Packing Buttons In Tkinter | Python Tkinter GUI Tutorial In Hindi #9

The **Button** widget is a standard Tkinter widget, which is used for various kinds of buttons. A button is a widget that is designed for the user to interact with (i.e., if a mouse click presses the button, some action might be started). They can also contain text and images like labels. While labels can display text in various fonts, a button can only display text in a single font. The text of a button can span more than one line.

A Python function or method can be associated with a button. This function or method will be executed if the button is pressed in some way. We can use *bd*, *bg*, *command*, *fg, text*, *image*, *relief,* etc., as attributes of the Button widget. Some important attributes are discussed below:

- **bg:** The normal background color displayed behind the label and indicator.

- **relief:** The type of the border of the frame. **Its default value is set to FLAT**. We can set it to any other styles, i.e., *FLAT*, *RAISED*, *SUNKEN*, *GROOVE*, *RIDGE*.
- **bd:** The size of the border in pixels. **Default is 2 pixels**.
- **font:** Text font to be used for the button's label.
- **image:** Instead of text, Image to be displayed on the button.
- **command:** Function or method to be called when the button is clicked. **Note: we only have to write the function or method name in the command attribute. We should not call the function in the attribute (i.e., If the function is *name()*, we just have to write *command=name*).**

Code is described below:

```
from tkinter import *

root =Tk()
root.geometry("655x333")

def hello():
    print("Hello tkinter Buttons")

def name():
    print("Name is harry")

frame = Frame(root, borderwidth=6, bg="grey", relief=SUNKEN)
frame.pack(side=LEFT, anchor="nw")

b1 = Button(frame, fg="red", text="Print now", command=hello)
b1.pack(side=LEFT, padx=23)
```

```
b2 = Button(frame, fg="red", text="Tell me name now", command=name)
b2.pack(side=LEFT, padx=23)

b3 = Button(frame, fg="red", text="Print now")
b3.pack(side=LEFT, padx=23)

b4 = Button(frame, fg="red", text="Print now")
b4.pack(side=LEFT, padx=23)
root.mainloop()
```

- Importing *tkinter* is the same as importing any other module in the Python code. Note that the name of the module in Python 2.x is '*Tkinter*', and in Python 3.x, it is '*tkinter*'.

```
from tkinter import *
```

- To create the main window, Tkinter offers a method, 'Tk'. To change the name of the window, you can change the className to the desired one.

```
root =Tk()
```

- To set the dimensions of the Tkinter window and to set the position of the main window on the user's desktop, the geometry() function is used. As in the example: the width is 655 pixels, and height is 333 pixels, so we can write the function as *geometry(655x333)*.

```
root.geometry("655x333")
```

- To define a function using 'def' (i.e., here, two functions, hello() and name() are defined) and use it in button attributes.

```
def hello():
    print("Hello tkinter Buttons")

def name():
    print("Name is harry")
```

- To take Frame variable as *frame* and set the attributes- *bg* (background color) = "**grey**", *borderwidth*(thickness of the Frame's border)=**6,** and *relief*=**SUNKEN**. Then the Frame *frame* must be packed (here, it is packed on the left side and north-west corner).

```
frame = Frame(root, borderwidth=6, bg="grey", relief=SUNKEN)
frame.pack(side=LEFT, anchor="nw")
```

- To take the Button variable (here we take four Button variables b1, b2, b3 & b4) and use attributes in the Button widget (i.e., for Button b1, the attributes are fg (foreground color) = "red", text= "Print now" and command= hello, such that when the Button name showing as "Print now" is clicked it will call the hello() function through the command "hello" attribute and it will print "Hello tkinter Buttons".

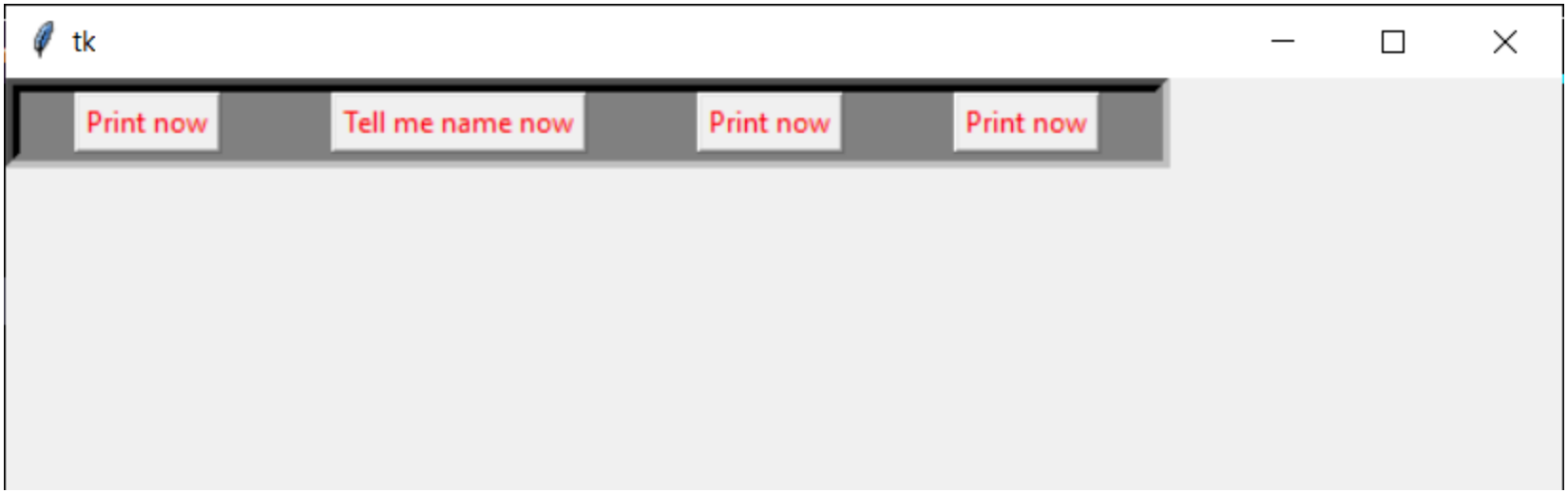
```
b1 = Button(frame, fg="red", text="Print now", command=hello)
b1.pack(side=LEFT, padx=23)
```

```
b2 = Button(frame, fg="red", text="Tell me name now", command=name)
b2.pack(side=LEFT, padx=23)

b3 = Button(frame, fg="red", text="Print now")
b3.pack(side=LEFT, padx=23)

b4 = Button(frame, fg="red", text="Print now")
b4.pack(side=LEFT, padx=23)
```

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

```
root.mainloop()
```

Output: The output of the code (or the GUI window) is given below:



Code as described/written in the video

Copy

```
from tkinter import *

root =Tk()
root.geometry("655x333")

def hello():
    print("Hello tkinter Buttons")

def name():
    print("Name is harry")

frame = Frame(root, borderwidth=6, bg="grey", relief=SUNKEN)
frame.pack(side=LEFT, anchor="nw")

b1 = Button(frame, fg="red", text="Print now", command=hello)
```

```
b1.pack(side=LEFT, padx=23)

b2 = Button(frame, fg="red", text="Tell me name now", command=name)
b2.pack(side=LEFT, padx=23)

b3 = Button(frame, fg="red", text="Print now")
b3.pack(side=LEFT, padx=23)

b4 = Button(frame, fg="red", text="Print now")
b4.pack(side=LEFT, padx=23)
root.mainloop()
```

Previous

Next

CodeWithHarry