack\_forget()

except:

pass

statsframe=Frame(root)

statsframe.pack()

yefrforat=0

bid=bookid.get()

libid=lid.get()

#[['\ufeffUser\_Lib\_ID', 'Book\_ID', 'Due\_Date\n'], ['4656', '1', '06-07-20\n'], ['2', '2', '06-07-20\n'], ['4656', '3', '06-07-20']]

if len(bid) == 0 or len(libid) == 0:

notfoundlbl=Label(statsframe,text='\n\nPlease enter all details', fg='Red')

notfoundlbl.pack()

return

cursor.execute('select \* from borrowedbooks where BOOKID = \'{}\''.format(bid))

l=cursor.fetchall()

if len(l) > 0:

req = list(l[0])

else:

notfoundlbl=Label(statsframe,text='\n\nBook not Found', fg='Red')

notfoundlbl.pack()

return

if str(l[0][0]) != libid:

notfoundurlbl=Label(statsframe,text='\n\nThis user did not borrow the mentioned book', fg='Red')

notfoundurlbl.pack()

return

now=datetime.now() + timedelta(days=18) #take in function

txtl=now.strftime("%d-%m-%y")

#MODIFY HERE TO ADD 18 DAYS

#SHOW MESSAGE THAT THE BOOK IS BORROWED, ASK FOR NEXT BOOK TO BORROW,

cursor.execute('update borrowedbooks set duedate = DATE\_ADD(duedate, INTERVAL 10 DAY) where bookid = {}'.format(bid))

mycon.commit()

cursor.execute('select duedate from borrowedbooks where BOOKID = \'{}\''.format(bid))

hts=cursor.fetchall()

txtl=hts[0][0]

sucmsg='\n\nThe Book has been Renewed \n Due Date: '+str(txtl)

borrowedlbl=Label(statsframe,text=sucmsg, fg='Green')

borrowedlbl.pack()

cursor.execute('select \* from users where LIBID= {}'.format(libid))

m=cursor.fetchall()

requsr = list(m[0])

cursor.execute('select BOOK\_Title from books where BOOKID = {}'.format(bid))

lmn=cursor.fetchall()

BKNM=lmn[0][0]

##########################\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_###############

ftxtborrowedbooksvariable='Dear '+requsr[2]+' '+requsr[3]+',\nYou have Renewed the Following Books\n\n'

renewedbooksvariable+=BKNM+'\t'+ 'Due Date: '+str(txtl)+'\n'

eml=requsr[6]

nextbk=Button(statsframe,text='Renew Another Book for Same User',command=nextbkfn)

nextbk.pack()

finshbtn=Button(statsframe,text='Finish and Send a Mail', command= finishandsendamail)

finshbtn.pack()

####################

lbl=Label(root,text='Renew Books \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)

lid.pack(side= LEFT)

bkidlbl=Label(detailstoprocessframe,text='Book ID')

bkidlbl.pack(side= LEFT)

bookid=Entry(detailstoprocessframe, width=20)

bookid.pack(side= LEFT)

renewedbooksvariable=''

statsframe=Frame(root)

statsframe.pack()

btnframe=Frame(root)# A FRAME

btnframe.pack(side=BOTTOM)

borrowbtn=Button(btnframe, text= 'Renew', command= renew)

borrowbtn.pack()

backbtn=Button(btnframe, text='Back', command= back)

backbtn.pack()

root.mainloop()

def overduebook():

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

def back():

lbl.pack\_forget()

lbldetails.pack\_forget()

backbtn.pack\_forget()

librarianspage()

odbks=''

cursor.execute('select \* from borrowedbooks where duedate < curdate()')

lsl=cursor.fetchall()

if len(lsl)>0:

for a in range(len(lsl)):

odbks+='\nUser Id : '+ str(lsl[a][0]) + '\nBook Id : '+ str(lsl[a][1])+'\n\n'

odbks+='\n---------------------------------------------------------------------------------------------------------------------------------------\n\n'

lbl=Label(root,text='Overdue Books \n\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING

lbl.pack(side= TOP)

lbldetails=Frame(root)

lbldetails.pack()

canvas = Canvas(lbldetails, width=650, height=400)

scroll\_y = Scrollbar(lbldetails, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack()

# 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')

backbtn=Button(root, text='Back', command = back)

backbtn.pack(side=BOTTOM)

root.mainloop()

############################### 3rd July 2020 #########################################################################

def messaging():

########################### 20th september 2020 ################################

#REPORT --- Problem with clear 10 messages

#even after clearing messages are displayed. they are to be rectified

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

def back():

global inboxframe

global sendframe

lbldetails.pack\_forget()

lbl.pack\_forget()

backbtn.pack\_forget()

try:

inboxframe.pack\_forget()

except:

pass

try:

manucomframe.pack\_forget()

except:

pass

try:

sendframe.pack\_forget()

except:

pass

librarianspage()

def inbox():

global overduelbl

global canvas

global scroll\_y

def clrmsgfn():

global manucomframe

global overduelbl

global overduelbl12

manucomframe.pack\_forget()

scroll\_y.pack\_forget()

cursor.execute('delete from messaging where toid = 2')

mycon.commit()

overduelbl.pack\_forget()

canvas.pack\_forget()

try:

overduelbl12.pack\_forget()

except:

pass

overduelbl=Label(inboxframe, text= 'All Messages are Cleared', fg='green')

overduelbl.pack()

manucomframe.pack\_forget()

def clr10msgfn():

global canvas

global overduelbl

global manucomframe

global scroll\_y

global overduelbl12

cursor.execute('delete from messaging where toid = 2')

mycon.commit()

manucomframe.pack\_forget()

canvas.pack\_forget()

scroll\_y.pack\_forget()

overduelbl.pack\_forget()

for a in range(5):

try:

lsl.pop(-1)

except:

break

if len(lsl)>0:

for a in lsl:

cursor.execute('insert into messaging values {}'.format(a))

mycon.commit()

overduelbl.pack\_forget()

try:

overduelbl12.pack\_forget()

except:

pass

overduelbl12=Label(inboxframe, text= 'Last 5 messages are Cleared\n', fg='green')

overduelbl12.pack()

odbks=''

if len(lsl)>0:

for a in range(len(lsl)):

odbks+='\n\n\nFROM: '+ str(lsl[a][1]) + '\tUSER Id: '+ str(lsl[a][0]) + '\nMESSAGE:\t\t\t\n\n'+ str(lsl[a][3])+'\n'

odbks+='---------------------------------------------------------------------------------------------------------------------------------------------------------\n'

canvas = Canvas(inboxframe, width=1000, height=400)

scroll\_y = Scrollbar(inboxframe, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack()

# 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')

manucomframe=Frame(root)

manucomframe.pack(side=BOTTOM)

clrmsg=Button(manucomframe, text='Clear All Messages', command=clrmsgfn)

clr10msg=Button(manucomframe, text='Clear Last 5 Messages', command=clr10msgfn)

clrmsg.pack()

clr10msg.pack()

else:

canvas.pack\_forget()

overduelbl=Label(inboxframe, text= 'There are no Messages')

overduelbl.pack()

global inboxframe

global sendframe

global manucomframe

try:

inboxframe.pack\_forget()

manucomframe.pack\_forget()

except:

pass

try:

sendframe.pack\_forget()

except:

pass

inboxframe=Frame(root)

inboxframe.pack(side=TOP)

lbl=Label(inboxframe,text='INBOX \n', fg='blue',font= "applecherry 25") #THIS HAS TO BE DETELED WHILE NAVIGATING

lbl.pack(side= TOP)

odbks=''

cursor.execute('select \* from messaging where toid = 2')

lsl=cursor.fetchall()

lsl=lsl[::-1]

if len(lsl)>0:

for a in range(len(lsl)):

odbks+='\n\n\nFROM: '+ str(lsl[a][1]) + '\tUSER Id: '+ str(lsl[a][0]) + '\n MESSAGE:\t\t\t\n\n '+ str(lsl[a][3])+'\n'

odbks+='---------------------------------------------------------------------------------------------------------------------------------------------------------\n'

canvas = Canvas(inboxframe, width=1000, height=400)

scroll\_y = Scrollbar(inboxframe, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack()

# 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')

manucomframe=Frame(root)

manucomframe.pack(side=BOTTOM)

clrmsg=Button(manucomframe, text='Clear All Messages', command=clrmsgfn)

clr10msg=Button(manucomframe, text='Clear Last 5 Messages', command=clr10msgfn)

clrmsg.pack()

clr10msg.pack()

else:

overduelbl=Label(inboxframe, text= 'There are no Messages')

overduelbl.pack()

def send():

global errorframe

def submsg():

global errorframe

errorframe.pack\_forget()

errorframe=Frame(sendframe)

errorframe.pack()

recid=nricofuserentry.get()

messagesent=TextArea.get("1.0",END)

if len(recid)==0:

notfoundurlbl=Label(errorframe,text='\n\nPlease Enter Recipient Library ID ', fg='Red')

notfoundurlbl.pack()

return

if len(messagesent)<=1:

notfoundurlbl=Label(errorframe,text='\n\nPlease Type your Message ', fg='Red')

notfoundurlbl.pack()

return

cursor.execute('select \* from users where LIBID= {}'.format(recid))

m=cursor.fetchall()

if len(m) > 0:

pass

else:

notfoundurlbl=Label(errorframe,text='\n\nUser not Found', fg='Red')

notfoundurlbl.pack()

return

if len(messagesent)>700:

notfoundurlbl=Label(errorframe,text='\n\nMax Characters Exceeded', fg='Red')

notfoundurlbl.pack()

return

sub='MESSAGE FROM LIBRARIAN'

msg='Dear '+str(m[0][2])+',\nYou have Recived a message from Librarian \nThe message is:\n\n'+str(messagesent)+'\n\nWarm Regards\nLibrary Team'

try:

with smtplib.SMTP('smtp.gmail.com', 587) as smtp:

smtp.ehlo()

smtp.starttls()

smtp.ehlo()

smtp.login('aministrator.library@gmail.com','scyjovpwhqzfbrur')

subject = sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

smtp.sendmail('aministrator.library@gmail.com',m[0][6],msg)

except:

pass

detailsofpersonframeforcreatingaccount.pack\_forget()

notfoundurlbl=Label(errorframe,text='\n\nYour Message has been sent', fg='Green')

notfoundurlbl.pack()

cursor.execute("insert into messaging values (2,'LIBRARIAN',{},'{}')".format(recid,messagesent))

mycon.commit()

root.mainloop()

global inboxframe

global sendframe

try:

inboxframe.pack\_forget()

manucomframe.pack\_forget()

except:

pass

try:

sendframe.pack\_forget()

except:

pass

sendframe=Frame(root)

sendframe.pack()

lbl=Label(sendframe,text='COMPOSE MAIL \n', fg='blue',font= "applecherry 25") #THIS HAS TO BE DETELED WHILE NAVIGATING

lbl.pack(side= TOP)

detailsofpersonframeforcreatingaccount=Frame(sendframe)

detailsofpersonframeforcreatingaccount.pack()

#TO ID

nricofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

nricofpersonframe.pack()

nricofuserlabel=Label(nricofpersonframe,text='\nRecipient Library ID \t')

nricofuserentry=Entry(nricofpersonframe,width=10)

nricofuserlabel.pack(side=LEFT)

nricofuserentry.pack(side=LEFT)

nricofuserlabel12=Label(nricofpersonframe,text='\t\t\t\t\t')

nricofuserlabel12.pack(side=LEFT)

#message

usernameofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

usernameofpersonframe.pack()

usernameofuserlabel=Label(usernameofpersonframe,text='\n\tYour Message \t')

TextArea = Text(usernameofpersonframe,selectborderwidth=3)

usernameofuserlabel.pack(side=LEFT)

TextArea.pack(expand=YES, fill=BOTH)

compusername=Label(usernameofpersonframe,text='Max 700 Characters\n\n\n')

compusername.pack(side=LEFT)

subbtn=Button(detailsofpersonframeforcreatingaccount, text='Send', command=submsg)

subbtn.pack()

errorframe=Frame(sendframe)

errorframe.pack()

root.mainloop()

lbl=Label(root,text='Messaging\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING

lbl.pack(side= TOP)

lbldetails=Frame(root)

lbldetails.pack()

btninbox=Button(lbldetails, text='Inbox', command=inbox)

btnsend=Button(lbldetails, text='Compose Mail', command=send)

btninbox.pack(side=BOTTOM)

btnsend.pack(side=BOTTOM)

backbtn=Button(root, text='Back', command = back)

backbtn.pack(side=BOTTOM)

root.mainloop()

def addnewbook():

####################### 4th Oct 2020 #######################

global errorframe

def back():

titleoffeedbackpage. pack\_forget()

detailsofpersonframe. pack\_forget()

suggestiongivingframe. pack\_forget()

errorframe.pack\_forget()

buttonsframe.pack\_forget()

librarianspage()

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

bookname = nameofuserentry.get()

author=nricofuserentry.get()

publisher=emailofuserentry.get()

availability='Available'

if len(bookname)==0 or len(author)==0 or len(publisher)==0:

plscheckentry=Label(errorframe,text='Please Give all necessary informations', fg='red')

plscheckentry.pack()

return

elif len(bookname)>40:

plscheckentry=Label(errorframe,text='Maximum Characters Exceeded for Book Title', fg='red')

plscheckentry.pack()

return

elif len(author)>40:

plscheckentry=Label(errorframe,text='Maximum Characters Exceeded for Author', fg='red')

plscheckentry.pack()

return

elif len(publisher)>100:

plscheckentry=Label(errorframe,text='Maximum Characters Exceeded for Publisher', fg='red')

plscheckentry.pack()

return

else:

cursor.execute("insert into books values ({},'{}','{}','{}','{}')".format(newbookid,

bookname,

author,

publisher,

availability))

mycon.commit()

detailsofpersonframe. pack\_forget()

suggestiongivingframe. pack\_forget()

submitbutton.pack\_forget()

plscheckentry=Label(errorframe,text='The New Book Has Been Uploaded\n\n\

Book Id: {}'.format(newbookid), fg='green', justify=LEFT)

plscheckentry.pack()

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

titleoffeedbackpage=Frame(root)

titleoffeedbackpage.pack()

feedbacktitletext = Label(titleoffeedbackpage, text = 'ADD NEW BOOK', fg='blue',font= "applecherry 35")

feedbacktitletext.pack(side= TOP)

suggestiongivingframe=Frame(root)

suggestiongivingframe.pack()

leaveyoursuggestiontext=Label(suggestiongivingframe, text='\n\n\n\n\nEnter Information About New Book Here\n\n\n')

leaveyoursuggestiontext.pack(side= LEFT)

suggestiongivingframe.pack()

detailsofpersonframe=Frame(root)

detailsofpersonframe.pack()

#name

nameframe=Frame(detailsofpersonframe)

nameframe.pack()

nameofuserlabel=Label(nameframe,text='Book Title ',justify=RIGHT)

nameofuserentry=Entry(nameframe,width=30)

nameofuserlabel.pack(side=LEFT)

nameofuserentry.pack(side=LEFT)

#nric

nricofpersonframe=Frame(detailsofpersonframe)

nricofpersonframe.pack()

nricofuserlabel=Label(nricofpersonframe,text='\nAuthor ',justify=RIGHT)

nricofuserentry=Entry(nricofpersonframe,width=30)

nricofuserlabel.pack(side=LEFT)

nricofuserentry.pack(side=LEFT)

#email

emailofpersonframe=Frame(detailsofpersonframe)

emailofpersonframe.pack()

emailofuserlabel=Label(emailofpersonframe,text='\nPublication ',justify=RIGHT)

emailofuserentry=Entry(emailofpersonframe,width=30)

emailofuserlabel.pack(side=LEFT)

emailofuserentry.pack(side=LEFT)

errorframe=Frame(root)

errorframe.pack()

buttonsframe=Frame(root)

buttonsframe.pack(side=BOTTOM)

submitbutton= Button(buttonsframe, text="Submit", command=submit)

submitbutton.pack()

backbutton = Button(buttonsframe, text="Back", command=back)

backbutton.pack()

root.mainloop()

def updateuser():

####################### 10th Oct 2020 #######################

global errorframe

def back():

global errorframe

global detailsofpersonframeforcreatingaccount

titleoffeedbackpage. pack\_forget()

errorframe.pack\_forget()

buttonsframe.pack\_forget()

try:

detailsofpersonframeforcreatingaccount.pack\_forget()

except:

pass

try:

otpevalfr.pack\_forget()

except:

pass

try:

updateuserframe.pack\_forget()

except:

pass

try:

deleteuserframe.pack\_forget()

except:

pass

librarianspage()

def updateuserdet():

global deleteuserframe

global otpevalfr

global detailsofpersonframeforcreatingaccount

global updateuserframe

global errorframe

global emailofpersonframe

try:

detailsofpersonframeforcreatingaccount.pack\_forget()

except:

pass

try:

otpevalfr.pack\_forget()

except:

pass

try:

updateuserframe.pack\_forget()

except:

pass

try:

deleteuserframe.pack\_forget()

except:

pass

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

######################## 11th Oct 2020#########################

def updatethatuser():

def goverifyotp():

def verifyotp():

global errorframe

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

otp = otpentry.get()

if len(otp)==0:

notfoundurlbl=Label(errorframe,text='\n\nPlease Enter OTP', fg='Red')

notfoundurlbl.pack()

return

elif str(val)!= otp:

notfoundurlbl=Label(errorframe,text='\n\nInvalid OTP', fg='Red')

notfoundurlbl.pack()

return

else:

cursor.execute("update users set First\_Name = '{}',Last\_Name = '{}', phone\_Number = '{}', Identity= '{}' , email\_id = '{}' where LIBID = {}".format(fname,lname,phone,nric,email,lid))

mycon.commit()

notfoundurlbl=Label(errorframe,text='\n\nAccount Has Been Updated', fg='Green')

notfoundurlbl.pack()

#################hkv####################

global emailofpersonframe

global detailsofpersonframeforcreatingaccount

global otpevalfr

global errorframe

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

fname=nameofuserentry.get()

lname=lnameofuserentry.get()

nric=nricofuserentry.get()

email=emailofuserentry.get()

phone=phoneofuserentry.get()

if len(fname)==0 or len(lname)==0 or len(phone)==0 or len(nric)==0 or len(email)==0:

enterallinfoerrormessagelabel=Label(errorframe,text='Please Enter all Necessary Details', fg='red')

enterallinfoerrormessagelabel.pack()

return

val=random.randint(1000,9999)

sub='OTP Verification'

msg='Dear '+fname+',\nYour OTP for Updating Your Libriary account is '+str(val)+'\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','scyjovpwhqzfbrur')

subject = sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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

detailsofpersonframeforcreatingaccount.pack\_forget()

otpevalfr = Frame(updateuserframe)

otpevalfr.pack()

labelforspace = Label( otpevalfr, text =" \n\n\nPlease Enter OTP Received by User").pack()

otpentry = Entry (otpevalfr,width=10)

otpentry.pack()

submitbtn=Button(otpevalfr,text='Submit',command=verifyotp)

submitbtn.pack()

global detailsofpersonframeforcreatingaccount

global emailofpersonframe

global errorframe

global emailofuserentry

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

lid=emailofuserentry.get()

if len(lid) == 0:

notfoundurlbl=Label(errorframe,text='\n\nPlease Enter Library ID of User', fg='Red')

notfoundurlbl.pack()

return

cursor.execute('select \* from users where LIBID= {}'.format(lid))

m=cursor.fetchall()

if len(m) == 0:

notfoundurlbl=Label(errorframe,text='\n\nInvalid Library ID', fg='Red')

notfoundurlbl.pack()

return

if int(lid) == 1 or int(lid) == 2:

notfoundurlbl=Label(errorframe,text='\n\nDefault Account Cannt be Updated', fg='Red')

notfoundurlbl.pack()

return

#NEW ENTRIES

emailofpersonframe.pack\_forget()

detailsofpersonframeforcreatingaccount=Frame(root)

detailsofpersonframeforcreatingaccount.pack()

instrlbl=Label(detailsofpersonframeforcreatingaccount,text='Update New User Details Here',font='verdana 15')

instrlbl.pack()

nameframe=Frame(detailsofpersonframeforcreatingaccount)

nameframe.pack()

nameofuserlabel=Label(nameframe,text=' \t First Name \t')

nameofuserentry=Entry(nameframe,width=20)

nameofuserlabel.pack(side=LEFT)

nameofuserentry.pack(side=LEFT)

lnameframe=Frame(detailsofpersonframeforcreatingaccount)

lnameframe.pack()

lnameofuserlabel=Label(lnameframe,text=' \t Last Name \t')

lnameofuserentry=Entry(lnameframe,width=20)

lnameofuserlabel.pack(side=LEFT)

lnameofuserentry.pack(side=LEFT)

#nric

nricofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

nricofpersonframe.pack()

nricofuserlabel=Label(nricofpersonframe,text='\nLast 4 characters NRIC/Passport\t')

nricofuserentry=Entry(nricofpersonframe,width=20)

nricofuserlabel.pack(side=LEFT)

nricofuserentry.pack(side=LEFT)

#phonenumber

phoneofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

phoneofpersonframe.pack()

phoneofuserlabel=Label(phoneofpersonframe,text='\n\tPhone Number\t')

phoneofuserentry=Entry(phoneofpersonframe,width=20)

phoneofuserlabel.pack(side=LEFT)

phoneofuserentry.pack(side=LEFT)

#email

emailofperson1frame=Frame(detailsofpersonframeforcreatingaccount)

emailofperson1frame.pack()

emailofuserlabel=Label(emailofperson1frame,text='\n\t\tEmail Address\t')

emailofuserentry=Entry(emailofperson1frame,width=30)

emailofuserlabel.pack(side=LEFT)

emailofuserentry.pack(side=LEFT)

submtbtn=Button(detailsofpersonframeforcreatingaccount,text='Submit',command=goverifyotp)

submtbtn.pack()

updateuserframe=Frame(root)

updateuserframe.pack()

global emailofpersonframe

emailofpersonframe=Frame(updateuserframe)

emailofpersonframe.pack()

instrlbl=Label(emailofpersonframe,text='Update User Details\nNOTE: Password can be only reseted by user through reset password in login page\n',font='verdana 20')

instrlbl.pack()

global emailofuserentry

emailofuserlabel=Label(emailofpersonframe,text='LIBRARY ID ')

emailofuserentry=Entry(emailofpersonframe,width=10)

emailofuserlabel.pack(side=LEFT)

emailofuserentry.pack(side=LEFT)

bufferlbl=Label(emailofpersonframe,text='\n\n')

bufferlbl.pack()

btnsubgo=Button(emailofpersonframe,text='Submit', command=updatethatuser)

btnsubgo.pack()

def deleteusr():

global deleteuserframe

global emailofpersonframe

global otpevalfr

global detailsofpersonframeforcreatingaccount

global updateuserframe

global errorframe

try:

detailsofpersonframeforcreatingaccount.pack\_forget()

except:

pass

try:

otpevalfr.pack\_forget()

except:

pass

try:

updateuserframe.pack\_forget()

except:

pass

try:

deleteuserframe.pack\_forget()

except:

pass

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

def deletethatuser():

global emailofpersonframe

def verifyotp():

global errorframe

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

otp = otpentry.get()

if len(otp)==0:

notfoundurlbl=Label(errorframe,text='\n\nPlease Enter OTP', fg='Red')

notfoundurlbl.pack()

elif str(val)!= otp:

notfoundurlbl=Label(errorframe,text='\n\nInvalid OTP', fg='Red')

notfoundurlbl.pack()

return

else:

cursor.execute("select \* from borrowedbooks where LIBID ={}".format(lid))

checklst=cursor.fetchall()

if len(checklst)>0:

notfoundurlbl=Label(errorframe,text='\n\nThe User has a Borrowed Book yet to be Returned\n Please Return All Books Borrowed by User and Try Again', fg='Red')

notfoundurlbl.pack()

return

cursor.execute("delete from users where LIBID ={}".format(lid))

mycon.commit()

cursor.execute("delete from reservations where LIBID ={}".format(lid))

mycon.commit()

cursor.execute("delete from messaging where Fromid ={} or toid = {}".format(lid,lid))

mycon.commit()

fhavpswd=open('./STORAGE/encrypteduserdetails.txt','rb')

lhavpswd=pickle.load(fhavpswd)

fhavpswd.close()

for a in range (len(lhavpswd)):

if str(lhavpswd[a][0]) == str(lid):

lhavpswd.pop(a)

break

fhavpswd=open('./STORAGE/encrypteduserdetails.txt','wb')

pickle.dump(lhavpswd,fhavpswd)

fhavpswd.close()

notfoundurlbl=Label(errorframe,text='\n\nAccount Has Been Deleted', fg='Green')

notfoundurlbl.pack()

global errorframe

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

lid=emailofuserentry.get()

if len(lid) == 0:

notfoundurlbl=Label(errorframe,text='\n\nPlease Enter Library ID of User', fg='Red')

notfoundurlbl.pack()

return

cursor.execute('select \* from users where LIBID= {}'.format(lid))

m=cursor.fetchall()

if len(m) == 0:

notfoundurlbl=Label(errorframe,text='\n\nInvalid Library ID', fg='Red')

notfoundurlbl.pack()

return

if int(lid) == 1 or int(lid) == 2:

notfoundurlbl=Label(errorframe,text='\n\nDefault Account Cannt be Deleted', fg='Red')

notfoundurlbl.pack()

return

val=random.randint(1000,9999)

sub='OTP Verification'

msg='Dear '+str(m[0][2])+',\nYour OTP for Deleting Your Libriary account is '+str(val)+'\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','scyjovpwhqzfbrur')

subject = sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

smtp.sendmail('aministrator.library@gmail.com',m[0][6],msg)

emailofpersonframe.pack\_forget()

otpevalfr = Frame(deleteuserframe)

otpevalfr.pack()

labelforspace = Label( otpevalfr, text =" \n\n\nPlease Enter OTP Received by User").pack()

otpentry = Entry (otpevalfr,width=10)

otpentry.pack()

submitbtn=Button(otpevalfr,text='Submit',command=verifyotp)

submitbtn.pack()

deleteuserframe=Frame(root)

deleteuserframe.pack()

emailofpersonframe=Frame(deleteuserframe)

emailofpersonframe.pack()

instrlbl=Label(emailofpersonframe,text='Delete User Account\n\n',font='verdana 20')

instrlbl.pack()

emailofuserlabel=Label(emailofpersonframe,text='LIBRARY ID ')

emailofuserentry=Entry(emailofpersonframe,width=10)

emailofuserlabel.pack(side=LEFT)

emailofuserentry.pack(side=LEFT)

bufferlbl=Label(emailofpersonframe,text='\n\n')

bufferlbl.pack()

btnsubgo=Button(emailofpersonframe,text='Submit', command=deletethatuser)

btnsubgo.pack()

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

titleoffeedbackpage=Frame(root)

titleoffeedbackpage.pack()

feedbacktitletext = Label(titleoffeedbackpage, text = 'UPDATE USER DETAILS / DELETE ACCOUNT', fg='blue',font= "applecherry 35")

feedbacktitletext.pack(side= TOP)

buttonsframe=Frame(root)

buttonsframe.pack(side=BOTTOM)

global errorframe

try:

errorframe.pack\_forget()

except:

pass

errorframe=Frame(root)

errorframe.pack()

addnewbannerbutton= Button(buttonsframe, text="Update User Details", command=updateuserdet)

addnewbannerbutton.pack()

deletebannerbutton= Button(buttonsframe, text="Delete User Account", command=deleteusr)

deletebannerbutton.pack()

backbutton = Button(buttonsframe, text="Back", command=back)

backbutton.pack()

root.mainloop()

def updateevents():

####################### 10th Oct 2020 #######################

global errorframe

global updateeventpagefrm

def back():

global deletebannerframe

global errorframe

global addbannerframe

titleoffeedbackpage. pack\_forget()

updateeventpagefrm.pack\_forget()

errorframe.pack\_forget()

buttonsframe.pack\_forget()

try:

deletebannerframe.pack\_forget()

except:

pass

try:

addbannerframe.pack\_forget()

except:

pass

librarianspage()

def addnewbanner():

global deletebannerframe

global addbannerframe

global errorframe

global updateeventpagefrm

try:

deletebannerframe.pack\_forget()

errorframe.pack\_forget()

except:

pass

try:

addbannerframe.pack\_forget()

except:

pass

def submsg():

global errorframe

global overduelbl

global addbannerframe

global updateeventpagefrm

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

filehvingbannerimg=open('./STORAGE/bannerssource.txt','rb')

lsl=pickle.load(filehvingbannerimg)

filehvingbannerimg.close()

bannerdesc=TextArea.get("1.0",END)

filenamegn=nameofuserentry.get()

frmtfilenamegn='./IMAGES/banners/'+filenamegn+'.png'

###checks

if len(filenamegn)==0 or len(bannerdesc)==0:

notfoundurlbl=Label(errorframe,text='\n\nPlease Enter All Details ', fg='Red')

notfoundurlbl.pack()

return

if len(bannerdesc)<=1:

notfoundurlbl=Label(errorframe,text='\n\nPlease Type Banner Description ', fg='Red')

notfoundurlbl.pack()

return

if len(bannerdesc)>=300:

notfoundurlbl=Label(errorframe,text='\n\nMaximum Characters Exceeded for Banner Description', fg='Red')

notfoundurlbl.pack()

return

try:

open(frmtfilenamegn)

except:

notfoundurlbl=Label(errorframe,text='\n\nFile Name is Incorrect', fg='Red')

notfoundurlbl.pack()

return

photobanner=PhotoImage(frmtfilenamegn)

if photobanner.height() > 397:

notfoundurlbl=Label(errorframe,text='\nMaximum Height Exceeded', fg='Red')

notfoundurlbl.pack()

return

if photobanner.width() > 1134:

notfoundurlbl=Label(errorframe,text='\nMaximum Width Exceeded', fg='Red')

notfoundurlbl.pack()

return

lsl.append((frmtfilenamegn,bannerdesc))

filehvingbannerimg=open('./STORAGE/bannerssource.txt','wb')

pickle.dump(lsl,filehvingbannerimg)

filehvingbannerimg.close()

plscheckentry=Label(errorframe,text='Banner Has been Added', fg='Green')

plscheckentry.pack()

addbannerframe.pack\_forget()

updateeventpagefrm=Frame(root)

updateeventpagefrm.pack()

odbks=''

for a in range(len(lsl)):

odbks+='\n\n\nBanner Number: '+ str(a+1) + '\n\nFile: '+ str(lsl[a][0]) + '\n\nBanner Description:\t\t\t\n\n '+ str(lsl[a][1])+'\n'

odbks+='---------------------------------------------------------------------------------------------------------------------------------------------------------\n'

canvas = Canvas(updateeventpagefrm, width=1000, height=400)

scroll\_y = Scrollbar(updateeventpagefrm, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack()

# 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()

updateeventpagefrm.pack\_forget()

addbannerframe=Frame(root)

addbannerframe.pack()

instrtxt='Please follow these instructions to add new banner\n\n\

--> The banner should be only a .png image \n\

--> Save the Image with an appropriate name\n\

--> The Size of Banner should be maximum \n\

\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\

--> Save the Image in LIBRARY >> IMAGES >> banners\n\

--> Now Enter Just the name of the image without directory names and extension\n\

--> In the next text box, give a small description about the banner'

instlbl=Label(addbannerframe,text=instrtxt,justify=LEFT)

instlbl.pack()

nameframe=Frame(addbannerframe)

nameframe.pack()

nameofuserlabel=Label(nameframe,text='File Name ')

nameofuserentry=Entry(nameframe,width=30)

nameofuserlabel.pack(side=LEFT)

nameofuserentry.pack(side=LEFT)

usernameofpersonframe=Frame(addbannerframe)

usernameofpersonframe.pack()

usernameofuserlabel=Label(usernameofpersonframe,text='\n\tBanner Description \t')

TextArea = Text(usernameofpersonframe,selectborderwidth=3)

usernameofuserlabel.pack(side=LEFT)

TextArea.pack(expand=YES, fill=BOTH)

compusername=Label(usernameofpersonframe,text='Max 300 Characters\n\n\n')

compusername.pack(side=LEFT)

subbtn=Button(usernameofpersonframe, text='Submit', command=submsg)

subbtn.pack()

root.mainloop()

def deletebanner():

global updateeventpagefrm

def deletethatnow():

global errorframe

global overduelbl

global updateeventpagefrm

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

bannerno=emailofuserentry.get()

filehvingbannerimg=open('./STORAGE/bannerssource.txt','rb')

lsl=pickle.load(filehvingbannerimg)

filehvingbannerimg.close()

if len(bannerno)==0:

plscheckentry=Label(errorframe,text='PLease Enter Banner Number', fg='red')

plscheckentry.pack()

return

try:

int(bannerno)

except:

plscheckentry=Label(errorframe,text='Invalid Entry for Banner Number', fg='red')

plscheckentry.pack()

return

if 1<= int(bannerno) <= len(lsl)+1:

lsl.pop((int(bannerno)-1))

filehvingbannerimg=open('./STORAGE/bannerssource.txt','wb')

pickle.dump(lsl,filehvingbannerimg)

filehvingbannerimg.close()

plscheckentry=Label(errorframe,text='Banner Has been Deleted', fg='Green')

plscheckentry.pack()

deletebannerframe.pack\_forget()

updateeventpagefrm=Frame(root)

updateeventpagefrm.pack()

odbks=''

for a in range(len(lsl)):

odbks+='\n\n\nBanner Number: '+ str(a+1) + '\n\nFile: '+ str(lsl[a][0]) + '\n\nBanner Description:\t\t\t\n\n '+ str(lsl[a][1])+'\n'

odbks+='---------------------------------------------------------------------------------------------------------------------------------------------------------\n'

canvas = Canvas(updateeventpagefrm, width=1000, height=400)

scroll\_y = Scrollbar(updateeventpagefrm, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack()

# 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')

else:

plscheckentry=Label(errorframe,text='Invalid Banner Number', fg='red')

plscheckentry.pack()

return

global deletebannerframe

global addbannerframe

global errorframe

try:

errorframe.pack\_forget()

errorframe=Frame(root)

errorframe.pack()

except:

pass

try:

deletebannerframe.pack\_forget()

except:

pass

try:

addbannerframe.pack\_forget()

except:

pass

updateeventpagefrm.pack\_forget()

deletebannerframe=Frame(root)

deletebannerframe.pack()

filehvingbannerimg=open('./STORAGE/bannerssource.txt','rb')

lsl=pickle.load(filehvingbannerimg)

filehvingbannerimg.close()

if len(lsl)==1:

plscheckentry=Label(errorframe,text='Minimum Number of Banner Reached', fg='red')

plscheckentry.pack()

return

emailofpersonframe=Frame(deletebannerframe)

emailofpersonframe.pack()

instructionlabel=Label(emailofpersonframe,text='Enter Details of Banner to Delete')

instructionlabel.pack()

emailofuserlabel=Label(emailofpersonframe,text='\nBanner Number ')

emailofuserentry=Entry(emailofpersonframe,width=30)

emailofuserlabel.pack(side=LEFT)

emailofuserentry.pack(side=LEFT)

bufferlbl=Label(emailofpersonframe,text='\n\n')

bufferlbl.pack()

btnsubgo=Button(emailofpersonframe,text='Submit', command=deletethatnow)

btnsubgo.pack()

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

titleoffeedbackpage=Frame(root)

titleoffeedbackpage.pack()

feedbacktitletext = Label(titleoffeedbackpage, text = 'UPDATE BANNER', fg='blue',font= "applecherry 35")

feedbacktitletext.pack(side= TOP)

updateeventpagefrm=Frame(root)

updateeventpagefrm.pack()

filehvingbannerimg=open('./STORAGE/bannerssource.txt','rb')

lsl=pickle.load(filehvingbannerimg)

filehvingbannerimg.close()

odbks=''

global overduelbl

global canvas

global scroll\_y

if len(lsl)>0:

for a in range(len(lsl)):

odbks+='\n\n\nBanner Number: '+ str(a+1) + '\n\nFile: '+ str(lsl[a][0]) + '\n\nBanner Description:\t\t\t\n\n '+ str(lsl[a][1])+'\n'

odbks+='---------------------------------------------------------------------------------------------------------------------------------------------------------\n'

canvas = Canvas(updateeventpagefrm, width=1000, height=400)

scroll\_y = Scrollbar(updateeventpagefrm, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack()

# 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')

buttonsframe=Frame(root)

buttonsframe.pack(side=BOTTOM)

errorframe=Frame(root)

errorframe.pack()

addnewbannerbutton= Button(buttonsframe, text="Add New Banner", command=addnewbanner)

addnewbannerbutton.pack()

deletebannerbutton= Button(buttonsframe, text="Delete Banner", command=deletebanner)

deletebannerbutton.pack()

backbutton = Button(buttonsframe, text="Back", command=back)

backbutton.pack()

root.mainloop()

def logout():

ans=tkinter.messagebox.askquestion('LIBRARY','Are you sure you want to log out?')

if ans == 'yes':

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

mainpage()

def searchbook():

welcomeframe.pack\_forget()

librarianframetop.pack\_forget()

librarianframetop.pack\_forget()

librarianframebottom.pack\_forget()

searchbookfn()

welcomeframe=Frame(root) # A FRAME HERE

welcomeframe.pack(side=TOP)

x=time.localtime()

if 0<=x[3]<=6 or 17<=x[3]<=23:

nom='Good Evening '

elif 7<=x[3]<=11:

nom='Good Morning '

else:

nom='Good Afternoon '

wlcmmsg=(nom+'LIBRARIAN')

welcomeusermsg=Label(welcomeframe, text=wlcmmsg, fg= 'Blue', font=' verdana 25', anchor=W)

welcomeusermsg.pack(side= LEFT)

spacelabel1=Label(welcomeframe,text='\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\n\n\n\n\n\n\n\n')

spacelabel1.pack(side=LEFT)

'''spaceframe1=Frame(framedetails)

spaceframe1.pack(side= LEFT)

spacelabel1=Label(spaceframe1,text=' ')

spacelabel1.pack(side=LEFT)'''

librarianframetop=Frame(root)

librarianframebottom=Frame(root)

librarianframetop.pack()

librarianframebottom.pack()

borrowbk=Frame(librarianframetop)

borrowbk.pack(side=LEFT)

borrowbkpic=PhotoImage(file='./IMAGES/Icons/borrow book.png')

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)

spacelabel1=Label(spaceframe1,text=' ')

spacelabel1.pack()

returnbk=Frame(librarianframetop)

returnbk.pack(side=LEFT)

returnbkpic=PhotoImage(file='./IMAGES/Icons/return book.png')

returnbkbtn=Button(returnbk,text='Open', image = returnbkpic,command= returnbook) # return BOOK

returnbkbtn.pack()

returnbklbl=Label(returnbk,text='Return Book',font= "timesnewroman 20")

returnbklbl.pack()

spaceframe2=Frame(librarianframetop)

spaceframe2.pack(side= LEFT)

spacelabel1=Label(spaceframe2,text=' ')

spacelabel1.pack()

reservebk=Frame(librarianframetop)

reservebk.pack(side=LEFT)

reservebkpic=PhotoImage(file='./IMAGES/Icons/reservation.png')

reservebkbtn=Button(reservebk,text='Open', image = reservebkpic,command= reservebook) # reserve BOOK

reservebkbtn.pack()

reservebklbl=Label(reservebk,text='Reserve Book',font= "timesnewroman 20")

reservebklbl.pack()

spaceframe3=Frame(librarianframetop)

spaceframe3.pack(side= LEFT)

spacelabel1=Label(spaceframe3,text=' ')

spacelabel1.pack()

renewbk=Frame(librarianframetop)

renewbk.pack(side=LEFT)

renewbkpic=PhotoImage(file='./IMAGES/Icons/reservation.png')

renewbkbtn=Button(renewbk,text='Open', image = renewbkpic,command= renewbook) # reserve BOOK

renewbkbtn.pack()

renewbklbl=Label(renewbk,text='Renew Book',font= "timesnewroman 20")

renewbklbl.pack()

spaceframe4=Frame(librarianframetop)

spaceframe4.pack(side= LEFT)

spacelabel1=Label(spaceframe4,text=' ')

spacelabel1.pack()

searchbk=Frame(librarianframetop)

searchbk.pack(side=LEFT)

searchbkpic=PhotoImage(file='./IMAGES/Icons/searchbookimage.png')

global currentlocationforserachbook

currentlocationforserachbook='LIBRARIAN'

searchbkbtn=Button(searchbk,text='Open', image = searchbkpic,command= searchbook) # search BOOK #CHANGE

searchbkbtn.pack()

searchbklbl=Label(searchbk,text='Search Book',font= "timesnewroman 20")

searchbklbl.pack()

spaceframe5=Frame(librarianframetop)

spaceframe5.pack(side= LEFT)

spacelabel1=Label(spaceframe5,text=' ')

spacelabel1.pack()

overduebk=Frame(librarianframetop)

overduebk.pack(side=LEFT)

overduebkpic=PhotoImage(file='./IMAGES/Icons/overdue books.png')

overduebkbtn=Button(overduebk,text='Open', image = overduebkpic,command= overduebook) # overdue BOOK #CHANGE

overduebkbtn.pack()

overduebklbl=Label(overduebk,text='Overdue Books',font= "timesnewroman 20")

overduebklbl.pack()

spaceframe6=Frame(librarianframetop)

spaceframe6.pack(side= LEFT)

spacelabel1=Label(spaceframe6,text=' ')

spacelabel1.pack()

############ ADD NEW HERE##############

################# 4th October 2020 ##################

#Space Frame 9

addnewfrm=Frame(librarianframebottom)

addnewfrm.pack(side=LEFT)

addnewpic=PhotoImage(file='./IMAGES/Icons/newbook.png')

addnewbtn=Button(addnewfrm,text='Open', image = addnewpic,command= addnewbook)

addnewbtn.pack()

newbklbl=Label(addnewfrm,text='Add New Books',font= "timesnewroman 20")

newbklbl.pack()

spaceframe9=Frame(librarianframebottom)

spaceframe9.pack(side= LEFT)

spacelabel1=Label(spaceframe9,text=' ')

spacelabel1.pack()

updatevenfrm=Frame(librarianframebottom)

updatevenfrm.pack(side=LEFT)

updateventpic=PhotoImage(file='./IMAGES/Icons/eventupdate.png')

updateeventbtn=Button(updatevenfrm,text='Open', image = updateventpic,command= updateevents)

updateeventbtn.pack()

updatelbl=Label(updatevenfrm,text='Update Banner',font= "timesnewroman 20")

updatelbl.pack()

spaceframe10=Frame(librarianframebottom)

spaceframe10.pack(side= LEFT)

spacelabel1=Label(spaceframe10,text=' ')

spacelabel1.pack()

updtuserfrm=Frame(librarianframebottom)

updtuserfrm.pack(side=LEFT)

usrupdateventpic=PhotoImage(file='./IMAGES/Icons/updateuser.png')

usrupdateeventbtn=Button(updtuserfrm,text='Open', image = usrupdateventpic,command= updateuser)

usrupdateeventbtn.pack()

updateusrlbl=Label(updtuserfrm,text='Update User Account',font= "timesnewroman 20")

updateusrlbl.pack()

spaceframe11=Frame(librarianframebottom)

spaceframe11.pack(side= LEFT)

spacelabel1=Label(spaceframe11,text=' ')

spacelabel1.pack()

######################################

notificationsfrm=Frame(librarianframebottom)

notificationsfrm.pack(side=LEFT)

notipic=PhotoImage(file='./IMAGES/Icons/inbox.png')

notibtn=Button(notificationsfrm,text='Open', image = notipic,command= messaging) # overdue BOOK #CHANGE

notibtn.pack()

notilbl=Label(notificationsfrm,text='Messaging',font= "timesnewroman 20")

notilbl.pack()

spaceframe7=Frame(librarianframebottom)

spaceframe7.pack(side= LEFT)

spacelabel1=Label(spaceframe7,text=' ')

spacelabel1.pack()

logoutfrm=Frame(librarianframebottom)

logoutfrm.pack(side=LEFT)

logoutpic=PhotoImage(file='./IMAGES/Icons/logout.png')

logoutbtn=Button(logoutfrm,text='Open', image = logoutpic,command= logout) # overdue BOOK #CHANGE

logoutbtn.pack()

logoutlbl=Label(logoutfrm,text='Log Out',font= "timesnewroman 20")

logoutlbl.pack()

spaceframe6=Frame(librarianframebottom)

spaceframe6.pack(side= LEFT)

spacelabel1=Label(spaceframe6,text=' ')

spacelabel1.pack()

root.mainloop()

########################### 21st september 2020 ################################

def userpage(wlcmmsg,uname,username,libid):

def latestarrivalspg():

welcomeframe.pack\_forget()

userframetop.pack\_forget()

userframebottom.pack\_forget()

def back():

latestarrivalsframetop.pack\_forget()

backbtn.pack\_forget()

latestarrivalstxt.pack\_forget()

userpage(currentlocationforserachbookwlcmmsgusefulforuseraccount,

currentlocationforserachbook,

currentlocationforserachbookwlcmmsgusefulforuseraccountusername,

currentlocationforserachbookuserlibraryid)

global latestarrivalsframetop

latestarrivalstxt = Label(root, text = 'Latest arrivals \n\n', fg='magenta',font= "applecherry 35")

latestarrivalstxt.pack(side=TOP)

latestarrivalsframetop = Frame(root)

latestarrivalsframetop.pack()

def openbook1():

webbrowser.open("https://manybooks.net/book/126571/read#epubcfi(/6/2[titlepage]!/4/1:0)")

book1=Frame(latestarrivalsframetop)

book1.pack(side = LEFT)

latestarrival1pic=PhotoImage(file='./IMAGES/latestarrivals/latestarrival1.png')

openbutton1=Button(book1,text='Open',image = latestarrival1pic,command=openbook1)

openbutton1.pack()

latestarrivallabel1=Label(book1,text='He fell in love with his wife',font= "timesnewroman 20")

latestarrivallabel1.pack()

spaceframe1=Frame(latestarrivalsframetop)

spaceframe1.pack(side= LEFT)

spacelabel1=Label(spaceframe1,text=' ')

spacelabel1.pack(side= LEFT)

def openbook2():

webbrowser.open("https://manybooks.net/book/123104/read#epubcfi(/6/2[titlepage]!/4/1:0)")

book2= Frame(latestarrivalsframetop)

book2.pack(side = LEFT)

lasestarrival2pic=PhotoImage(file='./IMAGES/latestarrivals/latestarrival2.png')

openbutton2=Button(book2,text='Open',image=lasestarrival2pic,command=openbook2)

openbutton2.pack()

latestarrivallabel2=Label(book2,text='the return of sherlock holmes',font='timesnewroman 20')

latestarrivallabel2.pack()

spaceframe2=Frame(latestarrivalsframetop)

spaceframe2.pack(side= LEFT)

spacelabel2=Label(spaceframe2,text=' ')

spacelabel2.pack(side= LEFT)

def openbook3():

webbrowser.open("https://manybooks.net/book/127828/read#epubcfi(/6/2[item3]!/4/2/1:0)")

book3= Frame(latestarrivalsframetop)

book3.pack(side = LEFT)

latestarrival3pic=PhotoImage(file='./IMAGES/latestarrivals/latestarrival3.png')

openbutton3=Button(book3,text='Open',image = latestarrival3pic,command=openbook3)

openbutton3.pack()

latestarrivallabel3=Label(book3,text='The invisible man',font= "timesnewroman 20")

latestarrivallabel3.pack()

spaceframe3=Frame(latestarrivalsframetop)

spaceframe3.pack(side= LEFT)

spacelabel3=Label(spaceframe3,text=' ')

spacelabel3.pack()

def openbook4():

webbrowser.open("https://manybooks.net/book/127823/read#epubcfi(/6/2[coverpage-wrapper]!/4/1:0)")

book4= Frame(latestarrivalsframetop)

book4.pack(side = LEFT)

latestarrival4pic=PhotoImage(file='./IMAGES/latestarrivals/latestarrival4.png')

openbutton4=Button(book4,text='Open',image = latestarrival4pic,command=openbook4)

openbutton4.pack()

latestarrivallabel4=Label(book4,text='The first men on the moon',font= "timesnewroman 20")

latestarrivallabel4.pack()

spaceframe4=Frame(latestarrivalsframetop)

spaceframe4.pack(side= LEFT)

spacelabel4=Label(spaceframe4,text=' ')

spacelabel4.pack(side = LEFT)

backbtn=Button(root, text='Back',command=back)

backbtn.pack()

root.mainloop()

def messaging():

welcomeframe.pack\_forget()

userframetop.pack\_forget()

userframebottom.pack\_forget()

def back():

global inboxframe

global sendframe

lbldetails.pack\_forget()

lbl.pack\_forget()

backbtn.pack\_forget()

try:

inboxframe.pack\_forget()

except:

pass

try:

manucomframe.pack\_forget()

except:

pass

try:

sendframe.pack\_forget()

except:

pass

userpage(currentlocationforserachbookwlcmmsgusefulforuseraccount,

currentlocationforserachbook,

currentlocationforserachbookwlcmmsgusefulforuseraccountusername,

currentlocationforserachbookuserlibraryid)

def inbox():

global overduelbl

global canvas

global scroll\_y

def clrmsgfn():

global manucomframe

global overduelbl

global overduelbl12

manucomframe.pack\_forget()

scroll\_y.pack\_forget()

cursor.execute('delete from messaging where toid = {}'.format(currentlocationforserachbookuserlibraryid))

mycon.commit()

overduelbl.pack\_forget()

canvas.pack\_forget()

try:

overduelbl12.pack\_forget()

except:

pass

overduelbl=Label(inboxframe, text= 'All Messages are Cleared', fg='green')

overduelbl.pack()

manucomframe.pack\_forget()

def clr10msgfn():

global canvas

global overduelbl

global manucomframe

global scroll\_y

global overduelbl12

cursor.execute('delete from messaging where toid = {}'.format(currentlocationforserachbookuserlibraryid))

mycon.commit()

manucomframe.pack\_forget()

canvas.pack\_forget()

scroll\_y.pack\_forget()

overduelbl.pack\_forget()

for a in range(5):

try:

lsl.pop(-1)

except:

break

if len(lsl)>0:

for a in lsl:

cursor.execute('insert into messaging values {}'.format(a))

mycon.commit()

overduelbl.pack\_forget()

try:

overduelbl12.pack\_forget()

except:

pass

overduelbl12=Label(inboxframe, text= 'Last 5 messages are Cleared\n', fg='green')

overduelbl12.pack()

odbks=''

if len(lsl)>0:

for a in range(len(lsl)):

odbks+='\n\n\nFROM: '+ str(lsl[a][1]) + '\nMESSAGE:\t\t\t\n\n'+ str(lsl[a][3])+'\n'

odbks+='---------------------------------------------------------------------------------------------------------------------------------------------------------\n'

canvas = Canvas(inboxframe, width=1000, height=400)

scroll\_y = Scrollbar(inboxframe, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack(side=LEFT)

# 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')

manucomframe=Frame(root)

manucomframe.pack(side=BOTTOM)

clrmsg=Button(manucomframe, text='Clear All Messages', command=clrmsgfn)

clr10msg=Button(manucomframe, text='Clear Last 5 Messages', command=clr10msgfn)

clrmsg.pack()

clr10msg.pack()

else:

canvas.pack\_forget()

overduelbl=Label(inboxframe, text= 'There are no Messages')

overduelbl.pack()

global inboxframe

global sendframe

global manucomframe

try:

inboxframe.pack\_forget()

manucomframe.pack\_forget()

except:

pass

try:

sendframe.pack\_forget()

except:

pass

inboxframe=Frame(root)

inboxframe.pack(side=TOP)

lbl=Label(inboxframe,text='INBOX \n', fg='blue',font= "applecherry 25") #THIS HAS TO BE DETELED WHILE NAVIGATING

lbl.pack(side= TOP)

odbks=''

cursor.execute('select \* from messaging where toid = {}'.format(currentlocationforserachbookuserlibraryid))

lsl=cursor.fetchall()

lsl=lsl[::-1]

if len(lsl)>0:

for a in range(len(lsl)):

odbks+='\n\n\nFROM: '+ str(lsl[a][1]) +'\nMESSAGE:\t\t\t\n\n'+ str(lsl[a][3])+'\n'

odbks+='---------------------------------------------------------------------------------------------------------------------------------------------------------\n'

canvas = Canvas(inboxframe, width=1000, height=400)

scroll\_y = Scrollbar(inboxframe, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack(side=LEFT)

# 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')

manucomframe=Frame(root)

manucomframe.pack(side=BOTTOM)

clrmsg=Button(manucomframe, text='Clear All Messages', command=clrmsgfn)

clr10msg=Button(manucomframe, text='Clear Last 5 Messages', command=clr10msgfn)

clrmsg.pack()

clr10msg.pack()

else:

overduelbl=Label(inboxframe, text= 'There are no Messages')

overduelbl.pack()

def send():

global errorframe

#################### 21st september 2020 ######################

def submsg():

global errorframe

errorframe.pack\_forget()

errorframe=Frame(sendframe)

errorframe.pack()

recid=var.get()

messagesent=TextArea.get("1.0",END)

if recid==0:

notfoundurlbl=Label(errorframe,text='\n\nPlease Select Recipient ', fg='Red')

notfoundurlbl.pack()

return

if len(messagesent)<=1:

notfoundurlbl=Label(errorframe,text='\n\nPlease Type your Message ', fg='Red')

notfoundurlbl.pack()

return

if len(messagesent)>700:

notfoundurlbl=Label(errorframe,text='\n\nMax Characters Exceeded', fg='Red')

notfoundurlbl.pack()

return

detailsofpersonframeforcreatingaccount.pack\_forget()

notfoundurlbl=Label(errorframe,text='\n\nYour Message has been sent', fg='Green')

notfoundurlbl.pack()

cursor.execute("insert into messaging values ({},'{}',{},'{}')".format(currentlocationforserachbookuserlibraryid,

currentlocationforserachbookwlcmmsgusefulforuseraccountusername,

recid,

messagesent))

mycon.commit()

root.mainloop()

global inboxframe

global sendframe

try:

inboxframe.pack\_forget()

manucomframe.pack\_forget()

except:

pass

try:

sendframe.pack\_forget()

except:

pass

sendframe=Frame(root)

sendframe.pack()

lbl=Label(sendframe,text='\nCOMPOSE MAIL \n', fg='blue',font= "applecherry 25") #THIS HAS TO BE DETELED WHILE NAVIGATING

lbl.pack(side= TOP)

detailsofpersonframeforcreatingaccount=Frame(sendframe)

detailsofpersonframeforcreatingaccount.pack()

var = IntVar()

nricofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

nricofpersonframe.pack()

nricofuserlabel=Label(nricofpersonframe,text='\nTo \t')

nricofuserlabel.pack()

radiobtn2star = Radiobutton(nricofpersonframe)

radiobtn2star.config(text='Librarian', variable=var, value=2)

radiobtn2star.pack(side=LEFT)

#message

usernameofpersonframe=Frame(detailsofpersonframeforcreatingaccount)

usernameofpersonframe.pack()

usernameofuserlabel=Label(usernameofpersonframe,text='\n\tYour Message \t')

TextArea = Text(usernameofpersonframe,selectborderwidth=3)

usernameofuserlabel.pack(side=LEFT)

TextArea.pack(expand=YES, fill=BOTH)

compusername=Label(usernameofpersonframe,text='Max 700 Characters\n\n\n')

compusername.pack(side=LEFT)

subbtn=Button(detailsofpersonframeforcreatingaccount, text='Send', command=submsg)

subbtn.pack()

errorframe=Frame(sendframe)

errorframe.pack()

root.mainloop()

lbl=Label(root,text='Messaging\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING

lbl.pack(side= TOP)

lbldetails=Frame(root)

lbldetails.pack()

btninbox=Button(lbldetails, text='Inbox', command=inbox)

btnsend=Button(lbldetails, text='Compose Mail', command=send)

btninbox.pack(side=BOTTOM)

btnsend.pack(side=BOTTOM)

backbtn=Button(root, text='Back', command = back)

backbtn.pack(side=BOTTOM)

root.mainloop()

def borrowbook():

welcomeframe.pack\_forget()

userframetop.pack\_forget()

userframebottom.pack\_forget()

def back():

lblb.pack\_forget()

lbldetails.pack\_forget()

backbtn.pack\_forget()

userpage(currentlocationforserachbookwlcmmsgusefulforuseraccount,

currentlocationforserachbook,

currentlocationforserachbookwlcmmsgusefulforuseraccountusername,

currentlocationforserachbookuserlibraryid)

cursor.execute('select \* from borrowedbooks where LIBID = {}'.format(libid))

ls=cursor.fetchall()

vblbks='Number of Books Borrowed: '+str(len(ls))+'\n\n'

for a in range(len(ls)):

cursor.execute('select BOOK\_Title from books where BOOKID = {}'.format(ls[a][1]))

bknm=cursor.fetchone()

vblbks+='Book Id: '+ str(ls[a][1])+'\nBook Title: '+ str(bknm[0])+ '\nDue Date: '+ str(ls[a][2])+'\n'

vblbks+='-------------------------------------------------------------------------------------------------------------------------------\n\n'

lblb=Label(root,text='Borrowed Books \n\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING

lblb.pack(side= TOP)

lbldetails=Frame(root)

lbldetails.pack()

canvas = Canvas(lbldetails, width=650, height=400)

scroll\_y = Scrollbar(lbldetails, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= vblbks, justify=LEFT)

overduelbl.pack()

# 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')

backbtn=Button(root, text='Back', command = back)

backbtn.pack(side=BOTTOM)

root.mainloop()

def reservebook():

welcomeframe.pack\_forget()

userframetop.pack\_forget()

userframebottom.pack\_forget()

def back():

lbl.pack\_forget()

lbldetails.pack\_forget()

backbtn.pack\_forget()

smallinfo.pack\_forget()

userpage(currentlocationforserachbookwlcmmsgusefulforuseraccount,

currentlocationforserachbook,

currentlocationforserachbookwlcmmsgusefulforuseraccountusername,

currentlocationforserachbookuserlibraryid)

odbks=''

cursor.execute('select \* from reservations where LIBID = {}'.format(libid))

lsl=cursor.fetchall()

if len(lsl)>0:

for a in range(len(lsl)):

bookidis=lsl[a][1]

cursor.execute('select BOOK\_Title from books where BOOKID = {}'.format(bookidis))

msn=cursor.fetchone()

odbks+='\t Book Id: '+ str(lsl[a][1])+'\t Book Title: '+ str(msn[0])+'\n'

odbks+='---------------------------------------------------------------------------------------------------------------------------------------------------------\n'

lbl=Label(root,text='Reserved Books \n\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING

lbl.pack(side= TOP)

lbldetails=Frame(root)

lbldetails.pack()

canvas = Canvas(lbldetails, width=650, height=400)

scroll\_y = Scrollbar(lbldetails, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack()

# 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')

smallinfo=Label(root,text='To Make Reservations, Please Contact Librarian')

smallinfo.pack()

backbtn=Button(root, text='Back', command = back)

backbtn.pack(side=BOTTOM)

root.mainloop()

def overduebook():

welcomeframe.pack\_forget()

userframetop.pack\_forget()

userframebottom.pack\_forget()

def back():

lbl.pack\_forget()

lbldetails.pack\_forget()

backbtn.pack\_forget()

userpage(currentlocationforserachbookwlcmmsgusefulforuseraccount,

currentlocationforserachbook,

currentlocationforserachbookwlcmmsgusefulforuseraccountusername,

currentlocationforserachbookuserlibraryid)

odbks=''

cursor.execute('select \* from borrowedbooks where duedate < curdate() and LIBID = {}'.format(libid))

lsl=cursor.fetchall()

if len(lsl)>0:

for a in range(len(lsl)):

bookidis=lsl[a][1]

cursor.execute('select BOOK\_Title from books where BOOKID = {}'.format(bookidis))

msn=cursor.fetchone()

odbks+='\t Book Id: '+ str(lsl[a][1])+'\t Book Title: '+ str(msn[0])+'\n'

odbks+='---------------------------------------------------------------------------------------------------------------------------------------------------------\n'

lbl=Label(root,text='Overdue Books \n\n', fg='blue',font= "applecherry 35") #THIS HAS TO BE DETELED WHILE NAVIGATING

lbl.pack(side= TOP)

lbldetails=Frame(root)

lbldetails.pack()

canvas = Canvas(lbldetails, width=650, height=400)

scroll\_y = Scrollbar(lbldetails, orient="vertical", command=canvas.yview)

frameforscroll = Frame(canvas)

canvas.create\_window(0, 0, anchor='nw', window=frameforscroll)

overduelbl=Label(frameforscroll, text= odbks, justify=LEFT)

overduelbl.pack(side=LEFT)

# 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')

backbtn=Button(root, text='Back', command = back)

backbtn.pack(side=BOTTOM)

root.mainloop()

def logout():

ans=tkinter.messagebox.askquestion('LIBRARY','Are you sure you want to log out?')

if ans == 'yes':

welcomeframe.pack\_forget()

userframetop.pack\_forget()

userframebottom.pack\_forget()

mainpage()

def searchbook():

welcomeframe.pack\_forget()

userframetop.pack\_forget()

userframebottom.pack\_forget()

searchbookfn()

welcomeframe=Frame(root) # A FRAME HERE

welcomeframe.pack(side=TOP)

welcomeusermsg=Label(welcomeframe, text=wlcmmsg, fg= 'Blue', font=' verdana 25', anchor=W)

welcomeusermsg.pack(side= LEFT)

spacelabel1=Label(welcomeframe,text='\t\t\t\t\t\t\t\t\t\t\t\t\t\t\n\n\n\n\n\n\n\n\n\n\n')

spacelabel1.pack(side=LEFT)

userframetop=Frame(root)

userframebottom=Frame(root)

userframetop.pack()

userframebottom.pack()

borrowbk=Frame(userframetop)

borrowbk.pack(side=LEFT)

borrowbkpic=PhotoImage(file='./IMAGES/Icons/borrow book.png')

borrowbkbtn=Button(borrowbk,text='Open', image = borrowbkpic,command= borrowbook) # BORROW BOOK

borrowbkbtn.pack()

borrowbklb=Label(borrowbk,text='View Borrowed Books',font= "timesnewroman 20")

borrowbklb.pack()

spaceframe1=Frame(userframetop)

spaceframe1.pack(side= LEFT)

spacelabel1=Label(spaceframe1,text=' ')

spacelabel1.pack()

overduebk=Frame(userframetop)

overduebk.pack(side=LEFT)

overduebkpic=PhotoImage(file='./IMAGES/Icons/overdue books.png')

overduebkbtn=Button(overduebk,text='Open', image = overduebkpic,command= overduebook) # overdue BOOK #CHANGE

overduebkbtn.pack()

overduebklbl=Label(overduebk,text='Overdue Books',font= "timesnewroman 20")

overduebklbl.pack()

spaceframe6=Frame(userframetop)

spaceframe6.pack(side= LEFT)

spacelabel1=Label(spaceframe6,text=' ')

spacelabel1.pack()

reservebk=Frame(userframetop)

reservebk.pack(side=LEFT)

reservebkpic=PhotoImage(file='./IMAGES/Icons/reservation.png')

reservebkbtn=Button(reservebk,text='Open', image = reservebkpic,command= reservebook) # reserve BOOK

reservebkbtn.pack()

reservebklbl=Label(reservebk,text='Reserved Book',font= "timesnewroman 20")

reservebklbl.pack()

spaceframe3=Frame(userframetop)

spaceframe3.pack(side= LEFT)

spacelabel1=Label(spaceframe3,text=' ')

spacelabel1.pack()

searchbk=Frame(userframetop)

searchbk.pack(side=LEFT)

searchbkpic=PhotoImage(file='./IMAGES/Icons/searchbookimage.png')

global currentlocationforserachbook

currentlocationforserachbook=uname

global currentlocationforserachbookuserlibraryid

currentlocationforserachbookuserlibraryid=libid

global currentlocationforserachbookwlcmmsgusefulforuseraccount

global currentlocationforserachbookwlcmmsgusefulforuseraccountusername

currentlocationforserachbookwlcmmsgusefulforuseraccountusername=username

currentlocationforserachbookwlcmmsgusefulforuseraccount=wlcmmsg

searchbkbtn=Button(searchbk,text='Open', image = searchbkpic,command= searchbook) # search BOOK #CHANGE

searchbkbtn.pack()

searchbklbl=Label(searchbk,text='Search Book',font= "timesnewroman 20")

searchbklbl.pack()

spaceframe5=Frame(userframetop)

spaceframe5.pack(side= LEFT)

spacelabel1=Label(spaceframe5,text=' ')

spacelabel1.pack()

latestarrfrmfrm=Frame(userframebottom)

latestarrfrmfrm.pack(side=LEFT)

latstarrpic=PhotoImage(file='./IMAGES/Icons/latestarrivals.png')

latestarrbtn=Button(latestarrfrmfrm,text='Open', image = latstarrpic,command= latestarrivalspg) # overdue BOOK #CHANGE

latestarrbtn.pack()

latestarrlbl=Label(latestarrfrmfrm,text='LatestArrivals',font= "timesnewroman 20")

latestarrlbl.pack()

spaceframe7=Frame(userframebottom)

spaceframe7.pack(side= LEFT)

spacelabel1=Label(spaceframe7,text=' ')

spacelabel1.pack()

###########

####################21st september 2020##############

notificationsfrm=Frame(userframebottom)

notificationsfrm.pack(side=LEFT)

notipic=PhotoImage(file='./IMAGES/Icons/inbox.png')

notibtn=Button(notificationsfrm,text='Open', image = notipic,command= messaging) # overdue BOOK #CHANGE

notibtn.pack()

notilbl=Label(notificationsfrm,text='Messaging',font= "timesnewroman 20")

notilbl.pack()

spaceframe7=Frame(userframebottom)

spaceframe7.pack(side= LEFT)

spacelabel1=Label(spaceframe7,text=' ')

spacelabel1.pack()

logoutfrm=Frame(userframebottom)

logoutfrm.pack(side=LEFT)

logoutpic=PhotoImage(file='./IMAGES/Icons/logout.png')

logoutbtn=Button(logoutfrm,text='Open', image = logoutpic,command= logout) # overdue BOOK #CHANGE

logoutbtn.pack()

logoutlbl=Label(logoutfrm,text='Log Out',font= "timesnewroman 20")

logoutlbl.pack()

spaceframe6=Frame(userframebottom)

spaceframe6.pack(side= LEFT)

spacelabel1=Label(spaceframe6,text=' ')

spacelabel1.pack()

root.mainloop()

##################################

def firstpageafterlogin():

lid = userlibid

usersname=''

theusername=''

cursor.execute('select \* from users where LIBID= {}'.format(lid))

m=cursor.fetchall()

if len(m) > 0:

requsr = list(m[0])

usersname = requsr[2] + ' '+ requsr[3]

theusername=requsr[1]

x=time.localtime()

if 0<=x[3]<=6 or 17<=x[3]<=23:

nom='Good Evening '

elif 7<=x[3]<=11:

nom='Good Morning '

else:

nom='Good Afternoon '

if theusername == 'LIBRARIAN':

wlcmmsg=(nom+'LIBRARIAN')

librarianspage()

else:

wlcmmsg=(nom+str(usersname))

userpage(wlcmmsg,usersname,theusername,lid)

def loginpage():

destmainpageto()

global frameloginbuttons

global canvasloginouter

global captchaframe

global frameerrors

global captchanumber #this is just a variable dont have to delete

def extractusers():

f=open('./STORAGE/encrypteduserdetails.txt','rb')

l=pickle.load(f)

unameandpass={}

for a in l:

currperson={str(a[0]):(a[1],a[2])}

unameandpass.update(currperson)

return unameandpass

def validate():

global frameerrors

try:

frameerrors. pack\_forget()

except:

pass

frameerrors=Frame(root)

frameerrors.pack()

uname = username.get()

pwd=password.get()

lid=libid.get()

personinfo=extractusers()

if lid in personinfo and uname == personinfo[lid][0] and pwd == personinfo[lid][1] and listoffilesofcaptcha[captchanumber][1]==captchaentry.get():

frameerrors. pack\_forget()

frameloginbuttons. pack\_forget()

canvasloginouter. pack\_forget()

captchaframe. pack\_forget()

global userlibid

userlibid= lid

firstpageafterlogin()

catpval=0

try:

catpval=captchaentry.get()

except:

pass

if catpval == 0:

try:

lblcaptchastats=Label(frameerrors,text='Please Enter Captcha', fg='red')

lblcaptchastats.pack()

except:

pass

if listoffilesofcaptcha[captchanumber][1]!=catpval:

try:

lblcaptchastats=Label(frameerrors,text='Wrong Captcha Please Retry', fg='red')

lblcaptchastats.pack()

except:

pass

if lid not in personinfo:

try:

lblpassstats=Label(frameerrors,text='Library ID does not exist', fg='red')

lblpassstats.pack()

except:

pass

elif uname != personinfo[lid][0] or pwd != personinfo[lid][1]:

try:

lblpassstats=Label(frameerrors,text='Invalid Credentials', fg='red')

lblpassstats.pack()

except:

pass

root.mainloop()

def forgotpass(): #ADD FUNCTION HERE

global invalidpasswordmessage

global invalidcaptchacodeerror

invalidpasswordmessage=0#this is for giving error message in login page

invalidcaptchacodeerror=0#this is for giving error message in login page

frameloginbuttons. pack\_forget()

canvasloginouter. pack\_forget()

captchaframe. pack\_forget()

try:

frameerrors. pack\_forget()

except:

pass

def back():

global errorstreams

global otpevalfr

detailsframe.pack\_forget()

backbtn.pack\_forget()

submitbtn.pack\_forget()

try:

errorstreams.pack\_forget()

except:

pass

try:

otpevalfr.pack\_forget()

except:

pass

loginpage()

def newpass():

global newpassentryconf

global newpassentry1

global subbtn

def evaluatepass():

global newpassentryconf

global newpassentry1

global subbtn

def tryloginnow():

errorstreams.pack\_forget()

passframe.pack\_forget()

loginpage() #CHANGE THIS HERE \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

global errorstreams

try:

errorstreams.pack\_forget()

except:

pass

errorstreams=Frame(root)

errorstreams.pack()

pwd=newpassentry1.get()

pwdconf=newpassentryconf.get()

countforpasswordcaptletter,countforpasswordsmallletter, countforpasswordnumber=0,0,0 #PASSWORD EVALUATION

for varia in pwd:

if ord(varia) in range(65,91):

countforpasswordcaptletter+=1

elif ord(varia) in range(97,123):

countforpasswordsmallletter+=1

elif ord(varia) in range(48,58):

countforpasswordnumber+=1

if countforpasswordcaptletter==0 or countforpasswordsmallletter == 0 or countforpasswordnumber ==0 or len(pwd)<8:

passwordinvalidlabel=Label(errorstreams,text='Your Password is weak', fg='red')#EVALUATION FOR STRONG PASSWORD

passwordinvalidlabel.pack()

return

if pwd != pwdconf:# #IF PASSWORD ENTERED MATCHEES WITH THE CONFIRMED ONE

passdoesnotmatchlabel=Label(errorstreams,text='Passwords Entered does not match', fg='red')

passdoesnotmatchlabel.pack()

return

#binary file

passdet = open('./STORAGE/encrypteduserdetails.txt','rb')

l=pickle.load(passdet)

passdet.close()

for a in range(len(l)):

if str(l[a][0]) == lid:

break

tup=list(l[a])

tup[2]=pwd

tup=tuple(tup)

l[a]=tup

passdet=open('./STORAGE/encrypteduserdetails.txt','wb')

pickle.dump(l,passdet)

passdet.close()

subbtn.pack\_forget()

passreset=Label(passframe, text='Your Password has been reset. Try Loging in to your account now').pack()

gobtn=Button(passframe,text='Go',command=tryloginnow).pack()

passframe=Frame(root)

passframe.pack()

newpass1=Label(passframe, text='Enter Your New Password').pack()

newpassentry1=Entry(passframe,width=20,show='\*')

newpassentry1.pack()

newpass2=Label(passframe, text='Confirm Your New Password').pack()

newpassentryconf=Entry(passframe,width=20,show='\*')

newpassentryconf.pack()

subbtn=Button(passframe,text='Submit', command=evaluatepass)

subbtn.pack()

def submit():

global lid

global otpentry

def verifyotp():

global errorstreams

try:

errorstreams.pack\_forget()

except:

pass

errorstreams=Frame(root)

errorstreams.pack()

otp=otpentry.get()

if str(val) != otp:

lber=Label(errorstreams,text='Incorrect OTP', fg='red')

lber.pack()

return

detailsframe.pack\_forget()

backbtn.pack\_forget()

submitbtn.pack\_forget()

otpevalfr.pack\_forget()

try:

errorstreams.pack\_forget()

except:

pass

newpass()

global errorstreams

try:

errorstreams.pack\_forget()

except:

pass

errorstreams=Frame(root)

errorstreams.pack()

m=emailentry.get()

lid=lidentry.get()

#lid=libraryid.get()

if len(m)==0 or len(lid)==0:

lber=Label(errorstreams,text='Please enter all details', fg='red')

lber.pack()

return

cursor.execute('select \* from users where LIBID = {}'.format(lid))

m1=cursor.fetchall()

if len(m1) > 0:

requsr = list(m1[0])

fname = requsr[2]

else:

lber=Label(errorstreams,text='Invalid Library ID', fg='red')

lber.pack()

return

if m1[0][6] != m:

lber=Label(errorstreams,text='Incorrect Email ID\nPlease use the Registered Email ID', fg='red')

lber.pack()

return

val=random.randint(1000,9999)

sub='OTP Verification'

msg='Dear '+str(fname)+',\nYour OTP for you Libriary account is '+str(val)+'\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','scyjovpwhqzfbrur')

subject = sub

body = msg

msg = f'Subject: {subject}\n\n{body}'

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

submitbtn.pack\_forget()

global otpevalfr

otpevalfr = Frame(root)

otpevalfr.pack()

labelforspace = Label( otpevalfr, text =" \n\n\nPlease Enter Your OTP which you recived").pack()

otpentry = Entry (otpevalfr,width=10)

otpentry.pack()

submitbbtn=Button(otpevalfr, text='Submit',command=verifyotp).pack()

detailsframe=Frame(root)

detailsframe.pack()

#libid

lbl=Label(detailsframe,text='Reset Password \n\n', fg='blue',font= "applecherry 35")

lbl.pack(side= TOP)

textforemail=Label(detailsframe,text="please enter your Email ID ").pack()

emailentry=Entry(detailsframe, width=20)

emailentry.pack()

textfoflid=Label(detailsframe,text="please enter your Library ID ").pack()

lidentry=Entry(detailsframe, width=20)

lidentry.pack()

submitbtn=Button(root,text='submit', command=submit)

submitbtn.pack()

backbtn=Button(root,text='Back', command=back)

backbtn.pack()

#Username and password

canvasloginouter = Frame(root,width=1000, height= 1000)

canvasloginouter.pack()

lbl=Label(canvasloginouter,text='Login Page \n\n', fg='blue',font= "applecherry 35")

lbl.pack(side= TOP)

lblid=Label(canvasloginouter,text=' Library ID')

libid=Entry(canvasloginouter,width=20)

lblusername=Label(canvasloginouter,text=' Username')

username=Entry(canvasloginouter,width=20)

lblpassword=Label(canvasloginouter,text=' Password')

password = Entry(canvasloginouter, show="\*", width=20)

lblid.pack(side= LEFT)

libid.pack(side= LEFT)

lblusername.pack(side= LEFT)

username.pack(side= LEFT)

lblpassword.pack(side= LEFT)

password.pack(side= LEFT)

frameloginbuttons = Frame(root,width=1000, height= 1000)

frameloginbuttons.pack(side = BOTTOM)

#captcha

captchaframe = Frame(root,width=1000, height= 1000)

captchaframe.pack()

captchanumber= random.randint(1,9)-1

photo=PhotoImage(file=listoffilesofcaptcha[captchanumber][0])

captchaphoto=Label(captchaframe,image=photo)

captchaentry = Entry(captchaframe, width=20)

captchaentry.insert(0, "Enter Captcha Here")

#Sumbit buttons

Backtomain=Button(frameloginbuttons, text = 'Back', command= loginpagetomain)

Backtomain.pack(side = BOTTOM)

submitcred=Button(frameloginbuttons, text = 'Submit', command= validate)

submitcred.pack(side = BOTTOM)

forgotpassbutton=Button(frameloginbuttons, text = 'Reset Password', command= forgotpass)

forgotpassbutton.pack(side = BOTTOM)

createnewaccount=Button(frameloginbuttons, text = 'Create Account', command= createaccountfromloginpage)

createnewaccount.pack(side = BOTTOM)

captchaphoto.pack(side= LEFT)

captchaentry.pack(side= BOTTOM)

root.mainloop()

def mainpage(): #BE AWARE MANY VARIABLES ARE CONVERTED TO LOCAL VARIABLE HERE

global root

global frame

global canvas #actually a frame

global spaceframe1

global loginframe

global spaceframe2

global aboutusframe

global buttonsframe

global forbannerimagelistindex

global forbannerimagelistlength

global framepicbanner

frame=Frame(root)

frame.pack()

lbl = Label(frame, text = 'Welcome To Our e-Library', fg='blue',font= "applecherry 61 italic")

lbl.pack(side= TOP)

#banner --> ' Change this entire place from gif to an user controlled arrow mark

def goleft():

global forbannerimagelistindex

global forbannerimagelistlength

global framepicbanner

if forbannerimagelistindex == 0:

forbannerimagelistindex=forbannerimagelistlength-1

else:

forbannerimagelistindex-=1

framepicbanner.pack\_forget()

framepicbanner=Frame(encloseframeforbanner)

framepicbanner.pack()

photobanner=PhotoImage(file=imagelist[forbannerimagelistindex][0])

bannerphoto=Label(framepicbanner,image=photobanner)

bannerphoto.pack()

root.mainloop()

def goright():

global forbannerimagelistindex

global forbannerimagelistlength

global framepicbanner

if forbannerimagelistindex == forbannerimagelistlength-1:

forbannerimagelistindex=0

else:

forbannerimagelistindex+=1

framepicbanner.pack\_forget()

framepicbanner=Frame(encloseframeforbanner)

framepicbanner.pack()

photobanner=PhotoImage(file=imagelist[forbannerimagelistindex][0])

bannerphoto=Label(framepicbanner,image=photobanner)

bannerphoto.pack()

root.mainloop()

filehavingbannerpics=open('./STORAGE/bannerssource.txt','rb')

imagelist = pickle.load(filehavingbannerpics)

filehavingbannerpics.close()

canvas = Frame(root)

canvas.pack()

photoarrorleft=PhotoImage(file="./IMAGES/Icons/arrowleft.png")

leftbtn=Button(canvas, text = 'Click Me !', image = photoarrorleft, command= goleft) #CHANGE COMMAND HERE #chaange frame name HERE TOO for next copy paste

leftbtn.pack(side = LEFT)

forbannerimagelistlength=len(imagelist)

forbannerimagelistindex=0

#update other files here

encloseframeforbanner=Frame(canvas)

encloseframeforbanner.pack(side = LEFT)

framepicbanner=Frame(encloseframeforbanner)

framepicbanner.pack()

photobanner=PhotoImage(file=imagelist[forbannerimagelistindex][0])

bannerphoto=Label(framepicbanner,image=photobanner)

bannerphoto.pack()

photoarrorright=PhotoImage(file="./IMAGES/Icons/arrowright.png")

rightbtn=Button(canvas, text = 'Click Me !', image = photoarrorright, command= goright) #CHANGE COMMAND HERE #chaange frame name HERE TOO for next copy paste

rightbtn.pack(side = LEFT)

#photo=PhotoImage(file=listoffilesofcaptcha[captchanumber][0])

#captchaphoto=Label(captchaframe,image=photo)

'''for k in range(0, 1): #can also change it to while true

for gif in giflist:

canvas.delete(ALL)

canvas.create\_image(width/2, height/2, image=gif)

canvas.update()

time.sleep(2)''' #change this for adding a button with moving images

#navigation buttons

buttonsframe=Frame(root, width=1000)

buttonsframe.pack(side=BOTTOM)

#frame to leave space

spaceframe1=Frame(buttonsframe)

spaceframe1.pack(side= LEFT)

spacelabel=Label(spaceframe1,text=' ')

spacelabel.pack()

#login to account

loginframe=Frame(buttonsframe)

loginframe.pack(side= LEFT) #IMPORTANT PLACE

photo = PhotoImage(file = "./IMAGES/Icons/login.png") #Change the command below to pass control to open new window

Button(loginframe, text = 'Click Me !', image = photo, command= loginpage).pack(side = TOP)

loginlabel=Label(loginframe, text='Login and View Account')

loginlabel.pack(side= BOTTOM)

#frame to leave space

spaceframe3=Frame(buttonsframe)

spaceframe3.pack(side= LEFT)

spacelabel3=Label(spaceframe3,text=' ')

spacelabel3.pack()

#SEARCH BOOK

global currentlocationforserachbook

currentlocationforserachbook='GENERAL'

searchbook=Frame(buttonsframe)

searchbook.pack(side= LEFT) #IMPORTANT PLACE

photosearchbook = PhotoImage(file = "./IMAGES/Icons/searchbookimage.png") #Change the command below to pass control to open new window

photosearchbutton=Button(searchbook, text = 'Click Me !', image = photosearchbook, command= searchbookfn) #CHANGE COMMAND HERE #chaange frame name HERE TOO for next copy paste

photosearchbutton.pack(side = TOP)

searchbooklabel=Label(searchbook, text='Search a Book') #for next copypaste change this value too

searchbooklabel.pack(side= BOTTOM)

#frame to leave space

spaceframe2=Frame(buttonsframe)

spaceframe2.pack(side= LEFT)

spacelabel2=Label(spaceframe2,text=' ')

spacelabel2.pack()

#ebooks

ebookframe=Frame(buttonsframe)

ebookframe.pack(side= LEFT) #IMPORTANT PLACE

photoebook = PhotoImage(file = "./IMAGES/Icons/ebook.png") #Change the command below to pass control to open new window

ebookbutton=Button(ebookframe, text = 'Click Me !', image = photoebook, command= ebookpage) #CHANGE COMMAND HERE #chaange frame name HERE TOO for next copy paste

ebookbutton.pack(side = TOP)

ebooklabel=Label(ebookframe, text='EBooks') #for next copypaste change this value too

ebooklabel.pack(side= BOTTOM)

#frame to leave space

spaceframe5=Frame(buttonsframe)

spaceframe5.pack(side= LEFT)

spacelabel5=Label(spaceframe5,text=' ')

spacelabel5.pack()

#Suggestions

feedbackframe=Frame(buttonsframe)

feedbackframe.pack(side= LEFT) #IMPORTANT PLACE

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 too

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()