fkos.py

```
1
2  from tkinter import*
3  from tkinter import messagebox
4  import os
5  from math import*
6  import urllib.parse
7  import webbrowser
8  import time
9
10
11   #  #6be4ce light colour    #21d1b3  dark colour       wallpaper colours
12   #  #007166  default theme
13
14
15 def global_click_handler(event):    #handles all clicks #removedclcikin out and checkin
   out for better usage and functionality in this code
16     widget = event.widget
17
18     if widget != searchbar:
19         if searchbar.get() == '':
20             searchbar.insert(0, 'Search Anything')
21             searchbar.config(fg='gray')
22             win.focus_set()
23     else:
24         if searchbar.get() == 'Search Anything':
25             searchbar.delete(0, END)
26             searchbar.config(fg='black')
27     if widget != signin:
28         if signin.get() == '':
29             signin.insert(0, 'Password')
30             signin.config(fg='gray', show='')
31     else:
32         if signin.get() == 'Password':
33             signin.delete(0, END)
34             signin.config(fg='black', show='*')
35     if widget not in (signin, searchbar):
36         win.focus_set()
37
38
39
40 def delsear():
41     searchbar.delete(0,END)
42     searchbar.insert(0,'Search Anything')
43     searchbar.config(fg="gray")
44     win.focus()
45 def search():
46     query = searchbar.get()
47     if query and query != 'Search Anything':
48         try:
49             url = "https://www.google.com/search?q=" + urllib.parse.quote(query)
   #convert the search in a+b+c format more url frendly
50             webbrowser.open_new_tab(url)     #open in new tab not new chrome
```

```
        except Exception as e:
            messagebox.showerror("Error", f"Could not open browser:\n{e}")
    else:
        messagebox.showerror("Error",'Search something')


count=0
def checkpass():
    global count
    count+=1
    passw=signin.get()
    if passw=='1234':
        time.sleep(2)
        signf.destroy()
        entryscreen.update()
        entryscreen.destroy()
    else:
        if count>2:
            turnoff()
        else:
            signin.delete(0,END)
            messagebox.showerror("Wrong Password",f'Enter correct password {3-count} tries
left')
def turnoff():
    messagebox.showerror("Turning off",'maximum amount of tries used')
    quit()



def calculate():
    ste=StringVar()
    ste.set('')
    global calpic

    def click(event):
        t = event.widget.cget('text')
        current = ste.get()

        try:
            if t == '=':
                expr = current.replace('x', '*').replace('÷', '/')
                result = eval(expr)
                ste.set(str(result))

            elif t in ['C', 'CE']:
                ste.set('')

            elif t == '⌫':
                ste.set(current[:-1])
                screen.update()

            elif t == '¹/ₓ':
                if current:
                    result = 1 / float(current)
```

```
                    ste.set(str(result))
                    screen.update()

            elif t == 'x²':
                if current:
                    result = float(current) ** 2
                    ste.set(str(result))
                    screen.update()

            elif t == '√x':
                if current:
                    num = float(current)
                    if num < 0:
                        ste.set("Error")
                        screen.update()
                    else:
                        ste.set(str(num ** 0.5))
                        screen.update()

            elif t == '+/-':
                if current:
                    num = float(current)
                    ste.set(str(-num))
                    screen.update()

            elif t == '%':
                if current:
                    num = float(current)
                    ste.set(str(num / 100))
                    screen.update()

            else:
                ste.set(current + t)
                screen.update()

        except Exception as e:
            ste.set("Error")
            screen.update()

    def click_simulate_equal():
        expr = ste.get().replace('x', '*').replace('÷', '/')
        try:
            result = eval(expr)
            ste.set(str(result))
            screen.update()
        except:
            ste.set("Error")
            screen.update()



    calc = Toplevel()
    calc.iconphoto(True,calpic)
    calc.title("Calculator")
```

```
    calc.config(bg="#00C4B0")
    calc.geometry("+800+350")

    screen = Entry(calc, font='lucida 28 bold', textvariable=ste,
justify=RIGHT,bg="#91FFF4",fg="#001D1A")
    screen.config(width=12)
    screen.grid(row=0, column=0, columnspan=4)


    buttons = [
        ['%', 'CE', 'C', '⌫'],
        ['¹/ₓ', 'x²', '√x', '÷'],
        ['7', '8', '9', 'x'],
        ['4', '5', '6', '-'],
        ['1', '2', '3', '+'],
        ['+/-', '0', '.', '=']
    ]

    for i in range(len(buttons)):
        for j in range(len(buttons[i])):
            b = Button(calc, text=buttons[i][j], font='lucida 20 bold', relief='flat',
bd=0, highlightthickness=0,

bg="#00C4B0",activebackground="#00C4B0",fg="#00312C",activeforeground='#00574E')
            b.grid(row=i+1, column=j)
            b.bind("<Button-1>", click)
    screen.focus_set()
    calc.bind("<Return>", lambda event: click_simulate_equal())




def fundang():    #afun easter egg just for whow may delete in future
    dangwin=Toplevel()
    dangwin.geometry("800x500+300+150")
    dangwin.config(bg="#007166")
    danglab=Label(dangwin,image=dangerpic,bg='#007166')
    danglab2=Label(dangwin,text="WHY DID YOU OPENED IT ",font=(30),bg='#007166')
    danglab3=Label(dangwin,text="WHY WHY WHY ... ",font=(30),bg='#007166')
    danglab4=Label(dangwin,text="WHY YOU KILLED ME ",font=(30),bg='#007166')
    danglab.pack(fill=BOTH)
    danglab2.pack(fill=BOTH)
    danglab3.pack(fill=BOTH)
    danglab4.pack(fill=BOTH)

    dangwin.update()  # Force window to render before freezing bcs as soon as fucntion is
loaded it frezzes bcs of sleep
    time.sleep(5)
    quit()


def open_folder(path):   #open actual path of folder
    explorer = Toplevel()
```

```
    explorer.title(f"Explorer - {os.path.basename(path)}")  # Set window title as folder name   f'{x}' is use to incoperate variable in strig
    explorer.geometry("800x500+300+150")                    # Size and position
    explorer.config(bg="#007166")                           # FKOS theme
    explorer.iconphoto(True, mypcpic)                       # Same icon as "My PC"
    current_path = StringVar(value=path)  # Track current folder path     does like intvar stringvar is use to store dynamic text

    listbox=Listbox(explorer,bg="#007166",font=('Segoe UI',18,'bold'),selectbackground='#21d1b3',relief="flat", highlightthickness=0, bd=0)
    listbox.pack(fill=BOTH, expand=True, padx=10, pady=10)
    #fill=both will allow to listbox to expand in both side in window
    #expand =true allow listbox to take as much space as required
    #pad x/y adds padding as before


    def list_folder(p):
        try:
            files=os.listdir(p)   #get the content of folder
        except:
            messagebox.showerror("ERROR",f"file {p} not found") #messagebox.showerror("title","message")
            return
        listbox.delete(0,END)     #clear old entries
        for f in files:
            listbox.insert(END,f)
        current_path.set(p)    #update tracker to show which folder you re in
        win.config(cursor='dotbox')


    def double_click(event):
        selected=listbox.get(listbox.curselection())    #what user double click
        new_path=os.path.join(current_path.get(),selected)     #create full path

        if os.path.isdir(new_path):    # checks if its a folder and if it is go inside
            list_folder(new_path)
        else:
            try:
                os.startfile(new_path)  #if files open by default
            except:
                messagebox.showerror("ERROR",f"File {new_path} can't be accessed")
    listbox.bind("<Double-Button-1>",double_click)
    list_folder(path)


def mypc():
    pc=Toplevel()
    fpic_local=PhotoImage(file='folder 100.png').zoom(2,2)
    pc.config(background='#007166')
    pc.geometry('1200x720+200+100')
    pc.title('My pc')
    pc.iconphoto(True,mypcpic)
    folA=Button(pc,
                image=fpic_local,
```

```
                bg="#007166",activebackground='#007166',
                relief='flat',bd=0,highlightthickness=0,
                command=lambda:open_folder(r'F:\folder A'),
                text='Folder A',fg='white',font=('Segoe UI',18,'bold'),compound='top',)
#open actual folder    r'F:/folderA' is use bcs \f is escape command
    folA.image = fpic_local   #preserve the image
    folA.place(x=45,y=30)
    folB=Button(pc,
                image=fpic_local,
                bg="#007166",activebackground='#007166',
                relief='flat',bd=0,highlightthickness=0,
                command=lambda:open_folder(r'F:\folder B'),
                text='Folder B',fg='white',font=('Segoe UI',18,'bold'),compound='top')
#open actual folder    r'F:/folderA' is use bcs \f is escape command
    folB.image = fpic_local   #preserve the image
    folB.place(x=345,y=30)



win=Tk()

walp=PhotoImage(file="wallpaper.png")
fpic=PhotoImage(file='folder.png')
mypcpic=PhotoImage(file='my pc.png')
calpic=PhotoImage(file="calculator.png")
searchpic=PhotoImage(file='search40.png')
dangerpic=PhotoImage(file="danger100.png")
userpic=PhotoImage(file='user logo.png')

win.attributes('-fullscreen',True)
wallpaper=Label(image=walp)
wallpaper.place(x=0,y=0)
icon=PhotoImage(file='logo.png')
win.iconphoto(True,icon)
win.title("FKOS")
tpc=Button(image=mypcpic,bg='#6be4ce',activebackground='#6be4ce',
           relief='flat',bd=0,highlightthickness=0,         #for making button flat
           text='My pc',font=('Segoe UI',9,'bold'),compound='top',
           command=mypc)
tpc.place(x=15,y=25)


tcalc=Button(image=calpic,bg='#6be4ce',activebackground='#6be4ce',
             relief='flat',bd=0,highlightthickness=0,         #for making button flat
             text='Calculator',font=('Segoe UI',9,'bold'),compound='top',command=calculate)
tcalc.place(x=15,y=150)


dang=Button(win,image=dangerpic,text="DON'T OPEN",font=('Segoe UI',9,'bold'),
             relief='flat',bd=0,highlightthickness=0,bg='#6be4ce',
             activebackground='#6be4ce',compound='top',command=fundang)
dang.place(x=15,y=275)


```

```
taskbar1=Label(win,bg="#37FFEB",height=37,width=130)
taskbar1.place(x=0,y=1043)

taskbar2=Label(win,bg="#37FFEB",height=37,width=145)
taskbar2.place(x=1000,y=1043)


searchbar=Entry(win,font=('Segoe UI',20,),fg="gray",
                relief='flat',bd=0,highlightthickness=0)
searchbar.insert(0,'Search Anything')
searchbar.config(width=23)
searchbar.place(x=740,y=1043)

delsearch=Button(win,text='X',font=("Verdana", 20, "bold"),fg="gray",
                 bg='#FFFFFF',activebackground='#FFFFFF',
                 relief='flat',bd=0,highlightthickness=0,command=delsear)
delsearch.place(x=1086,y=1043)

searchbut=Button(win,image=searchpic,bg='#37FFEB',activebackground='#37FFEB',
                 relief='flat',bd=0,highlightthickness=0,command=search)
searchbut.place(x=1136,y=1043)
searchbar.bind("<Return>", lambda event: search())


entryscreen=Frame(win,bg='#b1f8eb')
entryscreen.pack(fill=BOTH,expand=True)                  #   for experiment of login
screen simulation
user=Label(entryscreen,image=userpic,bg="#b1f8eb",text='Wellcome User',
           fg="#004639",font=('lucida handwriting', 20 ,'bold'),compound=TOP)
user.place(relx=0.5,rely=0.5,anchor=CENTER)

signf=Frame(entryscreen,bg='#b1f8eb')

signin=Entry(signf,fg='gray',bg='#6be4ce',font=('impact',18))
signin.insert(0,'Password')

signin.pack(side=LEFT)

signin.bind('<Return>',lambda event: checkpass())

sbut=Button(signf,fg="#1d5248",activeforeground='#1d5248',bg='#6be4ce',
            activebackground='#6be4ce',text='ENTER',font=('impact',13),
            relief='flat',bd=0,highlightthickness=0,command=checkpass)
sbut.pack(side=LEFT)
signf.place(relx=0.5,rely=0.6,anchor=CENTER)
signf.lift()
signin.lift()




win.config(cursor='dotbox')
win.bind_all('<Button-1>', global_click_handler, add='+')  #call all clicks

```

```
363
364 bl=Frame(entryscreen,bg='black')
365 bl.pack(fill=BOTH,expand=True)
366 logo=Label(bl,image=icon,bg='black')
367 logo.place(relx=0.5,rely=0.5,anchor=CENTER)
368 bl.lift()
369 logo.lift()
370 win.update()
371 time.sleep(2)
372 bl.destroy()
373
374
375 win.mainloop()
376 #all icons used are from icon8.com
```