

A

Project Report

Submitted in partial Fulfillment of the of python
project on

“SCIENTIFIC CALCULATOR”

BY

Navin Chandra (12001138)

Under the guidance of

Neha Bagga(dep of computer science and
engineering)

LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA, PUNJAB(INDIA)



L OVELY
P ROFESSIONAL
U NIVERSITY

Transforming Education Transforming India

ACKNOWLEDGEMENT

Foremost, we would like to express our sincere gratitude to our advisor Neha Bagga mam for the continuous support in our studies and project. His guidance helped us all the time to complete this project. We could not have imagined having better minor project mentors for our project. We would also like to thank our parents, brothers and sisters for having given us their undisputable support throughout, as always, for which our expression of thanks likewise does not suffice. Also, we thank our friends and classmates for supporting us.

Student Declaration

Name - Navin Chandra.

Registration no - 12001138

Roll no – - RK19TGB71

Subject - - INT213 (python programming)

SUMMARY

In this project we are going to build a scientific calculator with proper GUI using python by importing tkinter , math library, and various functions and some inbuilt functions by which we can perform various arithmetic operations and various scientific calculations .

Link of github for code=

<https://github.com/navinchandra514/scientific-calculator/blob/main/calcc.py>

Introduction

A calculator is a device that performs arithmetic operations on numbers. The simplest calculators can do only addition, subtraction, multiplication, and division, but scientific calculators can handle operations like, trigonometric functions, square root, modulo operations, factorial etc.

Here in this project we have created a scientific calculator using tkinter in python using various lines of code and now I am going to explain all those code with help of this report.

Explanation of code

First we have import the tkinter which is the standard GUI library for Python.

Then we have import math library to give access to the mathematical functions.

Then we have taken root as Tk() for an ordinary window and we have given title to our GUI and set some dimensions and some min and max size for better Interface, then we have created a box(entry widget) to take some input and show output from calculator.

Then we have created some button like {0,1,2,3,4,5,6,7,8,9,+,-,*,/,%,),(,.,00}

And to printed these buttons on screen , I give a parameter name command and to command I give a function “**click(“ “)**” and in command we have used an anonymous function because in command we can’t pass any functions, so to use those functions we have used an anonymous function called “lambda” which can take any number of arguments

Now in click() function

```
def click(to_print):
```

```
    old=box.get()
```

```
    box.delete(0, END)
```

```
    box.insert(0, old+to_print)
```

```
    return
```

here to_print is storing the inputs we are giving whether it is numbers, dots or any of the arithmetic operators , then using the function box.get() we can get the inputs of entry bar, and then we have used box.delete(0,END) to delete every thing from 0 to end and then box.insert(0 ,old+to_print) which start from 0th position in entry widget and combine the old variable and the new variable.

Now clear() function

```
def clear():
```

```
    box.delete(0, END)
```

```
    return
```

here clear function delete all the variable from the entry widget.

Now evaluate() function

```
def evaluate():
```

```
    ans=box.get()
```

```
    ans=eval(ans)
```

```
    box.delete(0, END)
```

```
    box.insert(0, ans)
```

here we will use eval function , now in ans it will read every thing from the entry widget , and when we press equal button then it will delete everything from the entry widget and insert the ans in the entry widet.

Now backspace function

```
def bksps():
```

```
    current=box.get()
```

```
    length=len(current)-1
```

```
    box.delete(length, END)
```

here using box.get() it will read the current variables in box and using length function it will calculate the length and subtract it by -1 and it will delete the element from the end.

Now scop() function means scientific operation

```
def scop(event):
```

```
    key=event.widget
```

```
    text=key['text']
```

```
    no=box.get()
```

```
    result=""
```

```
    if text=='log':
```

```
        result=str(m.log(float(no)))
```

```
    if text=='sin':
```

```
        result=str(m.sin(float(no)))
```

```
    if text=='cos':
```

```
        result=str(m.cos(float(no)))
```

```
    if text=='tan':
```

```
        result=str(m.tan(float(no)))
```

```
    if text=='1/x':
```

```
        result=str(1/(float(no)))
```

```
    if text=='deg':
```

```
        result=str(m.degrees(float(no)))
```

```
    if text=='Sqrt':
```

```
        result=str(m.sqrt(float(no)))
```

```
    if text=='x!':
```

```
        result=str(m.factorial(float(no)))
```

```
    box.delete(0, END)
```

```
    box.insert(0, result)
```


we have some button like {sin,cos,tan,log,deg,sqrt,x!,1/x}

there on left click we have pass scop function using bind method.

Now in scop function we have pass event which handle that what action needed to be perform

Now in key we have taken event.widget (deg,log,sqrt will come under this) and then in text we have have it as in string format

Now in no we have pass box.get() so we can get whatever value we have passed.

Now in next step if text = deg than it will take float of that number because it was in the string format , now after getting that no as float , we will apply the math function which we have imported in the starting and when math function is applied then it will again it will convert it into string format for further operations and then it will stored as result and for "1/x" we will use 1/float no.

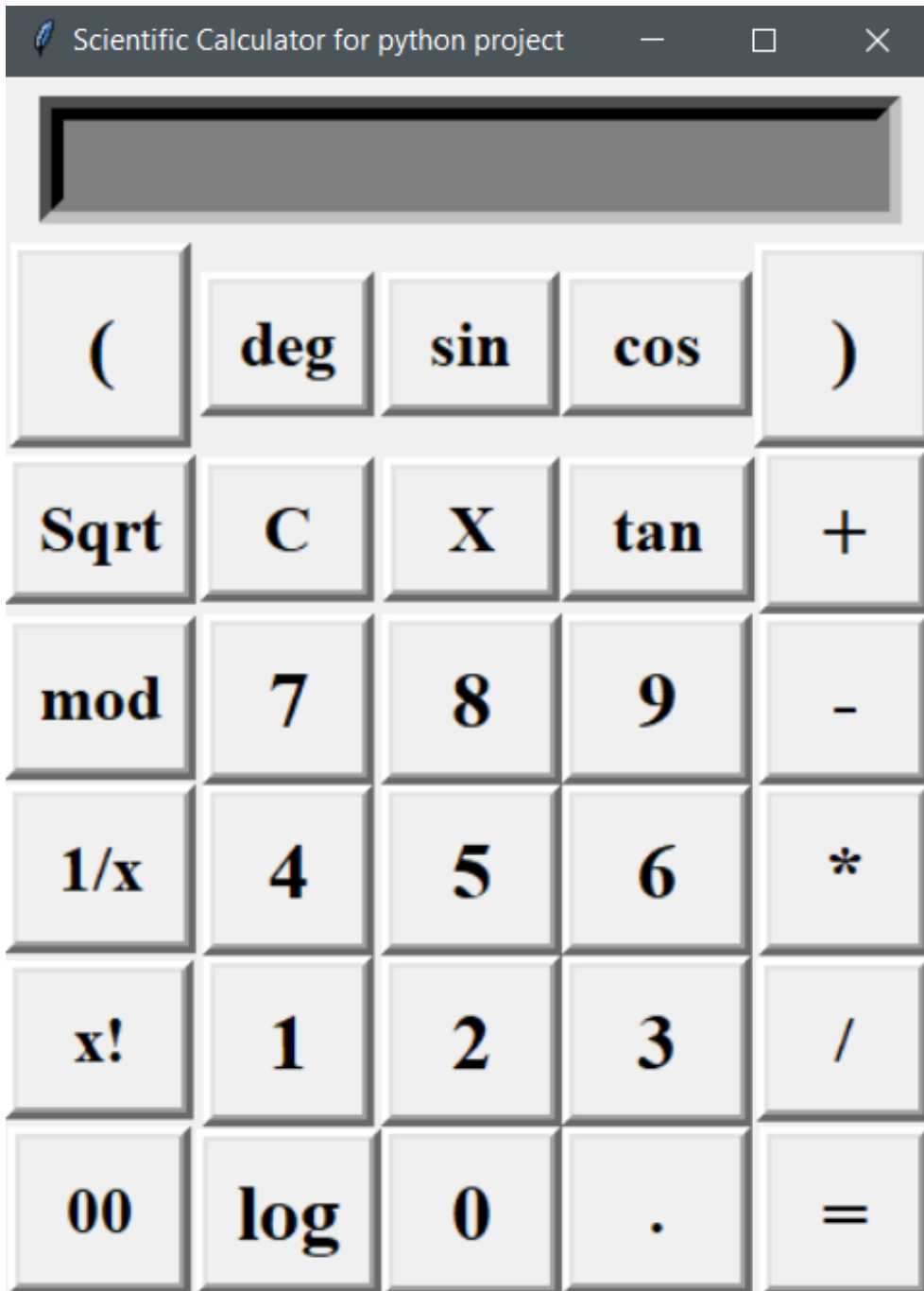
And at last in scop function the whole variables will be deleted and result will be inserted in the box.

Now whatever button we have made we have to pack or grid all that to be shown In the GUI as "**button.grid**" and we have pass the location for each button with the help of row and column as "**button.grid(row=1,column=1)**"

Link of github for code=

<https://github.com/navinchandra514/scientific-calculator/blob/main/calcc.py>

Output of the code .



Link of github for code=

<https://github.com/navinchandra514/scientific-calculator/blob/main/calcc.py>

Thank You