

CA-208 Python Lab
MCA (2021-24) , Semester –II

Python GUI (Tkinter) Presentation

Submitted to:

Prof. Anirban Choudhary sir

Submitted by:

Neelam(212120107)

My Application perform these task

- Calculate the value of given function $f(x)=ax^2+bx+c$
- Find the roots of the equation $f(x)=0$
- By taking input of x, a, b and c from user

Python GUI – tkinter

Python offers multiple options for developing **GUI (Graphical User Interface)**. Out of all the GUI methods, tkinter is the most commonly used method. **It is a standard Python interface to the Tk GUI toolkit shipped with Python.** Python with tkinter is the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.

To create a tkinter app :

1. Importing the module – tkinter
2. Create the main window (container)
3. Add any number of widgets to the main window
4. Apply the event Trigger on the widgets.

```
from tkinter import *  
  
root=Tk()  
  
root.mainloop()
```

There are two main methods used while creating the Python application with GUI.

1. Tk().

The basic code used to create the main window of the application is:

root = tkinter.Tk()

where root is the name of the main window object

2. **mainloop()**

There is a method known by the name `mainloop()` is used when your application is ready to run. `mainloop()` is an infinite loop used to run the application, wait for an event to occur and process the event as long as the window is not closed.

root.mainloop()

Tkinter also offers access to the geometric configuration of the widgets which can organize the widgets in the parent windows.

There are mainly three geometry manager classes :

1. **pack() method**

It organizes the widgets in blocks before placing in the parent widget.

2. **grid() method**

It organizes the widgets in grid (table-like structure) before placing in the parent widget.

3. **place() method**

It organizes the widgets by placing them on specific positions directed by the programmer.

Widgets which I have used in my application are :

1. **Button:**

To add a button in your application, this widget is used.

The general syntax is:

w = Button(master, option = value)

master is the parameter used to represent the parent window.

There are number of options which are used to change the format of the Buttons. Number of options can be passed as parameters separated by commas. Some of them are listed below.

- **activebackground** :- to set the background color when button is under the cursor.
- **activeforeground** :- to set the foreground color when button is under the cursor.
- **bg**:- to set the normal background color.
- **command**:- to call a function.
- **font**:- to set the font on the button label.
- **image**:- to set the image on the button.
- **width**:- to set the width of the button.
- **height**:- to set the height of the button.

```
Button( text="Solve for x", font="comicsans 10", relief=RAISED,
command=solve).place( x=60, y=210)
Button( text="Find roots", font="comicsans 10", relief=RAISED,
command=roots).place( x=160, y=210)
Button(text="Reset",relief=RAISED, font="comicsans 10", command=reset).place(
x=260, y=210)
Button( text="Plot Graph", font="comicsans 10", relief=RAISED,
command=roots).place( x=360, y=210)
```

2. Entry:

It is used to input the single line text entry from the user. For multi-line text input, Text widget is used.

The general syntax is:

$$w = \text{Entry}(\text{master}, \text{option} = \text{value})$$

master is the parameter used to represent the parent window.

There are number of options which are used to change the format of the widget. Number of options can be passed as parameters separated by commas. Some of them are listed below.

- **bd**:- to set the border width in pixels.
- **bg**:- to set the normal background color.
- **cursor**:- to set the cursor used.
- **command**:- to call a function.
- **highlightcolor**:- to set the color shown in the focus highlight.
- **width**:- to set the width of the button.
- **height**:- to set the height of the button.

```
x_value=IntVar()  
Entry( textvariable=x_value).place(x=150, y=70)  
a_value=IntVar()  
Entry( textvariable=a_value).place(x=150, y=100)  
b_value=IntVar()  
Entry( textvariable=b_value).place(x=150, y=130)  
c_value=IntVar()  
Entry( textvariable=c_value).place(x=150, y=160)
```

3. Frame:

It acts as a container to hold the widgets. It is used for grouping and organizing the widgets. The general syntax is:

$$w = \text{Frame}(\text{master}, \text{option} = \text{value})$$

master is the parameter used to represent the parent window.

There are number of options which are used to change the format of the widget. Number of options can be passed as parameters separated by commas. Some of them are listed below.

- **highlightcolor:** - To set the color of the focus highlight when widget has to be focused.
- **bd:** - to set the border width in pixels.
- **bg:-** to set the normal background color.
- **cursor:-** to set the cursor used.
- **width:-** to set the width of the widget.
- **height:** - to set the height of the widget.

4. Label:

It refers to the display box where you can put any text or image which can be updated any time as per the code.

The general syntax is:

$$w = \text{Label}(\text{master}, \text{option} = \text{value})$$

master is the parameter used to represent the parent window.

```
Label(root, text="f(x) = ax^2+bx+c", font="comicsans 15 bold ", pady=5).grid()
Label( text="Enter the value : ", font="comicsans 12 bold ").grid(row=1,column=0)
Label( text="x : ", font="comicsans 15").grid()
Label( text="a : " , font="comicsans 15").grid()
Label( text="b : " , font="comicsans 15").grid()
Label( text="c : " , font="comicsans 15").grid()
```

Using command

Every Button widget has a ***command*** attribute that you can assign to a function. Whenever the button is pressed, the function is executed.

```
def solve():
    x=x_value.get()
    a=a_value.get()
    b=b_value.get()
    c=c_value.get()
    f=a*x**2+b*x+c
    fx.set(f)

def roots():
    a=a_value.get()
    b=b_value.get()
    c=c_value.get()
    root1=(-b+np.lib.scimath.sqrt(b**2-(4*a*c)))/2*a
    root2=(-b-np.lib.scimath.sqrt(b**2-(4*a*c)))/2*a
    r1.set("{: .2f}".format(root1))
    r2.set("{: .2f}".format(root2))

def reset():
    x_value.set(0)
    a_value.set(0)
    b_value.set(0)
    c_value.set(0)
    fx.set(0)
    r1.set(0.0)
    r2.set(0.0)
```

```

Button( text="Solve for x", font="comicsans 10", relief=RAISED,
command=solve).place( x=60, y=210)
Button( text="Find roots", font="comicsans 10", relief=RAISED,
command=roots).place( x=160, y=210)
Button(text="Reset",relief=RAISED, font="comicsans 10", command=reset).place(
x=260, y=210)
Button( text="Plot Graph", font="comicsans 10", relief=RAISED,
command=roots).place( x=360, y=210)

```

My code:

```

import numpy as np
from tkinter import *

root=Tk()

root.geometry("550x450")
root.maxsize(550,450)
root.title("Play with quadratic function")

# function
def solve():
    x=x_value.get()
    a=a_value.get()
    b=b_value.get()
    c=c_value.get()
    f=a*x**2+b*x+c
    fx.set(f)

def roots():
    a=a_value.get()
    b=b_value.get()
    c=c_value.get()
    root1=(-b+np.lib.scimath.sqrt(b**2-(4*a*c)))/2*a
    root2=(-b-np.lib.scimath.sqrt(b**2-(4*a*c)))/2*a

```

```

r1.set("{: .2f}".format(root1))
r2.set("{: .2f}".format(root2))

def reset():
    x_value.set(0)
    a_value.set(0)
    b_value.set(0)
    c_value.set(0)
    fx.set(0)
    r1.set(0.0)
    r2.set(0.0)

Label(root, text="f(x) = ax^2+bx+c", font="comicsans 15 bold ", pady=5).grid()
Label( text="Enter the value : ", font="comicsans 12 bold ").grid(row=1,column=0)
Label( text="x : ", font="comicsans 15").grid()
Label( text="a : " , font="comicsans 15").grid()
Label( text="b : " , font="comicsans 15").grid()
Label( text="c : " , font="comicsans 15").grid()

x_value=IntVar()
Entry( textvariable=x_value).place(x=150, y=70)
a_value=IntVar()
Entry( textvariable=a_value).place(x=150, y=100)
b_value=IntVar()
Entry( textvariable=b_value).place(x=150, y=130)
c_value=IntVar()
Entry( textvariable=c_value).place(x=150, y=160)

Button( text="Solve for x", font="comicsans 10", relief=RAISED,
command=solve).place( x=60, y=210)
Button( text="Find roots", font="comicsans 10", relief=RAISED,
command=roots).place( x=160, y=210)
Button(text="Reset",relief=RAISED, font="comicsans 10", command=reset).place(
x=260, y=210)
Button( text="Plot Graph", font="comicsans 10", relief=RAISED,
command=roots).place( x=360, y=210)

fx=IntVar()

```



```

Label(text="Value of function f is ", font="comicsans 12 bold ").place( x=26,
y=260)
Label(padx=18, pady=18, borderwidth=1, relief=SUNKEN, textvariable=fx).place(
x=250, y=260)

r1=DoubleVar()
r2=DoubleVar()
Label(text="Roots of equation f(x) = 0 :", font="comicsans 12 bold ").place(
x=26, y=320)
Label(padx=15, pady=15, borderwidth=1, relief=SUNKEN, textvariable=r1).place(
x=250, y=320)
Label(text="and",).place( x=360, y=330)
Label(padx=15, pady=15, borderwidth=1, relief=SUNKEN, textvariable=r2).place(
x=420, y=320)

root.mainloop()

```

Output:

f(x) = ax²+bx+c

Enter the value :

x :

a :

b :

c :

Value of function f is

Roots of equation f(x) = 0 : and