from tkinter import \*  
from tkinter import ttk  
from PIL import Image, ImageTk  
import mysql.connector  
from tkinter import messagebox  
  
  
class Criminal:  
 def \_\_init\_\_(self, root):  
 self.root = root  
 self.root.geometry("1300x790+0+0")  
 self.root.title('CRIMINAL MANAGENET SYSTEM SOFTWARE')  
  
 # Variables  
 self.var\_case\_id = StringVar()  
 self.var\_criminal\_no = StringVar()  
 self.var\_name = StringVar()  
 self.var\_nickname = StringVar()  
 self.var\_arrest\_date = StringVar()  
 self.var\_date\_of\_crime = StringVar()  
 self.var\_address = StringVar()  
 self.var\_age = StringVar()  
 self.var\_occupation = StringVar()  
 self.var\_birthMark = StringVar()  
 self.var\_crime\_type = StringVar()  
 self.var\_father\_name = StringVar()  
 self.var\_gender = StringVar()  
 self.var\_wanted = StringVar()  
  
 lbl\_title = Label(self.root, text="CRIMINAL MANAGEMENT SYSTEM SOFTWARE", font=("times new roman", 40, "bold"),  
 bg="white", fg="blue")  
 lbl\_title.place(x=0, y=0, width=1300, height=70)  
  
  
 img\_frame=Frame(self.root,bd=2,relief=RIDGE,bg='white')  
 img\_frame.place(x=0,y=80,width=1550,height=200)  
  
 #1st  
 img\_logo = Image.open('police1.jpg')  
 img\_logo = img\_logo.resize((490, 160), Image.ANTIALIAS)  
 self.photo\_logo = ImageTk.PhotoImage(img\_logo)  
  
 self.logo = Label(img\_frame, image=self.photo\_logo)  
 self.logo.place(x=0, y=0, width=490, height=160)  
  
 #2nd  
 img\_logo2 = Image.open('police2.jpg')  
 img\_logo2 = img\_logo2.resize((490, 160), Image.ANTIALIAS)  
 self.photo\_logo2 = ImageTk.PhotoImage(img\_logo2)  
  
 self.logo2 = Label(img\_frame, image=self.photo\_logo2)  
 self.logo2.place(x=500, y=0, width=490, height=160)  
  
 #3rd  
  
 img\_logo3 = Image.open('police3.jpg')  
 img\_logo3 = img\_logo3.resize((490, 160), Image.ANTIALIAS)  
 self.photo\_logo3 = ImageTk.PhotoImage(img\_logo3)  
  
 self.logo3 = Label(img\_frame, image=self.photo\_logo3)  
 self.logo3.place(x=1000, y=0, width=280, height=160)  
  
  
  
 # Main Frame  
 Main\_frame = Frame(self.root, bd=2, relief=RIDGE, bg='white')  
 Main\_frame.place(x=10, y=200, width=1500, height=560)  
  
 # upper Frame  
 upper\_frame = LabelFrame(Main\_frame, bd=2, relief=RIDGE, bg='white', text='Criminal Information',  
 font=('times new roman', 11, 'bold'), fg='red')  
 upper\_frame.place(x=10, y=10, width=1480, height=270)  
  
 # Labels and Entry fields  
 caseid = Label(upper\_frame, text='Case ID:', font=('arial', 11, 'bold'), bg='white')  
 caseid.grid(row=0, column=0, padx=2, sticky=W)  
  
 caseentry = ttk.Entry(upper\_frame, textvariable=self.var\_case\_id, width=22, font=("arial", 11, "bold"))  
 caseentry.grid(row=0, column=1, padx=2, pady=10, sticky=W)  
  
 # Criminal NO  
 lbl\_criminal\_no = Label(upper\_frame, font=("arial", 12, "bold"), text="Criminal NO:", bg="white")  
 lbl\_criminal\_no.grid(row=0, column=2, sticky=W, padx=2, pady=7)  
  
 txt\_criminal\_no = ttk.Entry(upper\_frame, textvariable=self.var\_criminal\_no, width=22,  
 font=("arial", 11, "bold"))  
 txt\_criminal\_no.grid(row=0, column=3, padx=2, pady=7)  
  
 # Criminal Name  
 lbl\_Name = Label(upper\_frame, font=("arial", 12, "bold"), text="Criminal Name:", bg="white")  
 lbl\_Name.grid(row=1, column=0, sticky=W, padx=2, pady=7)  
  
 txt\_Name = ttk.Entry(upper\_frame, textvariable=self.var\_name, width=22, font=("arial", 11, "bold"))  
 txt\_Name.grid(row=1, column=1, sticky=W, padx=2, pady=7)  
  
 # NickName  
 lbl\_nickname = Label(upper\_frame, font=("arial", 12, "bold"), text="NickName:", bg="white")  
 lbl\_nickname.grid(row=1, column=2, sticky=W, padx=2, pady=7)  
  
 txt\_nickname = ttk.Entry(upper\_frame, textvariable=self.var\_nickname, width=22, font=("arial", 11, "bold"))  
 txt\_nickname.grid(row=1, column=3, padx=2, pady=7)  
  
 # Arrest Date  
 lbl\_arrestDate = Label(upper\_frame, font=("arial", 12, "bold"), text="Arrest Date:", bg="white")  
 lbl\_arrestDate.grid(row=2, column=0, sticky=W, padx=2, pady=7)  
  
 txt\_arrestDate = ttk.Entry(upper\_frame, textvariable=self.var\_arrest\_date, width=22, font=("arial", 11, "bold"))  
 txt\_arrestDate.grid(row=2, column=1, padx=2, pady=7)  
  
 # Date of Crime  
 lbl\_dateOfCrime = Label(upper\_frame, font=("arial", 12, "bold"), text="Date of Crime:", bg="white")  
 lbl\_dateOfCrime.grid(row=2, column=2, sticky=W, padx=2, pady=7)  
  
 txt\_dateOfCrime = ttk.Entry(upper\_frame, textvariable=self.var\_date\_of\_crime, width=22,  
 font=("arial", 11, "bold"))  
 txt\_dateOfCrime.grid(row=2, column=3, sticky=W, padx=2, pady=7)  
  
 # Address  
 lbl\_address = Label(upper\_frame, font=("arial", 12, "bold"), text="Address:", bg="white")  
 lbl\_address.grid(row=3, column=0, sticky=W, padx=2, pady=7)  
  
 txt\_address = ttk.Entry(upper\_frame, textvariable=self.var\_address, width=22, font=("arial", 11, "bold"))  
 txt\_address.grid(row=3, column=1, padx=2, pady=7)  
  
 # Age  
 lbl\_age = Label(upper\_frame, font=("arial", 12, "bold"), text="Age:", bg="white")  
 lbl\_age.grid(row=3, column=2, sticky=W, padx=2, pady=7)  
  
 txt\_age = ttk.Entry(upper\_frame, textvariable=self.var\_age, width=22, font=("arial", 11, "bold"))  
 txt\_age.grid(row=3, column=3, padx=2, pady=7)  
  
 # occupution  
 lbl\_occupution = Label(upper\_frame, font=("arial", 12, "bold"), text="Occupation:", bg="white")  
 lbl\_occupution.grid(row=4, column=0, sticky=W, padx=2, pady=7)  
  
 txt\_occupution = ttk.Entry(upper\_frame, textvariable=self.var\_occupation, width=22, font=("arial", 11, "bold"))  
 txt\_occupution.grid(row=4, column=1, padx=2, pady=7)  
  
 # birthMark  
 lbl\_birthMark = Label(upper\_frame, font=("arial", 12, "bold"), text="Birth Mark:", bg="white")  
 lbl\_birthMark.grid(row=4, column=2, sticky=W, padx=2, pady=7)  
  
 txt\_birthmark = ttk.Entry(upper\_frame, textvariable=self.var\_birthMark, width=22, font=("arial", 11, "bold"))  
 txt\_birthmark.grid(row=4, column=3, sticky=W, padx=2, pady=7)  
  
 # Crime Type  
 lbl\_crimeType = Label(upper\_frame, font=("arial", 12, "bold"), text="Crime Type:", bg="white")  
 lbl\_crimeType.grid(row=0, column=4, sticky=W, padx=2, pady=7)  
  
 txt\_crimeType = ttk.Entry(upper\_frame, textvariable=self.var\_crime\_type, width=22, font=("arial", 11, "bold"))  
 txt\_crimeType.grid(row=0, column=5, padx=2, pady=7)  
  
 # # Father Name  
 lbl\_fatherName = Label(upper\_frame, font=("arial", 12, "bold"), text="Father Name:", bg="white")  
 lbl\_fatherName.grid(row=1, column=4, sticky=W, padx=2, pady=7)  
  
 txt\_fatherName = ttk.Entry(upper\_frame, textvariable=self.var\_father\_name, width=22, font=("arial", 11, "bold"))  
 txt\_fatherName.grid(row=1, column=5, padx=2, pady=7)  
  
 # Radio Frame  
 radio\_frame\_gender = Frame(upper\_frame, bd=2, relief=RIDGE, bg='white')  
 radio\_frame\_gender.place(x=730, y=90, width=190, height=30)  
  
 lbl\_gender = Label(upper\_frame, font=("arial", 12, "bold"), text="Gender:", bg="white")  
 lbl\_gender.grid(row=2, column=4, sticky=W, padx=2, pady=7)  
  
 male = Radiobutton(radio\_frame\_gender, variable=self.var\_gender, text='Male', value='male',  
 font=('arial', 9, 'bold'), bg='white')  
 male.grid(row=2, column=5, pady=2, padx=5, sticky=W)  
 self.var\_gender.set('male')  
  
 female = Radiobutton(radio\_frame\_gender, variable=self.var\_gender, text='Female', value='female',  
 font=('arial', 9, 'bold'), bg='white')  
 female.grid(row=2, column=6, pady=2, padx=5, sticky=W)  
  
 # Most wantes  
  
 lbl\_wanted = Label(upper\_frame, font=("arial", 12, "bold"), text="Most Wanted:", bg="white")  
 lbl\_wanted.grid(row=3, column=4, sticky=W, padx=2, pady=7)  
  
 radio\_frame\_wanted = Frame(upper\_frame, bd=2, relief=RIDGE, bg='white')  
 radio\_frame\_wanted.place(x=730, y=130, width=190, height=30)  
  
 yes = Radiobutton(radio\_frame\_wanted, variable=self.var\_wanted, text='Yes', value='yes',  
 font=('arial', 9, 'bold'), bg='white')  
 yes.grid(row=2, column=5, pady=2, padx=5, sticky=W)  
 self.var\_wanted.set('yes')  
  
 no = Radiobutton(radio\_frame\_wanted, variable=self.var\_wanted, text='No', value='no', font=('arial', 9, 'bold'),  
 bg='white')  
 no.grid(row=2, column=6, pady=2, padx=5, sticky=W)  
  
 # Button Frame  
 button\_frame = Frame(upper\_frame, bd=2, relief=RIDGE, bg='white')  
 button\_frame.place(x=5, y=200, width=620, height=45)  
  
 btn\_add = Button(button\_frame, text="Record Save", command=self.add\_data, font=("arial", 13, "bold"), width=14,  
 bg='blue', fg='white')  
 btn\_add.grid(row=0, column=0, padx=3, pady=5)  
  
 btn\_update = Button(button\_frame, text="Update", command=self.update\_data, font=("arial", 13, "bold"), width=14,  
 bg='blue', fg='white')  
 btn\_update.grid(row=0, column=1, padx=1, pady=5)  
  
 btn\_delete = Button(button\_frame, text="Delete", command=self.delete\_data, font=("arial", 13, "bold"), width=14,  
 bg='blue', fg='white')  
 btn\_delete.grid(row=0, column=2, padx=1, pady=5)  
  
 btn\_clear = Button(button\_frame, text="Clear", command=self.reset\_data, font=("arial", 13, "bold"), width=14,  
 bg='blue', fg='white')  
 btn\_clear.grid(row=0, column=3, padx=1, pady=5)  
  
 img\_logo4 = Image.open('criminal1.jpg')  
 img\_logo4 = img\_logo4.resize((220, 220), Image.ANTIALIAS)  
 self.photo\_logo4 = ImageTk.PhotoImage(img\_logo4)  
  
 self.logo4 = Label(upper\_frame, image=self.photo\_logo4)  
 self.logo4.place(x=1000, y=0, width=220, height=220)  
  
 # down Frame  
 down\_frame = LabelFrame(Main\_frame, bd=2, relief=RIDGE, bg='white', text='Criminal Information Table',  
 font=('times new roman', 11, 'bold'), fg='red')  
 down\_frame.place(x=10, y=280, width=1480, height=270)  
  
 # Search Frame  
 search\_frame = LabelFrame(down\_frame, bd=2, relief=RIDGE, bg='white', text='Search Criminal Record',  
 font=('times new roman', 11, 'bold'), fg='red')  
 search\_frame.place(x=0, y=0, width=1470, height=60)  
  
 search\_by = Label(search\_frame, font=("arial", 11, "bold"), text="Search By:", fg="white", bg="red")  
 search\_by.grid(row=0, column=0, sticky=W, padx=5)  
  
 # search  
 self.var\_com\_search = StringVar()  
 com\_txt\_search = ttk.Combobox(search\_frame, textvariable=self.var\_com\_search, state="readonly",  
 font=("arial", 12, "bold"), width=18)  
 com\_txt\_search['value'] = ("Select Option", "Case\_id", "Criminal\_no")  
 com\_txt\_search.current(0)  
 com\_txt\_search.grid(row=0, column=1, sticky=W, padx=5)  
  
 self.var\_search = StringVar()  
 txt\_search = ttk.Entry(search\_frame, textvariable=self.var\_search, width=22, font=("arial", 11, "bold"))  
 txt\_search.grid(row=0, column=2, padx=5)  
  
 btn\_\_search = Button(search\_frame, text="Search", command=self.serach\_data, font=("arial", 11, "bold"),  
 width=14, bg="blue", fg="white")  
 btn\_\_search.grid(row=0, column=3, padx=5)  
  
 btn\_ShowAll = Button(search\_frame, text="Show All", command=self.fetch\_data, font=("arial", 11, "bold"),  
 width=14, bg="blue", fg="white")  
 btn\_ShowAll.grid(row=0, column=4, padx=5)  
  
 stayhome = Label(search\_frame, text="NATIONAL CRIME AGENCY", font=("times new roman", 30, "bold"), fg="crimson",  
 bg="white")  
 stayhome.place(x=10, y=280, width=600, height=30)  
  
 # img\_logo\_mask=Image.open(r"images\backgound.jpg")  
 # img\_logo\_mask=img\_logo\_mask.resize((50,50),Image.ANTIALIAS)  
 # self.photoimg\_logo\_mask=ImageTk.PhotoImage(img\_logo\_mask)  
  
 # self.logo=Label(search\_frame,image=self.photoimg\_logo\_mask)  
 # self.logo.place(x=900,y=0,width=50,height=30)  
  
 # ============= Criminal table========================  
 # Table Frame  
 table\_frame = Frame(down\_frame, bd=3, relief=RIDGE)  
 table\_frame.place(x=0, y=60, width=1170, height=170)  
  
 scroll\_x = ttk.Scrollbar(table\_frame, orient=HORIZONTAL)  
 scroll\_y = ttk.Scrollbar(table\_frame, orient=VERTICAL)  
  
 self.criminal\_table = ttk.Treeview(table\_frame, column=(  
 "1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14"), xscrollcommand=scroll\_x.set,  
 yscrollcommand=scroll\_y.set)  
  
 scroll\_x.pack(side=BOTTOM, fill=X)  
 scroll\_y.pack(side=RIGHT, fill=Y)  
  
 scroll\_x.config(command=self.criminal\_table.xview)  
 scroll\_y.config(command=self.criminal\_table.yview)  
  
 self.criminal\_table.heading('1', text='CaseId')  
 self.criminal\_table.heading("2", text="CrimeNo")  
 self.criminal\_table.heading("3", text="Criminal Name")  
 self.criminal\_table.heading("4", text="NickName")  
 self.criminal\_table.heading("5", text="ArrestDate")  
 self.criminal\_table.heading("6", text="CrimeOfDate")  
 self.criminal\_table.heading("7", text="Address")  
 self.criminal\_table.heading("8", text="Age")  
 self.criminal\_table.heading("9", text="Occupation")  
 self.criminal\_table.heading("10", text="Birth Mark")  
 self.criminal\_table.heading("11", text="Crime Type")  
 self.criminal\_table.heading("12", text="Father Name")  
 self.criminal\_table.heading("13", text="Gender")  
 self.criminal\_table.heading("14", text="Wanted")  
  
 self.criminal\_table['show'] = 'headings'  
  
 self.criminal\_table.column("1", width=100)  
 self.criminal\_table.column("2", width=100)  
 self.criminal\_table.column("3", width=100)  
 self.criminal\_table.column("4", width=100)  
 self.criminal\_table.column("5", width=100)  
 self.criminal\_table.column("6", width=100)  
 self.criminal\_table.column("7", width=100)  
 self.criminal\_table.column("8", width=100)  
 self.criminal\_table.column("9", width=100)  
 self.criminal\_table.column("10", width=100)  
 self.criminal\_table.column("11", width=100)  
 self.criminal\_table.column("12", width=100)  
 self.criminal\_table.column("13", width=100)  
 self.criminal\_table.column("14", width=100)  
  
 self.criminal\_table.pack(fill=BOTH, expand=1)  
 self.criminal\_table.bind("<ButtonRelease>", self.get\_cursor)  
 self.fetch\_data()  
  
 # \*\*\*\*\*\*\*\*\*\*\*\*\*Function Declarations\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
 def add\_data(self):  
 if self.var\_criminal\_no.get() == "" or self.var\_nickname.get() == "":  
 messagebox.showerror('Error', 'All Fields are required')  
 else:  
 try:  
 conn = mysql.connector.connect(host='localhost', user='root', password='root',  
 database='management')  
 my\_cursor = conn.cursor()  
 my\_cursor.execute('insert into criminal1 values(%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)', (  
  
 self.var\_case\_id.get(),  
 self.var\_criminal\_no.get(),  
 self.var\_name.get(),  
 self.var\_nickname.get(),  
 self.var\_arrest\_date.get(),  
 self.var\_date\_of\_crime.get(),  
 self.var\_address.get(),  
 self.var\_age.get(),  
  
 self.var\_occupation.get(),  
 self.var\_birthMark.get(),  
 self.var\_crime\_type.get(),  
 self.var\_father\_name.get(),  
 self.var\_gender.get(),  
 self.var\_wanted.get()  
  
 ))  
 conn.commit()  
 self.fetch\_data()  
 conn.close()  
 messagebox.showinfo('Success', 'Criminal record has been added!', parent=self.root)  
 except Exception as es:  
 messagebox.showerror('Error', f'Due To:{str(es)}', parent=self.root)  
  
 # fetch Data  
 def fetch\_data(self):  
 conn = mysql.connector.connect(host='localhost', user='root', password='root', database='management')  
 my\_cursor = conn.cursor()  
 my\_cursor.execute('select \* from criminal1')  
 data = my\_cursor.fetchall()  
 if len(data) != 0:  
 self.criminal\_table.delete(\*self.criminal\_table.get\_children())  
 for i in data:  
 self.criminal\_table.insert("", END, values=i)  
 conn.commit()  
 conn.close()  
  
 # Get Cursor  
  
 def get\_cursor(self, event=""):  
 cursor\_row = self.criminal\_table.focus()  
 content = self.criminal\_table.item(cursor\_row)  
 data = content['values']  
  
 self.var\_case\_id.set(data[0])  
 self.var\_criminal\_no.set(data[1])  
 self.var\_name.set(data[2])  
 self.var\_nickname.set(data[3])  
 self.var\_arrest\_date.set(data[4])  
 self.var\_date\_of\_crime.set(data[5])  
 self.var\_address.set(data[6])  
 self.var\_age.set(data[7])  
 self.var\_occupation.set(data[8])  
 self.var\_birthMark.set(data[9])  
 self.var\_crime\_type.set(data[10])  
 self.var\_father\_name.set(data[11])  
 self.var\_gender.set(data[12])  
 self.var\_wanted.set(data[13])  
  
 def update\_data(self):  
 if self.var\_criminal\_no.get() == "" or self.var\_nickname.get() == "":  
 messagebox.showerror('Error', 'All Fields are required')  
 else:  
 try:  
 update = messagebox.askyesno('Update', 'Are sure update this criminal record')  
 if update > 0:  
 conn = mysql.connector.connect(host='localhost', user='root', password='root',  
 database='management')  
 my\_cursor = conn.cursor()  
 my\_cursor.execute(  
 'update criminal1 set Criminal\_no=%s,Criminal\_name=%s,Nick\_name=%s,arrest\_date=%s,dateOfcrime=%s,address=%s,age=%s,occupation=%s,BirthMark=%s,crimeType=%s,fatherName=%s,gender=%s,wanted=%s where Case\_id=%s',  
 (  
  
 self.var\_criminal\_no.get(),  
 self.var\_name.get(),  
 self.var\_nickname.get(),  
 self.var\_arrest\_date.get(),  
 self.var\_date\_of\_crime.get(),  
 self.var\_address.get(),  
 self.var\_age.get(),  
 self.var\_occupation.get(),  
 self.var\_birthMark.get(),  
 self.var\_crime\_type.get(),  
 self.var\_father\_name.get(),  
 self.var\_gender.get(),  
 self.var\_wanted.get(),  
 self.var\_case\_id.get(),  
  
 ))  
 else:  
 if not update:  
 return  
 conn.commit()  
 self.fetch\_data()  
 conn.close()  
 messagebox.showinfo('success', 'Criminal record successfully updated', parent=self.root)  
 except Exception as es:  
 messagebox.showerror('Error', f'Due To:{str(es)}', parent=self.root)  
  
 # Delete  
 def delete\_data(self):  
 if self.var\_case\_id.get() == "":  
 messagebox.showerror('Error', "All fields are required")  
 else:  
 try:  
 Delete = messagebox.askyesno('Delele', 'Are you sure delete this criminal', parent=self.root)  
 if Delete > 0:  
 conn = mysql.connector.connect(host='localhost', user='root', password='root',  
 database='management')  
 my\_cursor = conn.cursor()  
 sql = 'delete from criminal1 where Case\_id=%s'  
 value = (self.var\_case\_id.get(),)  
 my\_cursor.execute(sql, value)  
 else:  
 if not Delete:  
 return  
 conn.commit()  
 self.fetch\_data()  
 conn.close()  
 messagebox.showinfo('Delete', 'Criminal record Successfully Deleted', parent=self.root)  
 except Exception as es:  
 messagebox.showerror('Error', f'Due To:{str(es)}', parent=self.root)  
  
 # reset  
  
 def reset\_data(self):  
 self.var\_case\_id.set("")  
 self.var\_criminal\_no.set("")  
 self.var\_name.set("")  
 self.var\_nickname.set("")  
 self.var\_arrest\_date.set("")  
 self.var\_date\_of\_crime.set("")  
 self.var\_address.set("")  
 self.var\_age.set("")  
 # self.var\_occupationcomb.set("Select ID Proof")  
 self.var\_occupation.set("")  
 self.var\_birthMark.set("")  
 self.var\_crime\_type.set("")  
 self.var\_father\_name.set("")  
 self.var\_gender.set("")  
 self.var\_wanted.set("")  
  
 # search  
 def serach\_data(self):  
 if self.var\_com\_search.get() == '' or self.var\_search.get() == '':  
 messagebox.showerror('Error', 'Please select option')  
 else:  
 try:  
 conn = mysql.connector.connect(host='localhost', user='root', password='root',  
 database='management')  
 my\_cursor = conn.cursor()  
 my\_cursor.execute('select \* from criminal1 where ' + str(self.var\_com\_search.get()) + " LIKE'%" + str(  
 self.var\_search.get() + "%'"))  
 rows = my\_cursor.fetchall()  
 if len(rows) != 0:  
 self.criminal\_table.delete(\*self.criminal\_table.get\_children())  
 for i in rows:  
 self.criminal\_table.insert("", END, values=i)  
 conn.commit  
 conn.close()  
 except Exception as es:  
 messagebox.showerror('Error', f'Due To:{str(es)}', parent=self.root)  
  
  
############################################  
if \_\_name\_\_ == "\_\_main\_\_":  
 root = Tk()  
 obj = Criminal(root)  
 root.mainloop()