Criminal.py

from tkinter import \*

from tkinter import Tk, ttk

import data

from PIL import Image, ImageTk

import mysql.connector

from tkinter import messagebox

class Criminal:

def \_\_init\_\_(self, root):

self.root = root

self.root.geometry('1530x790+0+0')

self.root.title('CRIMINAL INVESTIGATION WITH SUSPECT PREDICTION')

# 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 INVESTIGATION WITH SUSPECT PREDICTION',font=('times new roman',35,'bold'),bg='black',fg='gold')

lbl\_title.place(x=20,y=0,width=1530,height=70)

# logo

img\_logo=Image.open('images/6.jpeg')

img\_logo=img\_logo.resize((60,60))

self.photo\_logo=ImageTk.PhotoImage(img\_logo)

self.logo=Label(self.root,image=self.photo\_logo)

self.logo.place(x=70,y=5,width=40,height=60)

# Image\_Frame

img\_frame=Frame(self.root,bd=2,relief=RIDGE,bg='white')

img\_frame.place(x=0,y=70,width=1530,height=130)

img1=Image.open('images/7.jpeg')

img1=img1.resize((540,160))

self.photo1=ImageTk.PhotoImage(img1)

self.img\_1=Label(img\_frame,image=self.photo1)

self.img\_1.place(x=0,y=0,width=540,height=160)

# img2

img\_2=Image.open('images/1.jpeg')

img\_2=img\_2.resize((540,160))

self.photo2 = ImageTk.PhotoImage(img\_2)

self.img\_2 = Label(img\_frame,image=self.photo2)

self.img\_2.place(x=540, y=0, width=540, height=160)

#img3

img\_3 = Image.open('images/5.jpeg')

img\_3 = img\_3.resize((540, 160))

self.photo3 = ImageTk.PhotoImage(img\_3)

self.img\_3 = Label(img\_frame, image=self.photo3)

self.img\_3.place(x=1080, y=0, width=540, 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,text='Criminal Information',font=('times new roman',10,'bold'),fg='red', bg='white')

upper\_frame.place(x=15, y=10, width=1470, height=270)

# Label Entry

# Case id

caseid=Label(upper\_frame, text='Case ID:',font=('arial', 12, '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,sticky=W)

# Criminal N0

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, sticky=W, 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\_name = ttk.Entry(upper\_frame,textvariable=self.var\_arrest\_date, width=22, font=("arial", 11, "bold"))

txt\_name.grid(row=2, column=1, sticky=W, 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, sticky=W, 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, sticky=W, padx=2, pady=7)

# Occupation

lbl\_occupation = Label(upper\_frame, font=("arial", 12, "bold"), text=" Occupation:", bg="white")

lbl\_occupation.grid(row=4, column=0, sticky=W, padx=2, pady=7)

txt\_occupation = ttk.Entry(upper\_frame,textvariable=self.var\_occupation, width=22, font=("arial", 11, "bold"))

txt\_occupation.grid(row=4, column=1, sticky=W, padx=2, pady=7)

#Birthmark

lbl\_birthmark = Label(upper\_frame, font=("arial", 12, "bold"), text=" Birthmark:", 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, sticky=W, 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, sticky=W, padx=2, pady=7)

# gender

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)

# wanted

lbl\_wanted = Label(upper\_frame, font=("arial", 12, "bold"), text=" Wanted:", bg="white")

lbl\_wanted.grid(row=3, column=4, sticky=W, padx=2, pady=7)

# Radio Button gender

radio\_frame\_gender=Frame(upper\_frame,bd=2,relief=RIDGE,bg="white")

radio\_frame\_gender.place(x=740,y=80,width=190,height=30)

male=Radiobutton(radio\_frame\_gender,variable=self.var\_gender,text='Male',value='male',font=("arial", 8, "bold"),bg='white')

male.grid(row=0,column=0,pady=2,padx=5,sticky=W)

female = Radiobutton(radio\_frame\_gender,variable=self.var\_gender, text='female', value='female', font=("arial", 8, "bold"), bg='white')

female.grid(row=0, column=1, pady=2, padx=5, sticky=W)

# Radio Button wanted

radio\_frame\_wanted = Frame(upper\_frame, bd=2, relief=RIDGE, bg="white")

radio\_frame\_wanted.place(x=740, y=120, width=190, height=30)

yes = Radiobutton(radio\_frame\_wanted, variable=self.var\_wanted,text='Yes', value='yes', font=("arial", 8, "bold"), bg='white')

yes.grid(row=0, column=0, pady=2, padx=5, sticky=W)

no = Radiobutton(radio\_frame\_wanted, variable=self.var\_wanted ,text='No', value='no', font=("arial", 8, "bold"), bg='white')

no.grid(row=0, column=1, pady=2, padx=5, sticky=W)

# Buttons

button\_frame = Frame(upper\_frame, bd=2, relief=RIDGE, bg="white")

button\_frame.place(x=5, y=200, width=620, height=45)

# add button

btn\_add=Button(button\_frame,command=self.add\_data,text='Record Save',font=("arial", 12, "bold"),width=14,bg='blue',fg='white')

btn\_add.grid(row=0,column=0,padx=3,pady=5)

# update button

btn\_update = Button(button\_frame,command=self.update\_data, text='Update', font=("arial", 12, "bold"), width=14, bg='blue', fg='white')

btn\_update.grid(row=0, column=1, padx=3, pady=5)

# delete button

btn\_delete = Button(button\_frame,command=self.delete\_data, text='Delete', font=("arial", 12, "bold"), width=14, bg='blue', fg='white')

btn\_delete.grid(row=0, column=2, padx=3, pady=5)

# clear button

btn\_clear = Button(button\_frame,command=self.clear\_data, text='Clear', font=("arial", 12, "bold"), width=14, bg='blue', fg='white')

btn\_clear.grid(row=0, column=3, padx=3, pady=5)

# background right side image

img4 = Image.open('images/4.jpeg')

img4 = img4.resize((470, 245))

self.photo4 = ImageTk.PhotoImage(img4)

self.img\_4 = Label(upper\_frame, image=self.photo4)

self.img\_4.place(x=1000, y=0, width=470, height=245)

# Down Frame

down\_frame = LabelFrame(main\_frame, bd=2, relief=RIDGE, text='Criminal Information Data',

font=('times new roman', 10, 'bold'), fg='red', bg='white')

down\_frame.place(x=15, y=280, width=1470, height=270)

search\_frame = LabelFrame(down\_frame, bd=2, relief=RIDGE, text='Search Criminal Record',

font=('times new roman', 10, 'bold'), fg='red', bg='white')

search\_frame.place(x=0, y=0, width=1470, height=60)

search\_by= Label(search\_frame, font=("arial", 12, "bold"), text=" Search By:", bg="red",fg='white')

search\_by.grid(row=0, column=0, sticky=W, padx=5)

self.var\_com\_search=StringVar()

combo\_search\_box=ttk.Combobox(search\_frame, textvariable=self.var\_com\_search, font=("arial", 12, "bold"), width=18, state='readonly')

combo\_search\_box['value']=('Select Option','case\_id','Criminal\_no','Crime\_Type')

combo\_search\_box.current(0)

combo\_search\_box.grid(row=0,column=1,sticky=W ,padx=5)

self.var\_search=StringVar()

search\_txt= ttk.Entry(search\_frame,textvariable=self.var\_com\_search, width=18, font=("arial", 11, "bold"))

search\_txt.grid(row=0, column=2, sticky=W, padx=5)

# search button

btn\_search = Button(search\_frame, text='Search', font=("arial", 12, "bold"), width=14, bg='blue', fg='white')

btn\_search.grid(row=0, column=3, padx=3, pady=5)

# all button

btn\_all = Button(search\_frame, text='Show All', font=("arial", 12, "bold"), width=14, bg='blue', fg='white')

btn\_all.grid(row=0, column=4, padx=3, pady=5)

crimeagency = Label(search\_frame, font=("arial", 30, "bold"), text=" NATIONAL CRIME AGENCY", bg="white",fg='crimson')

crimeagency.grid(row=0, column=5, sticky=W, padx=50 )

# Table Frame

table\_frame=Frame(down\_frame,bd=2,relief=RIDGE)

table\_frame.place(x=0,y=60,width=1470,height=170)

# Scroll bar

scroll\_x = ttk.Scrollbar(table\_frame, orient=HORIZONTAL)

scroll\_y = ttk.Scrollbar(table\_frame, orient=VERTICAL)

self.criminal\_table=ttk.Treeview(table\_frame,columns=('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='Case id')

self.criminal\_table.heading('2', text='Crime No')

self.criminal\_table.heading('3', text='Criminal Name')

self.criminal\_table.heading('4', text='NickName')

self.criminal\_table.heading('5', text='Arrest Date')

self.criminal\_table.heading('6', text='Date of Crime')

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()

# Add function

def add\_data(self):

if self.var\_case\_id.get()=="":

messagebox.showerror('Error','All Fields are Required')

else:

try:

conn=mysql.connector.connect(host='localhost',username='root',password='Kishan@1703',database='crime')

my\_cursor=conn.cursor()

my\_cursor.execute('insert into criminal 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()

self.clear\_data()

conn.close()

messagebox.showinfo('Success','Criminal record has bee added')

except Exception as es:

messagebox.showerror('Error',f'Due To{str(es)}')

# fetch data

def fetch\_data(self):

conn = mysql.connector.connect(host='localhost', username='root', password='Kishan@1703', database='crime')

my\_cursor=conn.cursor()

my\_cursor.execute('select \* from criminal')

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])

# update

def update\_data(self):

if self.var\_case\_id.get()=="":

messagebox.showerror('Error','All Fields are required')

else:

try:

update=messagebox.askyesno('Update','Are you Sure! You want to Update this record')

if update>0:

conn = mysql.connector.connect(host='localhost', username='root', password='Kishan@1703', database='crime')

my\_cursor = conn.cursor()

my\_cursor.execute('update criminal 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()

self.clear\_data()

conn.close()

messagebox.showinfo('Success','Criminal records updated successfully')

except Exception as es:

messagebox.showerror('Error',f'Due To{str(es)}')

# delete

def delete\_data(self):

if self.var\_case\_id.get()=="":

messagebox.showerror('Error','All Fields are required')

else:

try:

delete = messagebox.askyesno('Delete', 'Are you Sure! You want to Delete this record')

if delete >0:

conn = mysql.connector.connect(host='localhost', username='root', password='Kishan@1703', database='crime')

my\_cursor = conn.cursor()

sql='delete from criminal 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()

self.clear\_data()

messagebox.showinfo('Success','Criminal record Deleted successfully')

except Exception as es:

messagebox.showerror('Error',f'Due To{str(es)}')

# clear

def clear\_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\_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 search\_data(self):

if self.var\_com\_search.get()=="":

messagebox.showerror('Error','All Fields are Required')

else:

try:

conn = mysql.connector.connect(host='localhost', username='root', password='Kishan@1703',

database='crime')

my\_cursor = conn.cursor()

my\_cursor=conn.cursor('select \* from criminal where' +str(self.var\_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)}')

if \_\_name\_\_ == "\_\_main\_\_":

root = Tk()

obj = Criminal(root)

root.mainloop()