Python | Simple GUI calculator using Tkinter

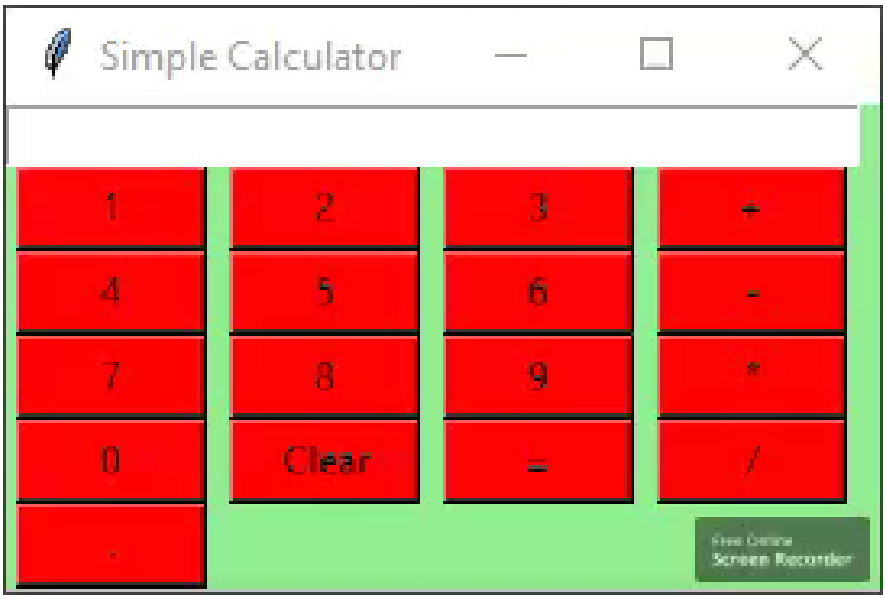
Prerequisite: Tkinter Introduction, lambda function

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.

Below is the implementation:

Python3

```
# Python program to create a simple GUI
# calculator using Tkinter

# import everything from tkinter module
from tkinter import *

# globally declare the expression variable
expression = ""


# Function to update expression
# in the text entry box
def press(num):
    # point out the global expression variable
    global expression

    # concatenation of string
    expression = expression + str(num)

    # update the expression by using set method
    equation.set(expression)


# Function to evaluate the final expression
```

```
def equalpress():
    # Try and except statement is used
    # for handling the errors like zero
    # division error etc.

    # Put that code inside the try block
    # which may generate the error
    try:

        global expression

        # eval function evaluate the expression
        # and str function convert the result
        # into string
        total = str(eval(expression))

        equation.set(total)

        # initialize the expression variable
        # by empty string
        expression = ""

    # if error is generate then handle
    # by the except block
    except:

        equation.set(" error ")
        expression = ""


# Function to clear the contents
# of text entry box
def clear():
    global expression
    expression = ""
    equation.set("")


# Driver code
if __name__ == "__main__":
    # create a GUI window
    gui = Tk()

    # set the background colour of GUI window
    gui.configure(background="light green")

    # set the title of GUI window
    gui.title("Simple Calculator")

    # set the configuration of GUI window
    gui.geometry("270x150")

    # StringVar() is the variable class
```

```
# we create an instance of this class
equation = StringVar()

# create the text entry box for
# showing the expression .
expression_field = Entry(gui, textvariable=equation)

# grid method is used for placing
# the widgets at respective positions
# in table like structure .
expression_field.grid(columnspan=4, ipadx=70)

# create a Buttons and place at a particular
# location inside the root window .
# when user press the button, the command or
# function affiliated to that button is executed .
button1 = Button(gui, text=' 1 ', fg='black', bg='red',
                 command=lambda: press(1), height=1, width=7)
button1.grid(row=2, column=0)

button2 = Button(gui, text=' 2 ', fg='black', bg='red',
                 command=lambda: press(2), height=1, width=7)
button2.grid(row=2, column=1)

button3 = Button(gui, text=' 3 ', fg='black', bg='red',
                 command=lambda: press(3), height=1, width=7)
button3.grid(row=2, column=2)

button4 = Button(gui, text=' 4 ', fg='black', bg='red',
                 command=lambda: press(4), height=1, width=7)
button4.grid(row=3, column=0)

button5 = Button(gui, text=' 5 ', fg='black', bg='red',
                 command=lambda: press(5), height=1, width=7)
button5.grid(row=3, column=1)

button6 = Button(gui, text=' 6 ', fg='black', bg='red',
                 command=lambda: press(6), height=1, width=7)
button6.grid(row=3, column=2)

button7 = Button(gui, text=' 7 ', fg='black', bg='red',
                 command=lambda: press(7), height=1, width=7)
button7.grid(row=4, column=0)

button8 = Button(gui, text=' 8 ', fg='black', bg='red',
                 command=lambda: press(8), height=1, width=7)
button8.grid(row=4, column=1)

button9 = Button(gui, text=' 9 ', fg='black', bg='red',
                 command=lambda: press(9), height=1, width=7)
button9.grid(row=4, column=2)

button0 = Button(gui, text=' 0 ', fg='black', bg='red',
```

```
                  command=lambda: press(0), height=1, width=7)
button0.grid(row=5, column=0)

plus = Button(gui, text=' + ', fg='black', bg='red',
              command=lambda: press("+"), height=1, width=7)
plus.grid(row=2, column=3)

minus = Button(gui, text=' - ', fg='black', bg='red',
              command=lambda: press("-"), height=1, width=7)
minus.grid(row=3, column=3)

multiply = Button(gui, text=' * ', fg='black', bg='red',
                  command=lambda: press("*"), height=1, width=7)
multiply.grid(row=4, column=3)

divide = Button(gui, text=' / ', fg='black', bg='red',
                  command=lambda: press("/"), height=1, width=7)
divide.grid(row=5, column=3)

equal = Button(gui, text=' = ', fg='black', bg='red',
              command=equalpress, height=1, width=7)
equal.grid(row=5, column=2)

clear = Button(gui, text='Clear', fg='black', bg='red',
              command=clear, height=1, width=7)
clear.grid(row=5, column='1')

Decimal= Button(gui, text='.', fg='black', bg='red',
                  command=lambda: press('.'), height=1, width=7)
Decimal.grid(row=6, column=0)
# start the GUI
gui.mainloop()
```