INTRODUCTION TO TKINTER

What is our GOAL for this CLASS?

In this class, we have learned about the Python -Tkinter () Module. We learned about graphical user interface with the Tkinter module and we created a GUI for BMI- Calculator.

What did we ACHIEVE in the class TODAY?

- Create Design
- Working of GUI
- Fully Functional BMI-Calculator

Which CONCEPTS/ CODING BLOCKS did we cover today?

- Tkinter Module
- Tkinter Widgets
- Functions on Tkinter Widget

How did we DO the activities?

GUI creation consist of two steps:

- Design
- Working

Design part can be accomplished with the below steps.

1. Import Tkinter module

2. Create your application window

3. Add widgets to the window

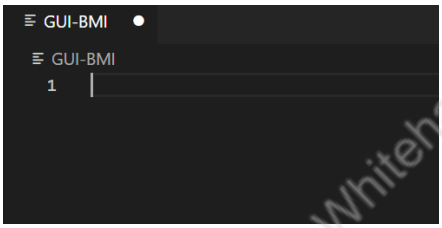
1. Create the widget
2. Place it on screen

4. Call mainloop

Working of GUI : Write a logic for how clicking a button(an action) will calculate your BMI.

Activity 1:

1. Open visual studio code and create one file with name GUI_BMIpy



2. Import tkinter module

```
from tkinter import *
```

3. Create a parent window.

```
window=Tk()
```

4. Create window, size, title and configure parent window.

```
window.title('BMI Calculator')
window.geometry("400x400")
window.configure(bg='lightcyan')
```

5. Create heading label using **Label ()** and then place it on parent window

```
heading_label=Label(window, text="BMI CALCULATOR", fg = "black", bg = "lightcyan", font=("Calibri", 20),bd=5)
heading_label.place(x=50, y=20)
```

6. Create a **Label ()**name label and place it on the parent window.

```
name_label=Label(window, text="Your Name", fg = "black", bg = "lightcyan", font=("Calibri", 12),bd=1)
name_label.place(x=20, y=90)
```

7. Create an Entry **()** entry box for user input and place it on screen.

```
username=Entry(window, text="", bd=2, width=22)
username.place(x=150, y=92)
```

8. Create **Label()** for Height label & **Entry ()** for height entry and place it on the parent window.

```
height_label=Label(window, text="Enter Height (cm)", fg = "black", bg = "lightcyan", font=("Calibri", 12))
height_label.place(x=20, y=140)
height_entry=Entry(window, text="", bd=2, width=15)
height_entry.place(x=150, y=142)
```

9. Create **Label()** Weight label & **Entry ()** weight entry and place it on the parent window.

```
weight_label=Label(window, text="Enter Weight (Kg)", fg = "black", bg = "lightcyan", font=("Calibri", 12))
weight_label.place(x=20, y=185)
weight_entry=Entry(window, text="", bd=2, width=15)
weight_entry.place(x=150, y=187)
```

10. Create **Button ()** widget to calculate BMI

```
calculate_button=Button(window,text="CALCULATE",fg = "black", bg = "cyan",bd=4,command=calculate_bmi)
calculate_button.place(x=20,y=250)
```

11. Create Frame Label (), place it on the parent window and then use **pack (),** to display content inside the frame Label.

```
result_frame = LabelFrame(window,text="Result", bg = "lightcyan", font=("Calibri", 12))
result_frame.pack(padx=20, pady=20)
result_frame.place(x=20,y=300)
```

12. Create a Label () result label() to show output inside result_frame.

```
result_label=Label(result_frame,text=" ", bg = "lightcyan", font=("Calibri", 12), width=33)
result_label.place(x=20,y=20)
result_label.pack()
```

13. Call the mainloop()
    - Window. mainloop ()
    - Create all widgets and then call mainloop.

```
window.mainloop()
```

Activity 2: Working of GUI

Write your event code before all widgets.

1. Write function for Calculate Button
    - Now get the user _weight,
    - Get the user height
    - Use BMI Formula
    - Round off Result
    - Get the name from user

```
def calculate_bmi():
    weight = int(weight_entry.get())
    height = int(height_entry.get())/100
    bmi = weight/(height*height)
    bmi = round(bmi, 1)
    name = username.get()
```

2. Destroy the result label so show output message.

```
result_label.destroy()
```

3. Write conditions to check BMI conditions.

- Create one blank variable
- Write if -else conditions as per below conditions
- **Underweight** = <18.5
- **Normal weight** = 18.5–24.9
- **Overweight** = 25–29.9
- **Obesity** = BMI of 30 or greater

```
msg=""

if bmi < 18.5:
    msg="you areUnderweight"
elif bmi > 18.5 and bmi <=24.9:
  msg="is in Normal Range"
elif bmi > 25 and bmi <=29.9:
  msg="you are Overweight"
elif bmi > 30:
  msg="you are Obese"
else:
  msg="Something Went Wrong"
```


4. Create a LABEL() to display output messages inside your result frame container.

```
output_message=Label(result_frame,text=name+", your BMI is "+str(bmi)+" and "+msg, bg = "lightcyan", font=("Calibri", 12), width=42)
output_message.place(x=20,y=40)
output_message.pack()
```

What's NEXT?

In the next class, we will learn about Tkinter graphical user interfaces based on Sockets.

Expand Your Knowledge:

Explore the Tkinter documentation here