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.

Importing tkinter is 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’.

```
import tkinter
```

There are two main methods used which the user needs to remember while creating the Python application with GUI.

1. **Tk(screenName=None, baseName=None, className=’Tk’, useTk=1):** To create a main window, tkinter offers a method ‘Tk(screenName=None, baseName=None, className=’Tk’, useTk=1)’. To change the name of the window, you can change the className to the desired one. The basic code used to create the main window of the application is:
   `m=tkinter.Tk() where m 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.
   `m.mainloop()`

```
import tkinter
m = tkinter.Tk()
'''
widgets are added here
'''
m.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 class.

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.

There are a number of widgets which you can put in your tkinter application. Some of the major widgets are explained below:

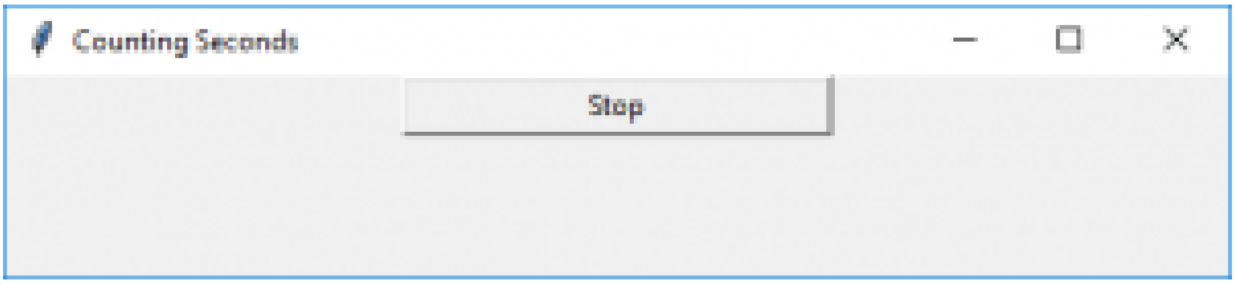
1. **Button**:To add a button in your application, this widget is used.
   The general syntax is:
   `w=Button(master, option=value)`

   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.

```
import tkinter as tk
r = tk.Tk()
r.title('Counting Seconds')
button=tk.Button(r,text='Stop',width=25,command=r.destroy)
button.pack()
r.mainloop()
```

Output:



2]Canvas: It is used to draw pictures and other complex layout like graphics, text and widgets.
The general syntax is:
`w = Canvas(master, option=value)`

master is the parameter used to represent the parent window.

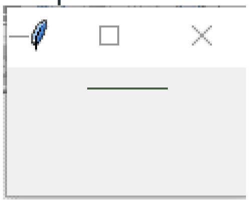
There are number of options which are used to change the format of the widget. Number of options can be passed as parameters separated by commas. Some of them are listed below.

- **bd**: to set the border width in pixels.

- **bg**: to set the normal background color.
- **cursor**: to set the cursor used in the canvas.
- **highlightcolor**: to set the color shown in the focus highlight.
- **width**: to set the width of the widget.
- **height**: to set the height of the widget.

```
from tkinter import *
master = Tk()
w = Canvas(master, width=40, height=60)
w.pack()
canvas_height=20
canvas_width=200
y = int(canvas_height / 2)
w.create_line(0, y, canvas_width, y )
mainloop()
```

Output:



3] **CheckButton:** To select any number of options by displaying a number of options to a user as toggle buttons. The general syntax is:

```
w = CheckButton(master, option=value)
```

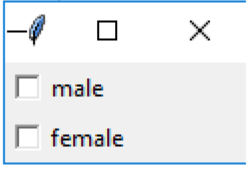
There are number of options which are used to change the format of this widget. Number of options can be passed as parameters separated by commas. Some of them are listed below.

- **Title**: To set the title of the widget.
- **activebackground**: to set the background color when widget is under the cursor.
- **activeforeground**: to set the foreground color when widget 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 widget.

```
from tkinter import *


master = Tk()
```

```
var1 = IntVar()

Checkbutton(master, text='male', variable=var1).grid(row=0,
sticky=W)

var2 = IntVar()

Checkbutton(master, text='female', variable=var2).grid(row=1,
sticky=W)

mainloop()
```

Output:



1. **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:
2. `w=Entry(master, option=value)`

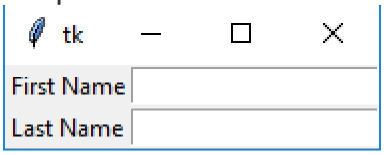
   master is the parameter used to represent the parent window.
   There are number of options which are used to change the format of the widget. Number of options can be passed as parameters separated by commas. Some of them are listed below.

   - **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.

```
from tkinter import *

master = Tk()

Label(master, text='First Name').grid(row=0)

Label(master, text='Last Name').grid(row=1)
```

```
e1 = Entry(master)

e2 = Entry(master)

e1.grid(row=0, column=1)

e2.grid(row=1, column=1)

mainloop()
```

Output:



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

```
w = Frame(master, option=value)
```

```
master is the parameter used to represent the parent window.
```

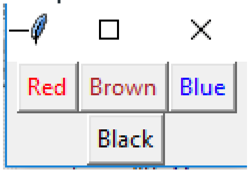
There are number of options which are used to change the format of the widget. Number of options can be passed as parameters separated by commas. Some of them are listed below.

- **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.

```
from tkinter import *



root = Tk()

frame = Frame(root)

frame.pack()
```

```
bottomframe = Frame(root)

bottomframe.pack( side = BOTTOM )

redbutton = Button(frame, text = 'Red', fg ='red')

redbutton.pack( side = LEFT)

greenbutton = Button(frame, text = 'Brown', fg='brown')

greenbutton.pack( side = LEFT )

bluebutton = Button(frame, text ='Blue', fg ='blue')

bluebutton.pack( side = LEFT )

blackbutton = Button(bottomframe, text ='Black', fg ='black')

blackbutton.pack( side = BOTTOM)

root.mainloop()
```

Output:



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=Label(master, option=value)`

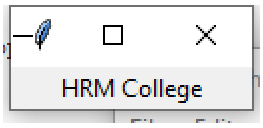
`master is the parameter used to represent the parent window.`

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

- **bg**: to set the normal background color.
- **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.

```
from tkinter import *

root = Tk()

w = Label(root, text='HRM College')

w.pack()

root.mainloop()
```

Output:



HRM College