PROJECT REPORT ON SCIENTIFIC CALCULATOR

Course Name : Python Programming

Course Code : INT213

Project Number : 17th

Project Statement : Design a Scientific Calculator with Proper GUI using Python

Submitted to:
Dr. Prateek Agrawal

Submitted by:
Team 17

1. Gurkirat kaur Suri
   RK21CRA17 (12107018)
2. Jain kirtan Sunil
   RK21CRB58 (12115299)
3. Sudhanshu Raj Tiwari
   RK21CRB34 (12103695)

ACKNOWLEDGEMENT

We would like to thank Dr. Prateek Agrawal, our Professor, for their great support and guidance in completing our project on "Scientific Calculator". It was a great learning experience.

We would also like to take this opportunity to express our gratitude towards Lovely Professional University that encourages us and provides us opportunities to display our hardwork and talent through amazing projects like this.

CONTENTS

INTRODUCTION

SCIENTIFIC CALCULATOR

The Calculator was written by Rolf Howarth in early 1996.
A fully featured scientific calculator with proper operator precedence is implemented, including trigonometry functions and logarithms, factorials, 12 levels of parenthesis, logs to base 2 (a handy function for information entropists!), bitwise logical operators, hex, octal, binary and ASCII display.
This Calculator application project is written in Python programming language.
It is a calculator (Scientific Calculator) application. This application is used for calculating maths functions easily.
In this application two types of calculators are there:

1. Standard Calculator
2. Scientific Calculator

The first one is quite simple to solve arithmetic operations and convert the result into either integer or float pointing number. And then the second one is scientific notation type maths functions like sin, cos, tan, log, etc. It is extremely useful to solve the odd math calculations in less time and in a simple manner and easy to use. Especially we used a menu bar with two items one is standard and second one is scientific.

After clicking the standard item it will show the standard calculator and after clicking the scientific item it will show the scientific calculator with standard also.

By using Tkinter in python, We have developed this application it is also converted into .exe file by using pyinstaller then it is now a desktop application.

Lastly it shows desktop icon in out system if you install it otherwise not show in your desktop.

OBJECTIVE

The Project Statement given is to - Design a Scientific Calculator with proper GUI using pyton.
The Scientific Calculator application should be able to perform the following functions:

1. Standard Calculator:
   - → Sum
   - → Subtraction
   - → Division
   - → Multiplication
   - → Modulo
2. Scientific Calculator:
   - → Trigonometric and logarithmic functions.
   - → Exponential and Inverse functions.

The GUI for this project will be implemented using Tkinter programming.
Tkinter is the standard GUI library for python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

DESIGN

The basic structural design of the Scientific Calculator includes the implementation of the basic functions of the calculator which are briefly described below.

Addition

The addition (sum function) is used by clicking on the "+" button or using the keyboard. The function results in a+b.

Subtraction

The subtraction (minus function) is used by clicking on the "-" button or using the keyboard. The function results in a-b.

Multiplication

The multiplication (times function) is used by clicking on the "x" button or using the keyboard. "*" key. The function results in a*b.

Division

The division (divide function) is used by clicking on the "/" button or using the keyboard "/" key. The function results in a/b.

Sign

The sign key (negative key) is used by clicking on the "(-)" button. The function results in -1*x.

Square
The square function is used by clicking on the "x^2" button or type "^2". The function results in x*x.

Square root
The square root function is used by clicking on the "√x" button or type "sqrt()". This represents x^.5 where the result squared is equal to x.

Raise to the power
The raise to the power (y raised to the x function) is used by clicking on the "y^x" button or type "^".

Natural Exponential
The natural exponential (e raised to the x) is used by clicking on the "e^x" button or type "exp()". The result is e (2.71828...) raised to x.

Logarithm
The logarithm (LOG) is used by clicking on the "LOG" button or type "Log()".

Natural logarithm
The natural logarithm (LN) is used by clicking on the (LN) button or type "Ln()".

Inverse
Multiplicative inverse (reciprocal function) is used by pressing the "1/x" button or typing "inv()". This function is the same as x^-1 or dividing 1 by the number.

Exponent
Numbers with exponents of 10 are displayed with

an "e", for example 4.5e+100 or 4.5e-100. This function represents 10^x. Numbers are automatically displayed in the format when the number is too large or too small for the display. To enter a number in this format use the exponent key "EEX". To do this enter the mantissa (the non exponent part) then press "EEX" or type "e" and then enter the exponent.

Factorial

The factorial function is used by clicking the "!" button or type "!".

PI

PI is a mathematical constant of the ratio of a circle's circumference to its diameter.

Python Tkinter GUI

Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps:

→ Import the Tkinter module.
→ Create the GUI application main window.
→ Add one or more of the above-mentioned widgets to the GUI application.
→ Enter the main event loop to take action against each event triggered by the user.

Some explanations for each button and the function which represents are the following:

1st Row

1. abs: The absolute value of a number (e.g. abs(-5) = 5).
2. mod: From modulo. It's the operation to find the remainder of the division of one number by another.
3. div: Floor division returns the result of the division rounded down to the nearest by another integer. In python we use the symbol // (e.g. 8 div 3 = 8//3 = 2).
4. x!: The factorial of the number x (e.g. 4! = 24).
5. e: The Euler's number. A mathematical constant approximately equal to 2.71828.

2nd Row

1. sin: sine of an angle θ in degrees (e.g. sin(90) = 1).
2. cos: cosine of an angle θ in degrees (e.g. cos(180) = -1).
3. tan: tangent of an angle θ in degrees (e.g. tan(45) = 1)

4. cot : Cotangent of an angle θ in degrees (e.g. cot(45) = 1/tan(45) = 1)

5. π : Archimedes' constant defined as the ratio of a circle's circumference to its diameter. It is approximately equal to 3.14159.

3rd Row

1. x^2 : x raised to the power of 2 (e.g. $4^2 = 16$).

2. x^3 : x raised to the power of 3 (e.g. $5^3 = 125$).

3. x^n : x raised to the power (e.g. $2^4 = 16$).

4. x^{-1} : x raised to the power of (-1). The inverse of number x (e.g. $2^{-1} = 0.5$)

5. 10^x : Powers of 10 (e.g. $10^3 = 1000$).

4th Row

1. $\sqrt[2]{}$: Square root of a number (e.g. $\sqrt[2]{144} = 12$).

2. $\sqrt[3]{}$: Cube root of a number (e.g. $\sqrt[3]{8} = 2$)

3. $\sqrt{}$: Any root of any number (e.g. $\sqrt[4]{16} = 2$).

4. $\log_{10}$: The logarithm of a number with base 10 (e.g. $\log_{10} 1000 = 3$)

5. ln : The logarithm of a number with base e (e.g. $\log_e e = \ln e = 1$)

5th Row

(	)	±	%	e^x

1. (: Left Parenthesis.

2.): Right Parenthesis.

3. ±: Change the sign of a number.

4. %: Find the percentage of a number (e.g. 5% = 0.05).

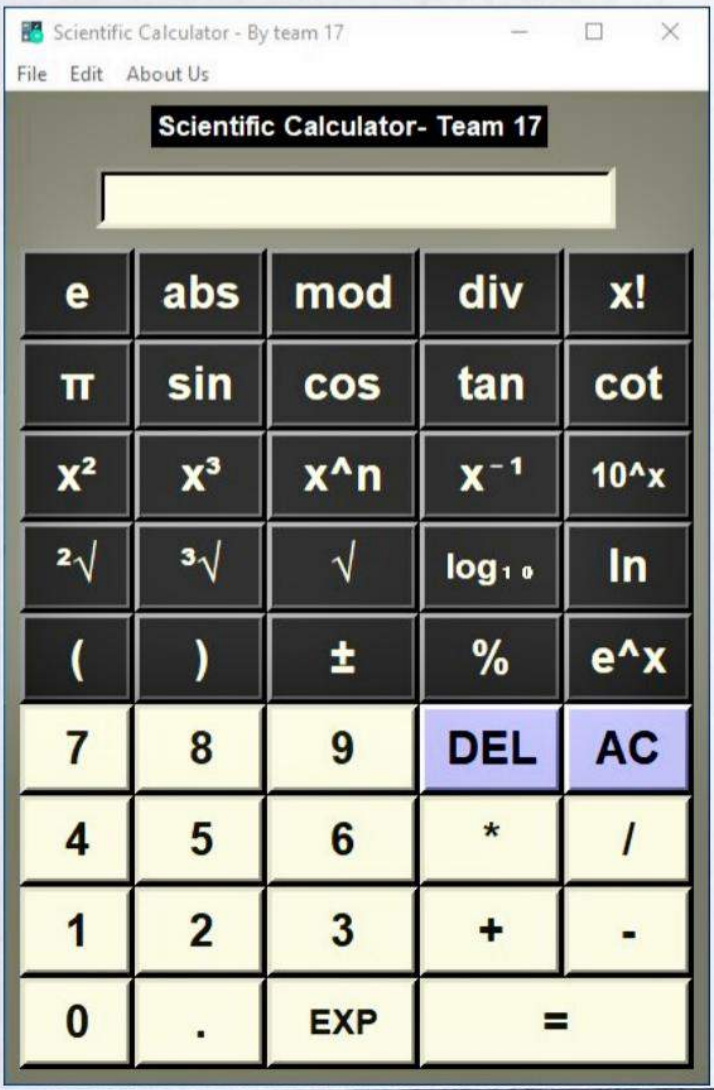
5. e^x: Exponential function (e.g. e^2 = approx. 7.389).

6th, 7th, 8th Row

7	8	9	DEL	AC
4	5	6	*	/
1	2	3	+	-
0	.	EXP	=	

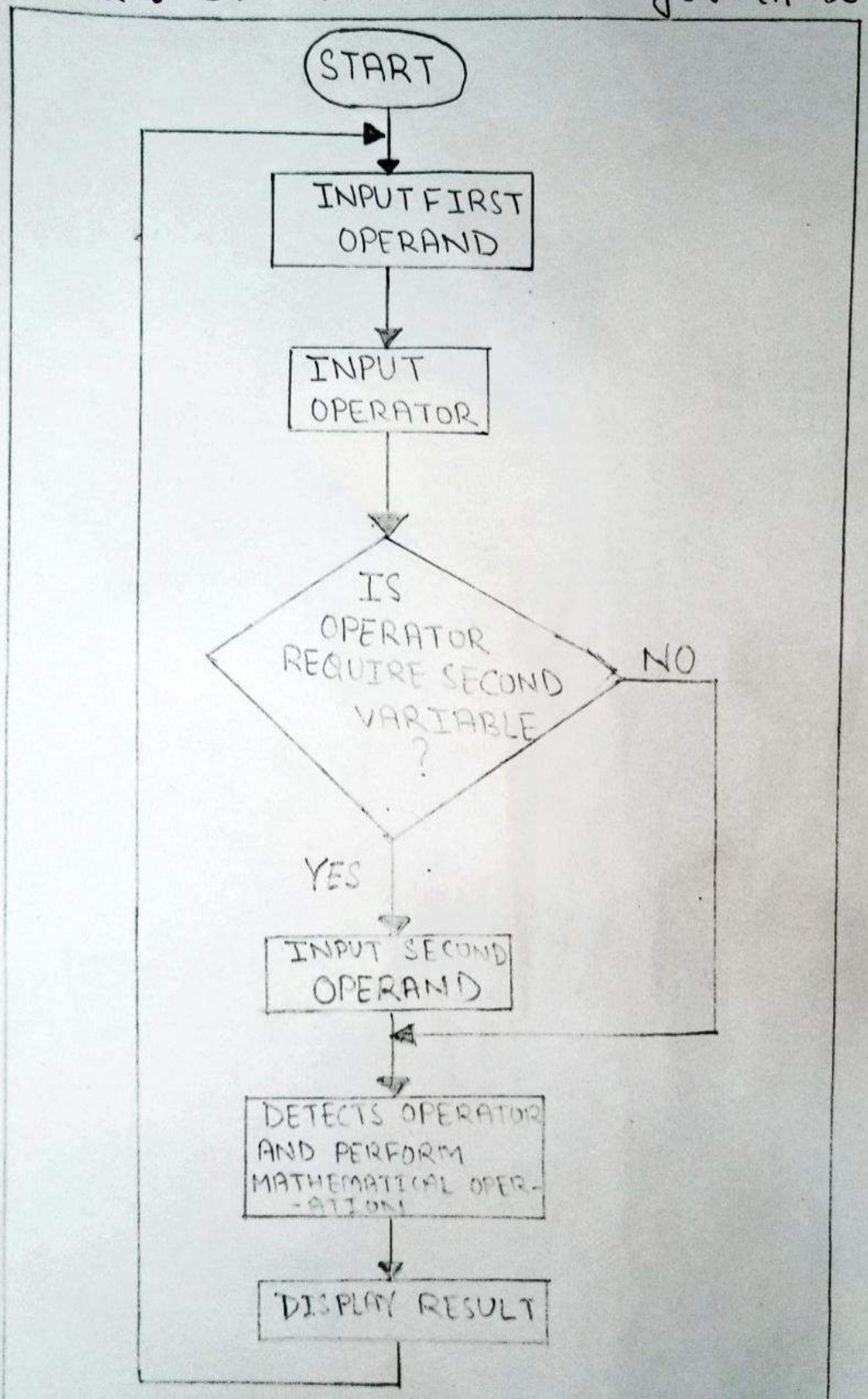
In these rows are :

→ The basic number buttons (0 to 9).
→ The basic math symbols (operators) (+, -, *, /).
→ The equal sign (=) and point (.).
→ Button DEL to delete one or more from the end of the entry.
→ Button AC to delete the whole entry.
→ EXP : Multiply any number with powers of 10 (e.g. 2 * 10^3 = 2000).



Complete GUI interface of our calculator

* Flow Chart Of Mathematical algorithms



* Pseudo algorithm for mathematical calculation performed by our GUI.

⇒ calc.operators(int a, int b and char operator)

```
if (operator = '+')
    display a + b
if (operator = '-')
    display a - b
if (operator = '*')
    display a * b.
if (operator = 'abs')
    display absolute value of arguments.
if (operator = 'sin')
    display sin(a)
if (operator = 'cos')
    display cos(a)
if (operator = 'mod')
    display modulus a % b
if (operator = 'tan')
    display tan(a)
if (operator = 'cot')
    display cot(a)
```

if (operator = x^2)
display x**2
if (operator = x^3)
display x**3.
if (operator = x^{-1})
display x**-1)
if (operator $\sqrt[2]{}$)
display $\sqrt[2]{x}$)
if (operator $\sqrt[3]{}$)
display $\sqrt[3]{x}$

PROJECT CODE AND FUNCTIONS

FUNCTIONS USED IN CODE IMPLEMENTATION

def button_click (char)

This function is used to add in the entry of text on the display. It makes use of global keyword so that the input can interact with all other functions of the calculator.

def button_clear_all ()

This function is used to clear the whole entry of text display. Global variables are implemented.

def button_delete ()

This function is used to delete one by one from the last in the entry of text on the display.

def factorial (n), def fact_func ()

These functions are used to calculate the factorial of a number.

def trig_sin(), def trig_cos(), def trig_tan(), def trig_cot()

These functions are used to calculate trigonometric number of an angle.

def square_root()

This function is used to find the square root of a number.

def third_root()

This function is used to find the cube root of a number.

def sign_change()

This function is used to change the sign of a number.

def percent()

This function is used to calculate the percentage of a number.

def button_equal()

This function or button is used to find the result of an operation.

DIFFERENT ATTRIBUTES USED IN CODE IMPLEMENTATION

Root is the Tkinter object for main GUI window.

root.configure()
It is used on any widget to change settings or to apply new settings.

root.iconphoto()
It is used to set the titlebar icon of the window.

root.geometry()
It is used to set the dimensions of the window.

root.eval()
It is used to place the Tkinter screen in center.

grid()
It defines the rows, columns and other properties of a button.

destroy()
used for closing the application.

add_command()
adds a menu item to the menu.

add_separator()
adds a separator line to the menu.

add_cascade ()
creates a new hierarchical menu by associating a given menu to a parent menu.

DIFFERENT HEADERS

-from tkinter import *
represents all the functions and built-in modules in the -tkinter library. By importing all the functions and methods, we can use the in-built functions or methods in a particular application without importing them implicitly.

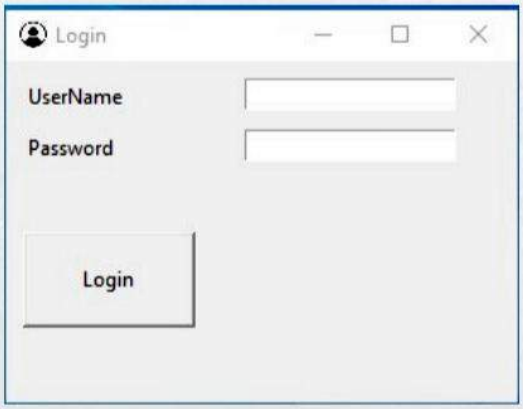
import tkinter.messagebox
The tkMessageBox module is used to display message boxes in your applications.

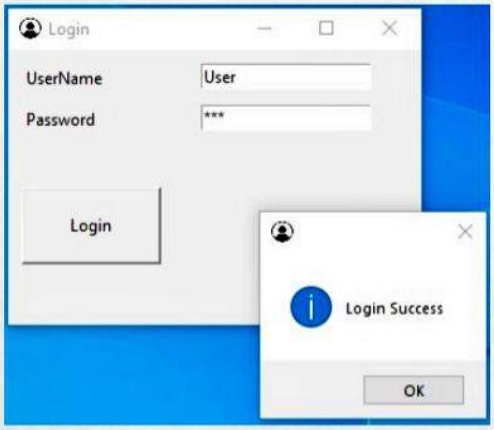
import math
It extends the list of mathematical functions

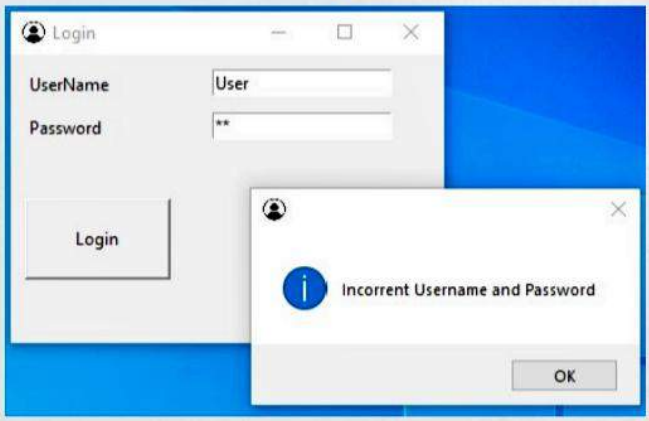
from mybanner import bannerTop
imports local python files to import banner of team 17 on stdout.

import login
to import a login window before entering the calculator.

SOME SNAPS OF CODE AND GUI IMPLEMENTATION

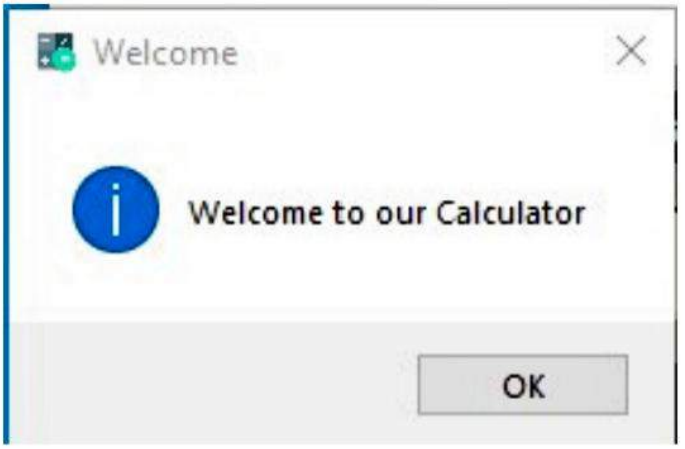




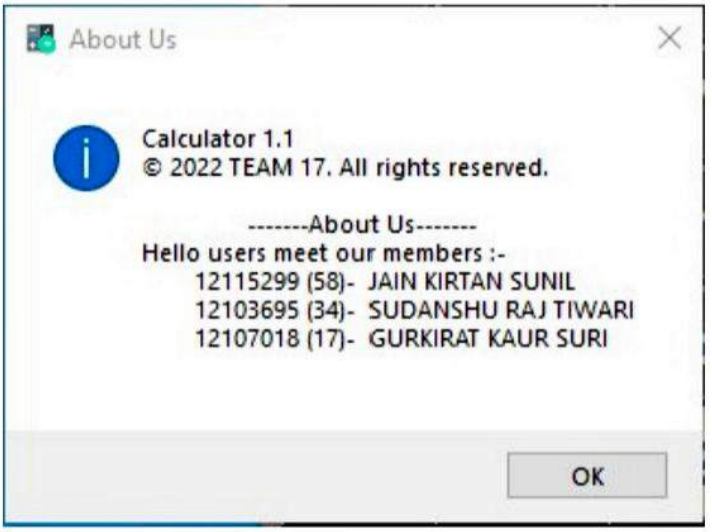


stdout while code implementation

welcome message box



about us message box



do you want to exit message box

CONCLUSION

This Project has really been faithful and informative. It has made us learn and understand the many trivial concepts of Python language. As we have used python Tkinter as a GUI, it provides various controls, such as buttons, labels and text boxes to build a user friendly Scientific Calculator.

The fast growing use of internet confirms the good future and scope of the proposed project. Finally it has taught us a valuable lifelong lessons about the improvements and working and interacting in a group.

REFERENCES

→ geeksforgeeks
→ Stackoverflow
→ Wikipedia
→ GitHub projects
→ python.org
→ python official documentation
→ Introduction to Programming Using Python by Y. Daniel Liang.