Python Programming
Unit 04 – Lecture 01 Notes
Tkinter Introduction and Widgets

Tofik Ali

February 14, 2026

Contents

1 Lecture Overview

Tkinter is Python's built-in GUI toolkit. In this unit we learn to build desktop applications such as:

- calculators,
- forms (registration, login),
- and small utilities.

This lecture introduces the Tkinter window, the event loop, and commonly used widgets.

2 Core Concepts

2.1 Basic Tkinter Program Structure

Every Tkinter app has:

- a root window (`tk.Tk()`),
- widgets attached to the window,
- and an event loop (`mainloop()`).

```python
import tkinter as tk

root = tk.Tk()
root.title("My App")
root.geometry("400x200")
root.mainloop()
```

2.2 Widgets

Widgets are UI components. Examples:

- `Label`: show text
- `Button`: clickable action
- `Entry`: single-line input
- `Text`: multi-line input/output
- `Listbox`: list selection
- `Canvas`: drawing
- `Checkbutton`, `Radiobutton`: choices

2.3 Control Variables: `StringVar`, `IntVar`

Tkinter provides variables that connect widget values to your code:

- `StringVar` for text
- `IntVar` for integers (often used for check/radio)

```python
name_var = tk.StringVar()
entry = tk.Entry(root, textvariable=name_var)
```

Now you can read `name_var.get()` and set `name_var.set("...")`.

2.4 The Event Loop

`mainloop()` listens for events:

- button clicks,
- key presses,
- window close,
- etc.

Without `mainloop()`, your window appears briefly (or not at all) and the program ends.

2.5 ttk Widgets

`tkinter.ttk` provides themed widgets like `Combobox` (dropdown). They often look nicer and more modern.

3 Demo Walkthrough

File: `demo/widgets_showcase.py`

This demo shows multiple widgets in one window. Focus on:

- how widget values are read (`StringVar.get()`),
- how button callbacks work,
- and how selection widgets (check/radio/combobox) store values.

4 Interactive Checkpoints (with Solutions)

Checkpoint 1 Solution

Question: Why do we need `root.mainloop()`?

Answer: It starts the Tkinter event loop, keeps the window alive, and processes user events.

Checkpoint 2 Solution

Question: Benefit of using `StringVar` with `Entry`?

Answer: It provides an easy way to get/set widget values and keeps the UI and code connected.

5 Practice Exercises (with Solutions)

Exercise 1: Simple Greeting GUI

Task: Create an Entry for name and a Button that prints `"Hello <name>"`.

Solution (idea):

```python
import tkinter as tk

root = tk.Tk()
name = tk.StringVar()

tk.Label(root, text="Name:").pack()
tk.Entry(root, textvariable=name).pack()

def greet():
    print("Hello", name.get())

tk.Button(root, text="Greet", command=greet).pack()
root.mainloop()
```

Exercise 2: Checkbox + Radio

Task: Add one checkbox "Agree" and two radio options "Male/Female". Print selected values when a button is clicked.

Solution (outline):

```python
agree = tk.IntVar()
gender = tk.StringVar(value="Male")

tk.Checkbutton(root, text="Agree", variable=agree).pack()
tk.Radiobutton(root, text="Male", variable=gender, value="Male").pack()
tk.Radiobutton(root, text="Female", variable=gender, value="Female").pack()
```

6 Exit Question (with Solution)

Question: 3 lines to create a window titled "Hello GUI" and keep it running.

Solution:

```python
root = tk.Tk()
root.title("Hello GUI")
root.mainloop()
```