import datetime

import os

import sys

import csv

from time import sleep

from datetime import date

import re

import webbrowser

import time

#logo shown in this system is the phrase "UTAR LIBRARY SYSTEM"

def logo():

print(' \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*')

print(' \* \_ \_ \_\_\_\_\_\_\_ \_\_\_\_\_ \_ \_\_\_\_\_ \_\_\_\_ \_\_\_\_\_ \_\_\_\_\_\_\_ \_\_ \_\_\_\_\_\_\_ \_\_\_\_\_\_\_ \_\_\_\_\_\_\_ \_\_\_\_\_\_ \_\_ \_\_ \*')

print(' \* | | | |\_\_ \_\_|/\ | \_\_ \ | | |\_ \_| \_ \| \_\_ \ /\ | \_\_ \ \ / / / \_\_\_\_\ \ / / \_\_\_\_|\_\_ \_\_| \_\_\_\_| \/ | \*')

print(' \* | | | | | | / \ | |\_\_) | | | | | | |\_) | |\_\_) | / \ | |\_\_) \ \\_/ / | (\_\_\_ \ \\_/ / (\_\_\_ | | | |\_\_ | \ / | \*')

print(' \* | | | | | | / /\ \ | \_ / | | | | | \_ <| \_ / / /\ \ | \_ / \ / \\_\_\_ \ \ / \\_\_\_ \ | | | \_\_| | |\/| | \*')

print(' \* | |\_\_| | | |/ \_\_\_\_ \| | \ \ | |\_\_\_\_ \_| |\_| |\_) | | \ \ / \_\_\_\_ \| | \ \ | | \_\_\_\_) | | | \_\_\_\_) | | | | |\_\_\_\_| | | | \*')

print(' \* \\_\_\_\_/ |\_/\_/ \\_\\_| \\_\ |\_\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_/|\_| \\_\/\_/ \\_\\_| \\_\ |\_| |\_\_\_\_\_/ |\_| |\_\_\_\_\_/ |\_| |\_\_\_\_\_\_|\_| |\_| \*')

print(' \* \*')

print(' \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*')

date\_and\_time = datetime.datetime.now()

print(" Current date and time:")

print(date\_and\_time.strftime(" %Y-%m-%d %H:%M:%S"))

# Main menu to choose whether admin login or student login or guest login.

def mainmenu():

#Print the option to let user choose

logo()

print('1.Admin Login')

print('2.Student Login')

print('3.Guest Login')

print('4.Quit')

print('')

selection\_a = input('Enter your selection :')

print('')

while selection\_a != '1' and selection\_a != '2' and selection\_a != '3' and selection\_a != '4':

print('Invalid input.Please insert 1-4')

time.sleep(0.5)

selection\_a = input('Enter your selection :')

#when choose 1,it will proceed to the page to let admin login to their account.

if selection\_a == '1':

os.system('cls')

adminloginaccount()

#when choose 2,it will proceed to the page to let student login to their account.

elif selection\_a == '2':

os.system('cls')

studentlogin()

#when choose 3,it will proceed to the page to let guest login.

elif selection\_a == '3':

os.system('cls')

guestlogin()

# Exit

elif selection\_a == '4':

print('THANKS FOR USING THE SYSTEM.')

time.sleep(0.7)

exit()

def account():

global name , password

#Input name and password to log in.

name = input ('Please key in your name : ')

password = input ('Please key in the password : ')

# Page to let admin login.

def adminloginaccount():

global status

logo()

selection = input('Are you an admin of this program? (1-Yes 2-No) :')

while selection != '1' and selection != '2':

print('Invalid choice..Please retry..')

time.sleep(0.5)

os.system('cls')

logo()

selection = input('Are you an admin of this program? (1-Yes 2-No) :')

if selection == '1':

with open ('account.txt' , 'r' , newline = '') as acc:

account\_reader = csv.DictReader (acc)

account()

for row in account\_reader:

# if name and password inserted are correct,it will proceed to admin feature.

if name == row['id'] and password == row ['password'] and row['status'] == '1':

status = row['status']

admin = True

break

# else it will just loop to the page to let admin retype their name and password.

else:

admin = False

if admin == True :

os.system('cls')

adminlogin()

else:

print ('The admin id or password is incorrect')

time.sleep(0.5)

os.system('cls')

adminloginaccount()#Try again to login

elif selection == '2':

os.system('cls')

mainmenu()

def studentlogin():

global status

logo()

account()

student = False

#Open file to check whether their name and password inserted is correct or not.

with open ('account.txt' , 'r' , newline = '') as acc:

readacc = csv.DictReader (acc)

name1 = name

password1 = password

for row in readacc:

if name1 == row['id'] and password1 == row['password'] and row['status'] == '0':

status = row['status']

student = True

#Variable of student will become true when their name and password is correct.

if student == True :

os.system('cls')

studentfeature()#Proceed to student feature page.

#Or else the system will ask them to sign up a new account or try again to log in.

elif student == False:

os.system('cls')

logo()

print('1.Sign up')

print('2.Try again')

#Loop until user key in a valid input.

selection\_d = input('Enter your selection :')

while selection\_d != '1' and selection\_d != '2':

print('Invalid input.Please retry.')

time.sleep(0.5)

selection\_d = input('Enter your selection :')

#If the user choose 1 , will let them start to create a brand new account.

if selection\_d == '1' :

os.system('cls')

studentsignup()#Let student sign up an account.

elif selection\_d == '2' :

os.system('cls')

studentlogin()#Go back to let student sign in again.

def studentsignup():

logo()

#Input a name and password they want.

name = input ('Name : ')

password = input ('Password : ')

fieldnames = ['id' , 'password' , 'status' , 'limit']

#Write the name and password to txt file.

with open ( 'account.txt' , 'a' , newline = '') as acc1:

readacc = csv.DictWriter(acc1 , fieldnames = fieldnames)

readacc.writerow ({'id' : '%s' % name , 'password' : '%s' % password , 'status' : '0' , 'limit' : '5'})

print('A new account is created !')

print('')

input('')

acc1.close()

os.system('cls')

studentfeature()#Proceed to student feature page.

# Admin feature

def adminlogin():

logo()

#Print the option to let admin choose the function he want to proceed.

print('Welcome back,please enter your choice.')

print('')

print('1.Add book')

print('2.Delete book')

print('3.Search book')

print('4.Book Library')

print('5.Approve Booking Request')

print('6.Log out')

print('')

selection\_b = input('Enter your selection :')

print('')

#Created to let they insert the selection again in case they insert an invalid input.

while selection\_b != '1' and selection\_b != '2' and selection\_b !='3' and selection\_b != '4' and selection\_b != '5' and selection\_b != '6':

print('Invalid input,please retry.')

time.sleep(0.6)

os.system('cls')

logo()

print('Welcome back,please enter your choice.')

print('')

print('1.Add book')

print('2.Delete book')

print('3.Search book')

print('4.Book Library')

print('5.Approve Booking Request')

print('6.Log out')

print('')

selection\_b = input('Enter your selection :')

print('')

#Proceed to add book.

if selection\_b == '1' :

os.system('cls')

addbook()

#Proceed to delete book.

elif selection\_b == '2':

os.system('cls')

deletebook()

#Proceed to search book.

elif selection\_b == '3':

os.system('cls')

search()

#Proceed to book library system.

elif selection\_b == '4':

os.system('cls')

loading()

booklibrary()

#Proceed to discussion room to approve user's booking request

elif selection\_b == '5':

os.system('cls')

discussionadmin()

#Back to main menu of the system.

elif selection\_b == '6':

os.system('cls')

mainmenu()

def studentfeature():

global borrow\_date , book\_limit

logo()

now = datetime.datetime.now()

account\_id = name

#Print the option of student feature.

print('1.Borrow book')

print('2.Return book')

print('3.Book library')

print('4.Discussion room')

print('5.Search Book')

print('6.Log out')

selection = input('Enter your selection: ')

#Loop when user key in an invalid input.

while selection != '1' and selection != '2' and selection != '3' and selection != '4' and selection != '5' and selection !='6':

print('Invalid input.Please retry')

time.sleep(0.5)

selection = input ('Enter your selection: ')

#Proceed to borrow book.

if selection == '1':

os.system('cls')

borrowbook()

#Proceed to return book.

elif selection == '2':

os.system('cls')

returnbook()

#Proceed to book library system.

elif selection == '3':

os.system('cls')

loading()

booklibrary()

#Proceed to discussion room booking system.

elif selection == '4':

os.system('cls')

discussionroom()

#Proceed to search book function.

elif selection == '5':

os.system('cls')

search()

#Back to main menu.

elif selection == '6':

os.system('cls')

mainmenu()

def guestlogin():

global status

status = 3 #Make a condition that guess is not 1(admin) or 0(student).

logo()

#Print the option of guest page.

print('1.Search book')

print('2.Book library')

print('3.Return to main menu')

print('')

selection\_c = input('Enter your selection :')

#Looping to make sure user key in a valid input

while selection\_c != '1' and selection\_c != '2' and selection\_c != '3':

print('Invalid input, please retry.')

time.sleep(0.5)

os.system('cls')

guestlogin()

#Proceed to search book.

if selection\_c == '1':

os.system('cls')

search()

#Proceed to book library system.

elif selection\_c == '2' :

os.system('cls')

loading()

booklibrary()

#Back to main menu of this system.

elif selection\_c == '3' :

os.system('cls')

mainmenu()

#Created to check whether the user is now a admin or student or guest.

def check():

if status == '1':#when an admin logged in , their status will become 1 automatically.

os.system('cls')

adminlogin()

elif status == '0':#when an student logged in , their status will become 0 automatically.

os.system('cls')

studentfeature()

else:#when an guest logged in , their status will become 3 automatically.

os.system('cls')

guestlogin()

#Search book function.

def search():

logo()

prompt = False

selection = input('Insert "1" to continue and "2" to back : ')

#Loop until user key in a valid input.

while selection != '1' and selection != '2' :

print('Invalid input.')

selection =input('Enter "1" to continue and "2" to back : ')

#If choose 1, user can start to insert the book id to search the book detail.

if selection == '1':

book = input ('ID: ')

#Read file to check whether book id inserted by user is correct or false.

with open ('book.txt' , 'r' ) as searchbook:

readbook = csv.reader (searchbook , delimiter = ',')

for row in readbook:

if book == row[0]:

prompt = True

#Print book details.

print (f'ID : {row[0]}\nTitle : {row[1]}\nAuthor : {row[2]}\nPublisher: {row[3]}\nPublication date:{row[4]}\nType:{row[6]}')

input('Press enter to continue')

os.system('cls')

check()#Back to previous page after searched book.

#Message pop out when book id inserted by user is non-detectable.

if prompt == False:

print('The book ID is invalid, please retry.')

time.sleep(0.5)

os.system('cls')

search()#Ask user to key in again the book id.

#Back to previous page.

elif selection == '2':

os.system('cls')

check()

#Display something like loading before page book library.

def loading():

logo()

for i in range(101):

print('\rLoading. . . %d' %i ,'%',end='')

time.sleep(0.015)

print('')

os.system('cls')

#A self-defined function that list out all the books in library.

def booklibrary():

logo()

#Print the option of book

print("Categories of books:")

print("\n1.Reference books \n2.Novels \n3.Dictionaries\n4.Back")

print("")

choice=input("Please enter your selection:")

#Looping to let user key in the option they want if they key in an invalid number accidentally.

while choice != '1' and choice != '2' and choice !='3' and choice != '4':

print('Invalid input.Please try again')

choice=input("Please enter your selection:")

os.system('cls')

if choice == '1':

logo()

#Branch of reference book

print("Choices of reference books: \n1.Chemistry \n2.Biology \n3.Mathematics \n4.Programming Concept \n5.Physics")

print("")

select=input("Please key in your selection:")

#Loop until user insert a valid input.

while select != '1' and select != '2' and select != '3' and select !='4' and select != '5':

print('Invalid input.Please try again')

select=input("Please key in your selection:")

os.system('cls')

#Print the title and author according to selection.

if select=='1':

logo()

with open ('book.txt' , 'r' ) as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'chemistry':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

elif select=='2':

logo()

with open ('book.txt', 'r') as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'biology':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

elif select=='3':

logo()

with open ('book.txt', 'r') as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'mathematics':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

elif select=='4':

os.system('cls')

logo()

with open ('book.txt', 'r') as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'programming':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

elif select=='5':

os.system('cls')

logo()

with open ('book.txt', 'r') as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'physics':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

elif choice == '2':

logo()

#Branch of novels.

print("Choices of novels: \n1.Science-fiction \n2.Fantasy \n3.Romances \n4.Action-adventure")

print("")

select1= input("Please enter your selection:")

#Loop until user key in a valid input.

while select1 != '1' and select1 != '2' and select1 !='3' and select1 != '4':

print('Invalid input.Please try again')

select1 =input("Please key in your selection:")

os.system('cls')

#Print the title and author of the books.

if select1=='1':

logo()

with open ('book.txt' , 'r' ) as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'science fiction':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

elif select1=='2':

logo()

with open ('book.txt' , 'r' ) as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'fantasy':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

elif select1=='3':

logo()

with open ('book.txt' , 'r' ) as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'romance':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

elif select1=='4':

logo()

with open ('book.txt' , 'r' ) as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'action story':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

#Print the title and author of dictionaries.

elif choice=='3':

logo()

print("Dictionaries:")

with open ('book.txt' , 'r' ) as library:

readlibrary = csv.reader (library , delimiter = ',')

for row in readlibrary:

if row[6] == 'dictionary':

print (f'ID: {row[0]}\nTittle: {row[1]}\nAuthor: {row[2]}\n')

os.system('pause')

os.system('cls')

booklibrary()

#Back to previous page.

elif choice == '4':

check()

def addbook():

logo()

list1 = []#empty list created.

with open ('book.txt' , 'r') as inp:

read = csv.reader (inp)

for row in read:

match = re.search (r'\w{1}' , row[0])

list1.append (match.group())#append the book id into list1.

#ID list show the ID for that particular book type and latest ID show the last id we are using .

print(' \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*')

print(' \* ID list \* The Latest ID using \*')

print(' \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*')#Count the number of alphabet to know the latest id.

print(' \* A - chemistry \* A%d \*' %(list1.count('A')))

print(' \* B - action story \* B%d \*' %(list1.count('B')))

print(' \* C - biology \* C%d \*' %(list1.count('C')))

print(' \* D - dictionary \* D%d \*' %(list1.count('D')))

print(' \* E - fantasy \* E%d \*' %(list1.count('E')))

print(' \* F - mathematics \* F%d \*' %(list1.count('F')))

print(' \* G - physics \* G%d \*' %(list1.count('G')))

print(' \* H - programming \* H%d \*' %(list1.count('H')))

print(' \* I - romance \* I%d \*' %(list1.count('I')))

print(' \* J - science fiction \* J%d \*' %(list1.count('J')))

print(' \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*')

print('')

print('Do you want to add a new book? 1- Yes 2- No')

selection = input('Enter your selection: ')

#Loop when user key in the wrong input.

while selection != '1' and selection != '2':

print('Invalid input.')

time.sleep(0.5)

selection = input('Enter your selection: ')

#Insert 1 to add book.

if selection == '1':

ID = input ('ID ( ACCORDING TO ID LIST,eg: A6): ')

title = input ('Title: ')

author = input ('Author: ')

publisher = input('Publisher :')

publish\_date = input('Publication date: ')

stock = int (input ('Stock: '))

book\_type = input ('Type ( ACCORDING TO ID LIST,eg: chemistry ) : ')

#Open txt to add new book into it.

with open ( 'book.txt' , 'a',newline='' ) as add:

writebook = csv.writer (add)

writebook.writerow (['%s' % ID, '%s' % title, '%s' % author, '%s' % publisher, '%s' % publish\_date , '%d' % stock , '%s' % book\_type])

print('A new book ID%s is added !'%ID)

os.system('pause')

os.system('cls')

adminlogin()

#Back to admin page.

elif selection == '2':

os.system('cls')

adminlogin()

def deletebook():

logo()

flag1 = False

#Option for delete book.

print('Do you want to delete a book? 1-Yes 2-No')

selection = input('Enter your selection:')

#Loop when user key in an invalid input.

while selection != '1' and selection != '2':

print('Invalid input,please retry.')

time.sleep(0.5)

selection = input('Enter your selection:')

#Proceed to let admin key in a book ID that he want to delete.

if selection == '1':

book = input ('ID: ')

with open ('book.txt' , 'r' ) as delete:

readbook = csv.reader (delete , delimiter = ',')

for row in readbook:

if book == row[0]:

flag1 = True

print (f'ID : {row[0]}\nTitle : {row[1]}\nAuthor : {row[2]}\nPublisher: {row[3]}\nPublication date:{row[4]}\nType:{row[6]}')#Print details of books.

if flag1 == True:

delete\_choice = input ('\nAre you sure \n1. Yes 2. No \nSelection: ')#Ask one more time for confirmation.

#Loop until user key in a valid input.

while delete\_choice !='1' and delete\_choice != '2':

print('Invalid input.')

delete\_choice = input ('\nAre you sure \n1. Yes 2. No \nSelection: ')

#COnfirm to delete that particular book.

if delete\_choice == '1':

#Process of rewrite the data inside the file.

with open('book.txt', 'r') as inp, open('book\_temporary.txt', 'w' , newline = '') as out:

write = csv.writer(out)

for row in csv.reader(inp):

if row[0] != book:

write.writerow (row)

with open('book\_temporary.txt', 'r') as inp, open('book.txt', 'w' , newline = '') as out:

read1 = csv.reader(inp)

write1 = csv.writer(out)

for row in read1:

write1.writerow (row)

os.system('cls')

adminlogin()#Back to admin login page after deleted that book.

#Back to delete book page when user choose 2 or anything.

else:

os.system('cls')

deletebook()

#This meesage shown when book ID inserted is incorrect.

else:

print('The book ID is invalid, please retry.')

time.sleep(0.5)

os.system('cls')

deletebook()

#Back to admin login page when user key in 2.

elif selection == '2':

os.system('cls')

adminlogin()

#To let admin approve the booking request.

def discussionadmin():

logo()

print('1.Approve Booking Request.\n2.Back\n')

choice = input('Enter your choice: ')

#Loop when user entered a wrong input.

while choice != '1' and choice != '2':

print('Invalid input,please retry')

time.sleep(0.5)

os.system('cls')

logo()

print('1.Approve Booking Request.\n2.Back\n')

choice = input('Enter your choice: ')

if choice == '1':

print("Details of students for booking discussion room:\n")

print ('Name\t\tStudent ID\tDate\t\tTime start\tTime end\tNumber of student\tEmail')

#Open file to read the detail of student.

with open ('discussionroominfo.txt','r') as dscsroom\_info:

readdiss= csv.reader(dscsroom\_info, delimiter=',')

linecount = 0

for row in readdiss:

#Print all details key in by student.

if linecount==0:

print (f'{row[0]} \t{row[1]} \t{row[2]} \t{row[3]} \t\t{row[4]} \t\t{row[5]}\t\t\t{row[6]}')

linecount= linecount+1

else:

print (f'{row[0]} \t{row[1]} \t{row[2]} \t{row[3]} \t\t{row[4]}\t\t{row[5]}\t\t\t{row[6]}')

#Input the student id that admin want to approve.

ID=input("\nEnter student's ID:")

flag = False

with open('discussionroominfo.txt','r')as info:

readdiss= csv.reader(info, delimiter=',')

for row in readdiss:

if ID == row[1]:

flag = True #webbrowser pop out when admin key in the correct student id and send an e-mail manually to that student.

website='https://mail.google.com/mail/u/0/#inbox?compose=new'

webbrowser.open\_new(website)

#Program start to delete the information of that particular student.

if flag == True:

with open('discussionroominfo.txt', 'r') as inp, open('discussionroominfo\_temp.txt', 'w' , newline = '') as out:

writediss = csv.writer(out)

for row in csv.reader(inp):

if row[1] != ID:

writediss.writerow (row)

with open('discussionroominfo\_temp.txt', 'r') as inp, open('discussionroominfo.txt', 'w' , newline = '') as out:

readdiss = csv.reader(inp)

writediss = csv.writer(out)

for row in readdiss:

writediss.writerow (row)

print('You have successfully aprrove the booking of discussion room.')

os.system('pause')

os.system('cls')

discussionadmin()

#Program loop back when admin key in a student id did not existed.

else:

print('The student ID is invalid')

os.system('pause')

os.system('cls')

discussionadmin()

#Back to admin login page.

elif choice == '2':

os.system('cls')

adminlogin()

def borrowbook():

now = datetime.datetime.now()

account\_id = name

logo()

selection = input('Insert "1" to continue and "2" to back : ')

#Looping until user key in a valid input.

while selection != '1' and selection != '2' :

print('Invalid input.')

selection = input('Enter "1" to continue and "2" to back')

#Starting to check whether that user is qualified to borrow book or not.

if selection == '1':

#Read the file.

with open ('account.txt' , 'r') as acc2:

readacc = csv.reader (acc2)

for row in readacc:

if name == row[0]:

book\_limit = int(row[3])

if book\_limit == 0:

#This message pop out when user is already reached their borrow limit.

print ('You have reached the book limit. \nPlease return books that has been borrowed in order to borrow more.')

input ('Press ENTER to continue')

studentfeature()#Back to student featured to ask them to use another function.

#Allow user to borrow book from library.

print ('IMPORTANT: The books are required to be returned within two weeks upon borrowing.')

book = input ('ID: ')

flag = False

#Read the file to search the book id inserted by user.

with open ('book.txt' , 'r' ) as book2:

readbook = csv.reader (book2 , delimiter = ',')

for row in readbook:

if book == row[0]:

if row[5] != '0':

flag = True

#Print the book detail.

print (f'\nID : {row[0]}\nTitle : {row[1]}\nAuthor : {row[2]}\nSubject: {row[6]}\nAvailability: Available')

elif row[5] == '0':

#For some case,the book id inserted by user is out of stock.

print (f'\nID : {row[0]}\nTitle : {row[1]}\nAuthor : {row[2]}\nSubject: {row[6]}\nAvailability: Unavailable')

null = input ('\nThe book is unavailable, please come back next time. \nPress ENTER to return to main menu.')

os.system('cls')

studentfeature()#Go back to previous page when the book is unavailable this time.

#When the book id is incorrect.

if flag == False:

input('The Book ID is incorrect.Press enter to try again.')

os.system('cls')

borrowbook()

borrow = input ('\nIs this the book that you want to borrow? \n1. Yes 2. No \nSelection: ')

while borrow != '1' and borrow != '2':

print('Invalid input,please try again.')

borrow = input ('\nIs this the book that you want to borrow? \n1. Yes 2. No \nSelection: ')

if borrow == '1':

with open('book.txt', 'r') as book3, open('book\_temporary.txt', 'w' , newline = '') as book4:

writebook = csv.writer(book4)

for row in csv.reader(book3):

if row[0] == book:

data = int (row[5])

if data > 0:

new\_data = data - 1

writebook.writerow ([row[0] , row[1] , row[2] , row[3] ,row[4], '%d' %new\_data, row[6]])

elif row[0] != book:

writebook.writerow (row)

#Back to student feature page.

elif borrow == '2':

os.system('cls')

studentfeature()

#Rewrite the data in book.txt by using 2 file method.

with open('book\_temporary.txt', 'r') as book5, open('book.txt', 'w' , newline = '') as book6:

readbook = csv.reader(book5)

writebook = csv.writer(book6)

for row in readbook:

writebook.writerow (row)

#Write the book reacord into history.txt to check the borrow record by user.

with open ('history.txt' , 'a' , newline = '') as history:

time = now.strftime("%Y-%m-%d")

writehistory = csv.writer(history , delimiter = ',' , quotechar = '"' , quoting = csv.QUOTE\_MINIMAL)

writehistory.writerow (['%s' %book , '%s' %account\_id , '1' , '%s' %time])

#Process to let the borrow limit substrate in account.txt

fieldnames = ['id' , 'password' , 'status' , 'limit']

with open ('account.txt' , 'r') as acc3 , open ('account\_temporary.txt' , 'w' , newline = '') as acc4:

readacc = csv.DictReader(acc3)

writeacc = csv.DictWriter(acc4 , fieldnames = fieldnames)

writeacc.writeheader()

for row in readacc:

if name == row['id']:

book\_limit -= 1

writeacc.writerow ({'id' : row['id'] , 'password' : row['password'] , 'status' : row['status'] , 'limit' : '%d' %book\_limit})

elif name != row['id']:

writeacc.writerow (row)

with open ('account\_temporary.txt' , 'r') as acc5 , open ('account.txt' , 'w' , newline = '') as acc6:

readacc = csv.DictReader(acc5)

writeacc = csv.DictWriter(acc6 , fieldnames = fieldnames)

writeacc.writeheader()

for row in readacc:

writeacc.writerow (row)

print ('\nThe book is borrowed and recorded in your account.')

input ('Press ENTER to return to main menu.')

os.system('cls')

studentfeature()#Back to previous page.

#Back to previous page when user don't want to borrow book from library anymore.

elif selection == '2':

os.system('cls')

studentfeature()

def returnbook():

import time

logo()

now = datetime.datetime.now()

account\_id = name

print('')

#An input that let user to go back previous page if they key in the wrong input.

selection = input('Insert "1" to continue and "2" to back : ')

#Loop until user key in a valid input.

while selection != '1' and selection != '2' :

print('Invalid input.')

selection = input('Enter "1" to continue and "2" to back : ')

if selection == '1':

#Show all the book borrowed by that user.

print ('Select which books you want to return.')

with open ('book.txt' , 'r' ) as book7 , open('history.txt' , 'r') as history2:

file2 = csv.reader (history2 , delimiter = ',')

file1 = csv.reader (book7 , delimiter = ',')

for file2\_row in file2:

if file2\_row[1] == account\_id and file2\_row[2] == '1':

for file1\_row in file1:

if file1\_row[0] == file2\_row[0]:

#Print all the details of the book.

print (f'\nID : ID{file1\_row[0]}\nTitle : {file1\_row[1]}\nAuthor : {file1\_row[2]}\nSubject: {file1\_row[3]}')

book7.seek (0,0)

break

return\_book = input ('\nID: ')

flag = False

#Try to check whether the book id key in by user is in borrow record.

with open ('history.txt' , 'r') as history3:

readhistory = csv.reader(history3)

for row in readhistory:

if row[0] == return\_book and account\_id == row[1]:

flag = True

data = row[3]

match = re.search (r'\d{4}-\d{2}-\d{2}' , data)

time1 = datetime.datetime.strptime(match.group() , '%Y-%m-%d').date()

year = int (time1.strftime('%Y'))

month = int (time1.strftime('%m'))

day = int (time1.strftime('%d'))

borrow\_date = date(year , month , day)

#else,loop back to return book to let user key in a correct input.

if flag == False:

print('The book ID is invalid.Please try again.')

time.sleep(0.5)

os.system('cls')

returnbook()

year2 = int (now.strftime ('%Y'))

month2 = int (now.strftime ('%m'))

day2 = int (now.strftime ('%d'))

return\_date = date(year2 , month2 , day2)

penalty\_date = return\_date - borrow\_date

penalty\_date2 = int (penalty\_date.days)

#If that user didn't return the book in a week, system will ask him/her to pay the penalty.

if penalty\_date2 > 7:

price = (penalty\_date2 - 8) \* 0.5 + 2

print ("You didn't return the book in a week.\nYou will be charged RM%.2f" %price)

website='https://www2.pbebank.com/myIBK/apppbb/servlet/BxxxServlet?RDOName=BxxxAuth&MethodName=login'#Webpage pop out for the payment.

webbrowser.open\_new(website)

print('Making transaction . . .')

time.sleep(10)

print('Payment received !\n')

#Make sure the user wanted to return the book.

returned = input ('\nAre you sure that you want to return this book? \n1. Yes 2. No \nSelection: ')

#Loop when user insert an invalid input.

while returned != '1' and returned != '2':

print('Invalid input.')

returned = input ('\nAre you sure? \n1. Yes 2. No \nSelection: ')

#If user decided to return the book, will start to rewrite the data in file .

if returned == '1':

with open('book.txt', 'r') as book8, open('book\_temporary.txt', 'w' , newline = '') as book9:

writebook = csv.writer(book9)

for row in csv.reader(book8):

if row[0] == return\_book:

data = float (row[5])

if data > 0:

new\_data = data + 1

writebook.writerow ([row[0] , row[1] , row[2] , row[3] ,row[4], '%d' %new\_data,row[6]])

elif row[0] != return\_book :

writebook.writerow (row)

with open('book\_temporary.txt', 'r') as inp, open('book.txt', 'w' , newline = '') as out:

readbook = csv.reader(inp)

writebook = csv.writer(out)

for row in readbook:

writebook.writerow (row)

with open ('history.txt' , 'r') as history3 , open ('history\_temporary.txt' , 'w' , newline = '') as history4:

time = now.strftime("%Y-%m-%d")

writehistory = csv.writer(history4)

for row in csv.reader(history3):

if row[0] == return\_book and row[1] == account\_id:

writehistory.writerow ([row[0] , row[1] , '0' ,row[3] , '%s' %time])

elif row[0] != return\_book or row[1] != account\_id:

writehistory.writerow (row)

with open('history\_temporary.txt', 'r') as history5, open('history.txt', 'w' , newline = '') as history6:

readhistory = csv.reader(history5)

writehistory = csv.writer(history6)

for row in readhistory:

writehistory.writerow (row)

fieldnames = ['id' , 'password' , 'status' , 'limit']

with open ('account.txt' , 'r') as inp , open ('account\_temporary.txt' , 'w' , newline = '') as out:

readacc = csv.DictReader(inp)

writeacc = csv.DictWriter(out , fieldnames = fieldnames)

writeacc.writeheader()

for row in readacc:

if name == row['id']:

book\_limit = int (row['limit'])

book\_limit += 1

writeacc.writerow ({'id' : row['id'] , 'password' : row['password'] , 'status' : row['status'] , 'limit' : '%d' %book\_limit})

elif name != row['id']:

writeacc.writerow (row)

with open ('account\_temporary.txt' , 'r') as inp , open ('account.txt' , 'w' , newline = '') as out:

readacc = csv.DictReader(inp)

writeacc = csv.DictWriter(out , fieldnames = fieldnames)

writeacc.writeheader()

for row in readacc:

writeacc.writerow (row)

#Successful to return the book

print ('\nThe book is returned.')

input ('Press ENTER to return to main menu.')

os.system('cls')

studentfeature()#Back to previous page.

#If user don't want to return the book , program will go back to previous page.

elif returned == '2':

os.system('cls')

studentfeature()

#Back to previous page.

elif selection == '2':

os.system('cls')

studentfeature()

def discussionroom():

logo()

#Print the rules of using discussion room.

print("\nRules & Regulations on the Use of Discussion Rooms:")

print('1.Booking duration is in 1 hour slot to a maximum of 2 hours slot.')

print('2.Each booking requires a minimum of 4 students and maximum of 6 student.')

print('3.No eating or drinking is allowed in Discussion Rooms.')

print('4.Students should not leave any personal belongings in the rooms,library will not be responsible for any loss of personal belongings')

print('5.Please ensure that the room is in good condition after used.')

print('6.BOOKING TIME FOR DISCUSSION ROOM IS FROM 12PM TO 9PM!')

selection = input('\nDo you want to book discussion room?\n1.Yes 2.No\nEnter your selection:')

#Loop until user key in a valid input.

while selection != '1' and selection != '2':

print('Invalid input.Please try again\n')

time.sleep(0.5)

os.system('cls')

logo()

print("\nRules & Regulations on the Use of Discussion Rooms:")

print('1.Booking duration is in 1 hour slot to a maximum of 2 hours slot.')

print('2.Each booking requires a minimum of 4 students and maximum of 6 student.')

print('3.No eating or drinking is allowed in Discussion Rooms.')

print('4.Students should not leave any personal belongings in the rooms,library will not be responsible for any loss of personal belongings')

print('5.Please ensure that the room is in good condition after used.')

print('6.BOOKING TIME FOR DISCUSSION ROOM IS FROM 12PM TO 9PM!')

selection = input('\nDo you want to book discussion room?\n1.Yes 2.No\nEnter your selection:')

#Choose 1 to start the booking process.

if selection == '1':

print("\nTo book our discussion room, please key in the details correctly as approval will be informed within 24 hours via email.")

print('')

date=input("Date:")

timestart=int(input("Time start(24 hours format):"))

#Loop when invalid time occur.

while timestart < 1200 or timestart >2100:

print('Sorry.Please follow the library operation hour.')

timestart=int(input("Time start(24 hours format):"))

timeend = int(input('Time end(24 hours format): '))

while timeend >2100 or timeend <1200:

print('Sorry.Please follow the library operation hour.')

timeend = int(input('Time end(24 hours format): '))

time1 = timeend - timestart

#Loop when time is less then 1 hour and more than 2 hour.

while time1 > 200 or time1 < 100:

print('Booking hour cannot exceed 2 hours and at least 1 hour.')

timestart=int(input("Time start(24 hours format):"))

while timestart < 1200:

print('Sorry.Please follow the library operation hour.')

timestart=int(input("Time start(24 hours format):"))

timeend = int(input('Time end(24 hours format): '))

while timeend >2100:

print('Sorry.Please follow the library operation hour.')

timeend = int(input('Time end(24 hours format): '))

time1 = timeend - timestart

name=input("Name:")

studentid=input("Student ID:")

email=input("Email address:")

num=int(input("Number of students:"))

#Loop when too many or too less student in a room.

while num<4 or num > 6:

print('Sorry,you are not allowed to book discussion room. ')

num=int(input("Number of students:"))

selection\_2 = input("\nEnter '1' to submit or enter '2' to cancel:")

#Loop when invalid choice occur.

while selection\_2 != '1' and selection\_2 != '2':

print('Please insert a valid input.')

selection\_2 = input("\nEnter '1' to submit or enter '2' to cancel:")

#Write the detail of the student into txt file.

if selection\_2 == '1':

fieldnames= ['Name','ID','Date','Time start','Time end','Number of Students','Email']

with open('discussionroominfo.txt','a',newline = '') as discussion:

writediss= csv.DictWriter(discussion, fieldnames=fieldnames)

writediss.writerow ({'Name':'%s'%name, 'ID':'%s'%studentid,'Date':'%s'%date,'Time start':'%d'%timestart,'Time end':'%d'%timeend,'Number of Students':'%s'%num,'Email':'%s'%email})

print("You have successfully submit your information.")

os.system('pause')

os.system('cls')

studentfeature()#Back to student feature.

#Back to discussion room main page.

elif selection\_2 == '2':

os.system('cls')

discussionroom()

#Back to student feature page.

elif selection == '2':

os.system('cls')

studentfeature()

logo()

quote = ["Reading is a basic tool in the living of a good life;",

"Today a reader,tomorrow a leader.",

"Work Hard Play Hard!"]

for line in quote:

for c in line:

print(c, end='')

sys.stdout.flush()

sleep(0.01)

print('')

print('')

os.system('pause')

os.system('cls')

mainmenu()