Objectives:

* TKINTER and Databases

**There are 2 challenge Exercises worth 50% each.**

**Project #1 (Using SQLite and** TKINTER**)**

**Table creation SQL Code, this code must be created on the database first.**

A close-up of a white background

Description automatically generated

**TKINTER GUI interface with database connection. Type the code below.**

A computer screen shot of a program code

Description automatically generatedA computer screen shot of a program

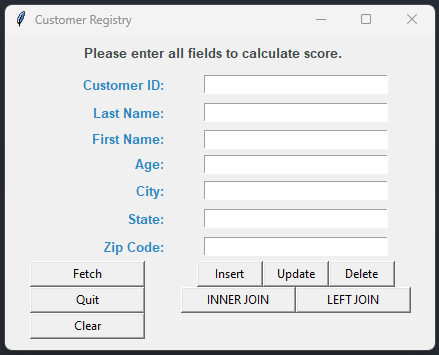
Description automatically generatedA computer screen shot of a program

Description automatically generatedA screen shot of a computer

Description automatically generated

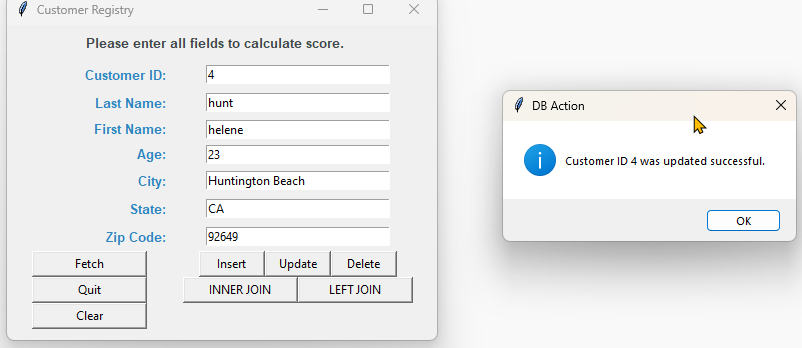
**Challenge Exercise #1:** Add the Address, City, State, and Zip code Fields to the Customer table and the Python Code.

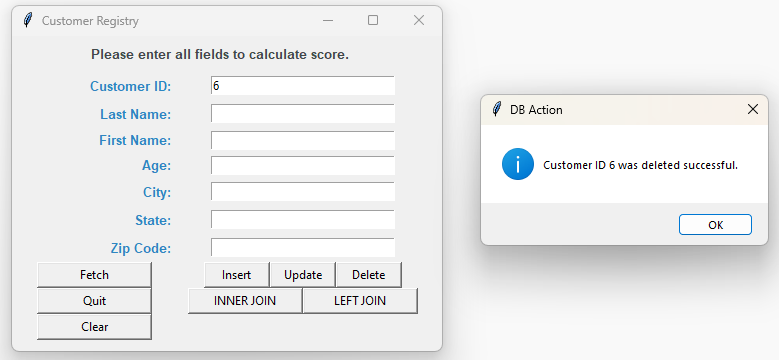
**#1 Print screen the running application below here.**



**Challenge Exercise #2:** The next task is to ensure your application will have the option to delete by the ID and update only the Last Name where the ID =?

**#2 Print screen the running application below here.**

****

****

**Copy and paste all your code below here.**

Code:

import tkinter as tk

import sqlite3 as sql

from tkinter import messagebox

#import sqlserver as ss

def insert\_data():

#id = txtid.get()

last\_name = txtln.get()

first\_name = txtfn.get()

age = txtage.get()

city = txtcity.get()

state = txtstate.get()

zip = txtzip.get()

#open connection

conn = sql.connect("C:\\Users\\17147\\Desktop\\SQLLite\\PythonClassDB.db")

crs = conn.cursor()

crs.execute("INSERT INTO customers (last\_name, first\_name, age, city, state, zipcode) VALUES(?,?,?,?,?,?)", (last\_name, first\_name, age, city, state, zip))

conn.commit() #Commit transaction

conn.close() #Close connection

messagebox.showinfo(title="DB Action", message="Insert successful.")

txtid.delete(0, tk.END)

txtln.delete(0, tk.END)

txtfn.delete(0, tk.END)

txtage.delete(0, tk.END)

txtcity.delete(0, tk.END)

txtstate.delete(0, tk.END)

txtzip.delete(0, tk.END)

txtid.focus\_set()

def update\_data():

id = txtid.get()

last\_name = txtln.get()

first\_name = txtfn.get()

age = txtage.get()

city = txtcity.get()

state = txtstate.get()

zip = txtzip.get()

conn = sql.connect("C:\\Users\\17147\\Desktop\\SQLLite\\PythonClassDB.db")

crs = conn.cursor()

crs.execute("UPDATE customers SET last\_name=?, first\_name=?, age=?, city=?, state=?, zipcode=? WHERE id=?", (last\_name, first\_name, age, city, state, zip, id))

conn.commit()

conn.close

messagebox.showinfo(title="DB Action", message=f"Customer ID {id} was updated successful.")

txtid.delete(0, tk.END)

txtln.delete(0, tk.END)

txtfn.delete(0, tk.END)

txtage.delete(0, tk.END)

txtcity.delete(0, tk.END)

txtstate.delete(0, tk.END)

txtzip.delete(0, tk.END)

txtid.focus\_set()

def delete\_data():

id = txtid.get()

conn = sql.connect("C:\\Users\\17147\\Desktop\\SQLLite\\PythonClassDB.db")

crs = conn.cursor()

crs.execute("DELETE FROM customers WHERE id=?", (id))

conn.commit()

conn.close

messagebox.showinfo(title="DB Action", message=f"Customer ID {id} was deleted successful.")

txtid.delete(0, tk.END)

txtln.delete(0, tk.END)

txtfn.delete(0, tk.END)

txtage.delete(0, tk.END)

txtcity.delete(0, tk.END)

txtstate.delete(0, tk.END)

txtzip.delete(0, tk.END)

txtid.focus\_set()

def fetch\_data():

conn = sql.connect("C:\\Users\\17147\\Desktop\\SQLLite\\PythonClassDB.db")

crs = conn.cursor()

crs.execute("SELECT \* FROM customers")

data = crs.fetchall()

for row in data:

print(row)

conn.close()

def fetch\_joindata1():

conn = sql.connect("C:\\Users\\17147\\Desktop\\SQLLite\\PythonClassDB.db")

crs = conn.cursor()

crs.execute("SELECT \* FROM customers c INNER JOIN orders o ON c.id = o.id")

data = crs.fetchall()

for row in data:

print(row)

conn.close()

def fetch\_joindata1():

conn = sql.connect("C:\\Users\\17147\\Desktop\\SQLLite\\PythonClassDB.db")

crs = conn.cursor()

crs.execute("SELECT \* FROM customers c LEFT JOIN orders o ON c.id = o.id LEFT JOIN representative r ON c.id = r.id")

data = crs.fetchall()

for row in data:

print(row)

conn.close()

def quit():

win.quit()

win.destroy()

def cleartext():

txtid.delete(0, tk.END)

txtfn.delete(0, tk.END)

txtln.delete(0, tk.END)

txtage.delete(0, tk.END)

txtcity.delete(0, tk.END)

txtstate.delete(0, tk.END)

txtzip.delete(0, tk.END)

txtid.focus\_set()

win = tk.Tk()

win.title('Customer Registry')

win.geometry("430x315")

# Instruction Label

frm1 = tk.Frame(win)

frm1.grid(column=0, row=0, ipadx=0, padx=0, pady=7, columnspan=2) #Label widge

lblheader = tk.Label(frm1, text = "Please enter all fields to calculate score.", font="Arial 10 bold", fg="#424949", anchor="w")

lblheader.grid()

# Input

lblid = tk.Label(win, text = "Customer ID: ", font="Arial 10 bold", width=20, anchor="e", fg="#2E86C1")

lblid.grid(column=0, row=1, ipadx = 0, padx = 0, pady=3) #Label widge

id = tk.StringVar() #Manage the Entry widget

txtid = tk.Entry(win, width=30, textvariable=id)

txtid.grid(column=1,row=1, ipadx = 0, padx = 0, pady=3)

txtid.focus\_set()

lblln = tk.Label(win, text = "Last Name: ", font="Arial 10 bold", width=20, anchor="e", fg="#2E86C1")

lblln.grid(column=0, row=2, ipadx = 0, padx = 0, pady=3) #Label widge

ln = tk.StringVar() #Manage the Entry widget

txtln = tk.Entry(win, width=30, textvariable=ln)

txtln.grid(column=1,row=2, ipadx = 0, padx = 0, pady=3)

lblfn = tk.Label(win, text = "First Name: ", font="Arial 10 bold", width=20, anchor="e", fg="#2E86C1")

lblfn.grid(column=0, row=3) #Label widge

fn = tk.StringVar() #Manage the Entry widget

txtfn = tk.Entry(win, width=30, textvariable=fn)

txtfn.grid(column=1,row=3, ipadx = 0, padx = 0, pady=3)

lblage = tk.Label(win, text = "Age: ", font="Arial 10 bold", width=20, anchor="e", fg="#2E86C1")

lblage.grid(column=0, row=4) #Label widge

age = tk.StringVar() #Manage the Entry widget

txtage = tk.Entry(win, width=30, textvariable=age)

txtage.grid(column=1, row=4, ipadx = 0, padx = 0, pady=3)

lblcity = tk.Label(win, text = "City: ", font="Arial 10 bold", width=20, anchor="e", fg="#2E86C1")

lblcity.grid(column=0, row=5, ipadx = 0, padx = 0, pady=3) #Label widge

city = tk.StringVar() #Manage the Entry widget

txtcity = tk.Entry(win, width=30, textvariable=city)

txtcity.grid(column=1, row=5, ipadx = 0, padx = 0, pady=3)

lblstate = tk.Label(win, text = "State: ", font="Arial 10 bold", width=20, anchor="e", fg="#2E86C1")

lblstate.grid(column=0, row=6, ipadx = 0, padx = 0, pady=3) #Label widge

state = tk.StringVar() #Manage the Entry widget

txtstate = tk.Entry(win, width=30, textvariable=state)

txtstate.grid(column=1, row=6, ipadx = 0, padx = 0, pady=3)

lblzip = tk.Label(win, text = "Zip Code: ", font="Arial 10 bold", width=20, anchor="e", fg="#2E86C1")

lblzip.grid(column=0, row=7, ipadx = 0, padx = 0, pady=3) #Label widge

zip = tk.StringVar() #Manage the Entry widget

txtzip = tk.Entry(win, width=30, textvariable=zip)

txtzip.grid(column=1, row=7, ipadx = 0, padx = 0, pady=3)

# lblDisplay = tk.Label(win, text="", font="Arial 10 bold", justify="center")

# lblDisplay.grid(column=0, row=7, padx=5, pady=12, columnspan=2)

btnfetch = tk.Button(win, text="Fetch", command=fetch\_data, width=15)

btnfetch.grid(column=0, row=9, ipadx = 0, padx = 0)

btnquit = tk.Button(win, text="Quit", command=quit, width=15)

btnquit.grid(column=0, row=10, ipadx = 0, padx = 0)

btnclear = tk.Button(win, text="Clear", command=cleartext, width=15)

btnclear.grid(column=0, row=11, ipadx = 0, padx = 0)

frm3 = tk.Frame(win)

frm3.grid(column=1, row=9, ipadx = 0, padx = 10)

btninsert = tk.Button(frm3, text="Insert", command=insert\_data, width=8)

btninsert.grid(column=0, row=0)

btnupdate = tk.Button(frm3, text="Update", command=update\_data, width=8)

btnupdate.grid(column=1, row=0)

btndelete = tk.Button(frm3, text="Delete", command=delete\_data, width=8)

btndelete.grid(column=2, row=0)

frm4 = tk.Frame(win)

frm4.grid(column=1, row=10, ipadx = 0, padx = 10)

btnjoin1 = tk.Button(frm4, text="INNER JOIN", command=fetch\_joindata1, width=15)

btnjoin1.grid(column=0, row=0, ipadx = 0, padx = 0)

btnjoin2 = tk.Button(frm4, text="LEFT JOIN", command=fetch\_joindata1, width=15)

btnjoin2.grid(column=1, row=0, ipadx = 0, padx = 0)

win.mainloop()

**Submit this document to Module 5 Class Exercise.**