```
from tkinter import *

from tkinter.ttk import *

from sqlite3 import *

from tkinter import messagebox as tkMessageBox

from tkinter.ttk import Combobox

import sqlite3

con=sqlite3.Connection('bus_database')

root=Tk()

root.title("BUS BOOKING SYSTEM")

root.geometry('1000x450')

photo = PhotoImage(file ="Bus.png")

Label(root,text="Bus Booking System",image = photo).place(x=50,y=10)

def anime(m,n,c,y='black'):

    style = Style()

    style.configure('TButton', font = ('calibri', m, 'bold'), borderwidth = str(n))

    style.map('TButton', foreground = [('active', '!disabled',c )],background = [('active',y)])

anime(20,4,'green')




def show_search_result(li,f,t):

    global root3

    root3=Tk()

    root3.title("BUS BOOKING SYSTEM")

    root3.geometry('1400x1000')

    photo = PhotoImage(file ="Bus.png")

    Label(root3,text="Bus Booking System",image = photo).place(x=220,y=10)

    Label(root3,text="Buses Available From-"+f+" To-"+t,font="Arial 20 bold 
underline").place(x=420,y=350)

    Label(root3,text="BUS ID",font="Arial 13 bold ").place(x=20+40,y=430)

    Label(root3,text="OPERATOR",font="Arial 13 bold ").place(x=233-50,y=430)
Label(root3,text="TYPE",font="Arial 13 bold ").place(x=316+30,y=430)

Label(root3,text="FROM",font="Arial 13 bold").place(x=439+40,y=430)

Label(root3,text="TO",font="Arial 13 bold").place(x=566+20,y=430)

Label(root3,text="DATE(YYYY/MM/DD)",font="Arial 13 bold").place(x=625+30,y=430)

Label(root3,text="DEPT. TIME",font="Arial 13 bold").place(x=808+30,y=430)

Label(root3,text="ARR. TIME",font="Arial 13 bold").place(x=931+25,y=430)

Label(root3,text="SEATS",font="Arial 13 bold").place(x=1054+20,y=430)

Label(root3,text="FARE(₹) ",font="Arial 13 bold").place(x=1177-20,y=430)

Label(root3,text="SELECT",font="Arial 13 bold").place(x=1300-40,y=430)



y=430

global i

#displaying matched records details

v=IntVar()

for i in range(0,len(li)):

   y=y+30

   Label(root3,text=li[i][0],font="Arial 12 bold").place(x=130-50,y=y)

   Label(root3,text=li[i][1],font="Arial 12 bold").place(x=220-50,y=y)

   Label(root3,text=li[i][2],font="Arial 12 bold").place(x=350-50,y=y)

   Label(root3,text=li[i][3],font="Arial 12 bold").place(x=530-50,y=y)

   Label(root3,text=li[i][4],font="Arial 12 bold").place(x=630-50,y=y)

   Label(root3,text=li[i][5],font="Arial 12 bold").place(x=740-50,y=y)

   Label(root3,text=li[i][6],font="Arial 12 bold").place(x=910-50,y=y)

   Label(root3,text=li[i][7],font="Arial 12 bold").place(x=1020-50,y=y)

   Label(root3,text=li[i][8],font="Arial 12 bold").place(x=1140-50,y=y)

   Label(root3,text=li[i][9],font="Arial 12 bold").place(x=1222-50,y=y)


   Radiobutton(root3,variable=v,value=1+int(i)).place(x=1290,y=y)


for i in range(0,len(li)):
        if(v.get()):
            break
    def book_seats(a,y):
        for i in range(0,len(li)):
            if(a.get()):
                break

        bus_id=a.get()-1

        seats=li[bus_id][8]
        seat_list=[0]*int(seats)

        j=0
        global dic
        dic={}
        Label(root3,text="Seats Selected").place(x=1120,y=y-120)
        total_seats=Entry(root3)
        total_seats.place(x=1200,y=y-120)

        while(j!=seats):
            y=y+10
            if(j%2==0):
                Checkbutton(root3,text=j+1,variable =
dic.update({j+1:0}),onvalue=1).place(x=1200,y=y+100)
            elif(j%2==1):
                Checkbutton(root3,text=j+1,variable = dic.update({j+1:0}),onvalue=1).place(x=1250,y=y+90)
            j=j+1
        Button(root3,text="CANCEL",command=lambda:root3.destroy()).place(x=1215,y=195)
        def add_passenger_details(ts,bus_no,y):
```

```
if(total_seats.get()==0 or total_seats.get()==''):
   tkMessageBox.showerror("OOPS","please Enter Number of Seats")




ts=total_seats.get()

root3.destroy()
i=int(int(bus_no.get())-1)





global root4
root4=Tk()
root4.geometry('420x300')
Label(root4,text="Passenger Details",font="Arial 17 bold underline").place(x=110,y=1)
Label(root4,text="Name",font="Arial 12 bold").place(x=85,y=35)
name=Entry(root4)
name.place(x=175,y=35)
Label(root4,text="Age",font="Arial 12 bold").place(x=85,y=70)
age=Entry(root4)
age.place(x=175,y=70)
Label(root4,text="Gender",font="Arial 12 bold").place(x=85,y=105)
gender=Combobox(root4,height=5,width=17,values=["Male","Female","Others"])
gender.place(x=175,y=105)
Label(root4,text="Phone_No",font="Arial 12 bold").place(x=85,y=140)
phone=Entry(root4)
phone.place(x=175,y=140)
Label(root4,text="Email_ID",font="Arial 12 bold").place(x=85,y=175)
email=Entry(root4)
```

```
    email.place(x=175,y=175)




    def confirm_passenger_info(ts):



      a=name.get()

      b=age.get()

      c=gender.get()

      contact=phone.get()

      mail=email.get()

      d=f     #fromCity

      e=t     #toCity

      date=li[i][4]

      price=li[i][9]

      time=li[i][5]

      left=int(li[i][8])-int(ts)


      x=(i+1,a,b,c,contact,mail,d,e,date,time,price)


      cur=con.cursor()


      if(a=='' or b=='' or c==''):

        tkMessageBox.showerror("OOPS","you can't leave any field empty")

      else:

        if(c!=d):

          #cur.execute("create table if not exists passenger_details(Booking_Id integer primary
key,Passenger_Name varchar(20),Age int(3),Gender varchar(8),Contact_Number int(10),SEATS
int(3),Email_Id varchar(40),From_City char(20),To_City char(20),Date char(12),Time char(7),Price
int(100000))")
```

```
                    cur.execute("create table if not exists passenger_details(Bus_Number integer,Booking_Id integer primary key,Passenger_Name varchar(20),Age int(3),Gender varchar(8),Contact_Number int(10),Email_Id varchar(40),From_City char(20),To_City char(20),Date char(12),Time char(7),Price int(100000))")
                    cur.execute("update bus_records set Number_Of_Seats=? where bus_id=?",(left,li[i][0]))
                    #cur.execute("CREATE TRIGGER IF NOT EXISTS seat_update AFTER INSERT ON PASSENGER_DETAILS FOR EACH ROW BEGIN UPDATE  BUS_RECORDS SET bus_records.Number_Of_Seats=bus_records.Number_Of_Seats-passenger_details.seats WHERE bus_records.From_City=passenger_details.from_city AND bus_records.To_City=passenger_details.to_city; END;")
                    cur.execute("insert into passenger_details(bus_number,Passenger_Name,Age,Gender,Contact_Number,Email_Id,From_City,To_City,Date,Time,Price) values(?,?,?,?,?,?,?,?,?,?,?)",x)


                    cur.execute("select * from passenger_details where Passenger_Name=? and Contact_Number=? and Age=?",(a,contact,b))
                    tkMessageBox.showinfo("Congrats!!","your seat has been booked")
                    con.commit()
                    detail=cur.fetchall()


                    cur.execute("update bus_records set number_of_seats=? where bus_id=?",(left,i+1))
                    tkMessageBox.showinfo("Details To Remember","Booking ID:"+str(detail[0][1])+"\nPassenger Name:"+str(detail[0][2])+"\nBoarding_City:"+str(detail[0][7])+"\nDeparture_Time:"+str(detail[0][9])+"\n\nHAVE A SAFE JOURNEY!!")
                    root4.destroy()
            root4.mainloop()
        anime(15,3,'Green','green')
        Button(root4,text="Cancel",command=lambda:root4.destroy()).place(x=50,y=230)
        Button(root4,text="Book Seat",command=lambda:confirm_passenger_info(ts)).place(x=220,y=230)
    anime(12,3,'Green','green')
    Button(root3,text="Submit & Proceed",command=lambda:add_passenger_details(total_seats,a,y)).place(x=1200,y=y-400)


  Button(root3,text="BOOK",command=lambda:book_seats(v,y)).place(x=1200,y=y+50)
```

```
    root3.mainloop()



def add_bus():
    root.destroy()
    root1=Tk()
    root1.title("BUS BOOKING SYSTEM")
    root1.geometry('1000x850')
    #root1.minsize(1220,500)
    photo = PhotoImage(file ="Bus.png")
    Label(root1,image = photo).place(x=50,y=10)
    Label(root1,text=" Enter The Details Of Bus Operator",font="Arial 20 bold 
underline").place(x=300,y=325)
    Label(root1,text="Full Name",font="Arial 11 bold ").place(x=390,y=380)
    name=Entry(root1)
    name.place(x=500,y=380)
    Label(root1,text="Contact",font="Arial 11 bold ").place(x=390,y=410)
    contact=Entry(root1)
    contact.place(x=500,y=410)
    Label(root1,text="Address",font="Arial 11 bold ").place(x=390,y=440)
    add=Entry(root1)
    add.place(x=500,y=440)
    def add_operator_details():
        Label(root1,text="Agency Name",font="Arial 11 bold ").place(x=390,y=520)
        agency=Entry(root1)
        agency.place(x=500,y=520)
        Label(root1,text="Bus Type",font="Arial 11 bold ").place(x=390,y=550)
        bus_type=Combobox(root1,height=5,width=17,values=['NON A/C Sleeper ','NON A/C 
Seater','A/C Sleeper','A/C Seater'])
        bus_type.place(x=500,y=550)
        Label(root1,text="From",font="Arial 11 bold ").place(x=390,y=580)
```

```
start=Combobox(root1,height=5,width=17,values=['Delhi','Agra','Gwalior','Indore','Jhansi','Gujarat','J
aipur','Kolkata','Kashmir','Lucknow','Udaipur','Kota','Guna','Mumbai','Hyderabad'])
        start.place(x=500,y=580)
        Label(root1,text="To",font="Arial 11 bold ").place(x=390,y=610)

stop=Combobox(root1,height=5,width=17,values=['Delhi','Agra','Gwalior','Indore','Jhansi','Gujarat','J
aipur','Kolkata','Kashmir','Lucknow','Udaipur','Kota','Guna','Mumbai','Hyderabad'])
        stop.place(x=500,y=610)
        Label(root1,text="Date(dd/mm/yyyy)",font="Arial 11 bold ").place(x=370,y=640)
        date=Combobox(root1,height=5,width=2,values=list(range(1,32)))
        date.place(x=500,y=640)
        month=Combobox(root1,height=5,width=2,values=list(range(1,13)))
        month.place(x=540,y=640)
        year=Combobox(root1,height=5,width=4,values=list(range(2020,3000)))
        year.place(x=580,y=640)
        Label(root1,text="Departure-Time",font="Arial 11 bold ").place(x=370,y=670)
        departure=Combobox(root1,height=5,width=7,values=['1','2','3','4','5','6','7','8','9','10','11','12'])
        departure.place(x=500,y=670)
        w1=Combobox(root1,height=5,width=5,values=['AM',"PM"])
        w1.place(x=570,y=670)
        Label(root1,text="Arrival-Time",font="Arial 11 bold ").place(x=370,y=700)
        arrival=Combobox(root1,height=5,width=7,values=['1','2','3','4','5','6','7','8','9','10','11','12'])
        arrival.place(x=500,y=700)
        w2=Combobox(root1,height=5,width=5,values=['AM',"PM"])
        w2.place(x=570,y=700)
        Label(root1,text="Seats",font="Arial 11 bold ").place(x=390,y=730)
        seats=Entry(root1)
        seats.place(x=500,y=730)
        Label(root1,text="Fare",font="Arial 11 bold ").place(x=390,y=760)
        rates=Entry(root1)
        rates.place(x=500,y=760)
```

```
def submit_details():
    a=agency.get()
    b=bus_type.get()
    c=start.get()
    d=stop.get()
    e=str(year.get())+"/"+str(month.get())+"/"+str(date.get())
    f=departure.get()+w1.get()
    g=arrival.get()+w2.get()
    h=(seats.get())
    i=(rates.get())
    x=(a,b,c,d,e,f,g,h,i)
    cur=con.cursor()


    if(a=='' or b=='' or c=='' or d=='' or e=='' or f=='' or g=='' or h=='' or i==''):
        tkMessageBox.showerror("OOPS","you can't leave any field empty")
    else:
        if(c!=d):
            cur.execute("create table if not exists bus_records(Bus_id integer primary key,Agency char(20),Bus_Type char(20),From_City char(20),To_City char(20),Date Date,Departure_Time char(5),Arrival_Time char(5),Number_Of_Seats int(100),Ticket_Price int(100000))")
            cur.execute("insert into bus_records(Agency,Bus_Type,From_City,To_City,Date,Departure_Time,Arrival_Time,Number_Of_Seats,Ticket_Price) values(?,?,?,?,?,?,?,?,?)",x)
            tkMessageBox.showinfo("congrats","your bus record is now in our database")
            con.commit()
            root1.destroy()
        else:
            tkMessageBox.showerror("Error","you can't choose same city")
    #root1.destroy()
anime(15,3,'blue','red')
Button(root1,text="Save And Continue",command=submit_details).place(x=650,y=640)
```

```
    Button(root1,text="Add Details",command=add_operator_details).place(x=460,y=480)
    root1.mainloop()



def search_bus():
    root.destroy()
    global root2
    root2=Tk()
    root2.title("BUS BOOKING SYSTEM")
    root2.geometry('910x500')
    photo = PhotoImage(file ="Bus.png")
    Label(root2,text="Bus Booking System",image = photo).place(x=10,y=10)
    Label(root2,text="  Bus Search Menu  ",font="Arial 20 bold underline").place(x=350,y=325)

    Label(root2,text="Bus Type",font="Arial 11 bold ").place(x=350,y=365)
    w3=Combobox(root2,height=5,width=19,values=['NON A/C Sleeper ','NON A/C Seater ','A/C
Sleeper ','A/C Seater','All Types'])
    w3.place(x=485,y=365)

    Label(root2,text="Select Boarding",font="Arial 11 bold ").place(x=350,y=400)

w1=Combobox(root2,height=5,width=19,values=['Delhi','Agra','Gwalior','Indore','Jhansi','Gujarat','Jai
pur','Kolkata','Kashmir','Lucknow','Udaipur','Kota','Guna','Mumbai','Hyderabad'])
    w1.place(x=485,y=400)

    Label(root2,text="Select Destination",font="Arial 11 bold ").place(x=350,y=435)

w2=Combobox(root2,height=5,width=19,values=['Delhi','Agra','Gwalior','Indore','Jhansi','Gujarat','Jai
pur','Kolkata','Kashmir','Lucknow','Udaipur','Kota','Guna','Mumbai','Hyderabad'])
    w2.place(x=485,y=435)
    def search():
```

```
    a=w1.get()
    b=w2.get()
    c=w3.get()
    cur=con.cursor()
    if a=='' or b=='':
      tkMessageBox.showerror("Error","Cant leave any field empty")

    else:
       if a!=b:
         cur.execute(''' SELECT count(name) FROM sqlite_master WHERE type='table' AND 
name='bus_records' ''')

         if cur.fetchone()[0]!=1 :
           tkMessageBox.showerror("Sorry", "Database is Empty\nNo DATA Available In Our 
Records")
           tkMessageBox.showerror("TO USE SEARCH BUS FEAUTRE", "Insert Records Of The Bus 
Operator In Database")
         elif(c=='All Types'):
           cur.execute("select count(*) from bus_records where From_City =? and To_City=?",(a,b))
           if cur.fetchone()[0]==0 :
             tkMessageBox.showerror("RESULT","No,("+c+") Bus Found For Destination-"+a+" To-
"+b)

           else:
             cur.execute("select * from bus_records where From_City =? and To_City=? and 
Bus_Type=?",(a,b,c))
             con.commit()
             e=cur.fetchall()
             root2.destroy()
             show_search_result(e,a,b)
         else:
           cur.execute("select count(*) from bus_records where From_City =? and To_City=? and 
Bus_Type=?",(a,b,c))
```

```
                if cur.fetchone()[0]==0 :
                    tkMessageBox.showerror("RESULT","No,("+c+") Bus Found For Destination-"+a+" To-
"+b)

                else:
                    cur.execute("select * from bus_records where From_City =? and To_City=? and
Bus_Type=?",(a,b,c))
                    con.commit()
                    e=cur.fetchall()
                    root2.destroy()
                    show_search_result(e,a,b)

            else:
               tkMessageBox.showerror("Oops","boarding and destination can't be the same")
    anime(20,4,'blue','blue')
    Button(root2,text="SEARCH",command=search).place(x=650,y=385)
    Button(root2,text="CLOSE",command=lambda:root2.destroy()).place(x=100,y=385)
    root2.mainloop()



Button(root,text="Add Bus",command=add_bus).place(x=350,y=320)
Button(root,text="Search Bus",command=search_bus).place(x=550,y=320)
root.mainloop()
```