1. What is Tkinter?

Explanation:

Tkinter is Python's standard GUI (Graphical User Interface) toolkit. It allows you to create windows, buttons, labels, text fields, etc., in a desktop application.

Example:

```
from tkinter import *


root = Tk()

root.title("My First Tkinter App")

root.geometry("300x200")

root.mainloop()
```

- Tk(): Initializes the main application window.
- title(): Sets window title.
- geometry(): Sets size as "widthxheight".
- mainloop(): Starts the event loop (keeps window open).

---

2. Widgets in Tkinter

Widgets are the building blocks of any GUI app.

Widget	**Purpose**
Label	Display text/image
Button	Trigger an action
Entry	Single-line text input
Text	Multi-line text input
Checkbutton	Checkbox for multiple options
Radiobutton	Single choice among options
Listbox	List of options

---

3. Label Widget

Explanation:

Used to display static text or images.

Example:

```
from tkinter import *


root = Tk()

root.title("Label Example")


Label(root, text="Welcome to Tkinter!").pack()

root.mainloop()
```

- pack(): Automatically places the widget (top-down).

---

4. Entry Widget

Explanation:

Used to take single-line text input from the user.

Example:

```
from tkinter import *


root = Tk()

root.title("Entry Example")


Label(root, text="Enter your name:").pack()

entry = Entry(root)

entry.pack()


root.mainloop()
```

---

5. Button Widget (with Action)

Explanation:

Used to perform an action when clicked using command.

Example:

```
from tkinter import *


def greet():
    name = entry.get()
    label.config(text=f"Hello {name}!")


root = Tk()
root.title("Button Example")


entry = Entry(root)
entry.pack()


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


label = Label(root)
label.pack()


root.mainloop()
```

---

6. Geometry Managers – pack(), grid(), place()

- ***pack() – Default top-down layout***

```
Label(root, text="Packed label").pack()
```

- ***grid() – Use rows and columns***

```
Label(root, text="Username").grid(row=0, column=0)
Entry(root).grid(row=0, column=1)
```

- ***place() – Absolute positioning***

```
Label(root, text="Fixed").place(x=50, y=30)
```

Sr. No.	Widget/Module	Description
1	**Button**	Used to display buttons in the application.
2	**Canvas**	Used to draw shapes like lines, ovals, polygons, and rectangles.
3	**Checkbutton**	Displays checkboxes for multiple selection options.
4	**Entry**	A single-line text field for user input.
5	**Frame**	A container widget to organize other widgets.
6	**Label**	Provides a single-line caption; can also display images.
7	**Listbox**	Displays a list of selectable options.
8	**Menubutton**	Displays menus in the application.
9	**Menu**	Contains and organizes menu commands under a Menubutton.
10	**Message**	Displays multiline text (usually non-editable).
11	**Radiobutton**	Displays radio buttons; allows only one option to be selected.
12	**Scale**	A slider widget for numeric value selection.
13	**Scrollbar**	Adds scroll functionality to other widgets like Text or Listbox.
14	**Text**	Allows multi-line text input or display.
15	**Toplevel**	Creates a separate window container.
16	**Spinbox**	An Entry variant that allows selecting from a fixed set of values.
17	**PanedWindow**	A container that can hold multiple panes, either vertically or horizontally.
18	**LabelFrame**	A container widget used to group and label related widgets.
19	**tkMessageBox**	Displays various types of message boxes like info, warning, or error.

CALCULATOR USING TKINTER:

```
from tkinter import *


def calculate():
    try:
        a = float(entry1.get())
        b = float(entry2.get())
        op = operator.get()
        if op == '+':
            result = a + b
        elif op == '-':
            result = a - b
        elif op == '*':
            result = a * b
        elif op == '/':
            result = a / b
        else:
            result = "Invalid operator"
        result_label.config(text=f"Result: {result}")
    except:
        result_label.config(text="Error in input")


root = Tk()
root.title("Simple Calculator")


Label(root, text="First Number").pack()
entry1 = Entry(root)
```

```
entry1.pack()

Label(root, text="Second Number").pack()
entry2 = Entry(root)
entry2.pack()

Label(root, text="Operator (+ - * /)").pack()
operator = Entry(root)
operator.pack()

Button(root, text="Calculate", command=calculate).pack()
result_label = Label(root, text="")
result_label.pack()

root.mainloop()
```

CALCULATOR USING COLOR AND FONT:

```
from tkinter import *

def calculate():
    try:
        a = float(entry1.get())
        b = float(entry2.get())
        op = operator.get()
        if op == '+':
            result = a + b
        elif op == '-':
            result = a - b
        elif op == '*':
            result = a * b
```

```
        elif op == '/':
            result = a / b
        else:
            result = "Invalid operator"
        result_label.config(text=f"Result: {result}", fg="green")
    except:
        result_label.config(text=" Error: Invalid input", fg="red")

root = Tk()
root.title("Simple Calculator")
root.geometry("350x300")
root.configure(bg="#f0f4f7")

# Heading
Label(root, text="Simple Calculator", font=("Arial", 16, "bold"), bg="#f0f4f7", fg="#333").grid(row=0, column=0, columnspan=2, pady=10)

# First Number
Label(root, text="Enter First Number:", font=("Arial", 12), bg="#f0f4f7").grid(row=1, column=0, sticky="e", padx=10, pady=5)
entry1 = Entry(root, font=("Arial", 12), width=15)
entry1.grid(row=1, column=1, pady=5)

# Second Number
Label(root, text="Enter Second Number:", font=("Arial", 12), bg="#f0f4f7").grid(row=2, column=0, sticky="e", padx=10, pady=5)
entry2 = Entry(root, font=("Arial", 12), width=15)
entry2.grid(row=2, column=1, pady=5)

# Operator
Label(root, text="Operator (+ - * /):", font=("Arial", 12), bg="#f0f4f7").grid(row=3, column=0, sticky="e", padx=10, pady=5)
```

```
operator = Entry(root, font=("Arial", 12), width=15)

operator.grid(row=3, column=1, pady=5)


# Calculate Button

Button(root, text="Calculate", font=("Arial", 12, "bold"), bg="#4CAF50", fg="white",
command=calculate).grid(row=4, column=0, columnspan=2, pady=15)


# Result Display

result_label = Label(root, text="", font=("Arial", 12, "bold"), bg="#f0f4f7")

result_label.grid(row=5, column=0, columnspan=2)


root.mainloop()
```

Practice Questions:

- Build a GUI to convert Celsius to Fahrenheit.
- Create a login form with entry fields for username and password. Show “Login successful” on correct input, else show “Try again”.
- Create a label and three buttons: Red, Green, Blue. Clicking a button should change the label's background color accordingly.
- Create a Text widget. Add a button that shows the total word count of the text entered.

PROJECT : CURRENCY CONVERTER USING TKINTER:

```
from tkinter import *

from tkinter import ttk


# Dummy conversion rates (for demonstration)

conversion_rates = {

   "USD": {"INR": 83.2, "EUR": 0.92, "USD": 1},

   "INR": {"USD": 0.012, "EUR": 0.011, "INR": 1},

   "EUR": {"USD": 1.09, "INR": 90.3, "EUR": 1}
```

```
}

def convert():
    try:
        amt = float(entry.get())
        from_curr = from_currency.get()
        to_curr = to_currency.get()
        rate = conversion_rates[from_curr][to_curr]
        result = amt * rate
        result_label.config(text=f"{amt} {from_curr} = {round(result, 2)} {to_curr}", fg="green")
    except:
        result_label.config(text="Error: Invalid input", fg="red")

root = Tk()
root.title("💱 Currency Converter")
root.geometry("360x300")
root.configure(bg="#eef2f3")

# Heading
Label(root, text="Currency Converter", font=("Arial", 16, "bold"), bg="#eef2f3", fg="#333").pack(pady=10)

# Amount input
Label(root, text="Enter Amount:", font=("Arial", 12), bg="#eef2f3").pack(anchor="w", padx=30)
entry = Entry(root, font=("Arial", 12), width=25)
entry.pack(pady=5)
```

```
# From currency
Label(root, text="From Currency:", font=("Arial", 12), bg="#eef2f3").pack(anchor="w", padx=30)
from_currency = ttk.Combobox(root, font=("Arial", 12), values=["USD", "INR", "EUR"], state="readonly")
from_currency.current(0)
from_currency.pack(pady=5)

# To currency
Label(root, text="To Currency:", font=("Arial", 12), bg="#eef2f3").pack(anchor="w", padx=30)
to_currency = ttk.Combobox(root, font=("Arial", 12), values=["USD", "INR", "EUR"], state="readonly")
to_currency.current(1)
to_currency.pack(pady=5)

# Convert button
Button(root, text="Convert", font=("Arial", 12, "bold"), bg="#4CAF50", fg="white", command=convert).pack(pady=15)

# Result
result_label = Label(root, text="", font=("Arial", 12, "bold"), bg="#eef2f3")
result_label.pack()

root.mainloop()
```

GITHUB LINK: https://github.com/Aditya2129939/Python-Training-BFGI