

Python | Simple GUI calculator using Tkinter

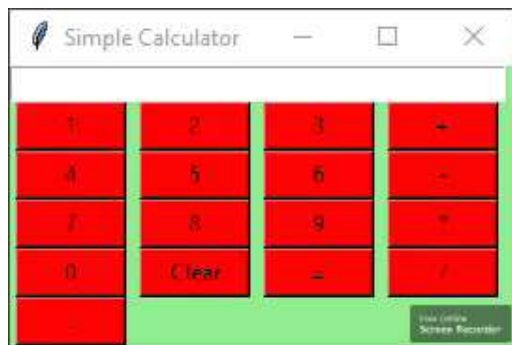
Prerequisite: [Tkinter Introduction](#), [lambda function](#)

Python offers multiple options for developing a 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 outputs the fastest and easiest way to create GUI applications. Creating a GUI using Tkinter is an easy task.

To create a Tkinter:

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.

Below is what the GUI looks like:



Let's create a GUI-based simple calculator using the Python Tkinter module, which can perform basic arithmetic operations addition, subtraction, multiplication, and division.

1. The code starts by importing the necessary modules.
2. The tkinter module provides all the basic functionality for creating graphical user interfaces.
3. Next, we create a global variable called expression which will store the result of the calculation.

4. We also create two functions to update and evaluate the expression.
5. Finally, we write driver code to initialize and manage our GUI window.
6. In order to create a simple calculator, we first need to define an expression variable.
7. This is done by using the global keyword and assigning it an empty string value ("").
8. Next, we create two functions to update and evaluate the expression.
9. The press function updates the contents of the text entry box while equalpress evaluates the final result of the calculation.
10. We next need to create a table-like structure in which our widgets will be placed.
11. We do this by using grid method which takes three arguments: columnspan , padx , and rowspan .
12. These parameters specify how many columns wide, how many rows high, and how many columns per row respectively should be used in our table layout.
13. We set columnspan to 4 , meaning that there will be four columns in our table, iPad width divided by 2 (70), multiplied by 1 for each row in our table (iPad height divided
14. The code creates a simple calculator using the Tkinter module.
15. First, the code imports everything from the Tkinter module.
16. Next, the code creates two global variables: expression and total.
17. The press() function is used to update the expression variable in the text entry box.
18. The equalpress() function is used to evaluate the final expression.
19. Finally, the clear() function is used to clear the contents of the text entry box.
20. Next, the driver code is created.
21. In this code, if __name__ == "__main__": is executed which will create a GUI window and set its background color to light green and its title to Simple Calculator.
22. Next, the geometry() method is used to set the size of the GUI window (270

23. The code starts with a few basic objects: a Button object, which has properties for text, font, background color, and command; and a grid object.
24. The first three buttons (button1 through button3) each have their own individual commands associated with them.
25. When the user clicks on one of these buttons, the corresponding command is executed.
26. For example, when the user clicks on button1, its command is to press the number 1 key.
27. Similarly, when the user clicks on button2's command, it will be to press the number 2 key; and so on.
28. Similarly, when the user clicks on button4's command (to increase the value by 1), its grid row and column values will be set to 3 and 0 respectively.
29. And finally when clicking on button5's command (to decrease the value by 1), its grid row and column values will be set to 2 and 1 respectively.
30. The code creates seven buttons, each with its own function.
31. When the user presses one of the buttons, the corresponding command is executed.
32. The first button, button1, has the function press(1).
33. When clicked, this button will execute the code lambda: press(1).
34. The second button, button2, has the function press(2), and so on.
35. When all seven buttons have been clicked, their functions will be executed in order.