

Digital Signal Processing Lab

Demo 16 (Tkinter)

Saad Zubairi
shz2020

October 15, 2025

Solution

This program creates a simple graphical user interface using Tkinter. It includes all the main widgets shown in the demo programs: Label, Button, Entry, and Scale, plus one extra widget: a Checkbutton.

The user types their name in the entry box, adjusts the volume using the slider, and can choose whether sound is enabled or not with the checkbutton. When the “Submit” button is pressed, a label displays a message with the name and current volume setting. The “Quit” button closes the window.

This small program shows how different Tkinter widgets can work together to make an interactive GUI.

```
1 import tkinter as Tk
2
3 def update_label():
4     name = entry.get()
5     if sound_var.get():
6         result.set(f"Hi {name}, volume set to {scale.get()}")
7     else:
8         result.set(f"Hi {name}, mute mode. Volume ignored.")
9
10 root = Tk.Tk()
11 root.title("Sample Tkinter GUI")
12 root.geometry("300x350")
13
14 # Tk variables
15 result = Tk.StringVar()
16 sound_var = Tk.BooleanVar(value=True)
17
18 # Widgets
19 label_title = Tk.Label(root, text="Welcome to My GUI", font=("Arial", 14, "bold")
20 )
21 label_name = Tk.Label(root, text="Enter your name:")
22 entry = Tk.Entry(root)
23 scale = Tk.Scale(root, from_=0, to=100, orient=Tk.HORIZONTAL, label="Volume")
24 check_sound = Tk.Checkbutton(root, text="Enable sound", variable=sound_var)
25 button = Tk.Button(root, text="Submit", command=update_label)
26 label_result = Tk.Label(root, textvariable=result, fg="blue")
27
28 # Layout
29 label_title.pack(pady=5)
30 label_name.pack()
31 entry.pack()
32 scale.pack()
33 check_sound.pack()
34 button.pack(pady=5)
35 label_result.pack(pady=10)
36 Tk.Button(root, text="Quit", command=root.quit, background="#DEA4A4", width=5).
37     pack(fill=Tk.X)
38
39 root.mainloop()
```