from tkinter import \*

from tkinter import ttk

from tkinter import messagebox as message

import consultas

class app:

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

self.wind = windows

self.wind.title("Aplicacion CRUD")

# Instanciando un objeto

self.query = consultas.query()

#Frame Label

frame = LabelFrame(self.wind, text="Registrar")

frame.grid(row=0, column=0, columnspan=3,ipadx=20)

Label(frame,text="Nombre: ").grid(row=1, column=1)

self.name = Entry(frame)

self.name.grid(row=1, column=2, pady=5)

Label(frame,text="Email: ").grid(row=2, column=1)

self.email = Entry(frame)

self.email.grid(row=2, column=2, pady=5)

#Btn enviar

ttk.Button(frame, text="Guardar", command=self.guardar).grid(row=3, columnspan=3, sticky=W + E)

self.tabla0 = ttk.Treeview(height=10, column=2)

self.tabla0.grid(row=4, column=0, columnspan=2)

self.tabla0.heading("#0",text="Nombre",anchor='center')

self.tabla0.heading("#1",text="Correo",anchor='center')

self.mostrar()

#CREANDO BOTON PARA ELIMINAR

ttk.Button(text="Eliminar", command=self.eliminar).grid(row=5, column = 0, sticky = W + E)

ttk.Button(text="Editar",command=self.actualizar).grid(row=5, column = 1, sticky = W + E)

def guardar(self):

name = self.name.get()

email = self.email.get()

if(email !='' and email !=''):

self.query.save(name,email)

message.showinfo(message="Datos almacenados", title="Guardados")

self.name.delete(0,END)

self.email.delete(0,END)

self.mostrar()

else:

message.showinfo(message="Ingrese los datos", title="Por favor")

def mostrar(self):

delete = self.tabla0.get\_children()

for elemento in delete:

self.tabla0.delete(elemento)

rows = self.query.read()

for row in rows:

self.tabla0.insert('',END, text=row[1],value=row[2])

def eliminar(self):

try:

email = self.tabla0.item(self.tabla0.selection())['values'][0]

self.query.delete(email)

self.mostrar()

except IndexError:

message.showinfo(message="Por favor selecciona un dato de la tabla", title="Error")

#Funcion que contiene mi ventana secundaria

def actualizar(self):

try:

self.previous\_name = self.tabla0.item(self.tabla0.selection())['text']

self.previous\_email = self.tabla0.item(self.tabla0.selection())['values'][0]

name\_set = StringVar()

name\_set.set(self.previous\_name)

email\_set = StringVar()

email\_set.set(self.previous\_email)

self.edit\_window = Toplevel()

self.edit\_window.title("Actualizar")

frame = LabelFrame(self.edit\_window, text='Actualizar')

frame.grid(row=0, column=0, ipadx=20)

Label(frame, text="Nombre: ").grid(row=1, column=1)

self.new\_name = Entry(frame, textvar=name\_set)

self.new\_name.grid(row=1,column=2,ipadx=20)

Label(frame, text='Email: ').grid(row=2, column=1)

self.new\_email = Entry(frame, textvar=email\_set)

self.new\_email.grid(row=2,column=2,ipadx=20)

ttk.Button(frame, text="Guardar", command=self.edit).grid(row=3, columnspan=2, sticky=W + E)

except IndexError:

message.showinfo(message="Por favor selecciona un dato de la tabla", title="Error")

#Funcion que permite hacer la actualización

def edit(self):

self.nombre=self.new\_name.get()

self.correo = self.new\_email.get()

if(self.nombre !='' and self.correo !=''):

self.query.update(self.nombre,self.correo,self.previous\_name,self.previous\_email)

self.mostrar()

self.edit\_window.destroy()

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

root = Tk()

Aplicacion = app(root)

root.mainloop()

import sqlite3

class query():

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

self.db = "database.db"

def ejecutar\_consultar(self,sql,parametros=()):

with sqlite3.connect(self.db) as conn:

cursor = conn.cursor()

result = cursor.execute(sql,parametros)

conn.commit()

return result

def save(self,name,email):

sql = "INSERT INTO empleados(Name,Email) VALUES(?,?)"

parametros =(name,email)

self.ejecutar\_consultar(sql, parametros)

def read(self):

sql = "SELECT \* FROM empleados ORDER BY ID ASC"

results = self.ejecutar\_consultar(sql)

return results

def delete(self,email):

sql = "DELETE FROM empleados WHERE Email=?"

parametros = (email,)

results = self.ejecutar\_consultar(sql,parametros)

def update(self,nombre\_nuevo,correo\_nuevo,name,email):#Le pasamos los 4 parametros para actualizar el registro

sql = "UPDATE empleados SET Name=?,Email=? WHERE Name=? AND Email=?"

parametros = (nombre\_nuevo,correo\_nuevo,name,email)

self.ejecutar\_consultar(sql,parametros)