74+戴一帆+211205102388

作业9

戴一帆

2022年5月15号

- 要求
- 基础题代码（低配版科学计算器）

要求

基础题要求：
编写一个GUI的工具，GUI功能如下：

1. GUI的外观设计可以自由设计，尽量美观；
2. 可以进行两个数值的加法和减法计算。
   附加题要求：GUI工具可以满足常规的科学计算器的功能。

基础题代码（低配版科学计算器）

```
import wx
import random

class Calculator(wx.Frame):
    calculation = ""  # 计算式初始化
    def __init__(self, parent, id):
        wx.Frame.__init__(self, parent, id, 'Calculator', size=(320, 470))
        panel = wx.Panel(self)
        self.inputField = wx.ComboBox(panel, value="", pos=(10, 0), size=(275, 150))
        self.inputField.Bind(wx.EVT_TEXT, self.OnKeyTyped)

        btnleftparenthesisbrackets = wx.Button(panel, label="(", pos=(0, 35), size=
(75, 50))
        self.Bind(wx.EVT_BUTTON,
self.leftparenthesisbrackets,btnleftparenthesisbrackets)

        btnrightparenthesisbrackets = wx.Button(panel, label=")", pos=(75, 35), size=
(75, 50))
        self.Bind(wx.EVT_BUTTON,
self.rightparenthesisbrackets,btnrightparenthesisbrackets)

        btnbackspace = wx.Button(panel, label="←", pos=(150, 35), size=(75, 50))
```

```
        self.Bind(wx.EVT_BUTTON, self.backspace, btnbackspace)

        btn1 = wx.Button(panel, label="1", pos=(0, 265), size=(75, 75))    # 设置按钮
        self.Bind(wx.EVT_BUTTON, self.one, btn1)    # 按钮事件的绑定

        btn2 = wx.Button(panel, label="2", pos=(75, 265), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.two, btn2)

        btn3 = wx.Button(panel, label="3", pos=(150, 265), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.three, btn3)

        btn4 = wx.Button(panel, label="4", pos=(0, 180), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.four, btn4)

        btn5 = wx.Button(panel, label="5", pos=(75, 180), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.five, btn5)

        btn6 = wx.Button(panel, label="6", pos=(150, 180), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.six, btn6)

        btn7 = wx.Button(panel, label="7", pos=(0, 95), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.seven, btn7)

        btn8 = wx.Button(panel, label="8", pos=(75, 95), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.eight, btn8)

        btn9 = wx.Button(panel, label="9", pos=(150, 95), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.nine, btn9)

        btn0 = wx.Button(panel, label="0", pos=(0, 350), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.zero, btn0)

        btnclr = wx.Button(panel, label="CLR", pos=(225, 35), size=(75, 50))
        self.Bind(wx.EVT_BUTTON, self.clear, btnclr)
        btnclr.SetBackgroundColour('Red')

        btnplus = wx.Button(panel, label="+", pos=(225, 350), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.plus, btnplus)

        btnminus = wx.Button(panel, label="-", pos=(225, 265), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.minus, btnminus)

        btnmultiply = wx.Button(panel, label="x", pos=(225, 180), size=(75, 75))
```

```
        self.Bind(wx.EVT_BUTTON, self.multiply, btnmultiply)

        btndivide = wx.Button(panel, label="/", pos=(225, 95), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.divide, btndivide)

        btnpoint = wx.Button(panel, label=".", pos=(75, 350), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.point, btnpoint)

        btnequal = wx.Button(panel, label="=", pos=(150, 350), size=(75, 75))
        self.Bind(wx.EVT_BUTTON, self.equal, btnequal)
        btnequal.SetBackgroundColour(random.choice(['Green', 'Yellow' ]))


    def OnKeyTyped(self, event):  # 支持直接键盘输入数据
       self.calculation = event.GetString()
    #对事件进行定义

    def leftparenthesisbrackets(self, event):
        self.calculation = self.calculation + "("
        self.inputField.SetValue(self.calculation)

    def rightparenthesisbrackets(self, event):
        self.calculation = self.calculation + ")"
        self.inputField.SetValue(self.calculation)

    def backspace(self, event):
        self.calculation = self.calculation[0:-1:1]
        self.inputField.SetValue(self.calculation)

    def one(self, event):
        self.calculation = self.calculation + "1"
        self.inputField.SetValue(self.calculation)
    def one(self, event):
        self.calculation = self.calculation + "1"
        self.inputField.SetValue(self.calculation)

    def two(self, event):
        self.calculation = self.calculation + "2"
        self.inputField.SetValue(self.calculation)

    def three(self, event):
        self.calculation = self.calculation + "3"
        self.inputField.SetValue(self.calculation)
```

```
    def four(self, event):
        self.calculation = self.calculation + "4"
        self.inputField.SetValue(self.calculation)

    def five(self, event):
        self.calculation = self.calculation + "5"
        self.inputField.SetValue(self.calculation)

    def six(self, event):
        self.calculation = self.calculation + "6"
        self.inputField.SetValue(self.calculation)

    def seven(self, event):
        self.calculation = self.calculation + "7"
        self.inputField.SetValue(self.calculation)

    def eight(self, event):
        self.calculation = self.calculation + "8"
        self.inputField.SetValue(self.calculation)

    def nine(self, event):
        self.calculation = self.calculation + "9"
        self.inputField.SetValue(self.calculation)

    def zero(self, event):
        self.calculation = self.calculation + "0"
        self.inputField.SetValue(self.calculation)

    def clear(self, event):
        self.calculation = ""
        self.inputField.SetValue(self.calculation)

    def plus(self, event):
        self.calculation = self.calculation + "+"
        self.inputField.SetValue(self.calculation)

    def minus(self, event):
        self.calculation = self.calculation + "-"
        self.inputField.SetValue(self.calculation)

    def multiply(self, event):
        self.calculation = self.calculation + "*"
```

```
        self.inputField.SetValue(self.calculation)

    def divide(self, event):
        self.calculation = self.calculation + "/"
        self.inputField.SetValue(self.calculation)

    def point(self, event):
        self.calculation = self.calculation + "."
        self.inputField.SetValue(self.calculation)



    def equal(self, event):
        try:
            result = eval(self.calculation)  # 计算式求值
            self.inputField.Insert(self.calculation, 0)  # 将历史记录加入下拉框
            self.inputField.SetValue(str(result))  # 输出框值为结果
        except Exception as e:
            self.inputField.SetValue('Input Illegal')
            print(e)
            return
        finally:
            self.calcuation = ""

#运行
if __name__ == '__main__':
    app = wx.App()
    frame = Calculator(parent=None, id=-1)
    frame.Show()
    app.MainLoop()
```

ps: pi = math.pi
e = math.e
math.log
math.sin
等用于科学计算器的编写