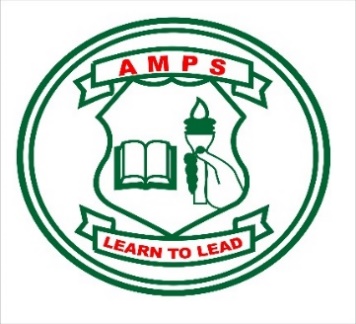
**ALWIN MEMORIAL PUBLIC SCHOOL**

**Indira Nagar, Selaiyur, Chennai – 600073**

****

**AISSCE 2020 – 2021**

CARGO MANAGEMENT SYSTEM

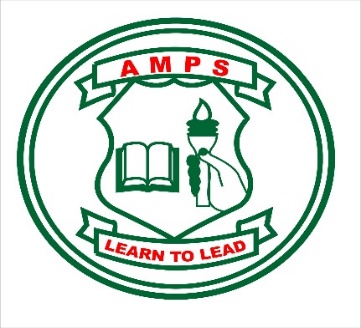
**NAME.………………………………………………**

**REGISTER NO……………………………………**

**STD.….…………………SEC……………………..SUBJECT.………………………………………….**

**ALWIN MEMORIAL PUBLIC SCHOOL**

**Indira Nagar, Selaiyur, Chennai – 600073**

****

**BONAFIDE CERTIFICATE**

**This is to certify that the project titled CARGO MANAGEMENT SYSTEM is a Bonafide work done by VISWANATHAN KRISHNAN in partial fulfillment of the requirements in computer science practical as prescribed by CBSE for AISSCE 2020 – 2021.**

**REGISTER NO. \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**TEACHER-IN-CHARGE PRINCIPAL**

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**ACKNOWLEDGEMENT**

It would be my utmost pleasure to thank to my teachers in computer science department for providing a helping hand in this project . Their valuable support , guidance and supervision are responsible for attaining its present form.

I would also like to thank all staffs , parents and friends for providing all necessities to complete my project successfully**.**

Last but not the least , I would like to thank to all who helped me directly or indirectly to complete this project

**CONTENTS**

* + ABSTRACT
  + REQUIREMENT ANALYSIS
    - DESIGN AND APPLICATION
    - HARDWARE AND SOFTWARE REQUIREMENTS
  + FLOW CHART
    - PROJECT FLOW
    - PROGRAM FLOW
  + SOURCE CODE
  + OUTPUTS
  + FUTURE ENHANCEMENT
  + BIBLIOGRAPHY

**ABSTRACT**

In recent days, shipping of goods is one of the important task in everyday life. So, while maintaining multiple shipments, we will not be able to remember all data.

At this part, Cargo Management system plays it role. Using this software, a person would be able to maintain data in tabular format. Also, a person would be able to add data, fetch data, update the details and finally delete the data.

This project contains 3 major tables namely:

* Import
* Export
* Employee Details

In this software, only one user (The Admin) will be able to register himself and use the software. Also, while logging in if username or password is incorrect, appropriate messages will be displayed.

**REQUIREMENT ANALYSIS**

DESIGN AND APPLICATION:

The windows of this project were built with an inbuilt library in python called the TKINTER. Using this library, many options like Window color, Window size, Buttons and Labels were implemented.

This project can mainly be used when we are about to handle large amount of shipping of goods. In today’s world, many freight forwarding companies encourage using this type of software as it would be easy for them to manage shipping details.

Frontend – Python

Backend - MySQL

SOFTWARE AND HARDWARE REQUIREMENTS:

Software’s:

PYTHON 3.7 and above

MYSQL 8.0

Hardware:

4 GB ram and above

Intel i3 processor

Graphic device – Intel® UHD Graphics

**FLOW CHART**

Project Flow:

Log Out

STOP

Enter Pin Number to retrieve Password

Access three tables

Insert, See, Update and Delete details

Invalid Username and Password

Valid Username and Password

Enter Username & Password

START

Program Flow:

LOGIN

LOGOUT

IMPORT

Delete Employee Details

Update Employee Details

Fetch Employee Details

Insert Employee Details

Insert Export Details

Delete Export Details

Update Export Details

Fetch Export Details

Delete Import Details

Update Import Details

Fetch Import Details

Insert Import Details

EMPLOYEE

EXPORT

SOURCE CODE :

#For executing and storing the data ,Please make sure that MYSQL installed in your device

#using pip install mysql-connector in the command prompt [command - pip install mysql-connector]

#Importing Libraries to be used

from tkinter import \*

from tkinter import ttk

from tkinter import messagebox

import mysql.connector as sql

import time

#Connecting to the database

connection = sql.connect(host='localhost',database='cms',user='root',password='mysql')

cursor = connection.cursor()

#Creating required database and tables

sql2="create table if not exists import (Billing\_ID int primary key,Carrier\_Name varchar(50),Origin varchar(50),Destination varchar(50),Container\_Dimensions varchar(50),Mode\_of\_Transport varchar(50),Payment\_Amount int)"

cursor.execute(sql2)

sql3="create table if not exists export (Billing\_ID int primary key,Carrier\_Name varchar(50),Origin varchar(50),Destination varchar(50),Container\_Dimensions varchar(50),Mode\_of\_Transport varchar(50),Payment\_Amount int)"

cursor.execute(sql3)

sql4="create table if not exists employee(Employee\_ID int primary key,Employee\_FName varchar(50),Employee\_LName varchar(50),Date\_of\_Birth date,Designation varchar(50),Employee\_PhNo bigint)"

cursor.execute(sql4)

#Registering New user

def register():

window.destroy()

time.sleep(1)

a=Tk()

def submit():

username=name.get()

password=passw.get()

np=key.get()

cn=na.get()

if name.get()=="":

messagebox.showerror(parent=a,title="ERROR",message="Please Enter your name")

elif passw.get()=="":

messagebox.showerror(parent=a,title="ERROR",message="Please Enter your password")

elif key.get()=="":

messagebox.showerror(parent=a,title="ERROR",message="Please Enter your pin number")

elif key.get()=="":

messagebox.showerror(parent=a,title="ERROR",message="Please Enter your Company Name")

elif key.get().isdigit()==False:

messagebox.showerror(parent=a,title="ERROR",message="Please Enter a Pin Number.\nYou have Entered an alphabet")

elif passw.get()!=conpas.get():

messagebox.showerror(parent=a,title="ERROR",message="Password's aren't same Please Check")

passw.delete(0,'end')

conpas.delete(0,'end')

elif passw.get()==conpas.get() and key.get().isdigit():

b=open("up.txt","w")

b.write(username+"\n")

b.write(password+"\n")

b.write(np+"\n")

b.write(cn)

b.close()

name.delete(0,'end')

passw.delete(0,'end')

conpas.delete(0,'end')

key.delete(0,'end')

na.delete(0,'end')

messagebox.showinfo(parent=a,title="SUCCESS",message="New User Registered successfully")

messagebox.showinfo(parent=a,title="Information",message="Please Relaunch your Application to save the details")

a.destroy()

else:

return

global username

global password

global name

global passw

def info():

messagebox.showinfo(parent=a,title="Information",message="Please enter a pin number . This is asked in order to fetch the password if you forget")

rl=Label(a,text='Register Yourself Here',font='Calibri 18',fg='lemon chiffon',bg='brown')

rl.place(x=100,y=20)

name\_label = Label(a, text = 'Username : ',font='14',bg='lemon chiffon')

name\_label.place(x=20,y=85)

name = Entry(a,text = "",font='14',width=14,justify=LEFT)

name.place(x=210,y=85)

passw\_label = Label(a,text = 'Password : ',font='14',bg='lemon chiffon')

passw\_label.place(x=20,y=125)

passw=Entry(a,text = "",show = '\*',font='14',width=14,justify=LEFT)

passw.place(x=210,y=125)

conpas\_label=Label(a,text = 'Confirm Password : ',font='14',bg='lemon chiffon')

conpas\_label.place(x=20,y=165)

conpas=Entry(a,text = "",show = '\*',font='14',width=14,justify=LEFT)

conpas.place(x=210,y=165)

key\_label = Label(a, text = 'Set a Pin Number : ',font='14',bg='lemon chiffon')

key\_label.place(x=20,y=205)

key = Entry(a,text = "",font='14',show = '\*',width=14,justify=LEFT)

key.place(x=210,y=205)

na\_label = Label(a, text = 'Company Name : ',font='14',bg='lemon chiffon')

na\_label.place(x=20,y=245)

na = Entry(a,text = "",font='14',width=14,justify=LEFT)

na.place(x=210,y=245)

c=Button(a,text=" ? ",command=info,font='Calibri 12',bg='lemon chiffon',bd=1)

c.place(x=373,y=204)

login\_btn=Button(a,text = 'Register',command=submit,bg='lemon chiffon',bd=1)

a.bind('<Return>', lambda i : submit())

login\_btn.place(x=280,y=290)

a.title("Register Yourself Here")

a.geometry("400x330+400+200")

a.resizable(0,0)

a.config(bg="brown")

a.mainloop()

#++++++++++++++++++++++++PROJECT BEGINS HERE++++++++++++++++++++++++#

#Creating main window

window=Tk()

#Assigning username and password with string variables

name\_var=StringVar()

passw\_var=StringVar()

#Retriving Username and Password

name=name\_var.get()

password=passw\_var.get()

#Opening and reading login credentials from text file

with open("up.txt", "r") as fd:

i = fd.read().splitlines()

if len(i)!=0:

name=i[0]

password=i[1]

np=i[2]

cn=i[3]

#Defining Login Button's command

def login():

#Verifying Username and Password

if (name\_entry.get()==i[0]) and (passw\_entry.get()==i[1]):

name\_entry.delete(0,'end')

passw\_entry.delete(0,'end')

messagebox.showinfo(parent=window,title="SUCCESS",message=("User Login successful.\n\nWelcome back "+i[0]))

#Destroying Main window

window.destroy()

win=Tk()

#Editing table Import

def imp():

a=Tk()

#Inserting into Import

def insert():

b=Tk()

def submit():

if (e2.get()=="" or e3.get()=="" or e4.get()=="" or e5.get()=="" or e6.get()=="" or e7.get()==""):

messagebox.showerror(parent=b,title="ERROR",message="Please Enter all the details")

else:

try:

sql="insert into import values (%s,%s,%s,%s,%s,%s,%s)"

values=e1.get(),e2.get(),e3.get(),e4.get(),e5.get(),e6.get(),e7.get()

cursor.execute(sql,values)

connection.commit()

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

messagebox.showinfo(parent=b,title="SUCCESS",message="Details inserted successfully")

b.destroy()

except:

messagebox.showerror(parent=b,title="ERROR",message="Couldn't insert the values")

l1=Label(b,text="Billing ID : ",font=["Algerian","18"],bg="lemon chiffon")

l1.place(x=10,y=80)

l2=Label(b,text="Carrier Name : ",font=["Algerian","18"],bg="lemon chiffon")

l2.place(x=10,y=145)

l3=Label(b,text="Origin : ",font=["Algerian","18"],bg="lemon chiffon")

l3.place(x=10,y=215)

l4=Label(b,text="Destination : ",font=["Algerian","18"],bg="lemon chiffon")

l4.place(x=10,y=290)

l5=Label(b,text="Container Dimensions : ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=10,y=355)

l6=Label(b,text="Mode of Transport : ",font=["Algerian","18"],bg="lemon chiffon")

l6.place(x=10,y=425)

l7=Label(b,text="Payment Amount : ",font=["Algerian","18"],bg="lemon chiffon")

l7.place(x=10,y=490)

e1=Entry(b,text="",font='Arial 14',width=14)

e1.place(x=370,y=80)

sql="select max(Billing\_ID) from import"

cursor.execute(sql)

a=cursor.fetchall()

if a[0][0]==None:

c=1000

e1["state"] = "normal"

e1.insert('end',c)

e1["state"] = "disabled"

else:

c=a[0][0]+1

e1["state"] = "normal"

e1.insert('end',c)

e1["state"] = "disabled"

e2=Entry(b,text="",font='Arial 14',width=14)

e2.place(x=370,y=150)

e3=Entry(b,text="",font='Arial 14',width=14)

e3.place(x=370,y=220)

e4=Entry(b,text="",font='Arial 14',width=14)

e4.place(x=370,y=290)

e5=Entry(b,text="",font='Arial 14',width=14)

e5.place(x=370,y=360)

e6=ttk.Combobox(b,font='Arial 14',width=12,state="readonly")

e6['values']=("Seaways","Airways","Roadways")

e6.place(x=370,y=430)

e7=Entry(b,text="",font='Arial 14',width=14)

e7.place(x=370,y=490)

l = Label(b, text = 'INSERT IMPORT DETAILS',font=["Lucida Calligraphy",'20'],bg='brown',fg='snow')

l.place(x=125,y=10)

add\_btn=Button(b,text = 'INSERT',command=submit,bg='gold',bd=1,height=2,width=10)

b.bind('<Return>', lambda i : submit())

add\_btn.place(x=550,y=490)

b.title("INSERT IMPORT DETAILS")

b.geometry("650x550+350+150")

b.resizable(0,0)

b.config(bg='brown')

b.mainloop()

#Updating Import

def update():

messagebox.showinfo(parent=a,title="INFORMATION",message="Enter Billing ID to fetch and update the details")

c=Tk()

def reset():

e1["state"]='normal'

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

def fetch():

sql="select \* from import where Billing\_ID='%s'"%(e1.get())

cursor.execute(sql)

rows=cursor.fetchall()

reset()

if len(rows)!=0:

for row in rows:

e1.insert('end',row[0])

e1["state"]='readonly'

e2.insert('end',row[1])

e3.insert('end',row[2])

e4.insert('end',row[3])

e5.insert('end',row[4])

e6["state"]='normal'

e6.insert('end',row[5])

e6["state"]='readonly'

e7.insert('end',row[6])

else:

messagebox.showerror(parent=c,title="Error",message="No details found with Billing ID you entered")

def submit():

try:

sql='update import set Carrier\_Name="%s",Origin="%s",Destination="%s",Container\_Dimensions="%s",Mode\_of\_Transport="%s",Payment\_Amount="%s" where Billing\_ID="%s"'%(e2.get(),e3.get(),e4.get(),e5.get(),e6.get(),e7.get(),e1.get())

cursor.execute(sql)

e1["state"]='normal'

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

messagebox.showinfo(parent=c,title="SUCCESS",message="Details updated successfully")

connection.commit()

except:

messagebox.showerror(parent=c,title="ERROR",message="Couldn't Update the details")

c.destroy()

l1=Label(c,text="Billing ID : ",font=["Algerian","18"],bg="lemon chiffon")

l1.place(x=10,y=80)

l2=Label(c,text="Carrier Name : ",font=["Algerian","18"],bg="lemon chiffon")

l2.place(x=10,y=145)

l3=Label(c,text="Origin : ",font=["Algerian","18"],bg="lemon chiffon")

l3.place(x=10,y=215)

l4=Label(c,text="Destination : ",font=["Algerian","18"],bg="lemon chiffon")

l4.place(x=10,y=290)

l5=Label(c,text="Container Dimensions : ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=10,y=355)

l6=Label(c,text="Mode of Transport : ",font=["Algerian","18"],bg="lemon chiffon")

l6.place(x=10,y=425)

l7=Label(c,text="Payment Amount : ",font=["Algerian","18"],bg="lemon chiffon")

l7.place(x=10,y=490)

e1=Entry(c,text="",font='Arial 14',width=14)

e1.place(x=370,y=80)

e2=Entry(c,text="",font='Arial 14',width=14)

e2.place(x=370,y=150)

e3=Entry(c,text="",font='Arial 14',width=14)

e3.place(x=370,y=220)

e4=Entry(c,text="",font='Arial 14',width=14)

e4.place(x=370,y=290)

e5=Entry(c,text="",font='Arial 14',width=14)

e5.place(x=370,y=360)

e6=ttk.Combobox(c,font='Arial 14',width=12,state="readonly")

e6['values']=("Seaways","Airways","Roadways")

e6.place(x=370,y=430)

e7=Entry(c,text="",font='Arial 14',width=14)

e7.place(x=370,y=490)

fet\_btn=Button(c,text = 'FETCH',command=fetch,bg='gold',bd=1,height=2,width=10)

fet\_btn.place(x=550,y=440)

upd\_btn=Button(c,text = 'UPDATE',command=submit,bg='gold',bd=1,height=2,width=10)

upd\_btn.place(x=550,y=490)

res=Button(c,text = 'RESET',command=reset,bg='gold',bd=1,height=2,width=10)

res.place(x=550,y=390)

l = Label(c, text = 'UPDATE IMPORT DETAILS',font=["Lucida Calligraphy",'20'],bg='brown',fg='snow')

l.place(x=125,y=10)

c.title("UPDATE IMPORT DETAILS")

c.geometry("650x550+350+150")

c.resizable(0,0)

c.config(bg='brown')

c.mainloop()

#Fetching from Import

def fetch():

d=Tk()

def fetchmany():

sql="select \* from import where %s='%s' order by %s asc"%(c1.get(),e8.get(),c1.get())

cursor.execute(sql)

row=cursor.fetchall()

if len(row)!=0:

tv.delete(\*tv.get\_children())

for i in row:

e8.delete(0,'end')

tv.insert('','end',values=i)

connection.commit()

elif e8.get()=="":

e8.delete(0,'end')

tv.delete(\*tv.get\_children())

messagebox.showerror(parent=d,title="Error",message="Please Enter Billing ID")

else:

e8.delete(0,'end')

tv.delete(\*tv.get\_children())

messagebox.showerror(parent=d,title="Error",message="No details found with "+c1.get()+" you entered")

def fetchall():

tv.delete(\*tv.get\_children())

sql='select \* from import order by Billing\_ID asc'

cursor.execute(sql)

row=cursor.fetchall()

for i in row:

e8.delete(0,'end')

tv.insert('','end',values=i)

connection.commit()

tv=ttk.Treeview(d,columns=("Billing ID","Carrier Name","Origin","Destination","Container Dimensions","Mode of Transport","Payment Amount"),height=24,selectmode='browse')

tv.place(x=20,y=70)

tv.heading("Billing ID",text="Billing ID")

tv.heading("Carrier Name",text="Carrier Name")

tv.heading("Origin",text="Origin")

tv.heading("Destination",text="Destination")

tv.heading("Container Dimensions",text="Container Dimensions")

tv.heading("Mode of Transport",text="Mode of Transport")

tv.heading("Payment Amount",text="Payment Amount")

tv['show']='headings'

tv.column("Billing ID",width=120)

tv.column("Carrier Name",width=140)

tv.column("Origin",width=120)

tv.column("Destination",width=120)

tv.column("Container Dimensions",width=120)

tv.column("Mode of Transport",width=120)

tv.column("Payment Amount",width=120)

l1=Label(d,text="Search by : ",font=["Lucida Calligraphy","14"],bg='gold')

l1.place(x=10,y=15)

c1=ttk.Combobox(d,font='Arial 14',width=12,state="readonly")

c1['values']=("Billing\_ID","Carrier\_Name","Origin","Destination","Container\_Dimensions","Mode\_of\_Transport","Payment\_Amount")

c1.place(x=150,y=15)

s=ttk.Scrollbar(d,orient="vertical",command=tv.yview)

tv.configure(yscroll=s.set)

s.pack(side=RIGHT,fill=Y)

e8=Entry(d,text="",font='Arial 14',width=14)

e8.place(x=320,y=17)

fetchall()

fet\_btn=Button(d,text = 'FETCH',command=fetchmany,bg='gold',bd=1,height=2,width=10)

fet\_btn.place(x=505,y=10)

if e8.get()!=None:

d.bind('<Return>', lambda i : fetchmany())

fa=Button(d,text="FETCH ALL",command=fetchall,bg='gold',bd=1,height=2,width=10)

fa.place(x=600,y=10)

d.title("FETCH IMPORT DETAILS")

d.geometry("920x600+200+100")

d.resizable(0,0)

d.config(bg='brown')

d.mainloop()

#Deleting from Import

def delete():

e=Tk()

def reset():

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

def fetch():

sql="select \* from import where Billing\_ID='%s'"%(e1.get())

cursor.execute(sql)

rows=cursor.fetchall()

reset()

if len(rows)!=0:

for row in rows:

e1.insert('end',row[0])

e2.insert('end',row[1])

e3.insert('end',row[2])

e4.insert('end',row[3])

e5.insert('end',row[4])

e6["state"]='normal'

e6.insert('end',row[5])

e6["state"]='readonly'

e7.insert('end',row[6])

else:

messagebox.showerror(parent=e,title="Error",message="No details found with Billing ID you entered")

def submit():

dele=messagebox.askyesno(parent=e,title="DELETE RECORD CONFIRMATION",message="Would you like to delete the selected Details?")

if dele>0:

sql='delete from import where Billing\_ID="%s"'%(e1.get())

cursor.execute(sql)

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

messagebox.showinfo(parent=e,title="SUCCESS",message="Details Deleted successfully")

connection.commit()

else:

return

e.destroy()

l1=Label(e,text="Billing ID : ",font=["Algerian","18"],bg="lemon chiffon")

l1.place(x=10,y=80)

l2=Label(e,text="Carrier Name : ",font=["Algerian","18"],bg="lemon chiffon")

l2.place(x=10,y=145)

l3=Label(e,text="Origin : ",font=["Algerian","18"],bg="lemon chiffon")

l3.place(x=10,y=215)

l4=Label(e,text="Destination : ",font=["Algerian","18"],bg="lemon chiffon")

l4.place(x=10,y=290)

l5=Label(e,text="Container Dimensions : ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=10,y=355)

l6=Label(e,text="Mode of Transport : ",font=["Algerian","18"],bg="lemon chiffon")

l6.place(x=10,y=425)

l7=Label(e,text="Payment Amount : ",font=["Algerian","18"],bg="lemon chiffon")

l7.place(x=10,y=490)

e1=Entry(e,text="",font='Arial 14',width=14)

e1.place(x=370,y=80)

e2=Entry(e,text="",font='Arial 14',width=14)

e2.place(x=370,y=150)

e3=Entry(e,text="",font='Arial 14',width=14)

e3.place(x=370,y=220)

e4=Entry(e,text="",font='Arial 14',width=14)

e4.place(x=370,y=290)

e5=Entry(e,text="",font='Arial 14',width=14)

e5.place(x=370,y=360)

e6=Entry(e,text="",font='Arial 14',width=14)

e6.place(x=370,y=430)

e7=Entry(e,text="",font='Arial 14',width=14)

e7.place(x=370,y=490)

l = Label(e, text = 'DELETE IMPORT DETAILS',font=["Lucida Calligraphy",'20'],bg='brown',fg='snow')

l.place(x=125,y=10)

fet\_btn=Button(e,text = 'FETCH',command=fetch,bg='gold',bd=1,height=2,width=10)

fet\_btn.place(x=550,y=440)

add\_btn=Button(e,text = 'DELETE',command=submit,bg='gold',bd=1,height=2,width=10)

add\_btn.place(x=550,y=490)

e.title("DELETE IMPORT DETAILS")

e.geometry("650x550+350+150")

e.resizable(0,0)

e.config(bg='brown')

e.mainloop()

def ret():

a.destroy()

l = Label(a, text = 'EDIT IMPORT DETAILS',font=["Lucida Calligraphy",'20'],bg='brown',fg='lemon chiffon')

l.place(x=150,y=50)

ins=Button(a,text = 'Insert Import Details',command=insert,bg='lemon chiffon',bd=1,height=4,width=20)

ins.place(x=10,y=250)

upd=Button(a,text = 'Fetch Import Details',command=fetch,bg='lemon chiffon',bd=1,height=4,width=20)

upd.place(x=170,y=250)

de=Button(a,text = 'Update Import Details',command=update,bg='lemon chiffon',bd=1,height=4,width=20)

de.place(x=330,y=250)

ins=Button(a,text = 'Delete Import Details',command=delete,bg='lemon chiffon',bd=1,height=4,width=20)

ins.place(x=490,y=250)

r=Button(a,text="Return to Main Window",command=ret,bg='brown',fg='lemon chiffon',bd=0,height=1)

r.place(x=500,y=10)

a.title("IMPORT TABLE EDIT PAGE")

a.geometry("650x550+350+150")

a.resizable(0,0)

a.config(bg='brown')

a.mainloop()

#Editing table Export

def exp():

a=Tk()

#Inserting into Export

def insert():

b=Tk()

def submit():

if (e2.get()=="" or e3.get()=="" or e4.get()=="" or e5.get()=="" or e6.get()=="" or e7.get()==""):

messagebox.showerror(parent=b,title="ERROR",message="Please Enter all the details")

else:

try:

sql="insert into export values (%s,%s,%s,%s,%s,%s,%s) "

values=e1.get(),e2.get(),e3.get(),e4.get(),e5.get(),e6.get(),e7.get()

cursor.execute(sql,values)

connection.commit()

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

messagebox.showinfo(parent=b,title="SUCCESS",message="Details inserted successfully")

b.destroy()

except:

messagebox.showerror(parent=b,title="ERROR",message="Couldn't Enter the details")

l1=Label(b,text="Billing ID : ",font=["Algerian","18"],bg="lemon chiffon")

l1.place(x=10,y=80)

l2=Label(b,text="Carrier Name : ",font=["Algerian","18"],bg="lemon chiffon")

l2.place(x=10,y=145)

l3=Label(b,text="Origin : ",font=["Algerian","18"],bg="lemon chiffon")

l3.place(x=10,y=215)

l4=Label(b,text="Destination : ",font=["Algerian","18"],bg="lemon chiffon")

l4.place(x=10,y=290)

l5=Label(b,text="Container Dimensions : ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=10,y=355)

l6=Label(b,text="Mode of Transport : ",font=["Algerian","18"],bg="lemon chiffon")

l6.place(x=10,y=425)

l7=Label(b,text="Payment Amount : ",font=["Algerian","18"],bg="lemon chiffon")

l7.place(x=10,y=490)

e1=Entry(b,text="",font='Arial 14',width=14)

e1.place(x=370,y=80)

sql="select max(Billing\_ID) from export"

cursor.execute(sql)

a=cursor.fetchall()

if a[0][0]==None:

c=1000

e1["state"] = "normal"

e1.insert('end',c)

e1["state"] = "disabled"

else:

c=a[0][0]+1

e1["state"] = "normal"

e1.insert('end',c)

e1["state"] = "disabled"

e2=Entry(b,text="",font='Arial 14',width=14)

e2.place(x=370,y=150)

e3=Entry(b,text="",font='Arial 14',width=14)

e3.place(x=370,y=220)

e4=Entry(b,text="",font='Arial 14',width=14)

e4.place(x=370,y=290)

e5=Entry(b,text="",font='Arial 14',width=14)

e5.place(x=370,y=360)

e6=ttk.Combobox(b,font='Arial 14',width=12,state="readonly")

e6['values']=("Seaways","Airways","Roadways")

e6.place(x=370,y=430)

e7=Entry(b,text="",font='Arial 14',width=14)

e7.place(x=370,y=490)

l = Label(b, text = 'INSERT EXPORT DETAILS',font=["Lucida Calligraphy",'20'],bg='brown',fg='snow')

l.place(x=125,y=10)

add\_btn=Button(b,text = 'INSERT',command=submit,bg='gold',bd=1,height=2,width=10)

b.bind('<Return>', lambda i : submit())

add\_btn.place(x=550,y=490)

b.title("INSERT EXPORT DETAILS")

b.geometry("650x550+350+150")

b.resizable(0,0)

b.config(bg='brown')

b.mainloop()

#Updating Export

def update():

messagebox.showinfo(parent=a,title="INFORMATION",message="Enter Billing ID to fetch and update the details")

c=Tk()

def reset():

e1["state"]='normal'

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

def fetch():

sql="select \* from export where Billing\_ID='%s'"%(e1.get())

cursor.execute(sql)

rows=cursor.fetchall()

reset()

if len(rows)!=0:

for row in rows:

e1.insert('end',row[0])

e1["state"]='readonly'

e2.insert('end',row[1])

e3.insert('end',row[2])

e4.insert('end',row[3])

e5.insert('end',row[4])

e6["state"]='normal'

e6.insert('end',row[5])

e6["state"]='readonly'

e7.insert('end',row[6])

else:

messagebox.showerror(parent=c,title="Error",message="No details found with Billing ID you entered")

def submit():

try:

sql='update export set Carrier\_Name="%s",Origin="%s",Destination="%s",Container\_Dimensions="%s",Mode\_of\_Transport="%s",Payment\_Amount="%s" where Billing\_ID="%s"'%(e2.get(),e3.get(),e4.get(),e5.get(),e6.get(),e7.get(),e1.get())

cursor.execute(sql)

e1["state"]='normal'

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

messagebox.showinfo(parent=c,title="SUCCESS",message="Details updated successfully")

connection.commit()

c.destroy()

except:

messagebox.showerror(parent=c,title="ERROR",message="Couldn't Update the details")

l1=Label(c,text="Billing ID : ",font=["Algerian","18"],bg="lemon chiffon")

l1.place(x=10,y=80)

l2=Label(c,text="Carrier Name : ",font=["Algerian","18"],bg="lemon chiffon")

l2.place(x=10,y=145)

l3=Label(c,text="Origin : ",font=["Algerian","18"],bg="lemon chiffon")

l3.place(x=10,y=215)

l4=Label(c,text="Destination : ",font=["Algerian","18"],bg="lemon chiffon")

l4.place(x=10,y=290)

l5=Label(c,text="Container Dimensions : ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=10,y=355)

l6=Label(c,text="Mode of Transport : ",font=["Algerian","18"],bg="lemon chiffon")

l6.place(x=10,y=425)

l7=Label(c,text="Payment Amount : ",font=["Algerian","18"],bg="lemon chiffon")

l7.place(x=10,y=490)

e1=Entry(c,text="",font='Arial 14',width=14)

e1.place(x=370,y=80)

e2=Entry(c,text="",font='Arial 14',width=14)

e2.place(x=370,y=150)

e3=Entry(c,text="",font='Arial 14',width=14)

e3.place(x=370,y=220)

e4=Entry(c,text="",font='Arial 14',width=14)

e4.place(x=370,y=290)

e5=Entry(c,text="",font='Arial 14',width=14)

e5.place(x=370,y=360)

e6=ttk.Combobox(c,font='Arial 14',width=12,state="readonly")

e6['values']=("Seaways","Airways","Roadways")

e6.place(x=370,y=430)

e7=Entry(c,text="",font='Arial 14',width=14)

e7.place(x=370,y=490)

fet\_btn=Button(c,text = 'FETCH',command=fetch,bg='gold',bd=1,height=2,width=10)

fet\_btn.place(x=550,y=440)

upd\_btn=Button(c,text = 'UPDATE',command=submit,bg='gold',bd=1,height=2,width=10)

upd\_btn.place(x=550,y=490)

res=Button(c,text = 'RESET',command=reset,bg='gold',bd=1,height=2,width=10)

res.place(x=550,y=390)

l = Label(c, text = 'UPDATE EXPORT DETAILS',font=["Lucida Calligraphy",'20'],bg='brown',fg='snow')

l.place(x=125,y=10)

c.title("UPDATE EXPORT DETAILS")

c.geometry("650x550+350+150")

c.resizable(0,0)

c.config(bg='brown')

c.mainloop()

#Fetching from Export

def fetch():

d=Tk()

def fetchmany():

sql="select \* from export where %s='%s' order by %s asc"%(c1.get(),e8.get(),c1.get())

cursor.execute(sql)

row=cursor.fetchall()

if len(row)!=0:

tv.delete(\*tv.get\_children())

for i in row:

tv.insert('','end',values=i)

connection.commit()

e8.delete(0,'end')

elif e8.get()=="":

e8.delete(0,'end')

tv.delete(\*tv.get\_children())

messagebox.showerror(parent=d,title="Error",message="Please Enter Billing ID")

else:

e8.delete(0,'end')

tv.delete(\*tv.get\_children())

messagebox.showerror(parent=d,title="Error",message="No details found with "+c1.get()+" you entered")

def fetchall():

tv.delete(\*tv.get\_children())

sql='select \* from export order by Billing\_ID asc'

cursor.execute(sql)

row=cursor.fetchall()

for i in row:

e8.delete(0,'end')

tv.insert('','end',values=i)

connection.commit()

tv=ttk.Treeview(d,columns=("Billing ID","Carrier Name","Origin","Destination","Container Dimensions","Mode of Transport","Payment Amount"),height=24,selectmode='browse')

tv.place(x=20,y=70)

tv.heading("Billing ID",text="Billing ID")

tv.heading("Carrier Name",text="Carrier Name")

tv.heading("Origin",text="Origin")

tv.heading("Destination",text="Destination")

tv.heading("Container Dimensions",text="Container Dimensions")

tv.heading("Mode of Transport",text="Mode of Transport")

tv.heading("Payment Amount",text="Payment Amount")

tv['show']='headings'

tv.column("Billing ID",width=120)

tv.column("Carrier Name",width=140)

tv.column("Origin",width=120)

tv.column("Destination",width=120)

tv.column("Container Dimensions",width=120)

tv.column("Mode of Transport",width=120)

tv.column("Payment Amount",width=120)

l1=Label(d,text="Search by : ",font=["Lucida Calligraphy","14"],bg='gold')

l1.place(x=10,y=15)

c1=ttk.Combobox(d,font='Arial 14',width=12,state="readonly")

c1['values']=("Billing\_ID","Carrier\_Name","Origin","Destination","Container\_Dimensions","Mode\_of\_Transport","Payment\_Amount")

c1.place(x=150,y=15)

s=ttk.Scrollbar(d,orient="vertical",command=tv.yview)

tv.configure(yscroll=s.set)

s.pack(side=RIGHT,fill=Y)

e8=Entry(d,text="",font='Arial 14',width=14)

e8.place(x=320,y=17)

fetchall()

fet\_btn=Button(d,text = 'FETCH',command=fetchmany,bg='gold',bd=1,height=2,width=10)

fet\_btn.place(x=505,y=10)

if e8.get()!=None:

d.bind('<Return>', lambda i : fetchmany())

fa=Button(d,text="FETCH ALL",command=fetchall,bg='gold',bd=1,height=2,width=10)

fa.place(x=600,y=10)

d.title("FETCH FROM EXPORT")

d.geometry("920x600+200+100")

d.resizable(0,0)

d.config(bg='brown')

d.mainloop()

#Deleting from Export

def delete():

e=Tk()

def reset():

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

def fetch():

sql="select \* from export where Billing\_ID='%s'"%(e1.get())

cursor.execute(sql)

rows=cursor.fetchall()

reset()

if len(rows)!=0:

for row in rows:

e1.insert('end',row[0])

e1["state"]='readonly'

e2.insert('end',row[1])

e3.insert('end',row[2])

e4.insert('end',row[3])

e5.insert('end',row[4])

e6["state"]='normal'

e6.insert('end',row[5])

e6["state"]='readonly'

e7.insert('end',row[6])

else:

messagebox.showerror(parent=e,title="Error",message="No details found with Billing ID you entered")

def submit():

dele=messagebox.askyesno(parent=e,title="DELETE RECORD CONFIRMATION",message="Would you like to delete the selected Details?")

if dele>0:

sql='delete from export where Billing\_ID="%s"'%(e1.get())

cursor.execute(sql)

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6["state"]='normal'

e6.delete(0,'end')

e6["state"]='readonly'

e7.delete(0,'end')

messagebox.showinfo(parent=e,title="SUCCESS",message="Details Deleted successfully")

connection.commit()

else:

return

e.destroy()

l1=Label(e,text="Billing ID : ",font=["Algerian","18"],bg="lemon chiffon")

l1.place(x=10,y=80)

l2=Label(e,text="Carrier Name : ",font=["Algerian","18"],bg="lemon chiffon")

l2.place(x=10,y=145)

l3=Label(e,text="Origin : ",font=["Algerian","18"],bg="lemon chiffon")

l3.place(x=10,y=215)

l4=Label(e,text="Destination : ",font=["Algerian","18"],bg="lemon chiffon")

l4.place(x=10,y=290)

l5=Label(e,text="Container Dimensions: ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=10,y=355)

l6=Label(e,text="Mode of Transport : ",font=["Algerian","18"],bg="lemon chiffon")

l6.place(x=10,y=425)

l7=Label(e,text="Payment Amount : ",font=["Algerian","18"],bg="lemon chiffon")

l7.place(x=10,y=490)

e1=Entry(e,text="",font='Arial 14',width=14)

e1.place(x=370,y=80)

e2=Entry(e,text="",font='Arial 14',width=14)

e2.place(x=370,y=150)

e3=Entry(e,text="",font='Arial 14',width=14)

e3.place(x=370,y=220)

e4=Entry(e,text="",font='Arial 14',width=14)

e4.place(x=370,y=290)

e5=Entry(e,text="",font='Arial 14',width=14)

e5.place(x=370,y=360)

e6=Entry(e,text="",font='Arial 14',width=14)

e6.place(x=370,y=430)

e7=Entry(e,text="",font='Arial 14',width=14)

e7.place(x=370,y=490)

l = Label(e, text = 'DELETE EXPORT DETAILS',font=["Lucida Calligraphy",'20'],fg='snow',bg='brown')

l.place(x=125,y=10)

fet\_btn=Button(e,text = 'FETCH',command=fetch,bg='gold',bd=1,height=2,width=10)

fet\_btn.place(x=550,y=440)

add\_btn=Button(e,text = 'DELETE',command=submit,bg='gold',bd=1,height=2,width=10)

add\_btn.place(x=550,y=490)

e.title("DELETE EXPORT DETAILS")

e.geometry("650x550+350+150")

e.resizable(0,0)

e.config(bg='brown')

e.mainloop()

def ret():

a.destroy()

l = Label(a, text = 'EDIT TABLE EXPORT',font=["Lucida Calligraphy",'20'],bg='brown',fg='lemon chiffon')

l.place(x=170,y=50)

ins=Button(a,text = 'Insert Export Details',command=insert,bg='lemon chiffon',bd=1,height=4,width=20)

ins.place(x=10,y=250)

upd=Button(a,text = 'Fetch Export Details',command=fetch,bg='lemon chiffon',bd=1,height=4,width=20)

upd.place(x=170,y=250)

de=Button(a,text = 'Update Export Details',command=update,bg='lemon chiffon',bd=1,height=4,width=20)

de.place(x=330,y=250)

ins=Button(a,text = 'Delete Export Details',command=delete,bg='lemon chiffon',bd=1,height=4,width=20)

ins.place(x=490,y=250)

r=Button(a,text="Return to Main Window",command=ret,bg='brown',fg='lemon chiffon',bd=0,height=1)

r.place(x=500,y=10)

a.title("EXPORT TABLE EDIT PAGE")

a.geometry("650x550+350+150")

a.resizable(0,0)

a.config(bg='brown')

a.mainloop()

#Editing table Employee

def emp():

a=Tk()

#Inserting New Employee

def insert():

b=Tk()

def submit():

if (e2.get()=="" or e3.get()=="" or e4.get()=="" or e5.get()=="" or e6.get()=="" ):

messagebox.showerror(parent=b,title="ERROR",message="Please Enter all the details")

else:

try:

sql="insert into employee values (%s,%s,%s,%s,%s,%s) "

values=e1.get(),e2.get(),e3.get(),e4.get(),e5.get(),e6.get()

cursor.execute(sql,values)

connection.commit()

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6.delete(0,'end')

messagebox.showinfo(parent=b,title="SUCCESS",message="New Details inserted successfully")

b.destroy()

except:

messagebox.showerror(parent=b,title="ERROR",message="Couldn't insert the values")

l1=Label(b,text="Employee ID : ",font=["Algerian","18"],bg="lemon chiffon")

l1.place(x=15,y=80)

l2=Label(b,text="Employee FirstName : ",font=["Algerian","18"],bg="lemon chiffon")

l2.place(x=15,y=145)

l3=Label(b,text="Employee LastName",font=["Algerian","18"],bg="lemon chiffon")

l3.place(x=15,y=215)

l4=Label(b,text="Date Of Birth(YYYY-MM-DD) : ",font=["Algerian","18"],bg="lemon chiffon")

l4.place(x=15,y=290)

l5=Label(b,text="Employee Designation : ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=15,y=355)

l5=Label(b,text="Employee Phone Number : ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=15,y=420)

e1=Entry(b,text="",font='Arial 14',width=14)

e1.place(x=380,y=80)

e2=Entry(b,text="",font='Arial 14',width=14)

e2.place(x=380,y=150)

e3=Entry(b,text="",font='Arial 14',width=14)

e3.place(x=380,y=220)

sql="select count(\*) from employee"

cursor.execute(sql)

a=cursor.fetchall()

if a[0][0]=="None":

c=1000

e1["state"] = "normal"

e1.insert('end',c)

e1["state"] = "disabled"

else:

c=a[0][0]+1000

e1["state"] = "normal"

e1.insert('end',c)

e1["state"] = "disabled"

e4=Entry(b,text="",font='Arial 14',width=14)

e4.place(x=380,y=290)

e5=Entry(b,text="",font='Arial 14',width=14)

e5.place(x=380,y=360)

e6=Entry(b,text="",font='Arial 14',width=14)

e6.place(x=380,y=425)

l = Label(b, text = 'CREATE A NEW EMPLOYEE',font=["Lucida Calligraphy",'20'],bg='brown',fg='snow')

l.place(x=125,y=10)

add\_btn=Button(b,text = 'INSERT',command=submit,bg='gold',bd=1,height=2,width=10)

b.bind('<Return>', lambda i : submit())

add\_btn.place(x=450,y=490)

b.title("CREATE NEW EMPLOYEE")

b.geometry("650x550+350+150")

b.resizable(0,0)

b.config(bg='brown')

b.mainloop()

#Updating Employee Details

def update():

messagebox.showinfo(parent=a,title="INFORMATION",message="Enter Employee ID to fetch and update the details")

c=Tk()

def reset():

e1["state"]='normal'

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6.delete(0,'end')

def fetch():

sql="select \* from employee where Employee\_ID='%s'"%(e1.get())

cursor.execute(sql)

rows=cursor.fetchall()

reset()

if len(rows)!=0:

for row in rows:

e1.insert('end',row[0])

e1["state"]='readonly'

e2.insert('end',row[1])

e3.insert('end',row[2])

e4.insert('end',row[3])

e5.insert('end',row[4])

e6.insert('end',row[5])

else:

messagebox.showerror(parent=c,title="Error",message="No details found with Employee ID you entered")

def submit():

try:

sql='update employee set Employee\_FName="%s",Employee\_LName="%s",Date\_of\_Birth="%s",Designation="%s",Employee\_PhNo="%s" where Employee\_ID="%s"'%(e2.get(),e3.get(),e4.get(),e5.get(),e6.get(),e1.get())

cursor.execute(sql)

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6.delete(0,'end')

messagebox.showinfo(parent=c,title="SUCCESS",message="Details updated successfully")

connection.commit()

except:

messagebox.showerror(parent=c,title="ERROR",message="Couldn't Update the details")

c.destroy()

l1=Label(c,text="Employee ID : ",font=["Algerian","18"],bg="lemon chiffon")

l1.place(x=15,y=80)

l2=Label(c,text="Employee FirstName : ",font=["Algerian","18"],bg="lemon chiffon")

l2.place(x=15,y=145)

l3=Label(c,text="Employee LastName : ",font=["Algerian","18"],bg="lemon chiffon")

l3.place(x=15,y=215)

l4=Label(c,text="Date Of Birth(YYYY-MM-DD) : ",font=["Algerian","18"],bg="lemon chiffon")

l4.place(x=15,y=290)

l5=Label(c,text="Designation : ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=15,y=355)

l6=Label(c,text="Employee PhNo : ",font=["Algerian","18"],bg="lemon chiffon")

l6.place(x=15,y=420)

e1=Entry(c,text="",font='Arial 14',width=14)

e1.place(x=380,y=80)

e2=Entry(c,text="",font='Arial 14',width=14)

e2.place(x=380,y=150)

e3=Entry(c,text="",font='Arial 14',width=14)

e3.place(x=380,y=220)

e4=Entry(c,text="",font='Arial 14',width=14)

e4.place(x=380,y=290)

e5=Entry(c,text="",font='Arial 14',width=14)

e5.place(x=380,y=360)

e6=Entry(c,text="",font='Arial 14',width=14)

e6.place(x=380,y=425)

fet\_btn=Button(c,text = 'FETCH',command=fetch,bg='gold',bd=1,height=2,width=10)

fet\_btn.place(x=550,y=365)

upd\_btn=Button(c,text = 'UPDATE',command=submit,bg='gold',bd=1,height=2,width=10)

upd\_btn.place(x=550,y=465)

res=Button(c,text = 'RESET',command=reset,bg='gold',bd=1,height=2,width=10)

res.place(x=550,y=415)

l = Label(c, text = 'UPDATE DETAILS OF AN EMPLOYEE',font=["Lucida Calligraphy",'20'],bg='brown',fg='snow')

l.place(x=50,y=10)

c.title("UPDATE EMPLOYEE DETAILS")

c.geometry("650x550+350+150")

c.resizable(0,0)

c.config(bg='brown')

c.mainloop()

#Fetching Employee Details

def fetch():

d=Tk()

def fetchmany():

sql="select \* from employee where %s='%s' order by %s asc"%(c1.get(),e8.get(),c1.get())

cursor.execute(sql)

row=cursor.fetchall()

if len(row)!=0:

tv.delete(\*tv.get\_children())

for i in row:

e8.delete(0,'end')

tv.insert('','end',values=i)

connection.commit()

elif e8.get()=="":

e8.delete(0,'end')

tv.delete(\*tv.get\_children())

messagebox.showerror(parent=d,title="Error",message="Please Enter Billing ID")

else:

e8.delete(0,'end')

tv.delete(\*tv.get\_children())

messagebox.showerror(parent=d,title="Error",message="No details found with "+c1.get()+" you entered")

def fetchall():

tv.delete(\*tv.get\_children())

sql='select \* from Employee order by Employee\_ID asc'

cursor.execute(sql)

row=cursor.fetchall()

for i in row:

e8.delete(0,'end')

tv.insert('','end',values=i)

connection.commit()

tv=ttk.Treeview(d,columns=("Employee ID","Employee FirstName","Employee LastName","Date Of Birth","Designation","Employee PhNo"),height=24,selectmode='browse')

tv.place(x=25,y=70)

tv.heading("Employee ID",text="Employee ID")

tv.heading("Employee FirstName",text="Employee FirstName")

tv.heading("Employee LastName",text="Employee LastName")

tv.heading("Date Of Birth",text="Date Of Birth")

tv.heading("Designation",text="Designation")

tv.heading("Employee PhNo",text="Employee PhNo")

tv['show']='headings'

tv.column("Employee ID",width=140)

tv.column("Employee FirstName",width=140)

tv.column("Employee LastName",width=140)

tv.column("Date Of Birth",width=140)

tv.column("Designation",width=140)

tv.column("Employee PhNo",width=140)

l1=Label(d,text="Search by : ",font=["Lucida Calligraphy","14"],bg='gold')

l1.place(x=10,y=15)

c1=ttk.Combobox(d,font='Arial 14',width=12,state="readonly")

c1['values']=("Employee\_ID","Employee\_FName","Employee\_LName","Date\_of\_Birth","Designation","Employee\_PhNo","Year(Date\_of\_Birth)","Month(Date\_of\_Birth)")

c1.place(x=150,y=15)

s=ttk.Scrollbar(d,orient="vertical",command=tv.yview)

tv.configure(yscroll=s.set)

s.pack(side=RIGHT,fill=Y)

e8=Entry(d,text="",font='Arial 14',width=14)

e8.place(x=320,y=17)

fetchall()

fet\_btn=Button(d,text = 'FETCH',command=fetchmany,bg='gold',bd=1,height=2,width=10)

fet\_btn.place(x=505,y=10)

if e8.get()!=None:

d.bind('<Return>', lambda i : fetchmany())

fa=Button(d,text="FETCH ALL",command=fetchall,bg='gold',bd=1,height=2,width=10)

fa.place(x=600,y=10)

d.title("FETCH FROM EMPLOYEE")

d.geometry("920x600+200+100")

d.resizable(0,0)

d.config(bg='brown')

d.mainloop()

#Removing an Employee

def delete():

e=Tk()

def reset():

e1["state"]='normal'

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6.delete(0,'end')

def fetch():

sql="select \* from employee where Employee\_ID='%s'"%(e1.get())

cursor.execute(sql)

rows=cursor.fetchall()

reset()

if len(rows)!=0:

for row in rows:

e1.insert('end',row[0])

e1["state"]='readonly'

e2.insert('end',row[1])

e3.insert('end',row[2])

e4.insert('end',row[3])

e5.insert('end',row[4])

e6.insert('end',row[5])

else:

messagebox.showerror(parent=e,title="Error",message="No details found with Employee ID you entered")

def submit():

dele=messagebox.askyesno(parent=e,title="DELETE RECORD CONFIRMATION",message="Would you like to delete the selected Details?")

if dele>0:

sql='delete from employee where Employee\_ID="%s"'%(e1.get())

cursor.execute(sql)

e1.delete(0,'end')

e2.delete(0,'end')

e3.delete(0,'end')

e4.delete(0,'end')

e5.delete(0,'end')

e6.delete(0,'end')

messagebox.showinfo(parent=e,title="SUCCESS",message="Details Deleted successfully")

connection.commit()

else:

return

e.destroy()

l1=Label(e,text="Employee ID : ",font=["Algerian","18"],bg="lemon chiffon")

l1.place(x=15,y=80)

l2=Label(e,text="Employee FirstName : ",font=["Algerian","18"],bg="lemon chiffon")

l2.place(x=15,y=145)

l3=Label(e,text="Employee LastName : ",font=["Algerian","18"],bg="lemon chiffon")

l3.place(x=15,y=215)

l4=Label(e,text="Date Of Birth(YYYY-MM-DD) : ",font=["Algerian","18"],bg="lemon chiffon")

l4.place(x=15,y=290)

l5=Label(e,text="Designation : ",font=["Algerian","18"],bg="lemon chiffon")

l5.place(x=15,y=355)

l6=Label(e,text="Employee PhNo : ",font=["Algerian","18"],bg="lemon chiffon")

l6.place(x=15,y=420)

e1=Entry(e,text="",font='Arial 14',width=14)

e1.place(x=380,y=80)

e2=Entry(e,text="",font='Arial 14',width=14)

e2.place(x=380,y=150)

e3=Entry(e,text="",font='Arial 14',width=14)

e3.place(x=380,y=220)

e4=Entry(e,text="",font='Arial 14',width=14)

e4.place(x=380,y=290)

e5=Entry(e,text="",font='Arial 14',width=14)

e5.place(x=380,y=360)

e6=Entry(e,text="",font='Arial 14',width=14)

e6.place(x=380,y=425)

l = Label(e, text = 'REMOVE AN EMPLOYEE',font=["Lucida Calligraphy",'20'],bg='brown',fg='snow')

l.place(x=125,y=10)

fet\_btn=Button(e,text = 'FETCH',command=fetch,bg='gold',bd=1,height=2,width=10)

fet\_btn.place(x=550,y=420)

add\_btn=Button(e,text = 'REMOVE',command=submit,bg='gold',bd=1,height=2,width=10)

add\_btn.place(x=550,y=470)

e.title("REMOVE AN EMLOYEE")

e.geometry("650x550+350+150")

e.resizable(0,0)

e.config(bg='brown')

e.mainloop()

def ret():

a.destroy()

l = Label(a, text = 'EDIT TABLE EMPLOYEE',font=["Lucida Calligraphy",'20'],bg='brown',fg='lemon chiffon')

l.place(x=150,y=50)

ins=Button(a,text = 'Insert Employee Details',command=insert,bg='lemon chiffon',bd=1,height=4,width=20)

ins.place(x=10,y=250)

upd=Button(a,text = 'Fetch Employee Details',command=fetch,bg='lemon chiffon',bd=1,height=4,width=20)

upd.place(x=170,y=250)

de=Button(a,text = 'Update Employee Details',command=update,bg='lemon chiffon',bd=1,height=4,width=20)

de.place(x=330,y=250)

ins=Button(a,text = 'Remove an Employee',command=delete,bg='lemon chiffon',bd=1,height=4,width=20)

ins.place(x=490,y=250)

r=Button(a,text="Return to Main Window",command=ret,bg='brown',fg='lemon chiffon',bd=0,height=1)

r.place(x=500,y=10)

a.title("EMPLOYEE TABLE EDIT PAGE")

a.geometry("650x550+350+150")

a.resizable(0,0)

a.config(bg='brown')

a.mainloop()

def logout():

dele=messagebox.askyesno(parent=win,title="LOG OUT CONFIRMATION",message="Are you sure you want to logout?")

if dele>0:

win.destroy()

else:

return

#Label and Buttons for second window

l = Label(win, text = 'CHOOSE YOUR TABLE HERE',font=["Lucida Calligraphy",'20'],bg='brown',fg='lemon chiffon')

l.place(x=100,y=50)

imp=Button(win,text = 'IMPORT',command=imp,bg='lemon chiffon',bd=1,height=4,width=20)

imp.place(x=250,y=150)

exp=Button(win,text = 'EXPORT',command=exp,bg='lemon chiffon',bd=1,height=4,width=20)

exp.place(x=250,y=250)

emp=Button(win,text = 'EMPLOYEE',command=emp,bg='lemon chiffon',bd=1,height=4,width=20)

emp.place(x=250,y=350)

v=Label(win,text="CMS version 1.1",bg="brown",font=(None,12))

v.place(x=520,y=520)

lo=Button(win,text="Logout",command=logout,bg='brown',fg='lemon chiffon',font="Arial 14 italic",bd=0,height=1)

lo.place(x=560,y=10)

win.title("CARGO MANAGEMENT SYSTEM EDIT PAGE")

win.geometry("650x550+350+150")

win.resizable(0,0)

win.config(bg='brown')

win.mainloop()

elif (name\_entry.get() =="") and (passw\_entry.get() ==""):

messagebox.showerror("Error","Please Enter your Username and Password")

elif (name\_entry.get() =="") :

messagebox.showerror("Error","Please Enter your Username")

elif (passw\_entry.get() =="") :

messagebox.showerror("Error","Please Enter your Password")

elif (name\_entry.get() !=i[0]) and (passw\_entry.get() !=i[1]):

messagebox.showerror("Error","The Username and Password you Entered is Incorrect")

name\_entry.delete(0,'end')

passw\_entry.delete(0,'end')

elif (passw\_entry.get() !=i[1]) :

passw\_entry.delete(0,'end')

messagebox.showerror("Error","The Password you Entered is Incorrect")

elif name\_entry.get() != i[0]:

messagebox.showerror("Error","The Username you Entered does not Exist")

name\_entry.delete(0,'end')

#About this software

def info():

at=Tk()

#Opening and reading a text file

a=open("info.txt","r")

c=a.read()

a.close()

info\_lbl=Label(at,text="About this Software : ",font=['Ink Free','20','bold'],bg='brown',fg='gold')

info\_lbl.place(x=10,y=10)

con=Text(at,font=['Ink Free','14','bold'],bg='brown',fg='snow',height=14,width=40,bd=0)

con.insert('end',c)

s=ttk.Scrollbar(at,orient="vertical",command=con.yview)

con.configure(yscroll=s.set)

s.pack(side=RIGHT,fill=Y)

con.place(x=10,y=60)

at.title("About this Software")

at.geometry("530x400+275+175")

at.resizable(0, 0)

at.config(bg='brown')

at.mainloop()

#Function for Forgot Password

def forpas():

at=Tk()

def submit():

if (npn.get()==i[2]):

messagebox.showinfo("Information","Your Password is "+i[1])

at.destroy()

elif (npn.get()) =="":

messagebox.showerror("Error","Please enter your Pin Number to fetch Password")

elif (npn.get()) !=i[2] :

messagebox.showerror("Error","The Pin Number you entered is incorrect")

at.destroy()

def info():

messagebox.showinfo(parent=at,title="Information",message="Please enter the pin number you entered while registering yourself")

wl=Label(at, text = 'Verify Your Pin Number',font='18',fg='lemon chiffon',bg='brown')

wl.place(x=110,y=30)

np\_label = Label(at, text = 'Pin Number : ',font='14',bg='lemon chiffon')

np\_label.place(x=50,y=125)

npn = Entry(at,text = "",show = '\*',font='Arial 14',width=14)

npn.place(x=195,y=125)

sub\_btn=Button(at,text = 'Verify Pin',command=submit,bg='lemon chiffon',bd=1)

at.bind('<Return>', lambda i : submit())

sub\_btn.place(x=250,y=200)

c=Button(at,text=" ? ",command=info,bg='lemon chiffon',bd=1)

c.place(x=360,y=127)

at.title("FORGOT PASSWORD")

at.geometry("400x300+400+200")

at.resizable(0, 0)

at.config(bg='brown')

at.mainloop()

#Defining Labels and Buttons for Login page

welcome\_label = Label(window, text ='Cargo Management System',font=['Ink Free','22','bold'],bg='brown',fg="lemon chiffon",width=38)

welcome\_label.place(x=5,y=40)

name\_label = Label(window, text = 'Username : ',font="Arial 16 bold italic",fg='lemon chiffon',bg='brown')

name\_label.place(x=70,y=320)

name\_entry = Entry(window,text='',font="Arial 15",width=14,justify=LEFT)

name\_entry.place(x=200,y=325)

passw\_label = Label(window,text = 'Password : ',font="Arial 16 bold italic",fg='lemon chiffon',bg='brown')

passw\_label.place(x=70,y=360)

passw\_entry=Entry(window,text='',show = '\*',font="Arial 15",width=14,justify=LEFT)

passw\_entry.place(x=200,y=365)

login\_btn=Button(window,text = 'Login Here',command=login,bg='lemon chiffon',bd=1,height=2,width=14)

login\_btn.place(x=475,y=320)

for\_btn=Button(window,text = 'Forgot Password',command=forpas,bg='lemon chiffon',bd=1,height=2,width=14)

for\_btn.place(x=475,y=270)

reg=Button(window,text='Register here',command=register,bg='lemon chiffon',bd=1,height=2,width=14)

reg.place(x=475,y=220)

if len(i)!=4:

login\_btn["state"]="disabled"

for\_btn["state"]="disabled"

if len(i)==4:

reg["state"]="disabled"

for\_btn["state"]="normal"

welcome\_label.config(text=i[3]+"\nCargo Management System")

v=Label(window,text="CMS version 1.1",bg="brown",font=(None,12))

v.place(x=520,y=520)

info=Button(window,text='Click here to view\nabout this software',command=info,bg='brown',fg="lemon chiffon",bd=0)

info.place(x=20,y=500)

if len(i)==4:

window.bind('<Return>', lambda i : login())

elif len(i)!=4:

window.bind('<Return>', lambda i : register())

#Inserting an Image

b=PhotoImage(file='E:\\My file 1\\CMS Project Files\\images.png')

lbl=Label(window,image=b)

lbl.place(x=140,y=150)

window.title("WELCOME TO CARGO MANAGEMENT SYSTEM")

window.geometry("650x550+350+150")

window.resizable(0,0)

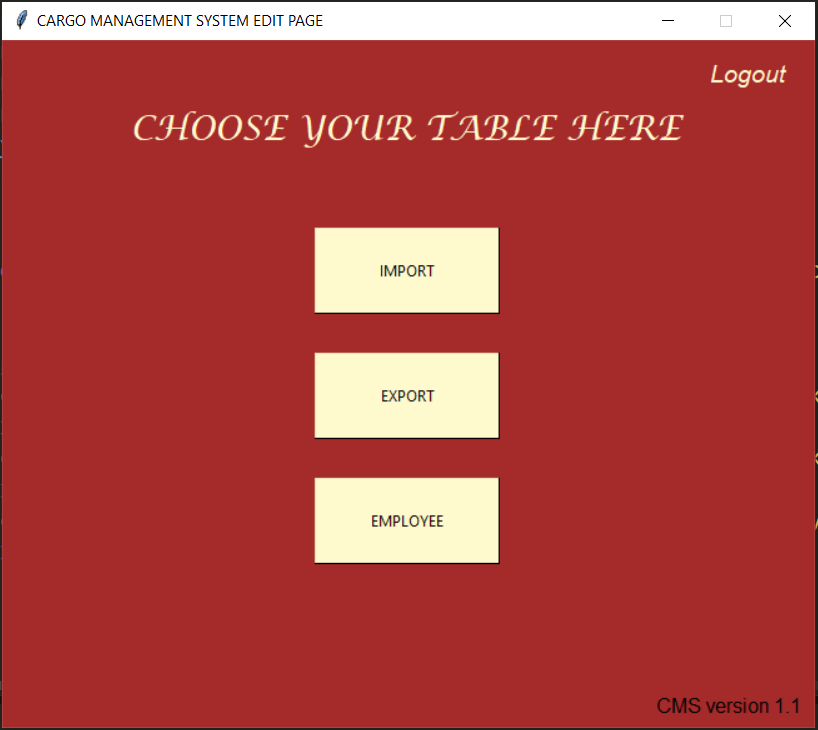
window.config(bg='brown')

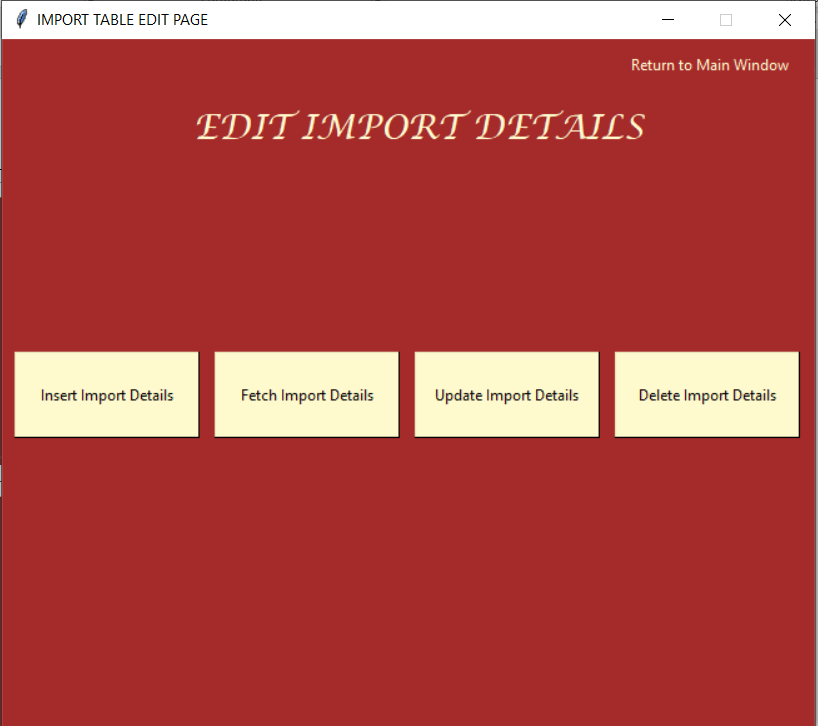
window.mainloop()

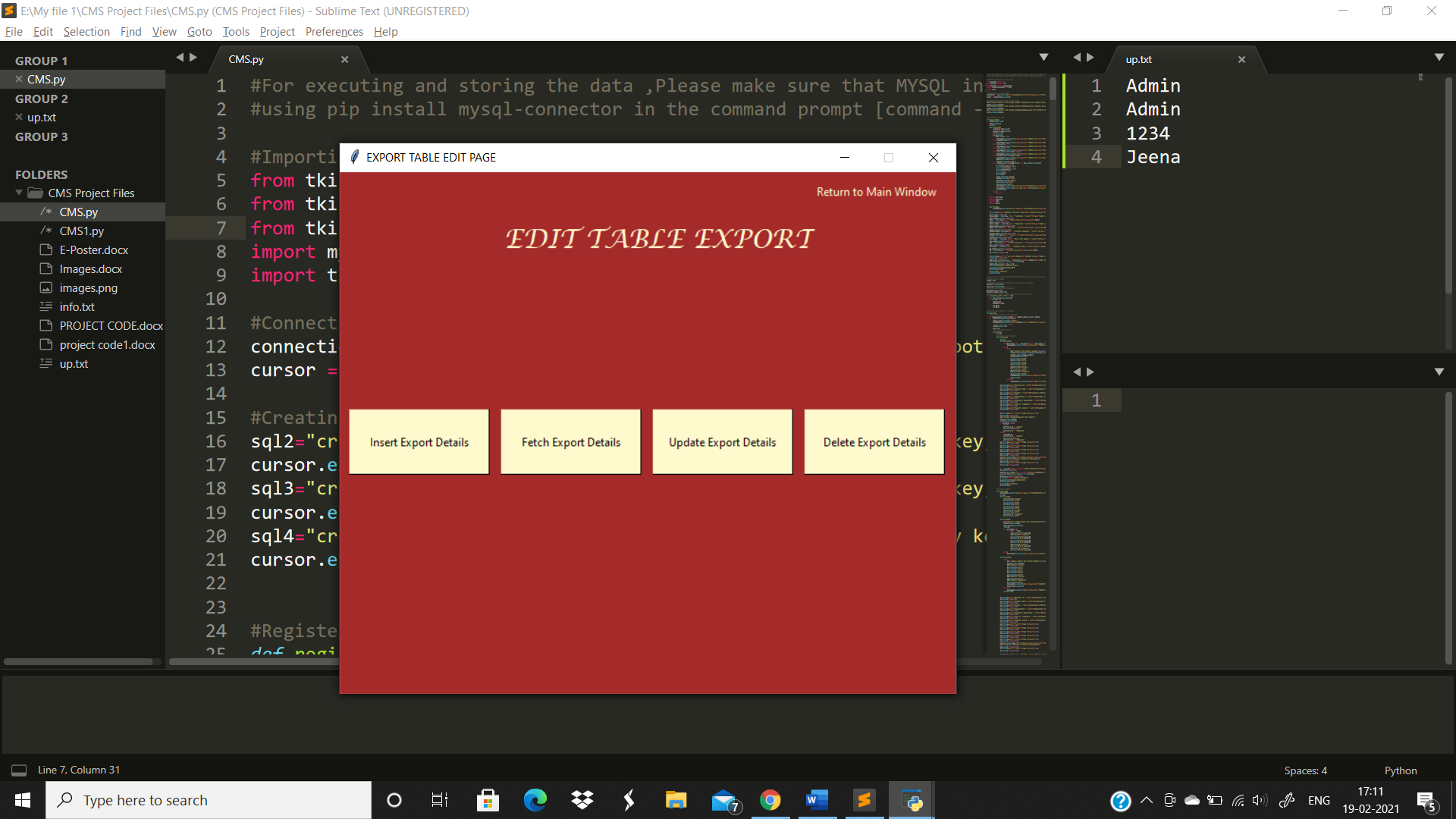
#=================PROGRAM OVER THANK YOU================#

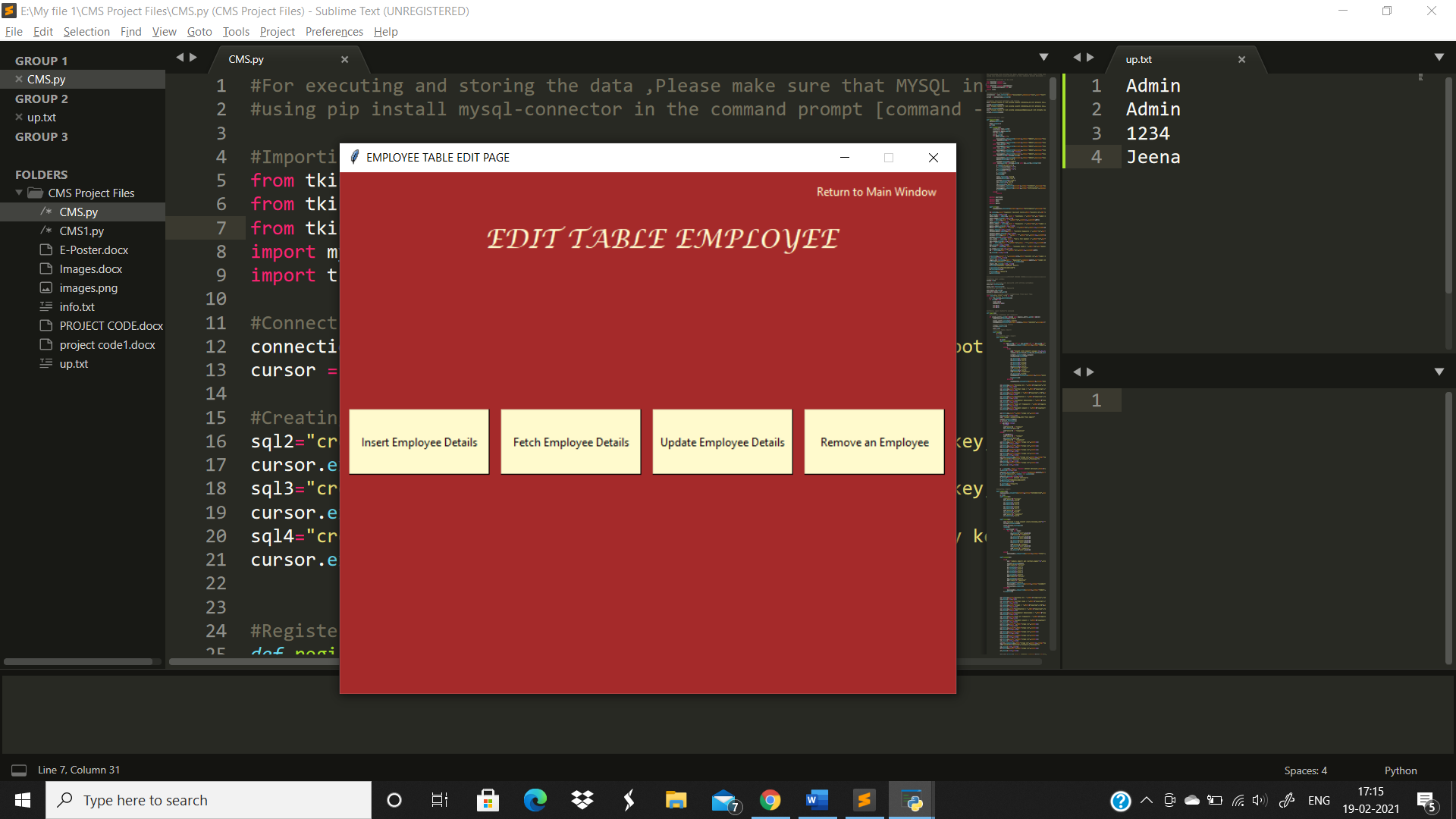
OUTPUTS

 1.Login Page

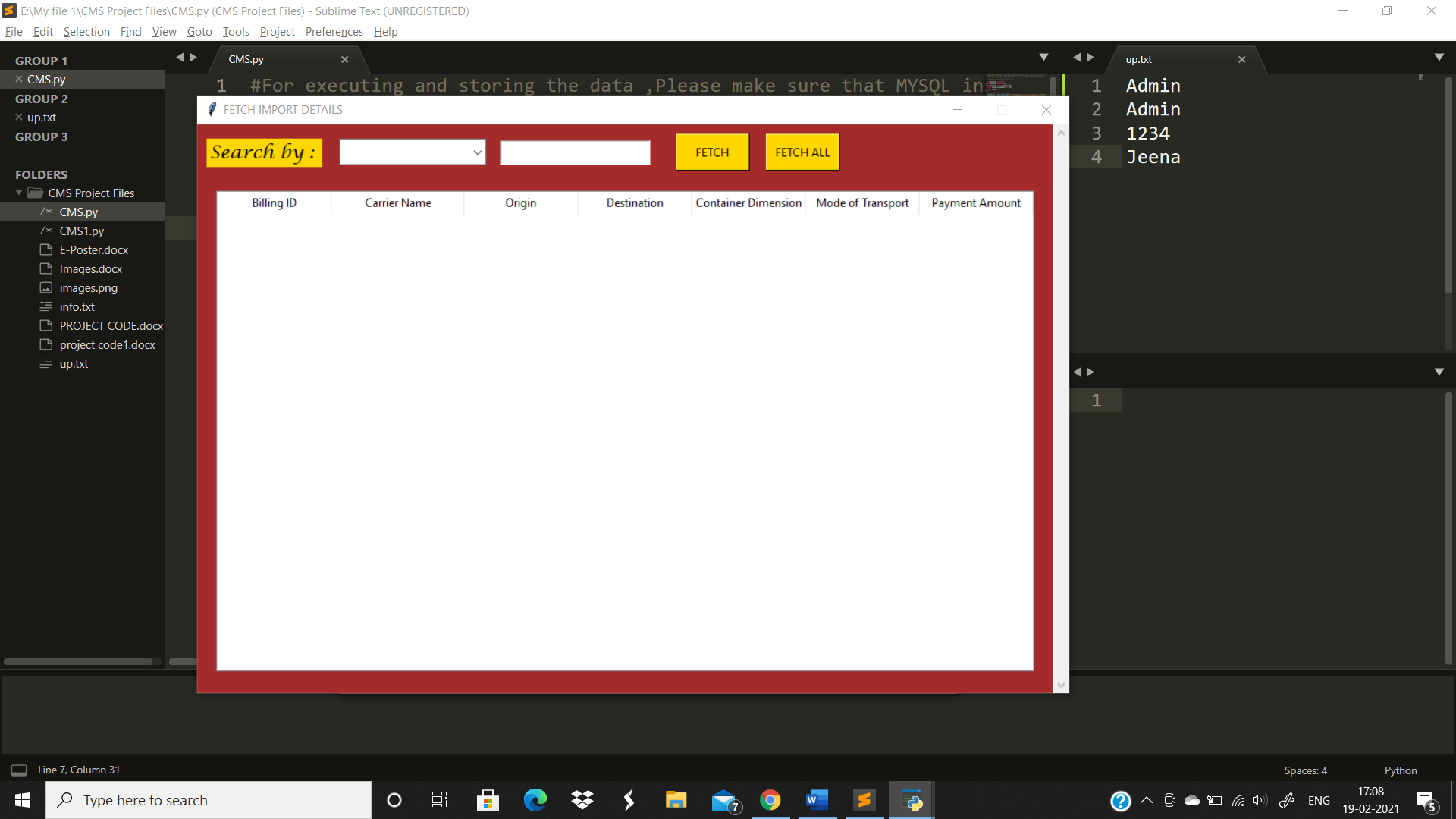
 2.Home Page

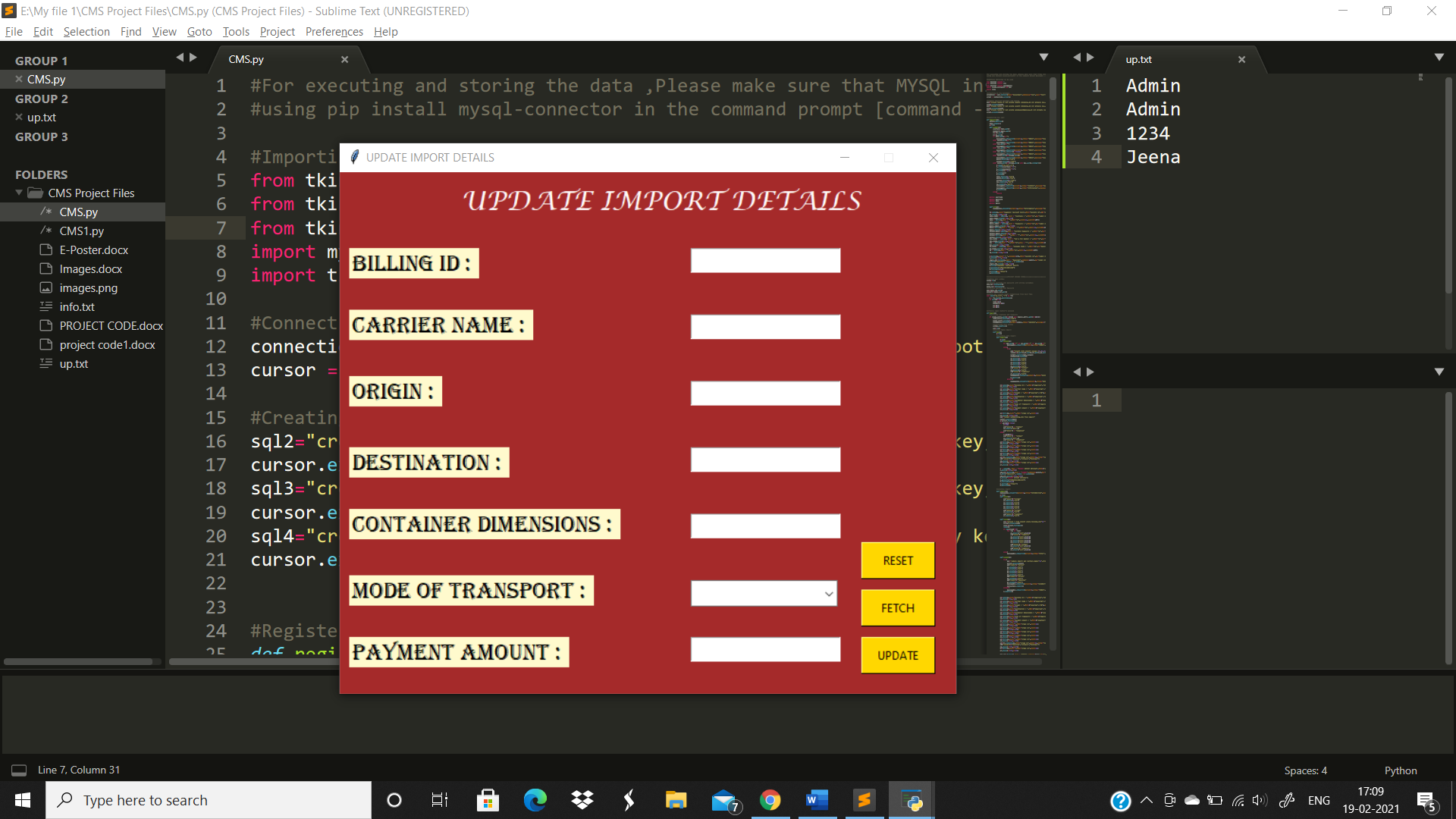
 3.Import Page

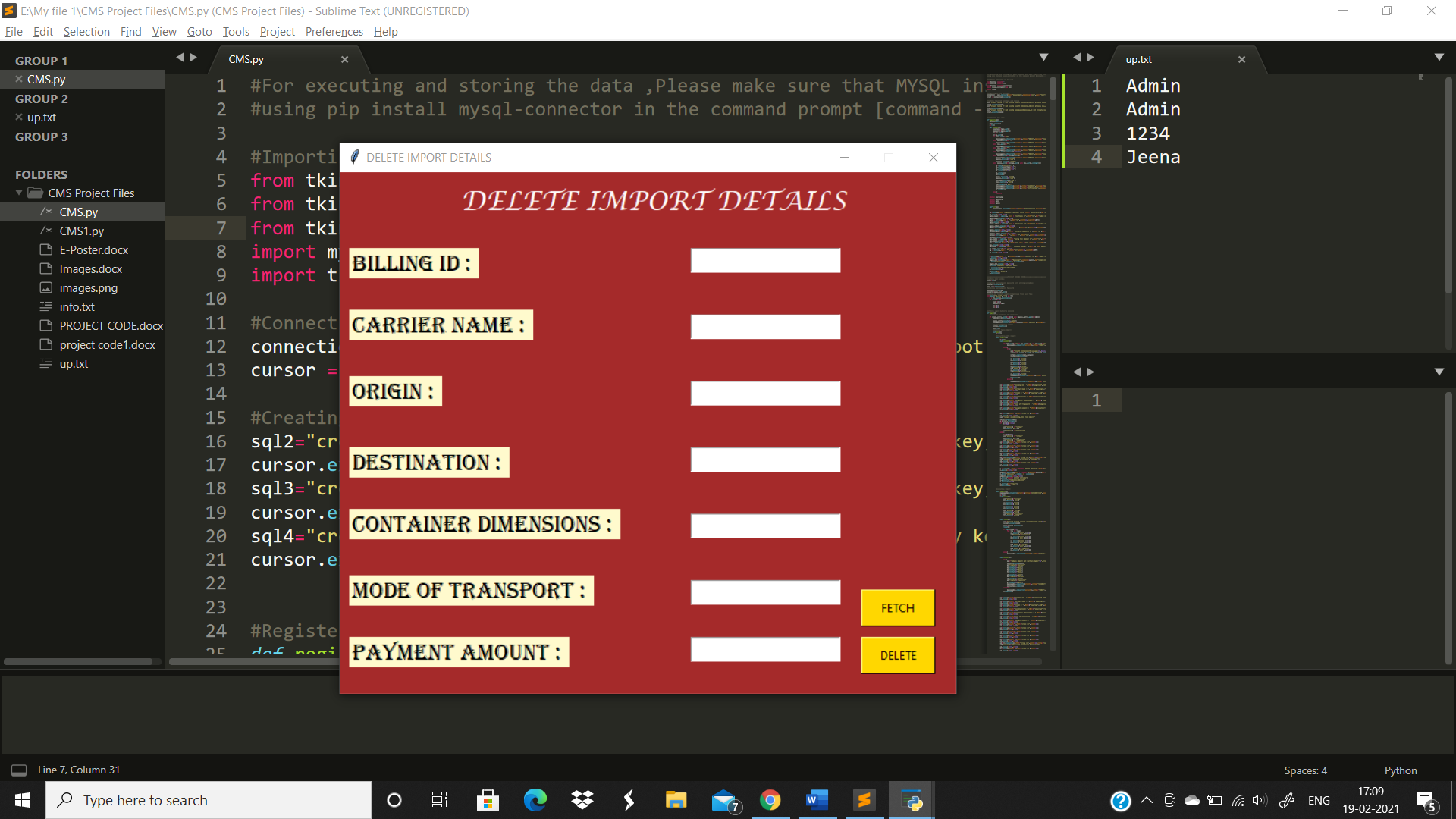
 4.Export Page

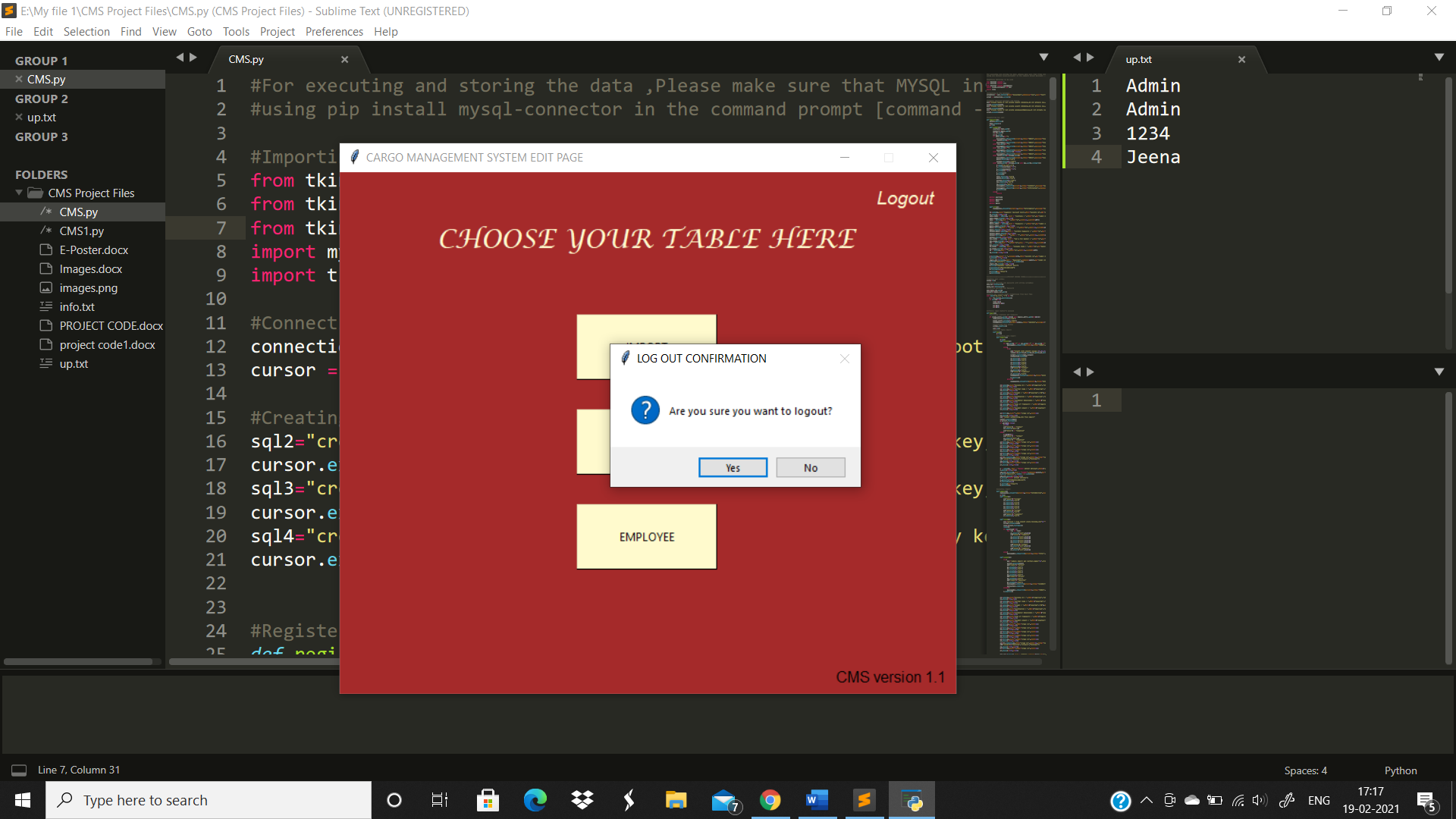
 5.Employee Page

 6.Insert Page

 7.Fetch Page

 8.Update Page

 9.Delete Page

 10.Logout Page

*Future Enhancements*

In future enhancements, features like show password, Login History, Change user details and other features will be added.

Show password is used here so that user can see what he/she has typed. But this should be used only when needed as there are chances that someone may see your password and access the data.

Login History is used here so that user can see if someone else has tried to login this software. This software will store the username and password entered everytime.

If the user wants to change his username password or pin number, this software will let you do so. This can be done by entering the pre-existing pin number which will take the user to a new window to make changes in his personal details.

Also, if the user is not going to use his account anymore, this software lets him/her delete the account along with their data.

*BIBLIOGRAPHY*

Python tutorial:

* <https://www.realpython.com>

Interfacing MySQL with Python:

* <https://www.dev.mysql.com>