3 starter codes necessary

1. Import tkinter
2. Window =tkinter.Tk() ## the name can be anything, window, win, root is common (opens a window)
3. window.title("LCD screen control") ##change the title of your window. Default is tk if this command is not run.
4. window.mainloop() ## name and then mainloop. This has to be at the end

You can write

Import tkinter as tk. ## this allows you to write tk instead of tkinter so
window=tk.Tk()

Create labels

Go to tkinter file and search for what you want
Tk : label, buttons, radio buttons
Ttk: another folder with GUI stuff

First import the file

From tkinter import ttk

Command

ttk.Label(window, text="type your name:") ## which window and what text you want

Need to tell where to put the label

You can use pack or grid
If you use pack command is

ttk.Label(window, text= "xyz").pack() ## this puts it in the center and then thats the size of the window

If you use grid command

ttk.Label(window, text= "xyz").grid(row=0,column=0)

If you want to name the label:

name_label= ttk.Label(window, text= 'xyz')
name _label.grid(row=0,column=0)

To get another label right below it

age_label=ttk.label(window, text='enter your age')
age_label.grid(row=1,column=0)

To get the label to position to the right or left

name_label.grid(row=0,column=0,sticky=tk.w) ## w for west

To create an entry box

name_entrybox= ttk.Entry(win, width= 16)
name _entrybox.grid(row=0, column=1)

To store entry

name_var=tk.StringVar()
name_entrybox= ttk.Entry(win, width= 16, textvariable = name_var)

name _entrybox.grid(row=0, column=1)

To create a button

submit_button=ttk.Button(win, text='Submit')

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

To make the button take action

```
Def action(): ## define the action
    Username = name_var.get()
    Userage = age_var.get()
    print(f'{username} is {userage} years old')
submit_button=ttk.Button(win,text='submit',command=action)
```

For multiple selection, or dropdown menu, create a combobox

```
gender_combobox= ttk.Combobox(win,width=16)
gender_combobox['values']=('Male','Female','other')
gender_combobox.grid(rows=3,column=0)
```

For users not to type in combobox space

```
gender_combobox= ttk.Combobox(win,width=16, textvariable=gender_var,
state='readonly')
```

To focus the cursor

```
name_entrybox.focus()
```

To have a option already selected on combobox

```
gender_combobox.current(0)
```

To create a radio button

```
usertype=tk.StringVar()
radiobtn1=ttk.Radiobutton(win, text='Student',value='Student',variable=usertype)
radiobtn1.grid(row=3,column=0)

radiobtn2=ttk.Radiobutton(win, text='Teacher',value='Teacher',variable=usertype)
radiobtn2.grid(row=3,column=1)
```

To create a check button

```
checkbtn_var=tk.IntVar()
checkbtn=ttk.Checkbutton(win, text='subscribe to our newsletter',
variable=checkbtn_var)
checkbtn.grid(row=4,columnspan=3)
```

To save the info on a file

```
        def action():
          username=name_var.get()
         userage = age_var.get()
          print(f'{username} is {userage} years old')
          user_gender=gender_var.get()
            user_type=usertype.get()
  if checkbtn_var.get()==0:
    subscribed ='NO'
  else:
    subscribed = 'YES'
  print(user_gender,user_type,subscribed)

  with open('file.txt','a') as f:
    f.write(f'{username},{userage},{user_gender},{user_type},{subscribed},\n')
```

To delete the entry once the button is pressed

Use the delete command in the action space

To change the color of the label once the button is pressed

```
name_label.configure(foreground='Blue')
```

To pick your color you need the hex value

Here is a good website: https://htmlcolorcodes.com/color-picker/

Then the code will change to this

```
name_label.configure(foreground='#991E1E') ## the number is the generated hex value
```

For this to work, the widget has to be generated with tk, not ttk