#Frontend

from tkinter import\*

import tkinter.messagebox

import studentdatabase1

class Student:

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

self.root = root

self.root.title("Students Database Management System")

self.root.geometry("1350x7500+0+0")

self.root.config(bg="cadet blue")

StdID = StringVar()

Firstname = StringVar()

Lastname = StringVar()

DoB = StringVar()

Age = StringVar()

Gender = StringVar()

Address = StringVar()

Mobile = StringVar()

#==========================Frame======================================#

def iExit():

iExit = tkinter.messagebox.askyesno("Students Database Management System", "Confirm if you want to exit")

if iExit > 0:

root.destroy()

return

def clearData():

self.txtStdID.delete(0,END)

self.txtFname.delete(0,END)

self.txtLname.delete(0,END)

self.txtDob.delete(0,END)

self.txtAge.delete(0,END)

self.txtGen.delete(0,END)

self.txtAdd.delete(0,END)

self.txtMob.delete(0,END)

def addData():

if(len(StdID.get()) !=0):

studentdatabase1.addStdRec(StdID.get(), Firstname.get(), Lastname.get(), DoB.get(), Age.get(), Gender.get(), Address.get(), Mobile.get())

studentlist.delete(0,END)

studentlist.insert(END,(StdID.get(), Firstname.get(), Lastname.get(), DoB.get(), Age.get(), Gender.get(), Address.get(), Mobile.get()))

def DisplayData():

studentlist.delete(0,END)

for row in studentdatabase1.viewData():

studentlist.insert(END,row,str(""))

def StudentRec(event):

global sd

searchStd = studentlist.curselection()

sd = studentlist.get(searchStd)

self.txtStdID.delete(0,END)

self.txtStdID.insert(END,sd[1])

self.txtFname.delete(0,END)

self.txtFname.insert(END,sd[2])

self.txtLname.delete(0,END)

self.txtLname.insert(END,sd[3])

self.txtDob.delete(0,END)

self.txtDob.insert(END,sd[4])

self.txtAge.delete(0,END)

self.txtAge.insert(END,sd[5])

self.txtGen.delete(0,END)

self.txtGen.insert(END,sd[6])

self.txtAdd.delete(0,END)

self.txtAdd.insert(END,sd[7])

self.txtMob.delete(0,END)

self.txtMob.insert(END,sd[8])

def DeleteData():

if(len(StdID.get()) !=0):

studentdatabase1.deleteRec(sd[0])

clearData()

DisplayData()

def searchData():

studentlist.delete(0,END)

for row in studentdatabase1.searchData(StdID.get(), Firstname.get(), Lastname.get(), DoB.get(), Age.get(), Gender.get(), Address.get(), Mobile.get()):

studentlist.insert(END,row,str(""))

def update():

if(len(StdID.get()) !=0):

studentdatabase1.deleteRec(sd[0])

if(len(StdID.get()) !=0):

studentdatabase1.addStdRec(StdID.get(), Firstname.get(), Lastname.get(), DoB.get(), Age.get(), Gender.get(), Address.get(), Mobile.get())

studentlist.delete(0,END)

studentlist.insert(END,(StdID.get(), Firstname.get(), Lastname.get(), DoB.get(), Age.get(), Gender.get(), Address.get(), Mobile.get()))

#==========================Frame======================================#

MainFrame = Frame(self.root, bg="cadet blue")

MainFrame.grid()

TitFrame = Frame(MainFrame, bd=2, padx=54, pady=8, bg="Ghost White", relief=RIDGE)

TitFrame.pack(side=TOP)

self.lblTit = Label(TitFrame ,font=('arial', 47,'bold'),text= "Student Database Management System", bg="Ghost White")

self.lblTit.grid()

ButtonFrame = Frame(MainFrame, bd=2, width=1350, height=70, padx=20, pady=20, bg="cadet blue", relief=RIDGE)

ButtonFrame.pack(side=BOTTOM)

DataFrame = Frame(MainFrame, bd=1, width=1400, height=500, padx=20, pady=20, bg="cadet blue", relief=RIDGE)

DataFrame.pack(side=BOTTOM)

DataFrameLeft = LabelFrame(DataFrame, bd=1, width=1000, height=600, padx=20, pady=20, bg="Ghost White", relief=RIDGE, font=('arial', 20,'bold'), text="Student Info\n")

DataFrameLeft.pack(side=LEFT)

DataFrameRight = LabelFrame(DataFrame, bd=1, width=450, height=300, padx=31, pady=3, bg="Ghost White", relief=RIDGE, font=('arial', 20,'bold'), text="Student Details\n")

DataFrameRight.pack(side=RIGHT)

#==========================Frame1======================================#

self.lblStdID = Label(DataFrameLeft ,font=('arial', 20,'bold'), padx=2, pady=2, text= "Student ID:", bg="Ghost White")

self.lblStdID.grid(row=0, column=0, sticky=W)

self.txtStdID = Entry(DataFrameLeft ,font=('arial', 20,'bold'), textvariable=StdID, width=39)

self.txtStdID.grid(row=0, column=1)

self.lblFname = Label(DataFrameLeft ,font=('arial', 20,'bold'), padx=2, pady=2, text= "FirstName:", bg="Ghost White")

self.lblFname.grid(row=1, column=0, sticky=W)

self.txtFname = Entry(DataFrameLeft ,font=('arial', 20,'bold'), textvariable=Firstname, width=39)

self.txtFname.grid(row=1, column=1)

self.lblLname = Label(DataFrameLeft ,font=('arial', 20,'bold'), padx=2, pady=2, text= "LastName:", bg="Ghost White")

self.lblLname.grid(row=2, column=0, sticky=W)

self.txtLname = Entry(DataFrameLeft ,font=('arial', 20,'bold'), textvariable=Lastname, width=39)

self.txtLname.grid(row=2, column=1)

self.lblDob = Label(DataFrameLeft ,font=('arial', 20,'bold'), padx=2, pady=2, text= "Date of Birth:", bg="Ghost White")

self.lblDob.grid(row=3, column=0, sticky=W)

self.txtDob = Entry(DataFrameLeft ,font=('arial', 20,'bold'), textvariable=DoB, width=39)

self.txtDob.grid(row=3, column=1)

self.lblAge = Label(DataFrameLeft ,font=('arial', 20,'bold'), padx=2, pady=2, text= "Age:", bg="Ghost White")

self.lblAge.grid(row=4, column=0, sticky=W)

self.txtAge = Entry(DataFrameLeft ,font=('arial', 20,'bold'), textvariable=Age, width=39)

self.txtAge.grid(row=4, column=1)

self.lblGen = Label(DataFrameLeft ,font=('arial', 20,'bold'), padx=2, pady=2, text= "Gender:", bg="Ghost White")

self.lblGen.grid(row=5, column=0, sticky=W)

self.txtGen = Entry(DataFrameLeft ,font=('arial', 20,'bold'), textvariable=Gender, width=39)

self.txtGen.grid(row=5, column=1)

self.lblAdd = Label(DataFrameLeft ,font=('arial', 20,'bold'), padx=2, pady=2, text= "Address:", bg="Ghost White")

self.lblAdd.grid(row=6, column=0, sticky=W)

self.txtAdd = Entry(DataFrameLeft ,font=('arial', 20,'bold'), textvariable=Address, width=39)

self.txtAdd.grid(row=6, column=1)

self.lblMob = Label(DataFrameLeft ,font=('arial', 20,'bold'), padx=2, pady=2, text= "Mobile:", bg="Ghost White")

self.lblMob.grid(row=7, column=0, sticky=W)

self.txtMob = Entry(DataFrameLeft ,font=('arial', 20,'bold'), textvariable=Mobile, width=39)

self.txtMob.grid(row=7, column=1)

#==========================Frame2======================================#

scrollbar = Scrollbar (DataFrameRight)

scrollbar.grid(row=0, column=1, sticky='ns')

studentlist = Listbox(DataFrameRight, width=41, height=16, font=('arial', 12, 'bold'), yscrollcommand=scrollbar.set)

studentlist.bind('<<ListboxSelect>>', StudentRec)

studentlist.grid(row=0, column=0, padx=8)

scrollbar.config(command = studentlist.yview)

#==========================Frame3======================================#

self.btnAddData = Button(ButtonFrame, text="Add New", font=('arial', 12, 'bold'), height=1, width=10, bd=4, command=addData)

self.btnAddData.grid(row=0, column=0)

self.btnAddData = Button(ButtonFrame, text="Display", font=('arial', 12, 'bold'), height=1, width=10, bd=4, command=DisplayData)

self.btnAddData.grid(row=0, column=1)

self.btnAddData = Button(ButtonFrame, text="Clear", font=('arial', 12, 'bold'), height=1, width=10, bd=4, command=clearData)

self.btnAddData.grid(row=0, column=2)

self.btnAddData = Button(ButtonFrame, text="Delete", font=('arial', 12, 'bold'), height=1, width=10, bd=4, command=DeleteData)

self.btnAddData.grid(row=0, column=3)

self.btnAddData = Button(ButtonFrame, text="Search", font=('arial', 12, 'bold'), height=1, width=10, bd=4, command=searchData)

self.btnAddData.grid(row=0, column=4)

self.btnAddData = Button(ButtonFrame, text="Update", font=('arial', 12, 'bold'), height=1, width=10, bd=4, command=update)

self.btnAddData.grid(row=0, column=5)

self.btnAddData = Button(ButtonFrame, text="Exit", font=('arial', 12, 'bold'), height=1, width=10, bd=4, command=iExit)

self.btnAddData.grid(row=0, column=6)

if \_\_name\_\_=='\_\_main\_\_':

root = Tk()

application = Student(root)

root.mainloop()