In [6]:

```
import sqlite3 as lite
from sqlite3 import Error
import tkinter as tk
from glob import glob



class App:
    # window
    def __init__(self, root):
        self.root = root
        self.root.geometry("200x200")
        self.root['bg'] = "cyan"
        self.butnew("Create Database", Win1)
        self.butnew("Create Table", Win2)

    def butnew(self, text, _class):
        tk.Button(self.root, text = text, command= lambda: self.new_window(_class)).pac
k()

    def new_window(self, _class):
        self.new = tk.Toplevel(self.root)
        _class(self.new)


class Win:
    fields = []

    def show_db(self):
        for file in glob("*.db"):
            self.lb.insert(tk.END, file)

    def mk_db(self):
        db = self.e.get()
        if db.endswith(".db"):
            pass
        else:
            db = db + ".db"
        try:
            conn = lite.connect(db)
            if db in self.lb.get(0, tk.END):
                pass
            else:
                self.lb.insert(tk.END, db)
            return conn
        except Error as e:
            print(e)
        finally:
            self.db.set("")
            conn.close()

    def mk_tb(self, dbn, tbn):
        self.conn = lite.connect(dbn.get())
        self.cur = self.conn.cursor()
        Win.fields = "".join(Win.fields)
        self.cur.execute("""create table {} (
        {} );""".format(tbn, Win.fields))
        Win.fields = []
        self.conn.close()
```

```
    def mk_fl(self):
        Win.fields.append(self.efl.get())
        self.vfl.set("")

    def show_selection(self):
        x = self.lb.curselection()[0]
        x = self.lb.get(x)
        self.dbn.set(x)

class Win1(Win):
    def __init__(self, root):
        self.root = root
        self.root.geometry("400x300")
        self.root['bg'] = "pink"
        self.label()
        self.entry() # it creates e entry to create db
        self.button() # create database
        self.listbox()

    def label(self):
        self.l = tk.Label(self.root, text="Create a db [insert the name]")
        self.l.pack()

    def entry(self):
        self.db = tk.StringVar()
        self.e = tk.Entry(self.root, textvariable=self.db)
        self.e.pack()

    def button(self):
        """Create db"""
        self.b = tk.Button(self.root, text="Create DB", command= lambda: self.mk_db())
        self.b.pack()

    def listbox(self):
        self.lb = tk.Listbox(self.root)
        self.lb.pack()
        self.show_db()
        print(self.lb.curselection())
        self.lb.bind("<Double-Button>", lambda x: self.show_selection())


class Win2(Win):
    def __init__(self, root):
        Win.__init__(self)
        self.root = root
        self.root.geometry("400x500")
        self.root['bg'] = "green"
        self.listbox()
        self.db_name_widgets()
        self.tb_name_widgets()
        self.fields_widgets()
        self.btn_create_table()

    def listbox(self):
        self.lb = tk.Listbox(self.root)
        self.lb.pack()
        self.show_db()
        print(self.lb.curselection())
        self.lb.bind("<Double-Button>", lambda x: self.show_selection())
```

```
    def db_name_widgets(self):
        # label and Entry for Database name
        self.ldbname = tk.Label(self.root, text="Insert Database name")
        self.ldbname.pack()
        self.dbn = tk.StringVar()
        self.edb = tk.Entry(self.root, textvariable = self.dbn)
        self.edb.pack()

    def tb_name_widgets(self):
        self.ltbname = tk.Label(self.root, text="Insert Table name")
        self.ltbname.pack()
        self.tbn = tk.StringVar()
        self.etb = tk.Entry(self.root, textvariable = self.tbn)
        self.etb.pack()

    def fields_widgets(self):
        self.lflname = tk.Label(self.root, text="Insert Fields name and type\n followeb
by a comma, one by one,\nclicking once for each field.")
        self.lflname.pack()
        self.vfl = tk.StringVar()
        self.efl = tk.Entry(self.root, textvariable = self.vfl)
        self.efl.pack()
        self.bfl = tk.Button(self.root, text="Create Field", command= lambda: self.mk_f
l())
        self.bfl.pack()

    def btn_create_table(self):
        self.btb = tk.Button(self.root, text="Create Table", command= lambda: self.mk_t
b(self.dbn, self.tbn))
        self.btb.pack()



root = tk.Tk()
win = App(root)
root.mainloop()
```

()
()