Calculator Project Code

```
from tkinter import*
import math

root = Tk()
root.title("Pycharm Calculator")
root.resizable(width= False, height= False)
root.geometry("470x600+600+200")

calc = Frame(root,bd=2, pady=5, bg='gainsboro', relief = RIDGE)
calc.grid()

class Calc():
        def __init__(self):
            self.total = 0
            self.current=""
            self.input_value = True
            self.check_sum= False
            self.op = ""
            self.result = False
        def numberEnter(self, num):
            self.result = False
            firstnum = txtDisplay.get()
            secondnum = str(num)
            if self.input_value:
                self.current = secondnum
                self.input_value = False
            else:
                if secondnum == '.':
                    if secondnum in firstnum:
                        return
                self.current = firstnum + secondnum
            self.display(self.current)

        def sum_of_total(self):
            self.result = True
            self.current = float(self.current)
            if self.check_sum == True:
                self.valid_function()
            else:
                self.total = float(txtDisplay.get())

        def display(self, value):
            txtDisplay.delete(0, END)
            txtDisplay.insert(0, value)

        def valid_function(self):
            if self.op == "add":
                self.total += self.current
            if self.op == "sub":
                self.total -= self.current
            if self.op == "multi":
                self.total *= self.current
            if self.op == "divide":
```

```
            self.total /= self.current

        self.input_value = True
        self.check_sum = False
        self.display(self.total)

    def operation(self, op):
        self.current = float(self.current)
        if self.check_sum:
            self.valid_function()
        elif not self.result:
            self.total = self.current
            self.input_value = True
        self.check_sum = True
        self.op = op
        self.result = False

    def Clear_Entry(self):
        self.result = False
        self.current = "0"
        self.display(0)
        self.input_value = True

    def all_Clear_Entry(self):
        self.Clear_Entry()
        self.total=0

    def delectBS(self):
        numLen = len(txtDisplay.get())
        txtDisplay.delete(numLen - 1, 'end')
        if numLen == 1:
            txtDisplay.insert(0, "0")
    def mathPM(self):
        self.result = False
        self.current = -(float(txtDisplay.get()))
        self.display(self.current)
    def squared(self):
        self.result = False
        self.current = math.sqrt(float(txtDisplay.get()))
        self.display(self.current)
    def cos(self):
        self.result = False
        self.current = math.cos(math.radians(float(txtDisplay.get())))
        self.display(self.current)

    def tan(self):
        self.result = False
        self.current = math.tan(math.radians(float(txtDisplay.get())))
        self.display(self.current)

    def sin(self):
        self.result = False
        self.current = math.sin(math.radians(float(txtDisplay.get())))
        self.display(self.current)

    def cosh(self):
        self.result = False
        self.current = math.cosh(math.radians(float(txtDisplay.get())))
        self.display(self.current)
```

```
        def tanh(self):
            self.result = False
            self.current = math.tanh(math.radians(float(txtDisplay.get())))
            self.display(self.current)

        def acos(self):
            self.result = False
            self.current = math.acos(math.radians(float(txtDisplay.get())))
            self.display(self.current)

        def atan(self):
            self.result = False
            self.current = math.atan(math.radians(float(txtDisplay.get())))
            self.display(self.current)

        def asin(self):
            self.result = False
            self.current = math.asin(math.radians(float(txtDisplay.get())))
            self.display(self.current)

        def sinh(self):
            self.result = False
            self.current = math.sinh(math.radians(float(txtDisplay.get())))
            self.display(self.current)

        def log10(self):
            self.result = False
            self.current = math.log10(float(txtDisplay.get()))
            self.display(self.current)

        def exp(self):
            self.result = False
            self.current = math.exp(float(txtDisplay.get()))
            self.display(self.current)

        def pi(self):
            self.result = False
            self.current = math.pi
            self.display(self.current)

        def tau(self):
            self.result = False
            self.current = math.tau
            self.display(self.current)

        def factorial(self):
            self.result = False
            self.current = math.factorial(int(txtDisplay.get()))
            self.display(self.current)




added_value = Calc()

txtDisplay = Entry(calc, font=('arial', 16,'bold'),bd=20, width=28,
justify=RIGHT)
txtDisplay.grid(row=0, column=0, columnspan=4, pady=1)
```

```
numberpad="789456123"
i=0
btn = []
for j in range (4,7):
    for k in range (3):
        btn.append(Button (calc, width=6, height=2, font=('arial', 16,
'bold'),bd=4,bg='yellow', text=numberpad[i]))
        btn[i].grid(row=j,column=k,pady=1)
        btn[i]["command"]= lambda x = numberpad[i]:
added_value.numberEnter(x)
        i+= 1
#========================================================================
btnDelete = Button(calc, width=6, height=2, text = "DEL", font=('arial',
16, 'bold'),bd=4,bg="gainsboro",
                   command=added_value.delectBS).grid(row=1, column=0,
pady=1)
btnClear = Button(calc, width=6, height=2, text = "C", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                   command=added_value.all_Clear_Entry).grid(row=1,
column=1, pady=1)
btnClearAll = Button(calc, width=6, height=2, text = "CE", font=('arial',
16, 'bold'),bd=4,bg="gainsboro",
                   command=added_value.Clear_Entry).grid(row=1, column=2,
pady=1)
btnPM = Button(calc, width=6, height=2, text = chr(177), font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                   command=added_value.mathPM).grid(row=1, column=3,
pady=1)
#========================================================================
btnSq = Button(calc, width=6, height=2, text = "√", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                   command=added_value.squared).grid(row=2, column=0,
pady=1)
btnCos = Button(calc, width=6, height=2, text = "Cos", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                   command=added_value.cos).grid(row=2, column=1, pady=1)
btnSin = Button(calc, width=6, height=2, text = "Sin", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                   command=added_value.sin).grid(row=2, column=2, pady=1)
btnTan= Button(calc, width=6, height=2, text = "Tan", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                   command=added_value.tan).grid(row=2, column=3, pady=1)
#========================================================================
btnAdd = Button(calc, width=6, height=2, text = "+", font=('arial', 16,
'bold'),bd=4,bg="blue",
                   command = lambda:
added_value.operation("add")).grid(row=6, column=3, pady=1)
btnSub = Button(calc, width=6, height=2, text = "-", font=('arial', 16,
'bold'),bd=4,bg="blue",
                   command = lambda:
added_value.operation("sub")).grid(row=4, column=3, pady=1)
btnMult = Button(calc, width=6, height=2, text = "*", font=('arial', 16,
'bold'),bd=4,bg="blue",
                   command = lambda:
added_value.operation("multi")).grid(row=5, column=3, pady=1)
btnDiv= Button(calc, width=6, height=2, text = chr(247), font=('arial', 16,
'bold'),bd=4,bg="blue",
                   command = lambda:
```

```
added_value.operation("divide")).grid(row=7, column=3, pady=1)

#====================================================================
btnzero = Button(calc, width=6, height=2, text = "0", font=('arial', 16,
'bold'),bd=4,bg="yellow",
                    command=lambda: added_value.numberEnter(0)).grid(row=7,
column=0, pady=1)
btnDot = Button(calc, width=6, height=2, text = ".", font=('arial', 16,
'bold'),bd=4,bg="red",
                    command=lambda: added_value.numberEnter(".")).grid(row=7
,column=1, pady=1)
btnEquls = Button(calc, width=6, height=2, text = "=", font=('arial', 16,
'bold'),bd=4,bg="red",
                    command=added_value.sum_of_total).grid(row=7, column=2,
pady=1)

btnCosh = Button(calc, width=6, height=2, text = "Cosh", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                    command=added_value.cosh).grid(row=3, column=0, pady=1)
btnSinh = Button(calc, width=6, height=2, text = "Sinh", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                    command=added_value.sinh).grid(row=3, column=1, pady=1)
btnTanh= Button(calc, width=6, height=2, text = "Tanh", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                    command=added_value.tanh).grid(row=3, column=2, pady=1)


btnlog10= Button(calc, width=6, height=2, text = "log", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                    command=added_value.log10).grid(row=3, column=3, pady=1)
btnexp= Button(calc, width=6, height=2, text = "exp", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                    command=added_value.exp).grid(row=1, column=5, pady=1)
btnpi= Button(calc, width=6, height=2, text = "π", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                    command=added_value.pi).grid(row=2, column=5, pady=1)
btnfactorial= Button(calc, width=6, height=2, text = "!", font=('arial',
16, 'bold'),bd=4,bg="gainsboro",
                    command=added_value.factorial).grid(row=7, column=5,
pady=1)

btnarcCos = Button(calc, width=6, height=2, text = "acos", font=('arial',
16, 'bold'),bd=4,bg="gainsboro",
                    command=added_value.acos).grid(row=4, column=5, pady=1)
btnarcSin = Button(calc, width=6, height=2, text = "aSin", font=('arial',
16, 'bold'),bd=4,bg="gainsboro",
                    command=added_value.asin).grid(row=5, column=5, pady=1)
btnarcTan= Button(calc, width=6, height=2, text = "aTan", font=('arial',
16, 'bold'),bd=4,bg="gainsboro",
                    command=added_value.atan).grid(row=6, column=5, pady=1)
btnarcTan= Button(calc, width=6, height=2, text = "aTan", font=('arial',
16, 'bold'),bd=4,bg="gainsboro",
                    command=added_value.atan).grid(row=6, column=5, pady=1)
btntau= Button(calc, width=6, height=2, text = "2π", font=('arial', 16,
'bold'),bd=4,bg="gainsboro",
                    command=added_value.tau).grid(row=3, column=5, pady=1)

root.mainloop()
```