INDUSTRIAL TRAINING DAILY DIARY
DAY 06

01 June, 2025
Topic : Learning making GUI using tkinter library in python

Introduction :

Python Tkinter is a standard GUI (Graphical User Interface) library for Python which provides a fast and easy way to create desktop applications. Tkinter provides a variety of widgets like buttons, labels, text boxes, menus and more that can be used to create interactive user interfaces. Tkinter supports event-driven programming, where actions are taken in response to user events like clicks or keypresses.

Create First Tkinter GUI Application

To **create a Tkinter Python app**, follow these basic steps:

1.**Import the tkinter module**: Import the tkinter module, which is necessary for creating the GUI components.

2.**Create the main window (container)**: Initialize the main application window using the Tk() class.

3.**Set Window Properties**: We can set properties like the title and size of the window.

4.**Add widgets to the main window**: We can add any number of widgets like buttons, labels, entry fields, etc., to the main window to design the interface.

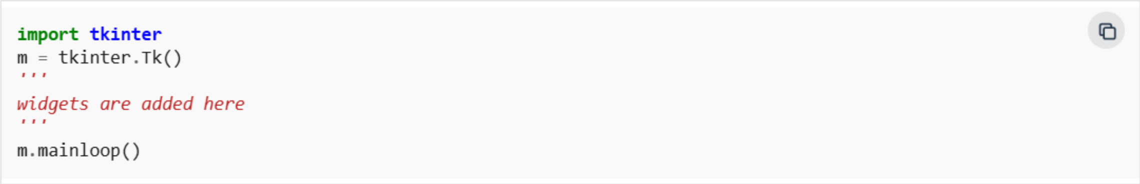
5.**Pack Widgets:** Use geometry managers like pack(), grid() or place() to arrange the widgets within the window.

6.**Apply event triggers to the widgets**: We can attach event triggers to the widgets to define how they respond to user interactions.

Widgets:

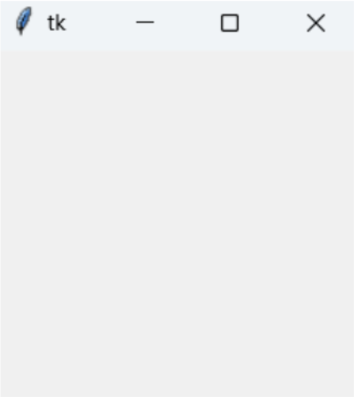
The fundamental building blocks of a Tkinter GUI are called widgets. These are graphical elements that users interact with, such as buttons, labels, text entry fields, frames, canvases, and more. Each widget is an object with its own properties and methods.

Example:



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

Output



Event-Driven Programming:

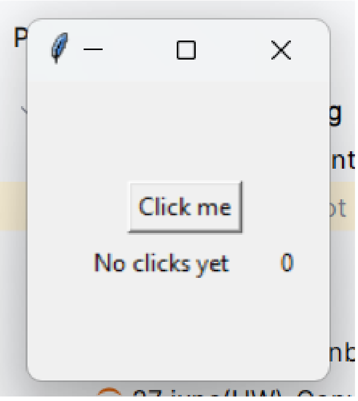
Tkinter operates on an event-driven paradigm. This means that the application waits for user interactions (events) like button clicks, key presses, or mouse movements. When an event occurs, a predefined function (callback) associated with that event is executed. The `mainloop()` method is crucial here, as it continuously listens for and processes these events.

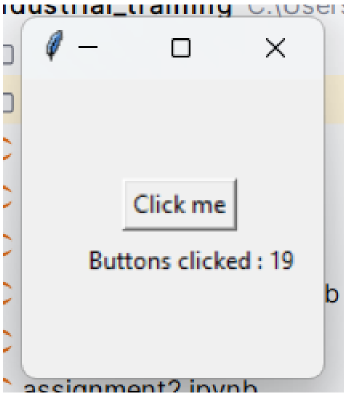
Few examples of tkinter GUIs

1. A GUI to check number of clicks

```
from tkinter import *

from win32con import LOGON32_LOGON_NETWORK

screen = Tk()

def button_press():
    x = l2.cget("text")    # For extracting text from labels, we use cget().  in Entry we use get()
    o = int(x)+1
    l1.config(text = "Buttons clicked : ")
    l2.config(text=str(o))
```

```
screen.geometry("150x150")

B1 = Button(screen,text = "Click me",command = button_press)
B1.place(x=50,y=50)

l1 = Label(screen, text = "No clicks yet")
l1.place(x=30,y=80)

l2 = Label(screen, text = " 0 ")
l2.place(x=120,y=80)

screen.mainloop()
```

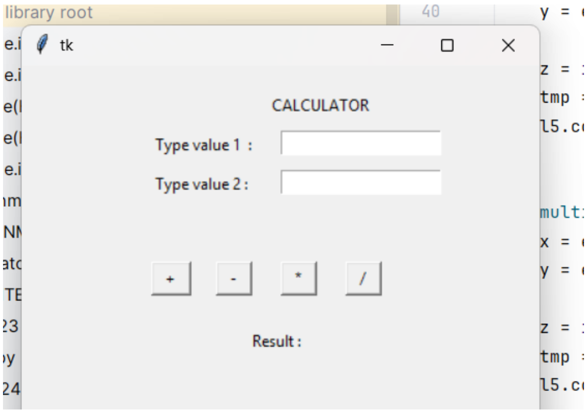


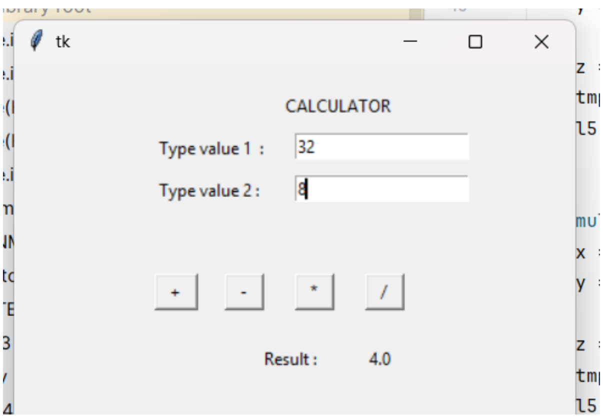


2. GUI of a simple Calculator

```
from tkinter import *
screen = Tk()

def add():
   x = e1.get()
   y = e2.get()

   z = int(x) + int(y)
   tmp = str(z)
   l5.config(text=tmp)


def subtract():
   x = e1.get()
   y = e2.get()

   z = int(x) - int(y)
   tmp = str(z)
   l5.config(text=tmp)


def multiply():
   x = e1.get()
   y = e2.get()

   z = int(x)*int(y)
   tmp = str(z)
```

```
    l5.config(text=tmp)

def divide():
    x = e1.get()
    y = e2.get()

    z = int(x)/int(y)
    tmp = str(z)
    l5.config(text=tmp)

screen.geometry("400x450")

l1 = Label(screen,text = "CALCULATOR")
l1.place(x=190,y=20)

l2 = Label(screen,text = "Type value 1  : ")
l2.place(x=100,y=50)

e1 = Entry(screen)
e1.place(x=200,y=50)

l3 = Label(screen,text = "Type value 2 : ")
l3.place(x=100,y=80)

e2 = Entry(screen)
e2.place(x=200,y=80)

b1 = Button(screen , text = '  +  ' , command  = add)
b1.place(x=100,y=150)

b2 = Button(screen , text = '  -  ' , command = subtract)
b2.place(x=150,y=150)

b3 = Button(screen , text = '  *  ' , command = multiply)
b3.place(x=200,y=150)

b4 = Button(screen , text = '  /  ', command = divide)
b4.place(x=250,y=150)

l4 = Label(screen, text = "Result : ")
l4.place(x= 175,y=200)

l5 = Label(screen, text = '')
l5.place(x=250,y=200)

screen.mainloop()
```





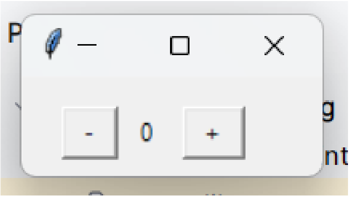
Type value 1 : 32

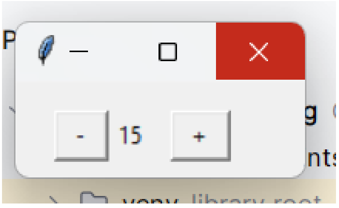
Type value 2 : 8

Result : 4.0

3. GUI of a counter

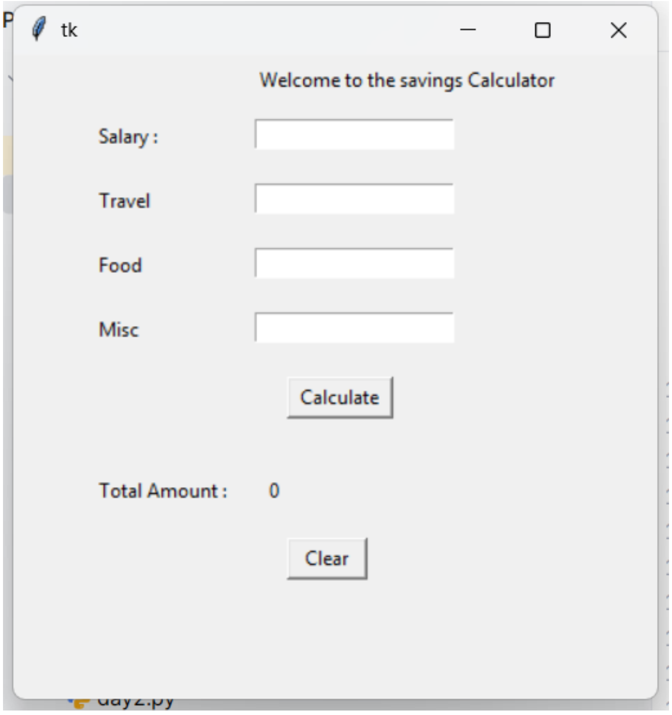
```
from tkinter import *
screen = Tk()

def subtract():
    num = l1.cget("text")
    n = int(num)-1
    l1.config(text = str(n))

def add():
    num = l1.cget("text")
    n = int(num) +1
    l1.config(text = str(n))


screen.geometry("150x50")

b1 = Button(screen,text = '  -  ', command = subtract)
b1.place(x =20 , y =15 )

l1 = Label(screen,text = '  0  ')
l1.place(x =50 , y = 17)

b1 = Button(screen,text = '  +  ',command = add)
b1.place(x =80 , y =15 )

screen.mainloop()
```





4. GUI of Savings Calculator

```python
from tkinter import *

screen = Tk()
def calculate():
    s = e1.get()
    t = e2.get()
    f = e3.get()
    m = e4.get()
    rem = int(s) - int(t) - int(f) - int(m)
    l7.config(text = str(rem))

def clear():
    l7.config(text = ' 0 ')


screen.geometry("400x400")
l1  =  Label(screen,text = "Welcome to the savings Calculator")
l1.place(x=150,y=5)

l2  =  Label(screen,text = "Salary : ")
l2.place(x=50,y=40)

l3  =  Label(screen,text = "Travel")
l3.place(x=50,y=80)

l4  =  Label(screen,text = "Food")
l4.place(x=50,y=120)

l5  =  Label(screen,text = "Misc")
l5.place(x=50,y=160)

e1 = Entry(screen)
e1.place(x=150,y=40)

e2 = Entry(screen)
e2.place(x=150,y=80)

e3 = Entry(screen)
e3.place(x=150,y=120)

e4 = Entry(screen)
e4.place(x=150,y=160)

b1 = Button(screen , text = " Calculate " , command = calculate)
b1.place(x=170 ,y=200 )

l6 = Label(screen,text="Total Amount : ")
l6.place(x = 50,y = 260)

l7 = Label(screen,  text = "  0  ")
l7.place(x = 150, y = 260)

b2 = Button(screen , text = "  Clear  ", command = clear)
b2.place(x = 170, y = 300 )


screen.mainloop()
```



Welcome to the savings Calculator

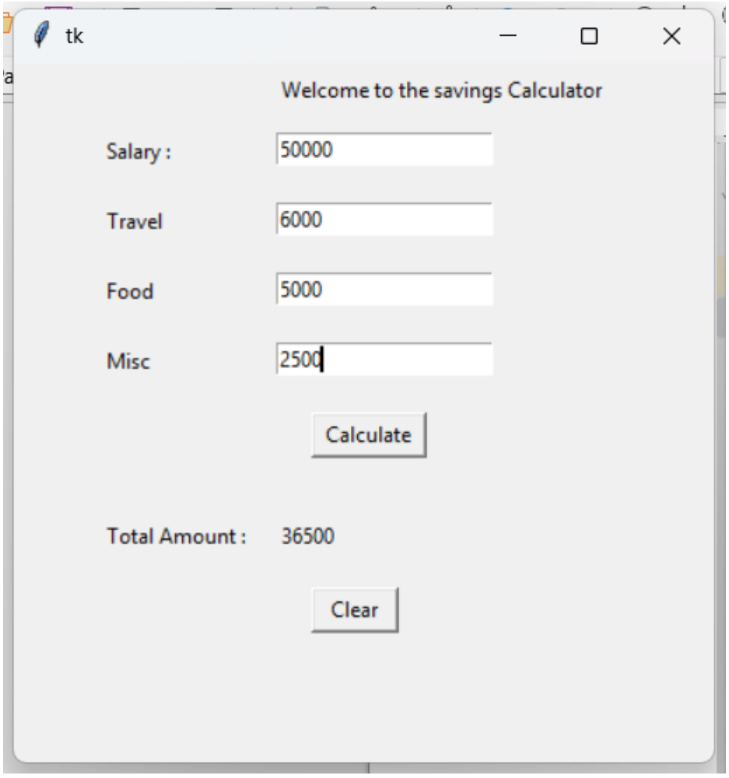
Salary :

Travel

Food

Misc

Total Amount : 0



Welcome to the savings Calculator

Salary : 50000

Travel 6000

Food 5000

Misc 2500

Total Amount : 36500