**Course Design A Report**

***Title of the Project: Calculator***



|  |  |
| --- | --- |
| **Major :** | **Software Engineering** |
| **Class :** | **2019** |
| **Name :** | **HOSEN ARAFAT** |
| **ID No :** | **199076003** |

**School of Computer Science and Technology**

**December 14, 2021**

***CONTENTS***

**Ⅰ. Environment configuration**

**1. Download and install Pycharm (The whole process needs network) …** 5

1.1．Open the official website of Pycharm, Click DOWNLOAD to download. ( <https://www.jetbrains.com/pycharm/)>.................................................................. 5

1.2． Click the download button under community to download……………………….. 5

1.3． run the installation package (in administrator mode)…………………………………. 6

1.4． always click ‘next' to install………………………………………………………………………… 7

1.5．click all of the chooses……………………………………………………………………………….. 8

1.6． Download successful…………………………………………………………………………………. 8

**2. Create a project**

2.1．Click ‘i confrim...’ to enter the pycharm ……………………………………………………… 9

2.2．Click new project to create a project…………………………………………………………. 10

2.3．Choose the environment. The default is OK. Click create to create it…………. 11

2.4．After entering, delete all codes in main.py first…………………………………………. 11

**Ⅱ.Development design**

**1.Basic configuration**

1.1. The pyinstaller library needs to be installed in this experiment. We need

to import it first. First, click file - > setting to open the configuration

interface and select Python interpreter under project………………………………. 12

1.2．Click the configure button to select Add to add compiler………………………... 14

1.3．Select the python folder location in the base interpreter……………………….. 15

1.4．After selecting the compiler, click '+' to add the library…………………………… 16

1.5．Select the pyinstaller library to download………………………………………………. 18

1.6. download succeeded………………………………………………………………………………. 20

**2.Interface writing**

2.1. Add the Tkinter library that comes with the python compiler……………………. 21

2.2. Add a form, set its title, length, width, and display……………………………………. 21

2.3. Click Run - > run... And select the main.py file to compile……………………....... 22

2.4. Add input text box (display box)……………………………………………………………….. 24

2.5. Add buttons for 1-9 and other operations……………………………………….……….. 24

2.6. Add bigger length, width, and display……………………………………………………….. 25

2.7. Add bigger buttons for 1-9 and other operations………………………………………. 27

2.8. Add buttons border for 1-9 and other operations……………………………………... 28

2.9. Add buttons background color for 1-9 and other operations……………………… 30

**3.Logic design**

3.1 Write the number enter function of the input box, and add the change

function to the 1-9 button. When the button inputs a number,

it will be updated in the input box in real time…………………………………………….. 32

3.2 Write the sum of total function……………………………………………………………………. 34

3.3. Write the Clear Entry and All Clear Entry function……………………………………… 36

3.4 Compiling calculation function add, subtract, multiply and divide,

and equal function, and write code, click the button to call the

corresponding function………………………………………………………………………………. 38

3.5. Write the mathPM function………………………………………………………………………. 40

3.6. Write the Squared function……………………………………………………………………….. 42

3.7. Run the program for calculator related operations……………………………………… 46

1) “add” 10+4=14………………………………………………………………………………………… 46

2) “sub” 50 – 6 = 44……………………………………………………………………………………… 48

3) “multi” 16 \* 2 = 32…………………………………………………………………………………. 49

4) “divide” 25 / 5 = 5…………………………………………………………………………………….. 50

5) “mathPM” 69 ±………………………………………………………………………………………... 52

6) “Squared” 9√…………………………………………………………………………………………….. 53

**III. Packaged applications**

1. Click terminal to open the terminal……………………………………………………………….. 54

2. Enter the code "pyinstaller –onefile main.py " package "………………………………. 54

3. Package succeeded……………………………………………………………………………………….. 55

4.Open the main.exe file in the dist folder………………………………………………………… 55

**IV.Coedes**………………………………………………………………………………………………………………………… 56

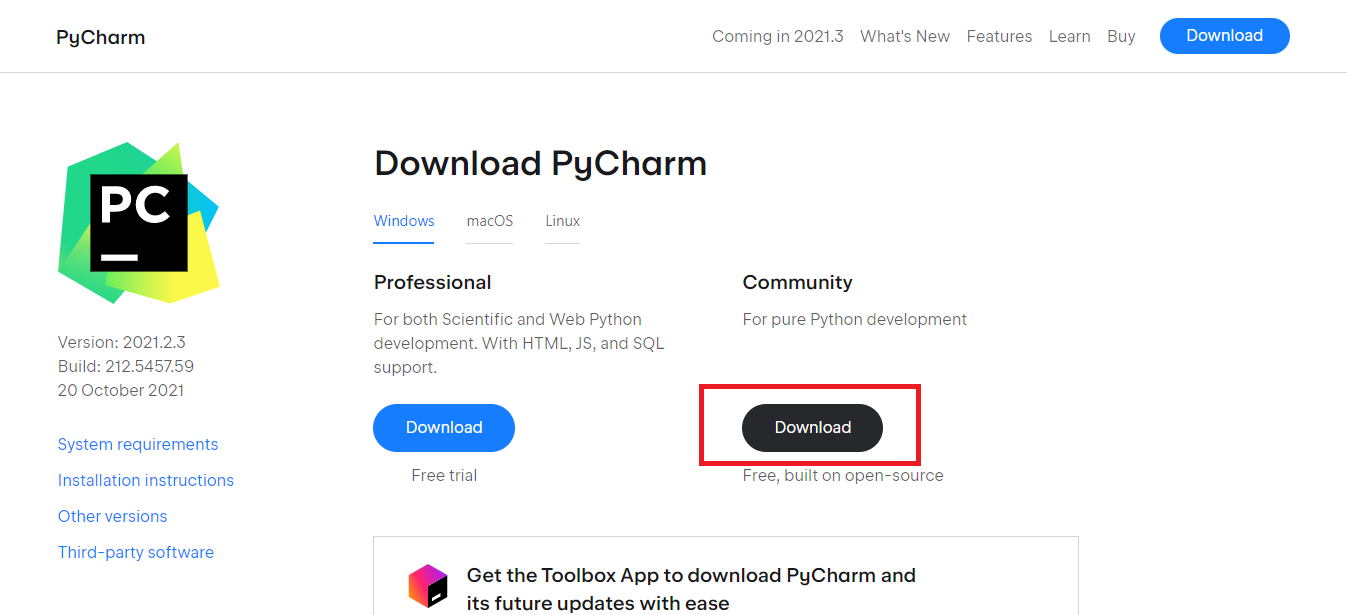
**Ⅰ. Environment configuration**

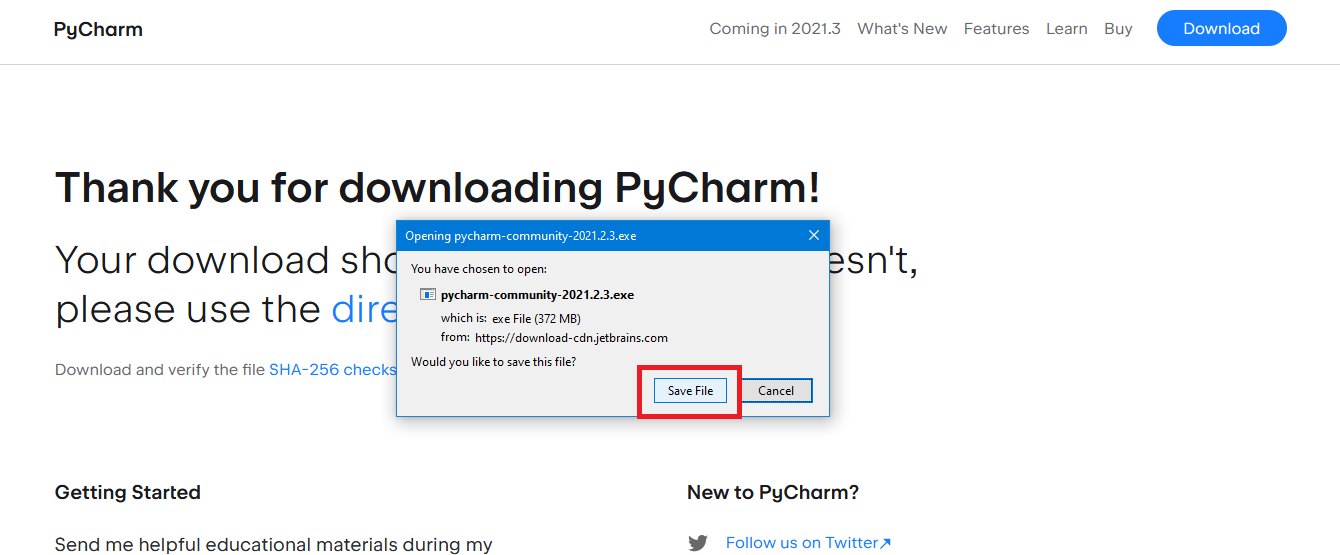
1. Download and install Pycharm (The whole process needs network)

1.1. Open the official website of Pycharm, Click DOWNLOAD to download. ( <https://www.jetbrains.com/pycharm/>）

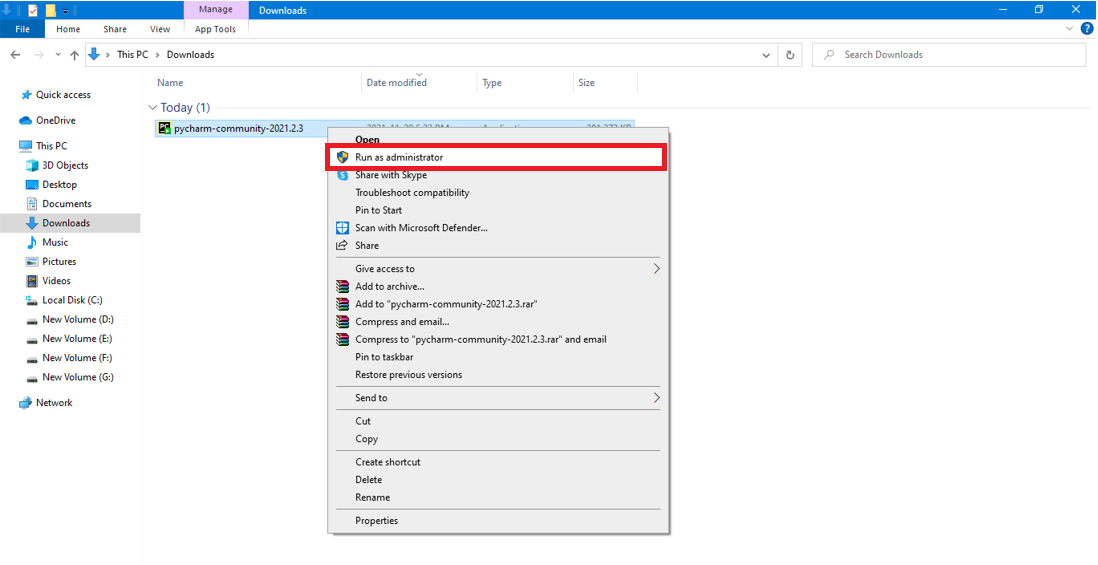


1.2. Click the download button under community to download

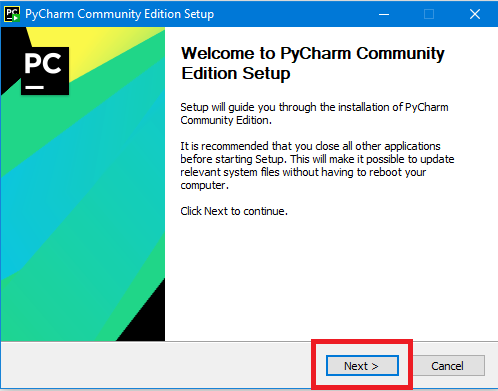


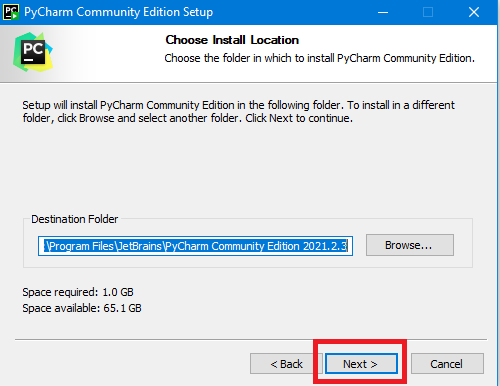
s

1.3. Run the installation package (in administrator mode).

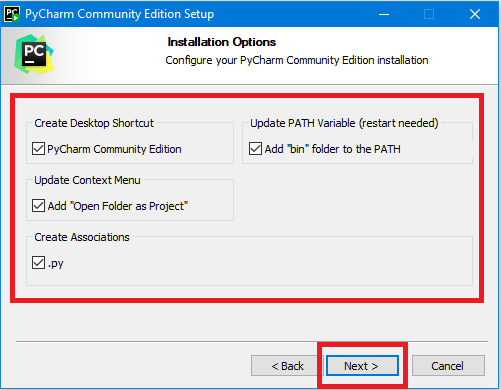


1.4． always click ‘next' to install.

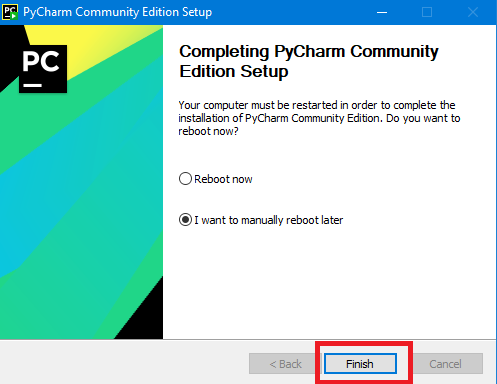




1.5．click all of the chooses

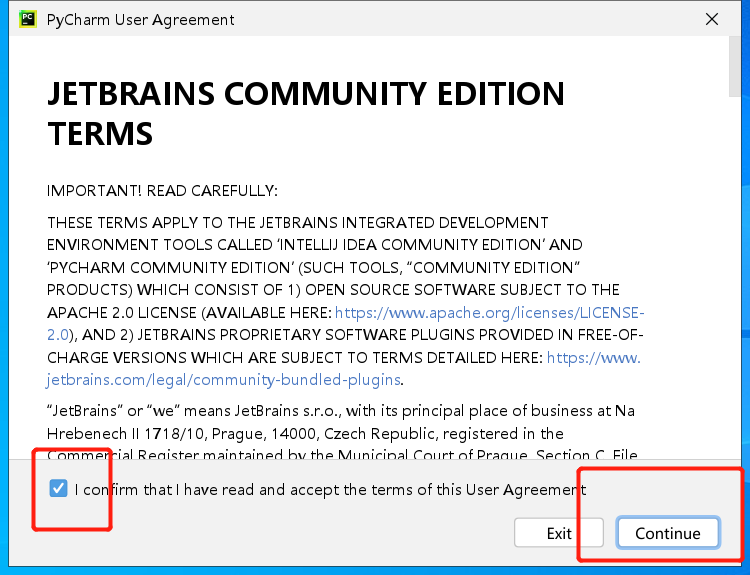


1.6． Download successful

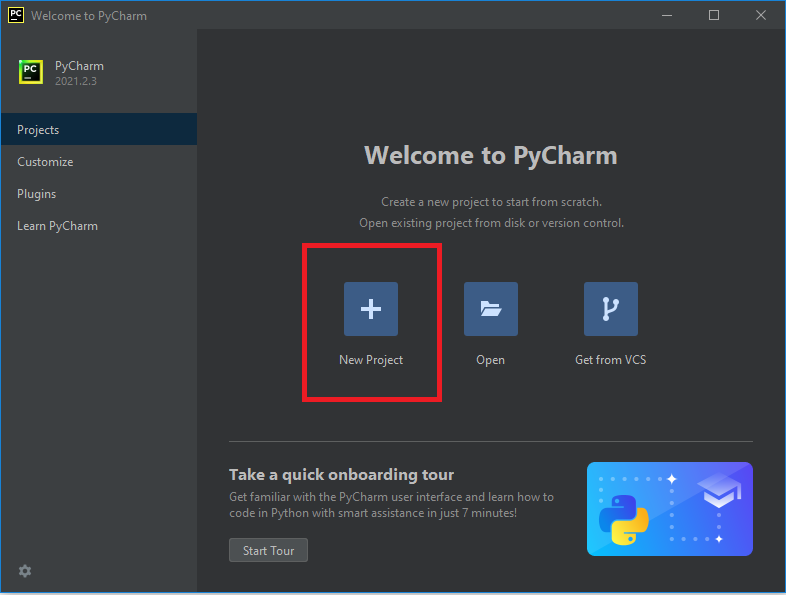


2. Create a project

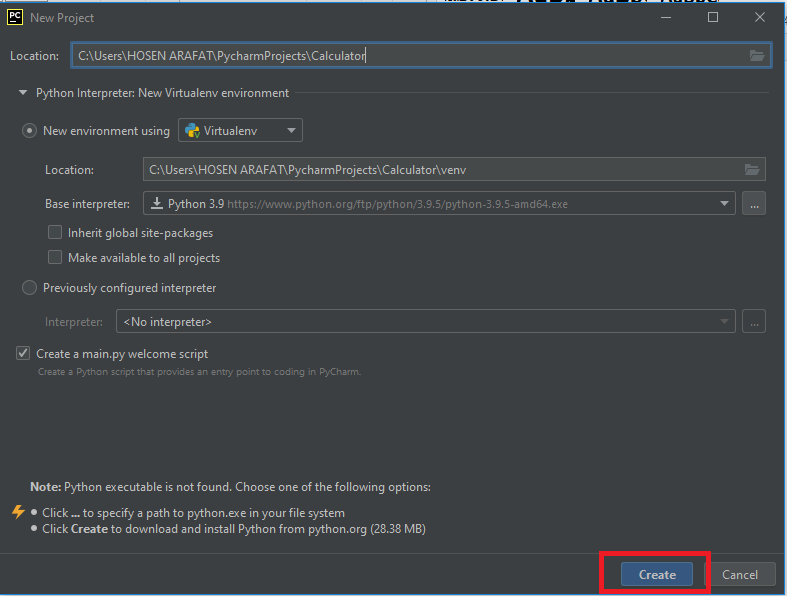
2.1．Click ‘i confrim...’ to enter the pycharm



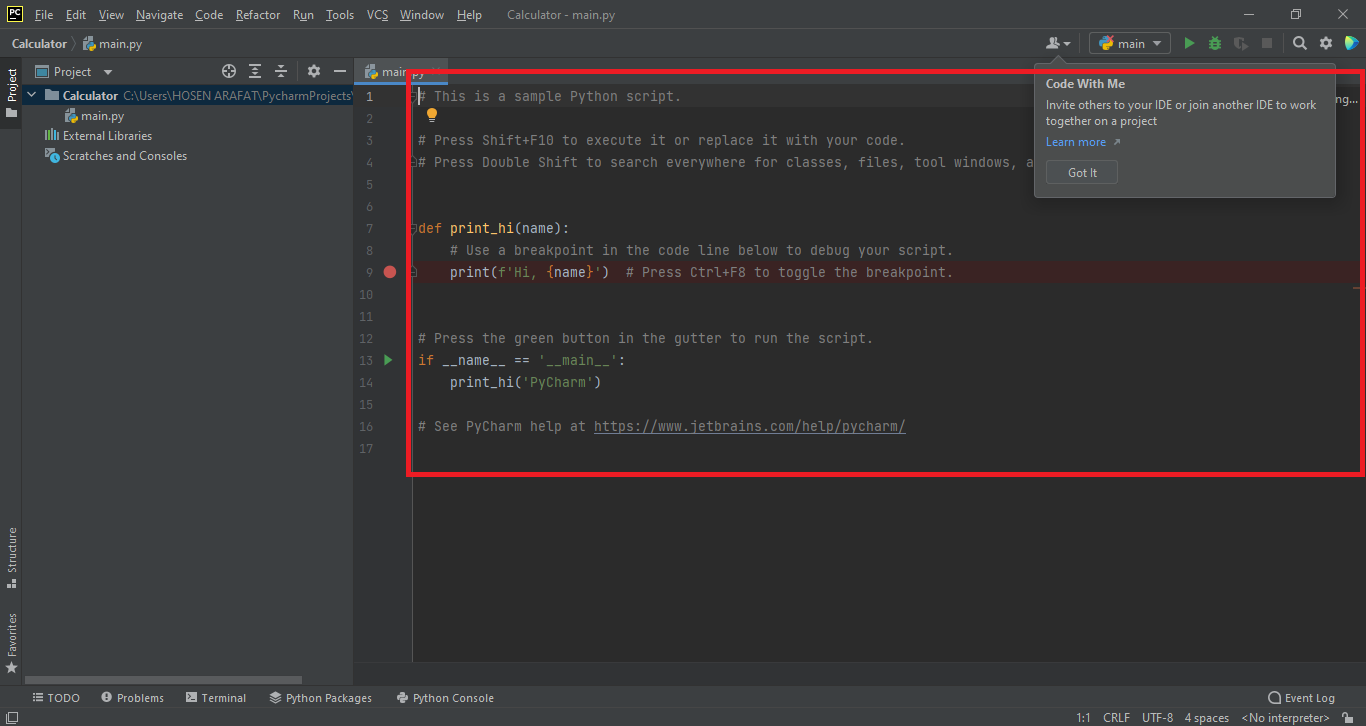
2.2．Click new project to create a project



2.3．Choose the environment. The default is OK. Click create to create it



2.4．After entering, delete all codes in main.py first

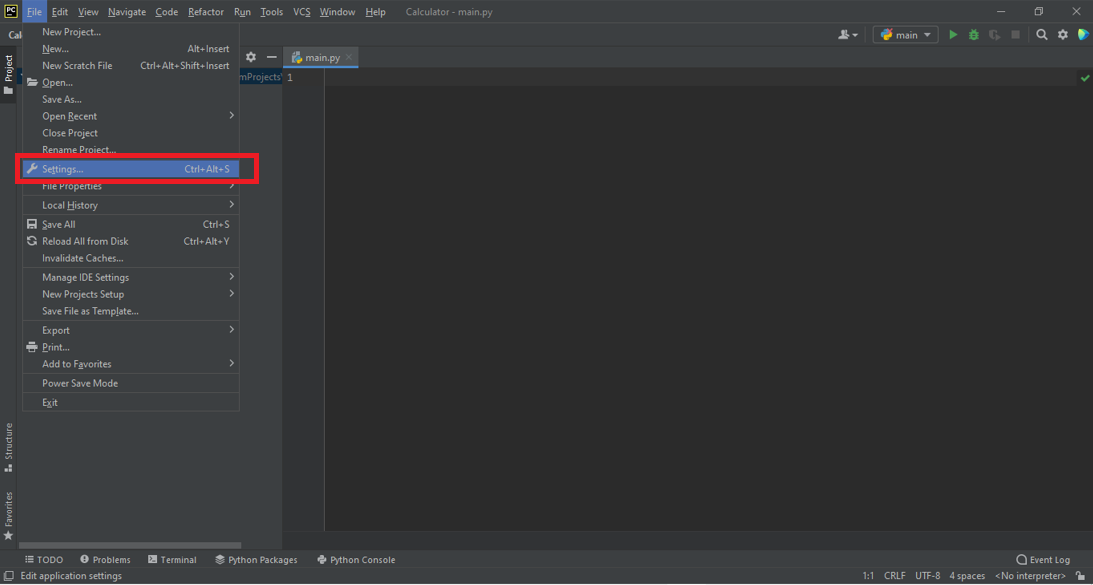


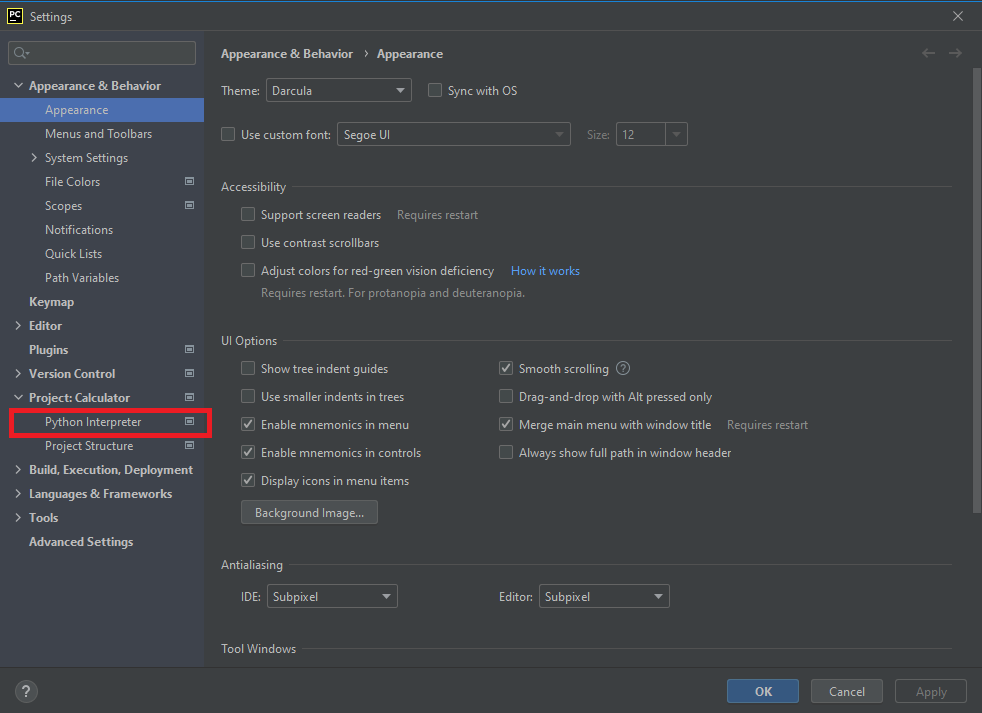


**Ⅱ.Development design**

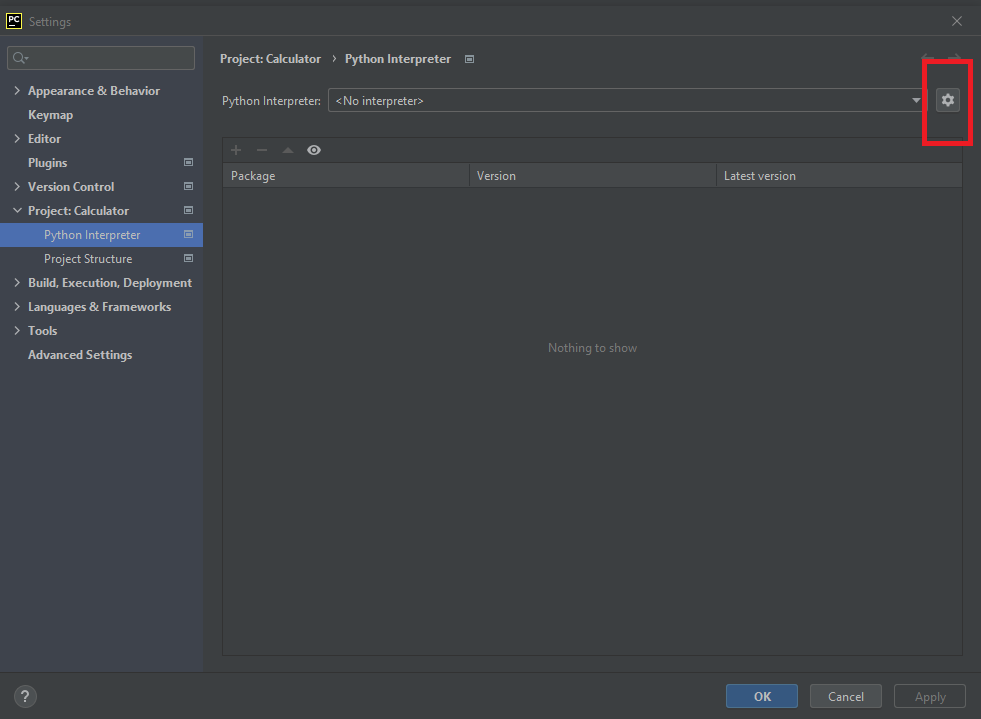
1.Basic configuration

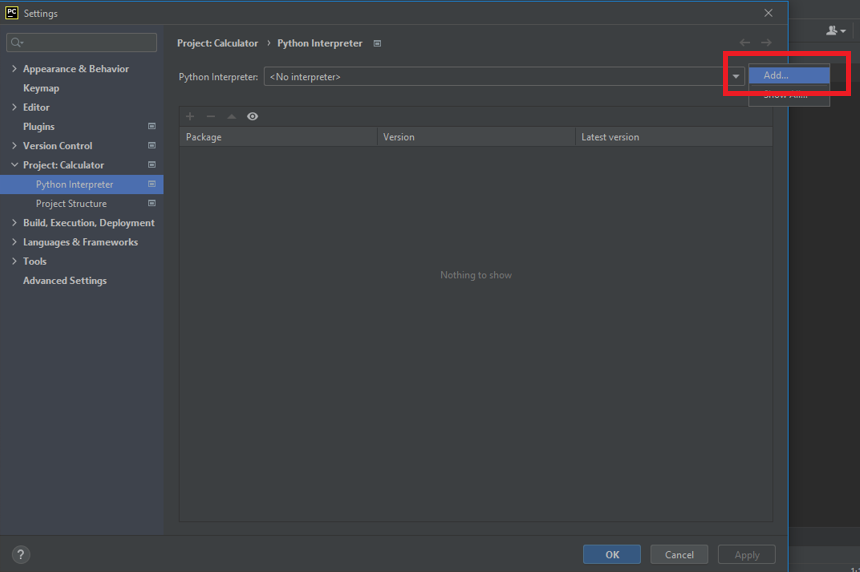
1.1. The pyinstaller library needs to be installed in this experiment. We need to import it first. First, click file - > setting to open the configuration interface and select Python interpreter under project



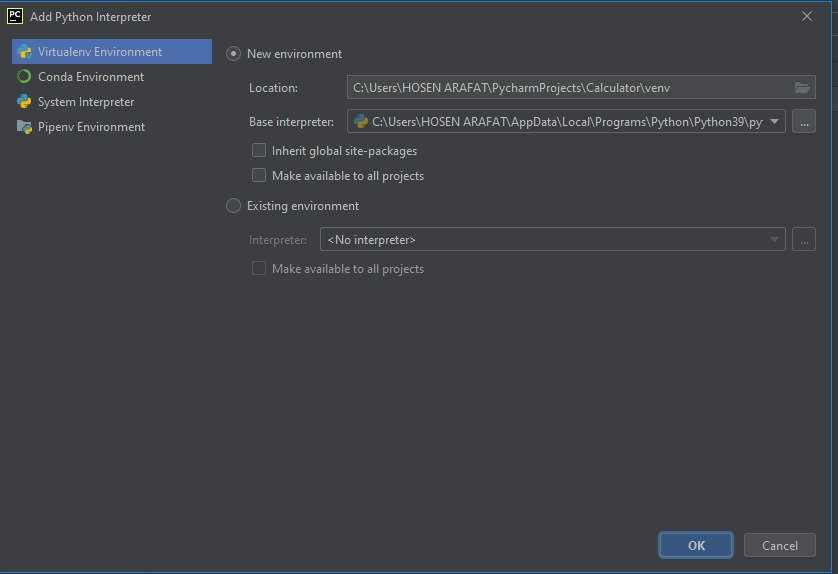


1.2．Click the configure button to select Add to add compiler.

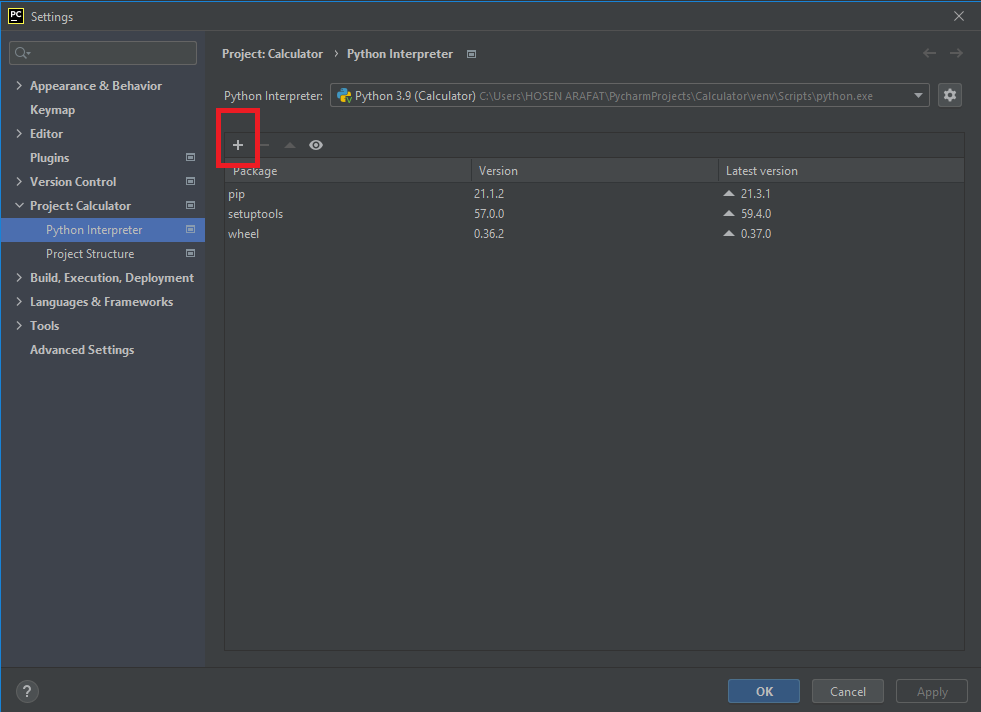


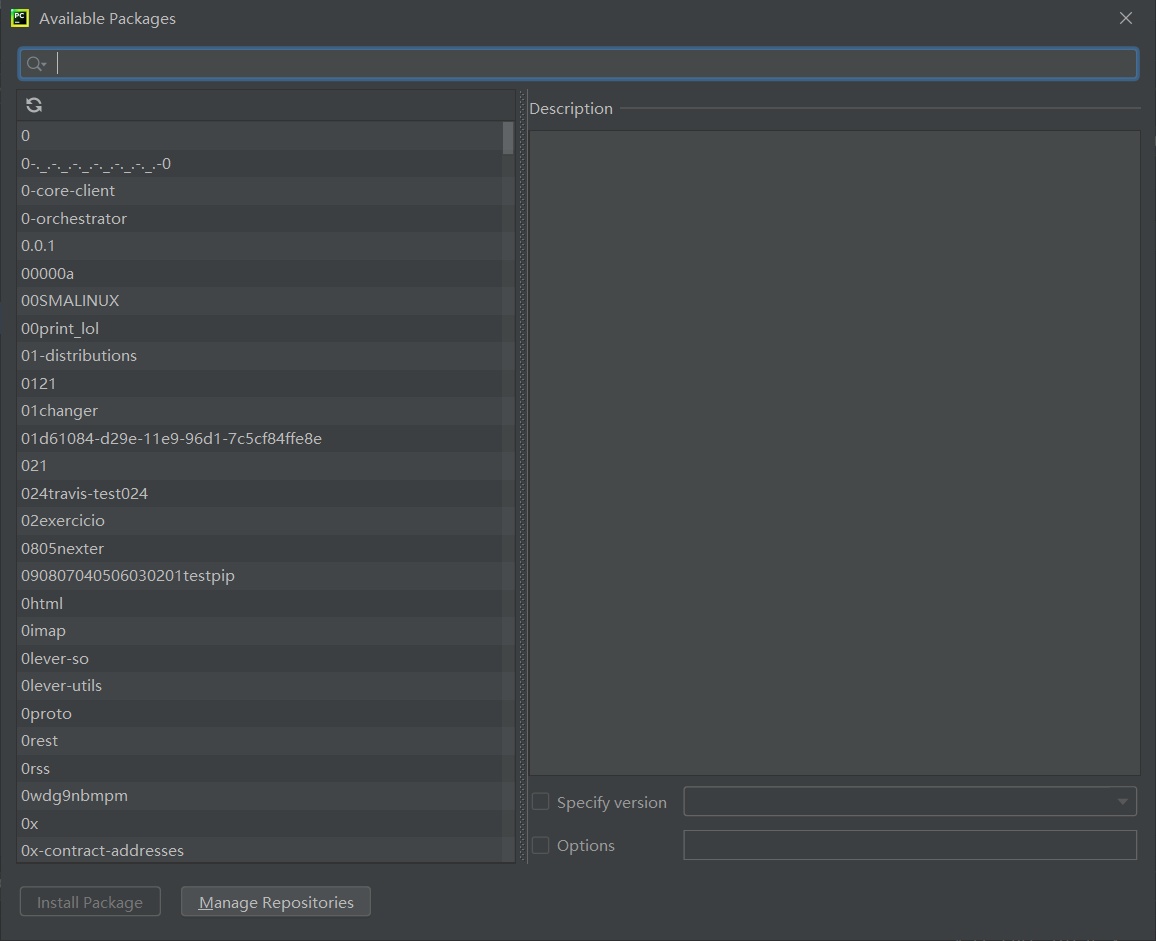


1.3．Select the python folder location in the base interpreter



1.4．After selecting the compiler, click '+' to add the library



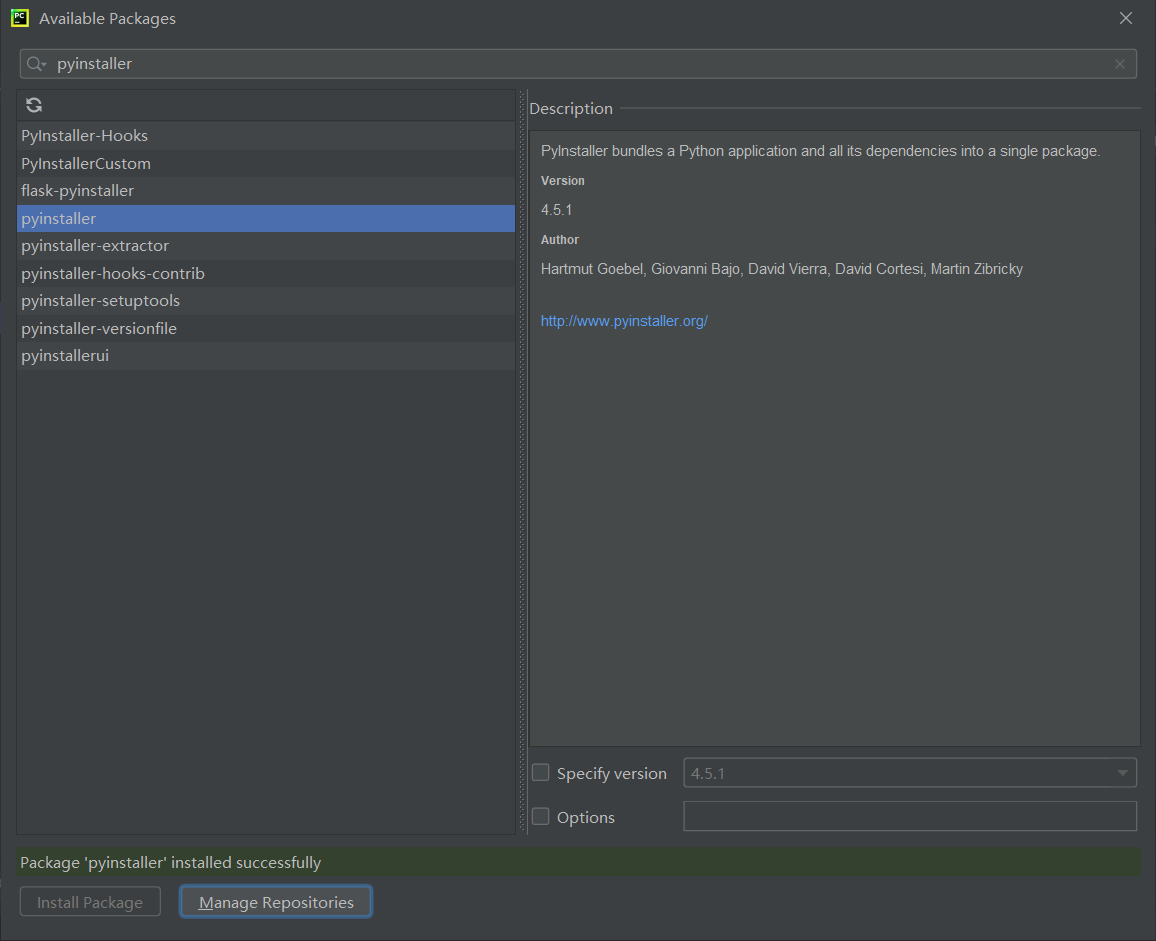


1.5．Select the pyinstaller library to download



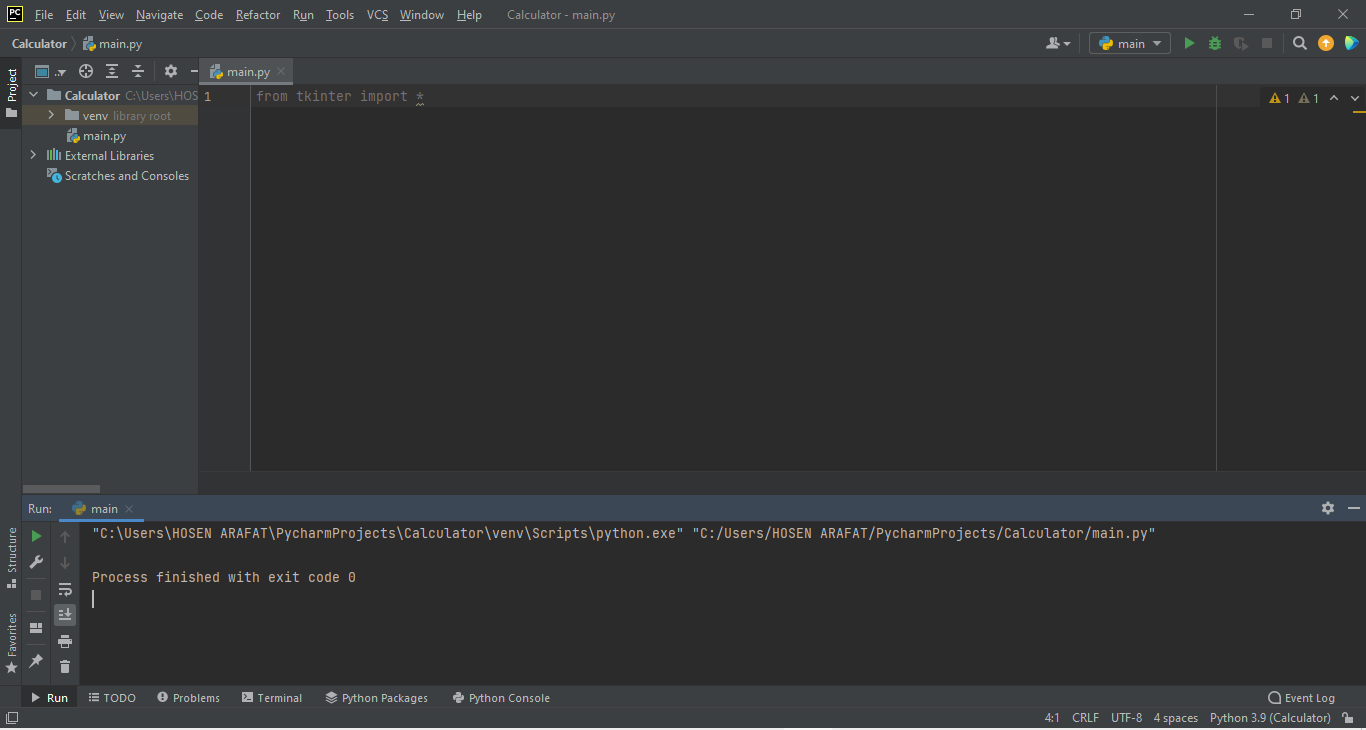


1.6. Download succeeded

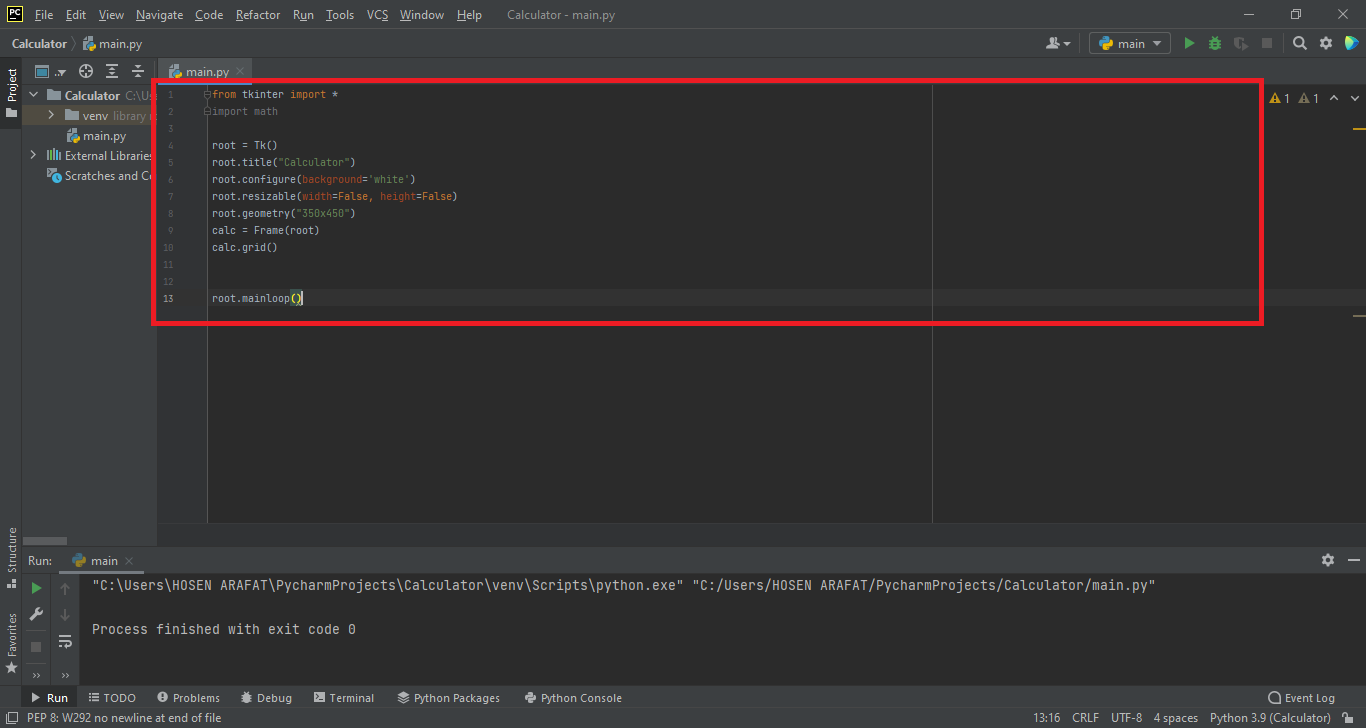


## 2.Interface writing

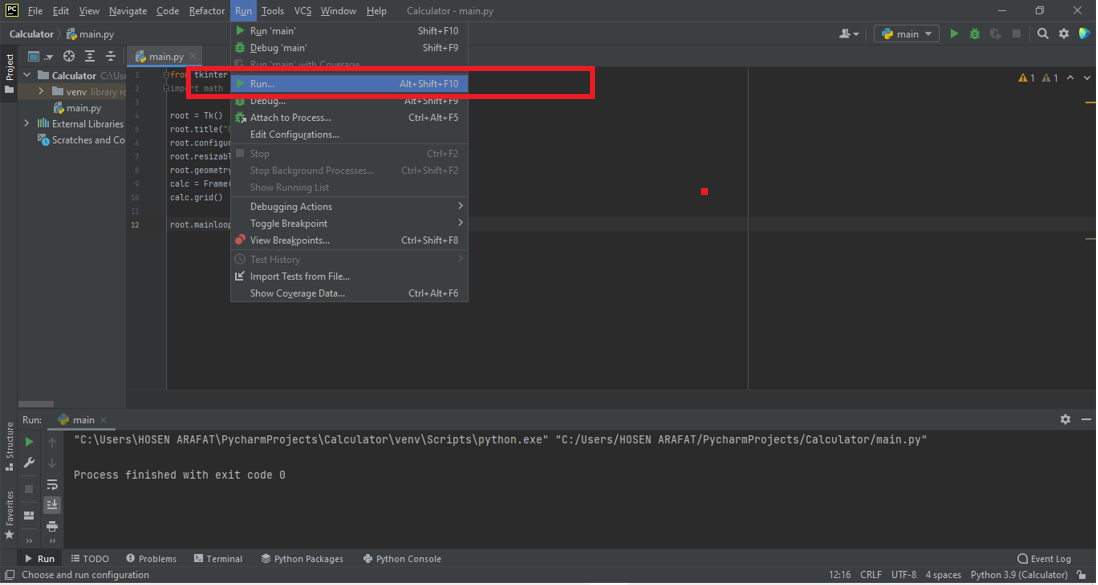
### 1.1. Add the Tkinter library that comes with the python compiler

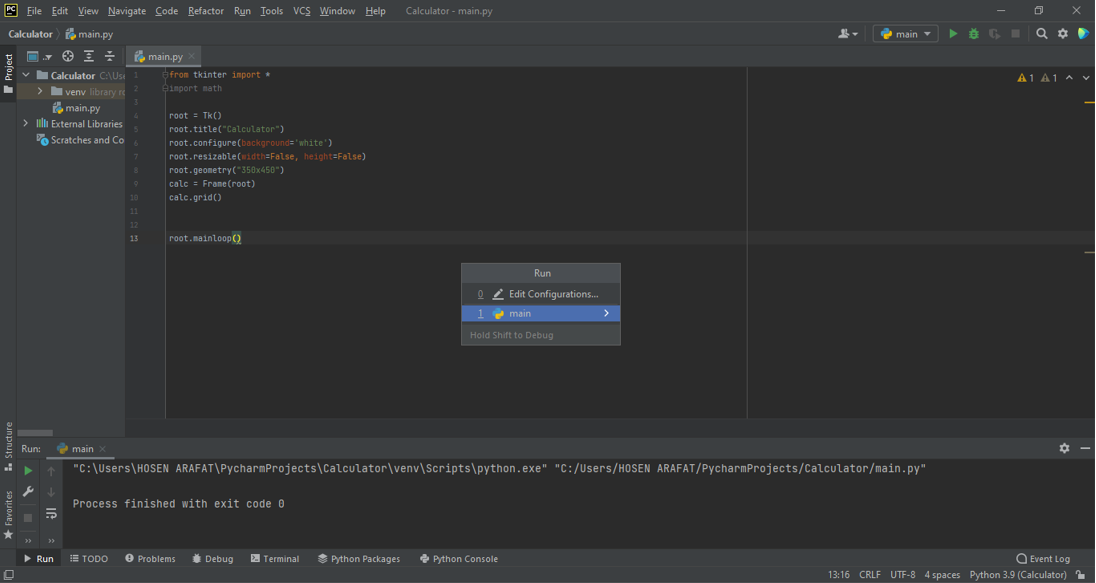


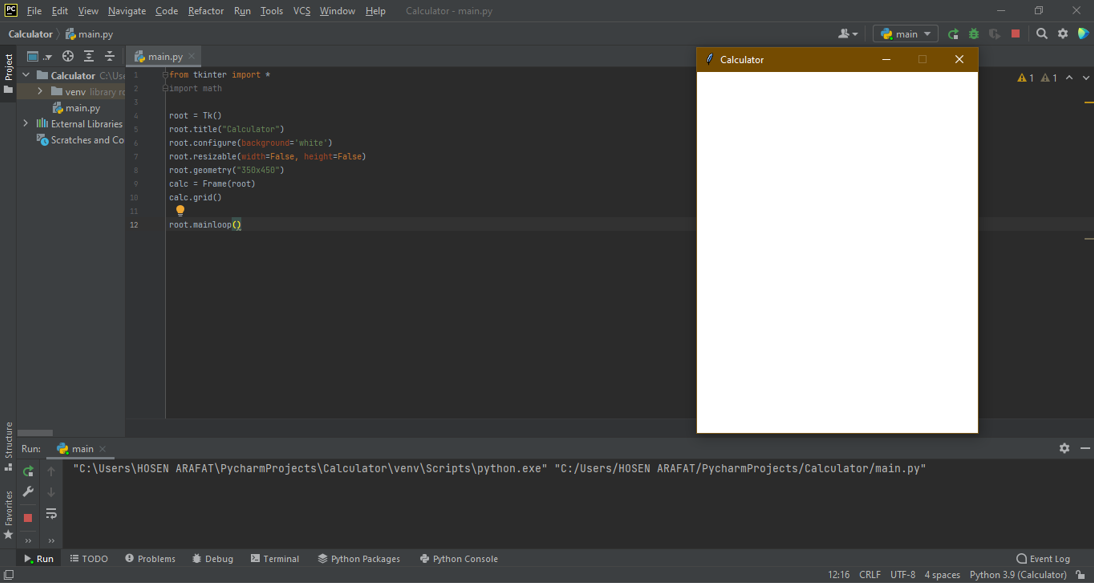
1.2. Add a form, set its title, length, width, and display



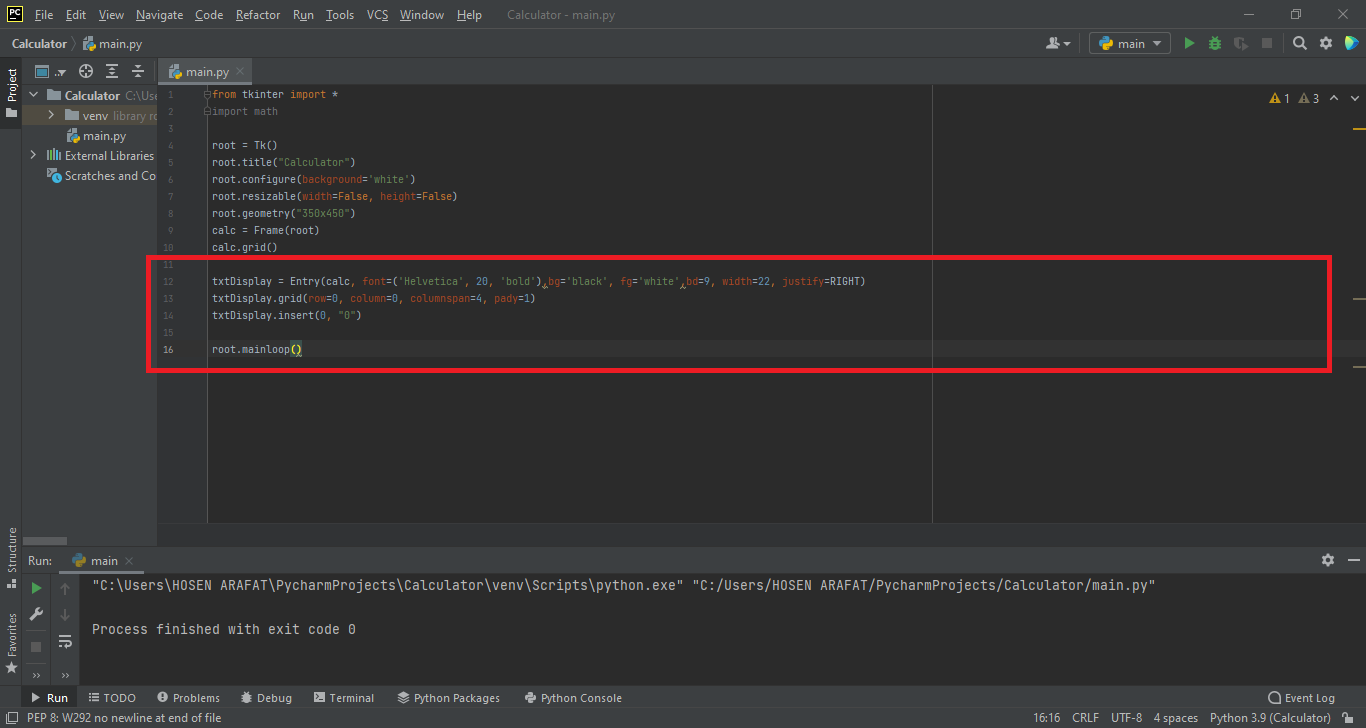
### 1.3. Click Run - > run... And select the main.py file to compile

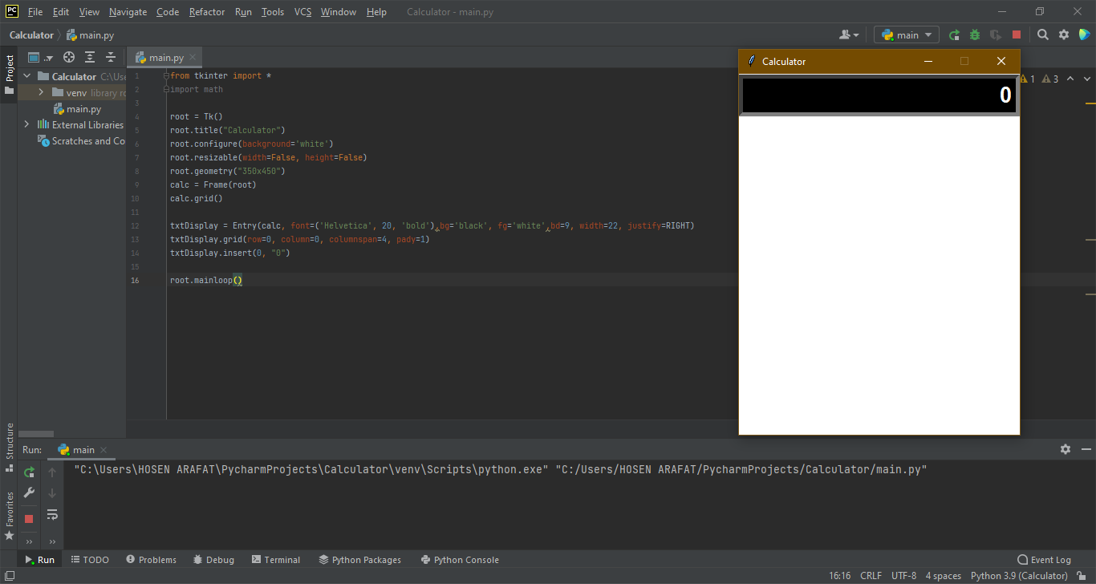




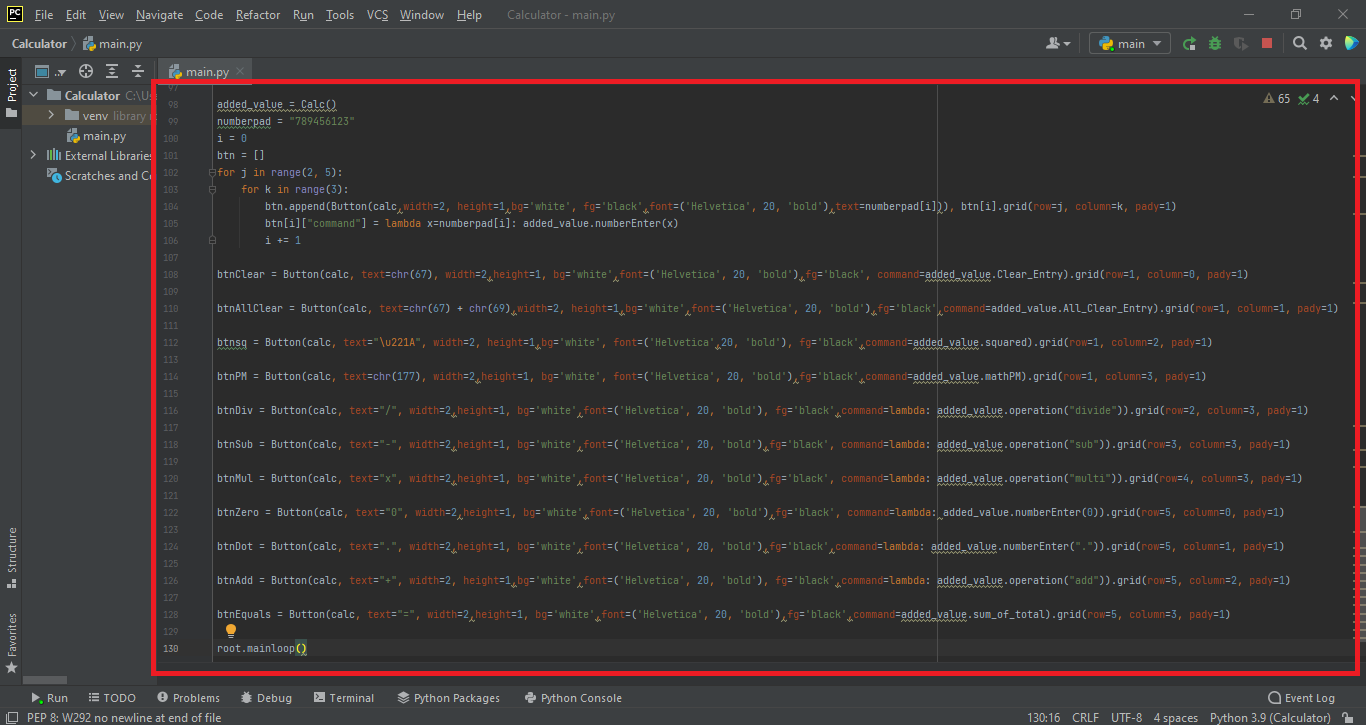


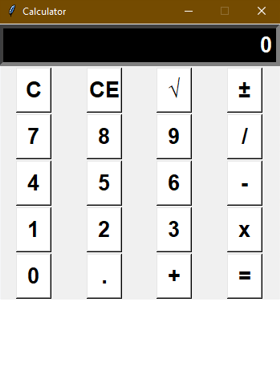
### 1.4. Add input text box (display box)



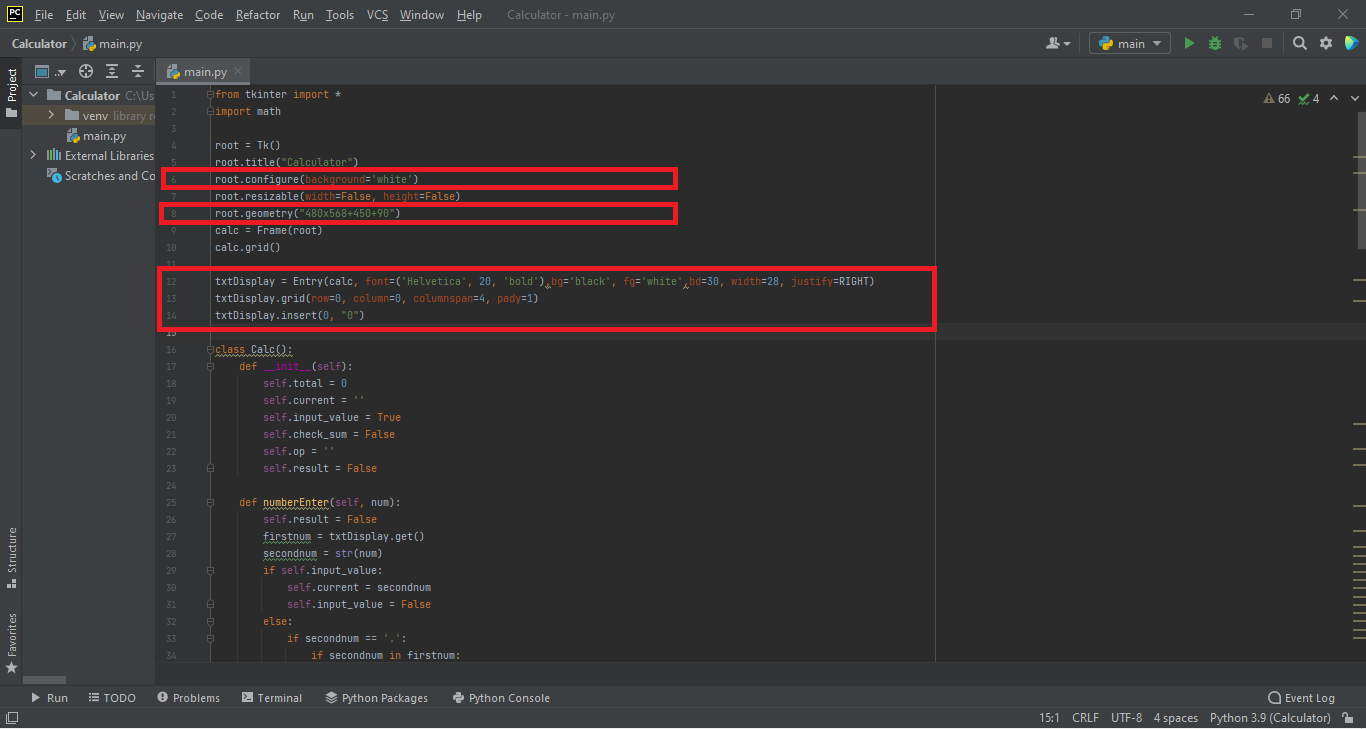


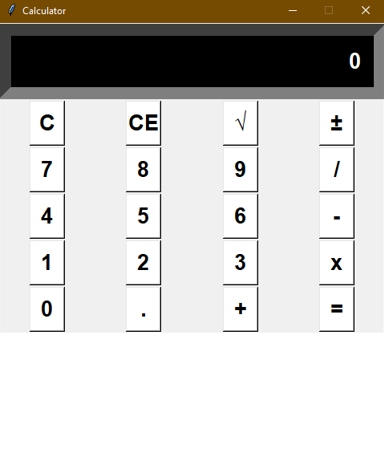
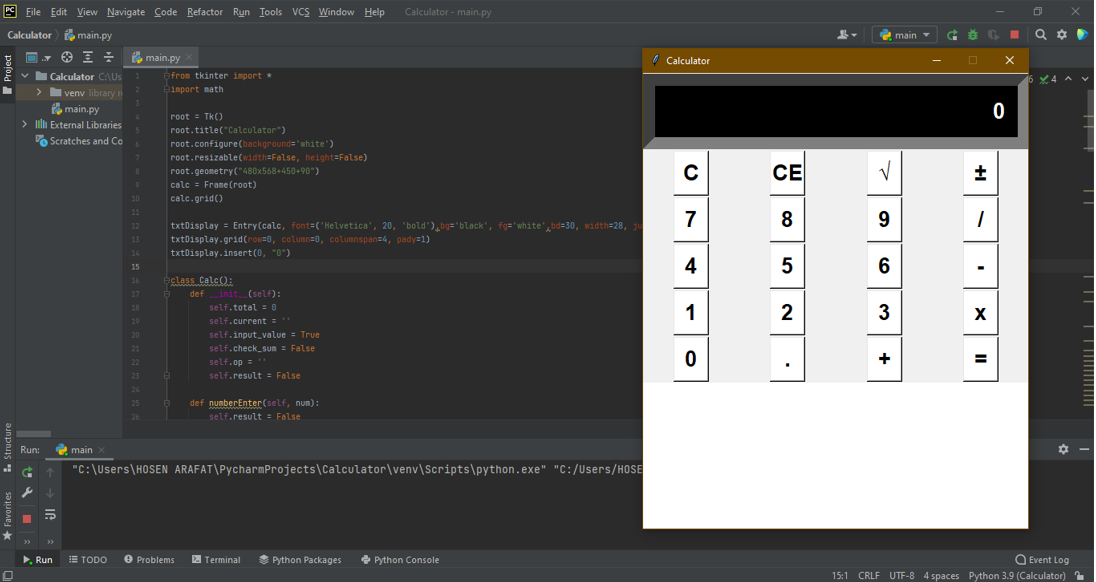
### 1.5. Add buttons for 1-9 and other operations



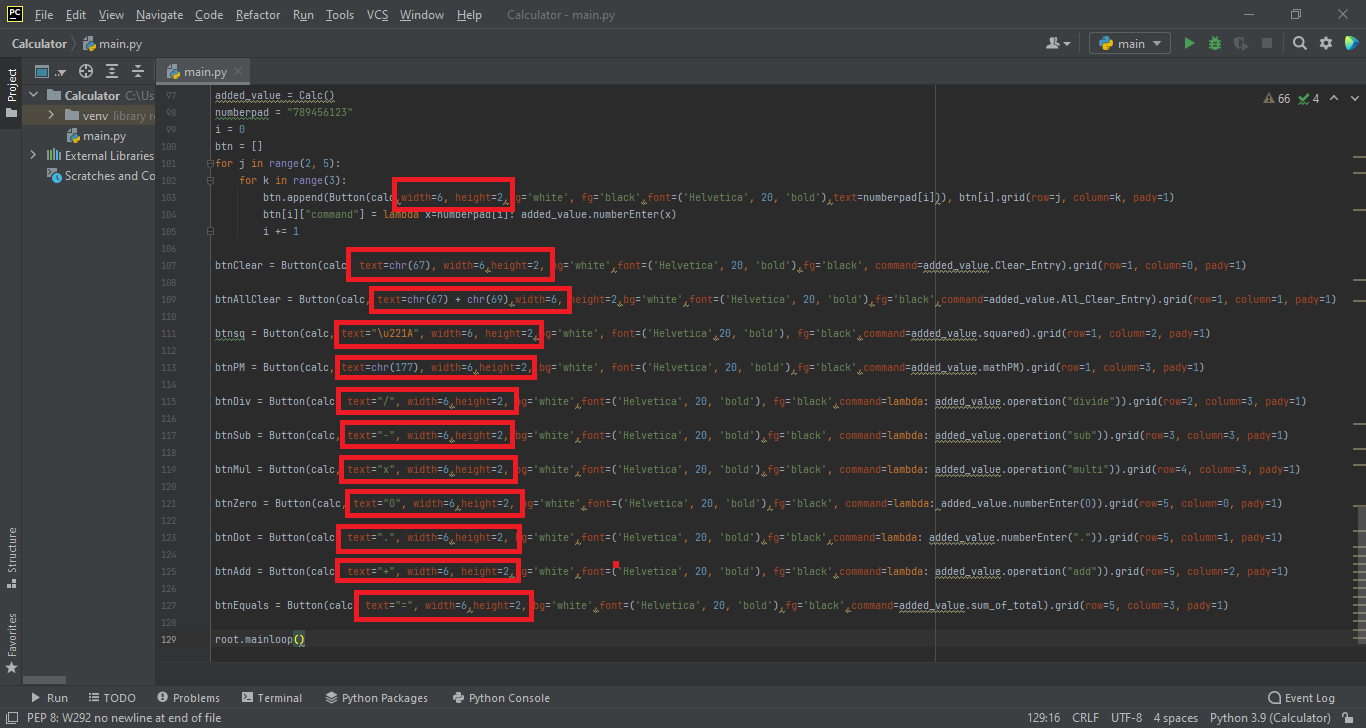


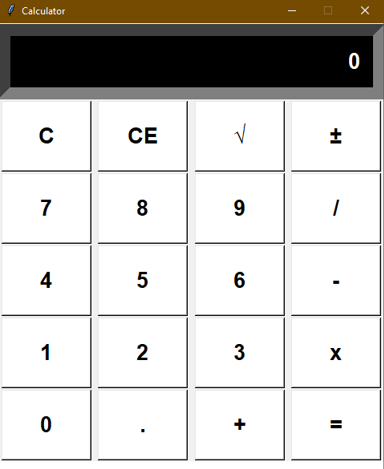
1.6. Add bigger length, width, and display



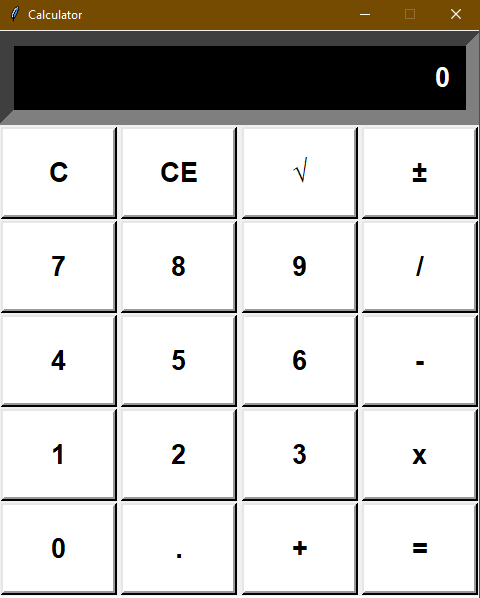
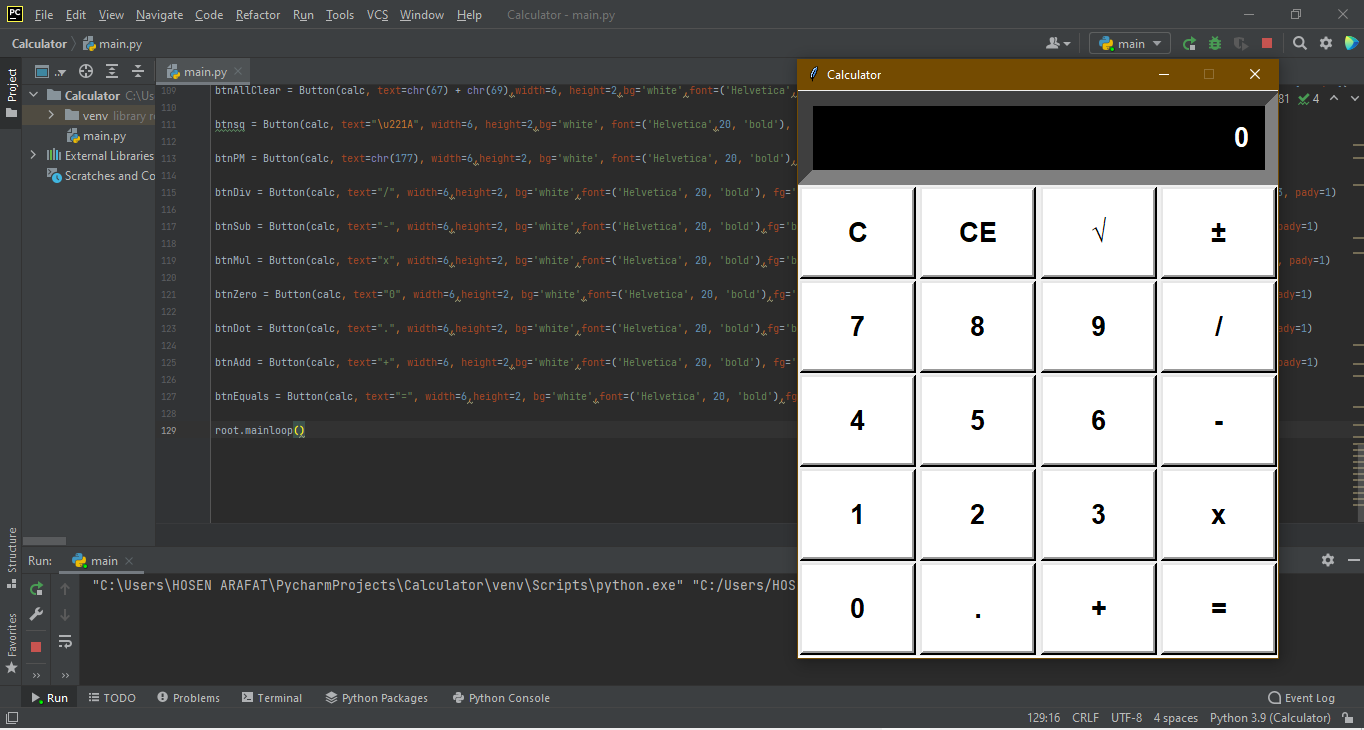
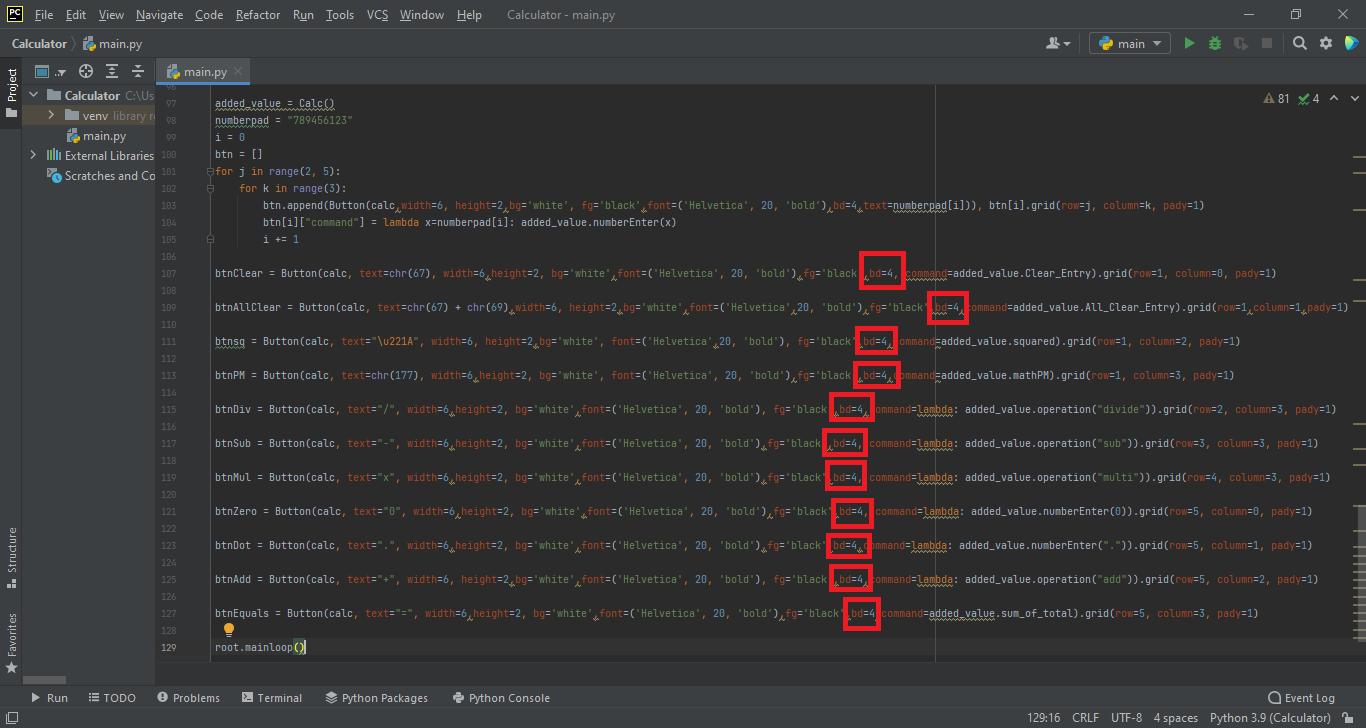


1.7. Add bigger buttons for 1-9 and other operations



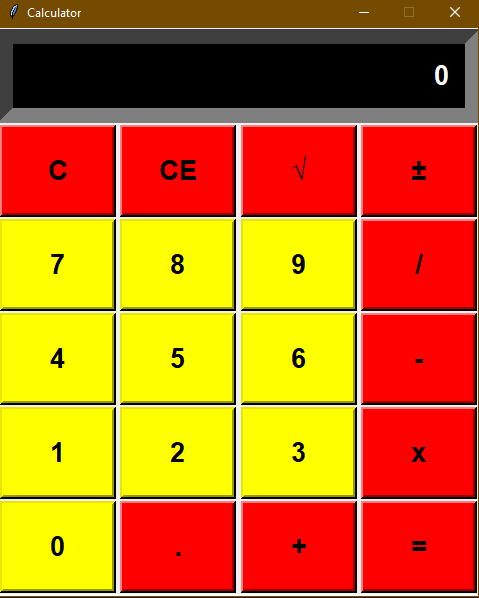
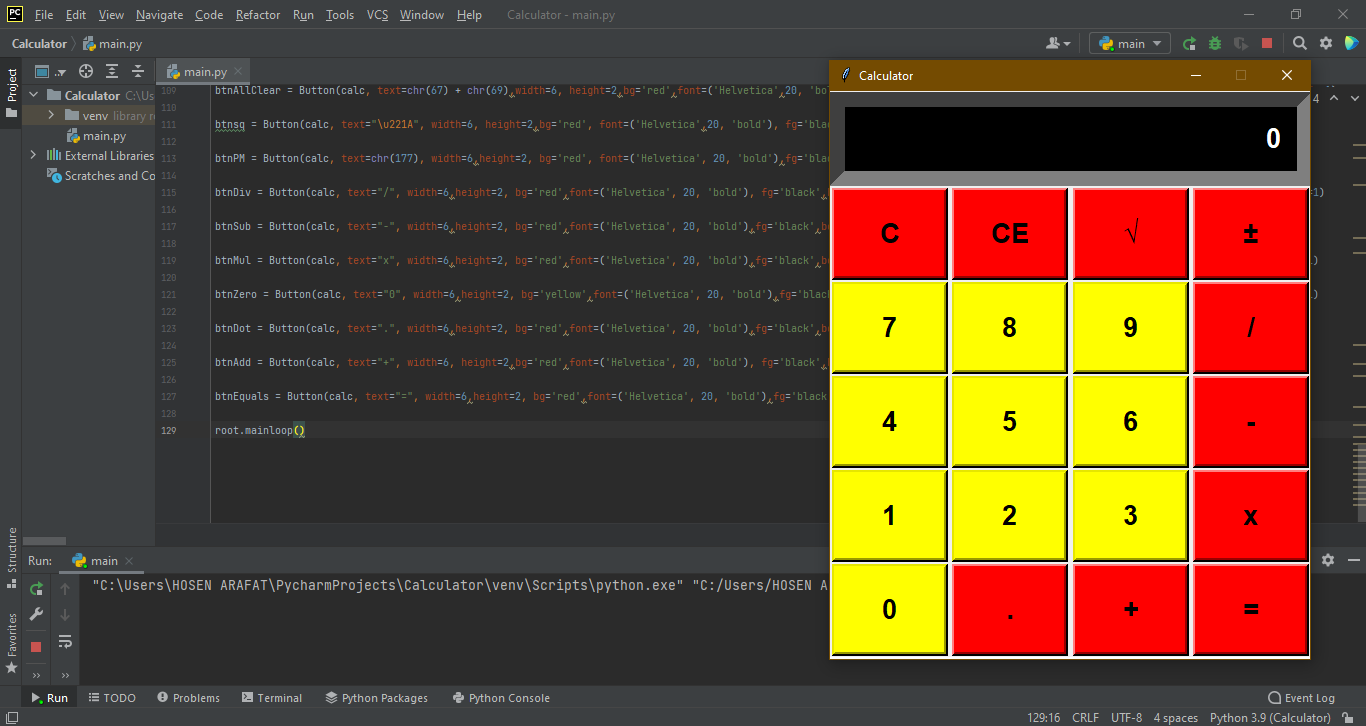


1.8. Add buttons border for 1-9 and other operations



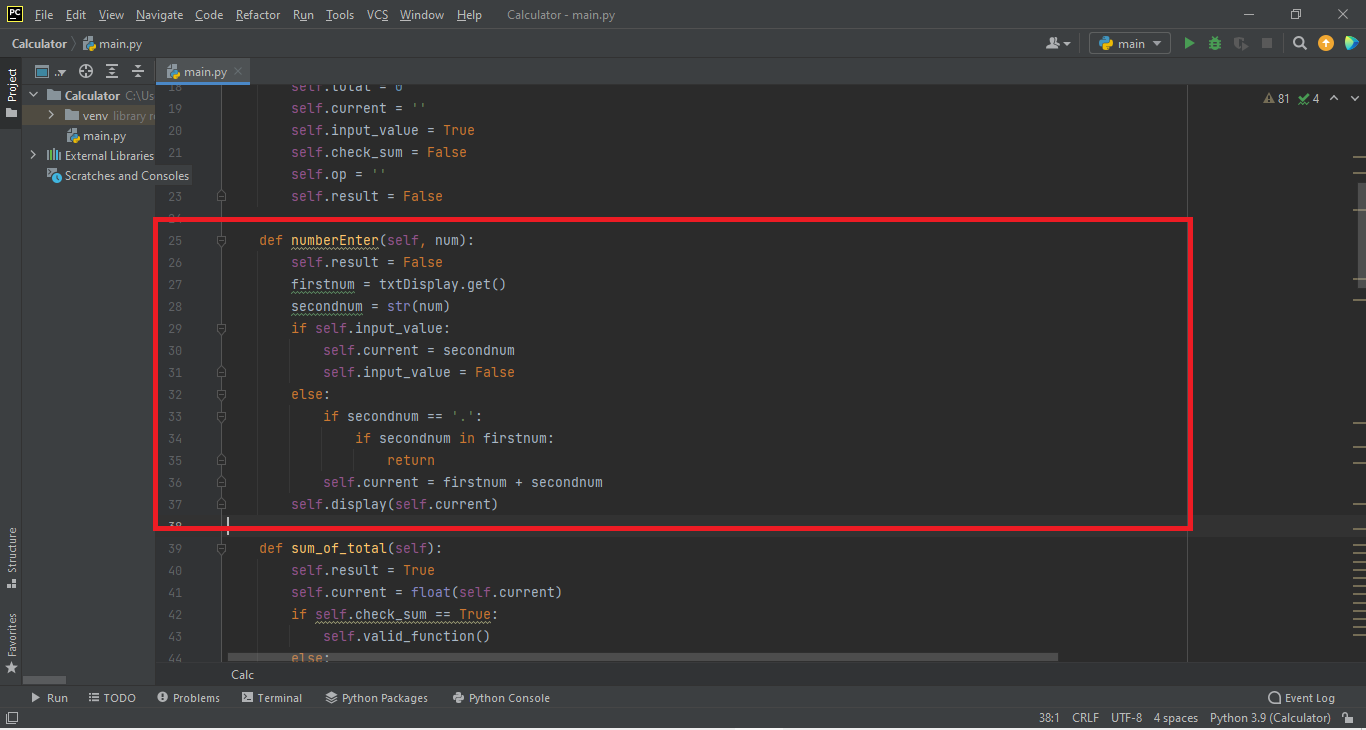
1.9. Add buttons background color for 1-9 and other operations

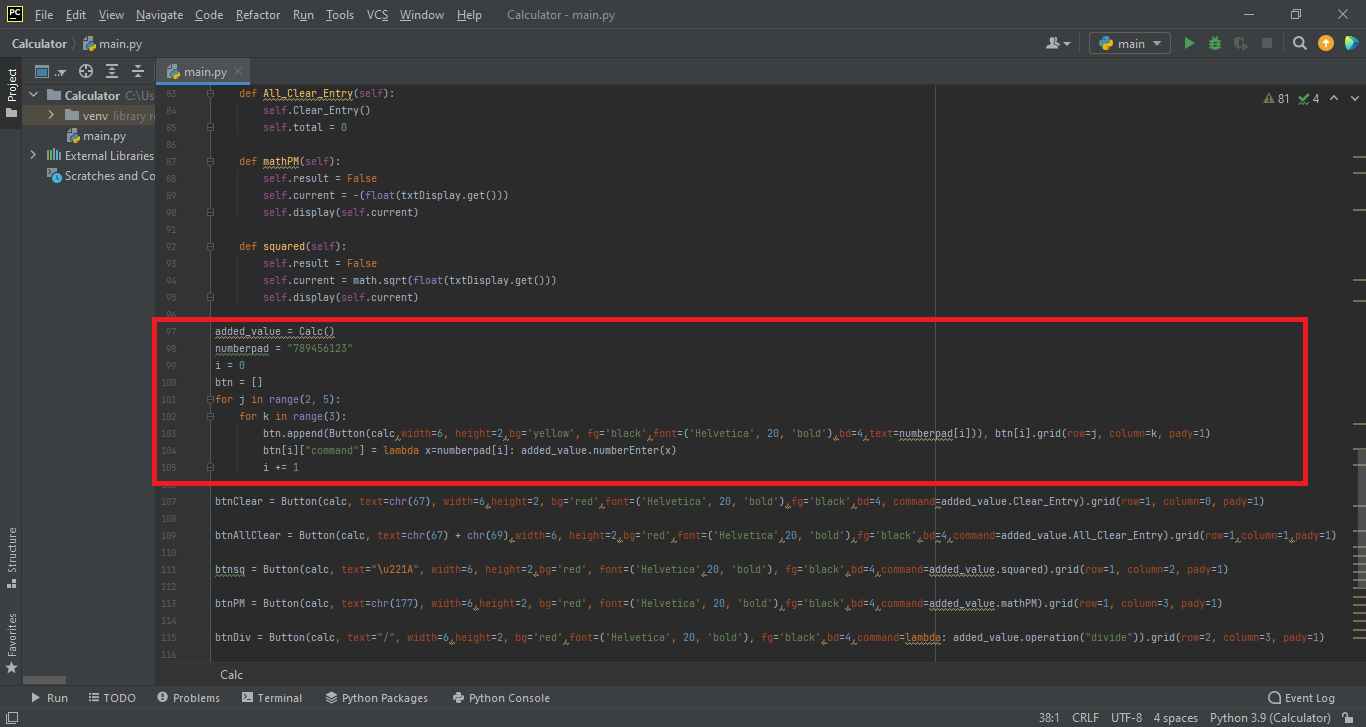


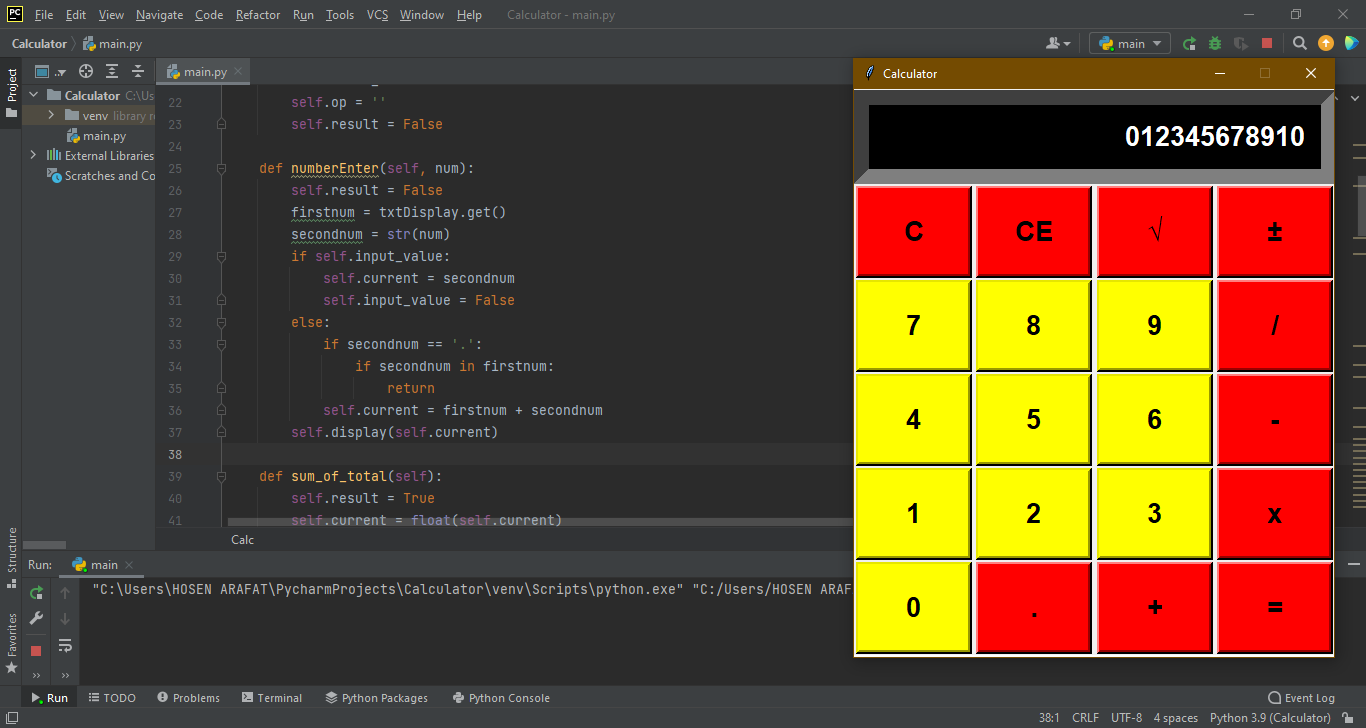


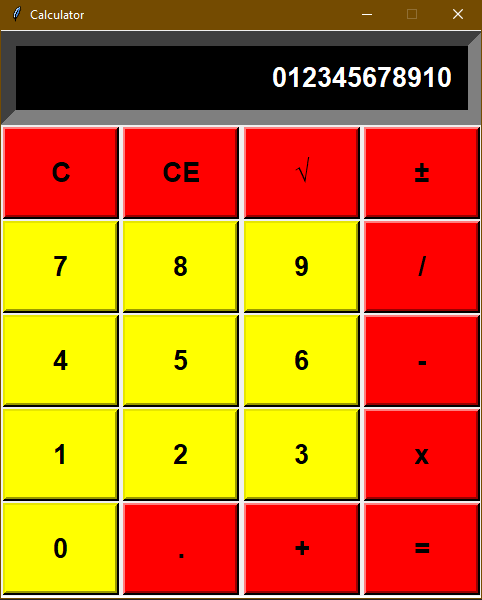
## 3.logic design

### 3.1 Write the number enter function of the input box, and add the change function to the 1-9 button. When the button inputs a number, it will be updated in the input box in real time.

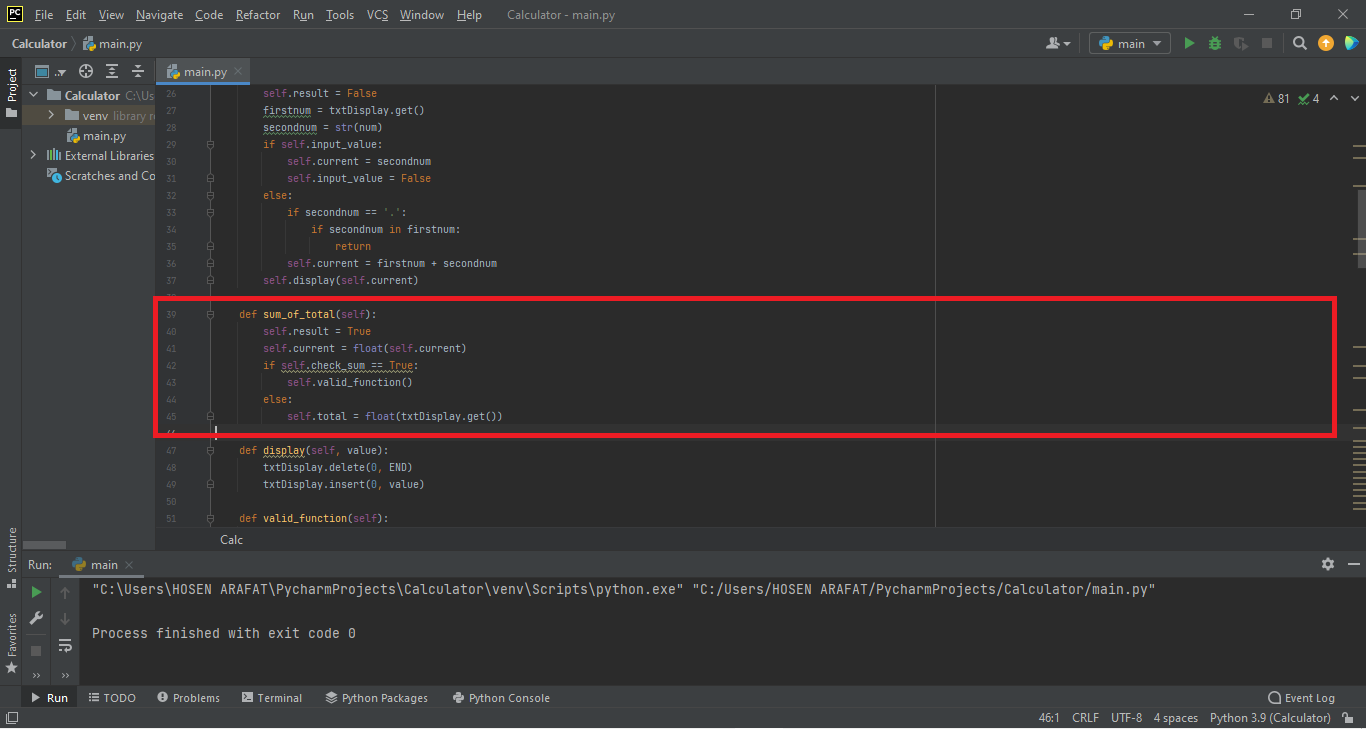


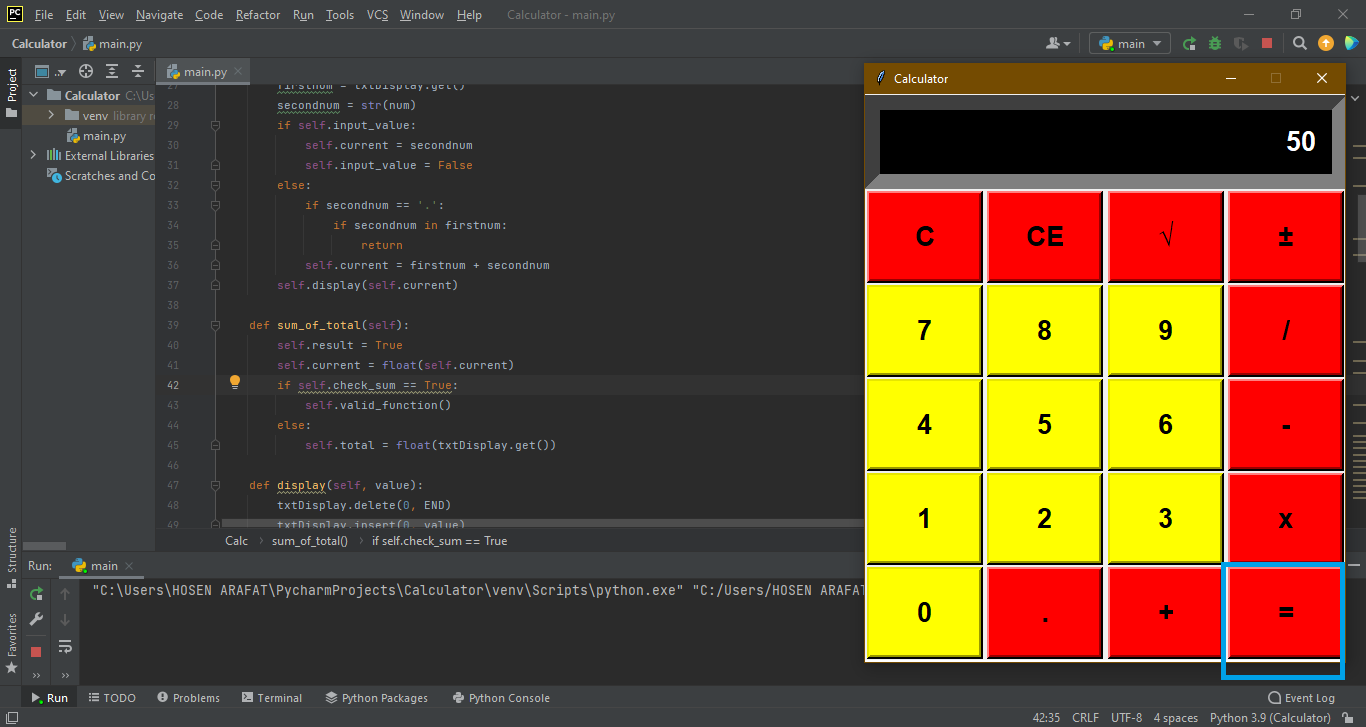
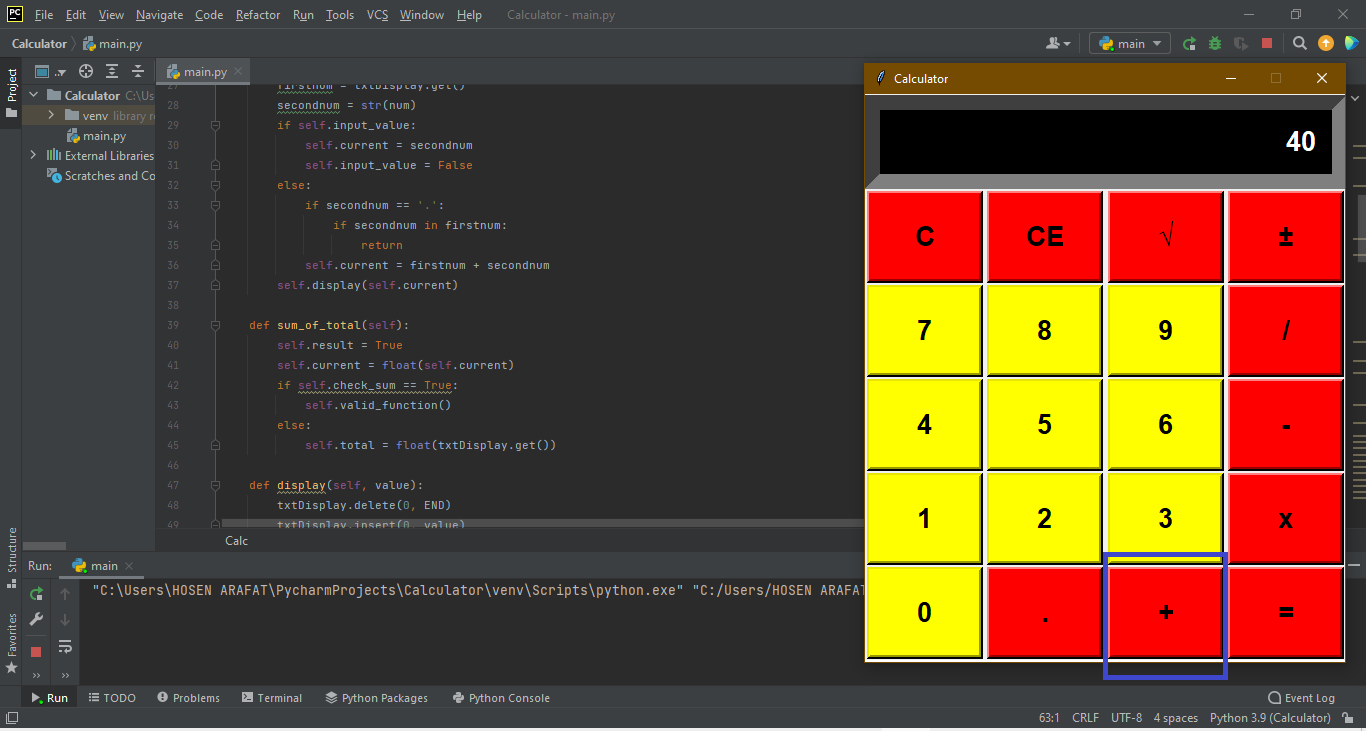


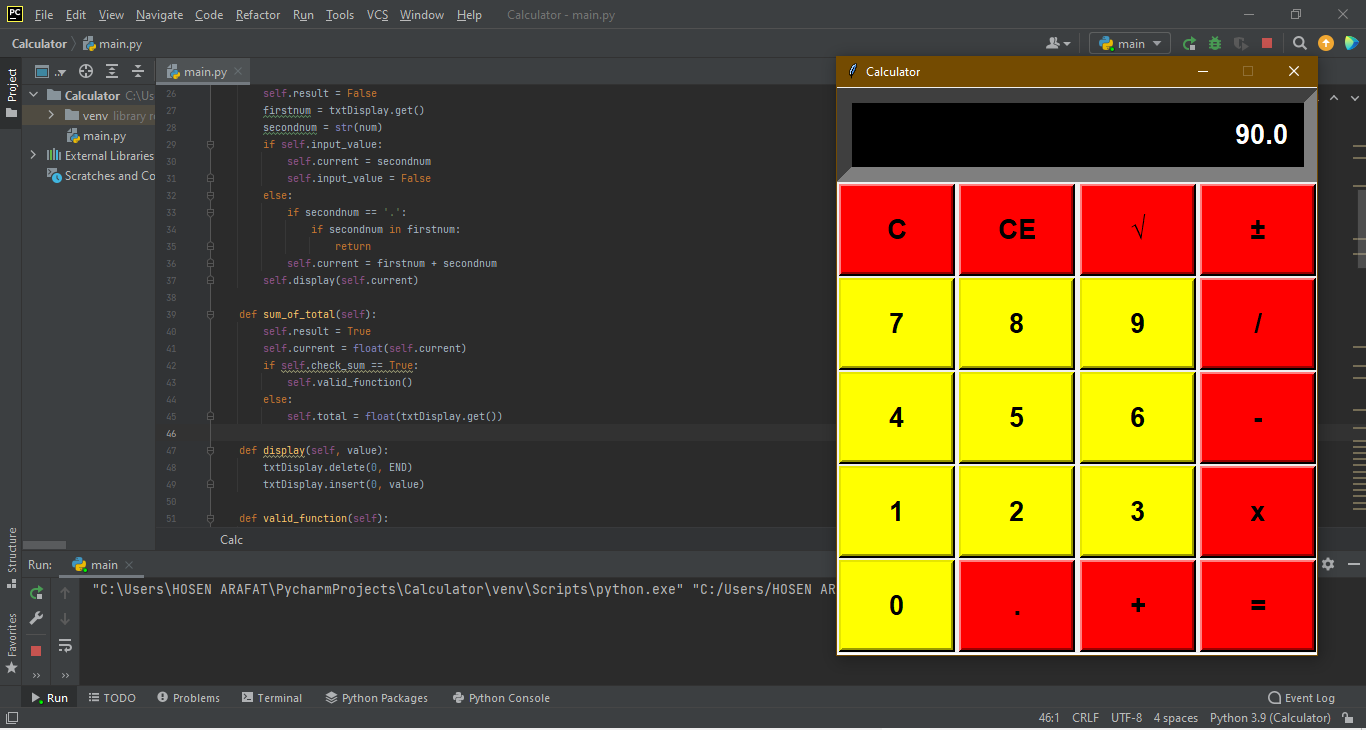




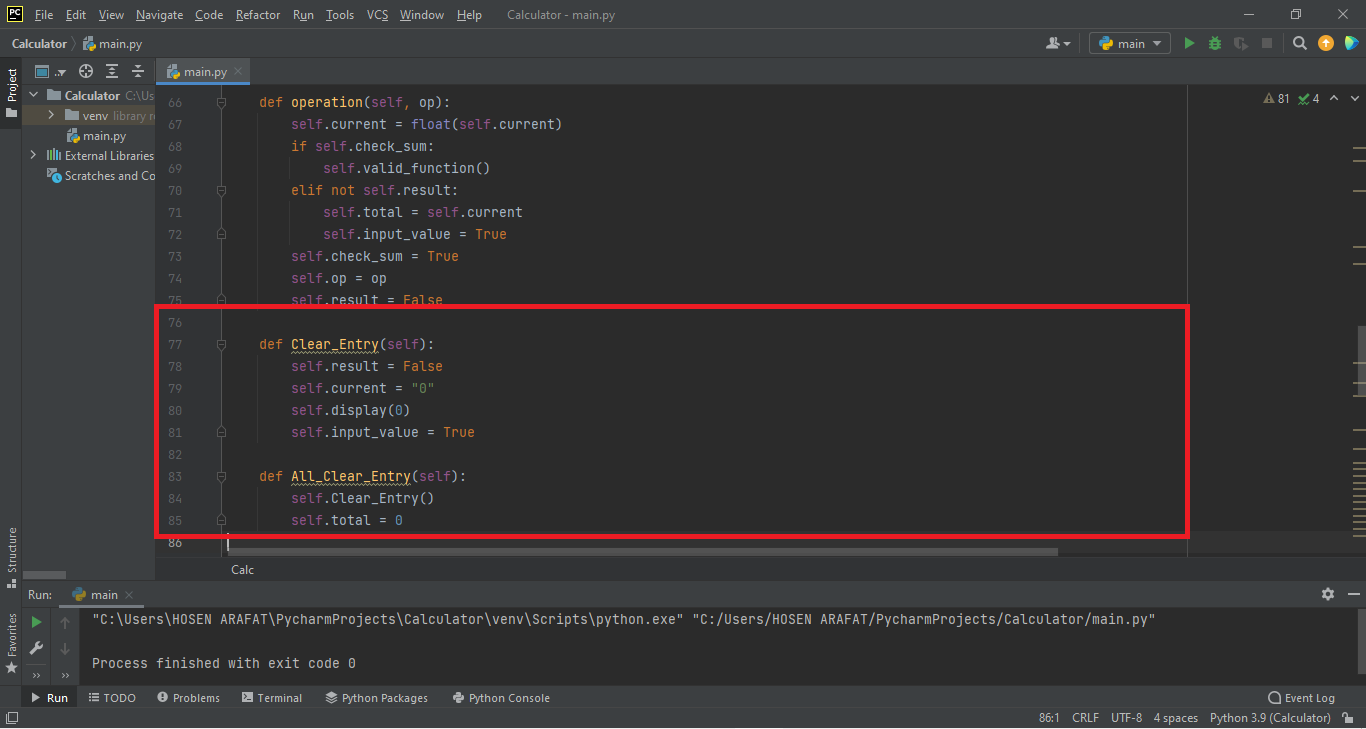
3.2 Write the sum of total function.

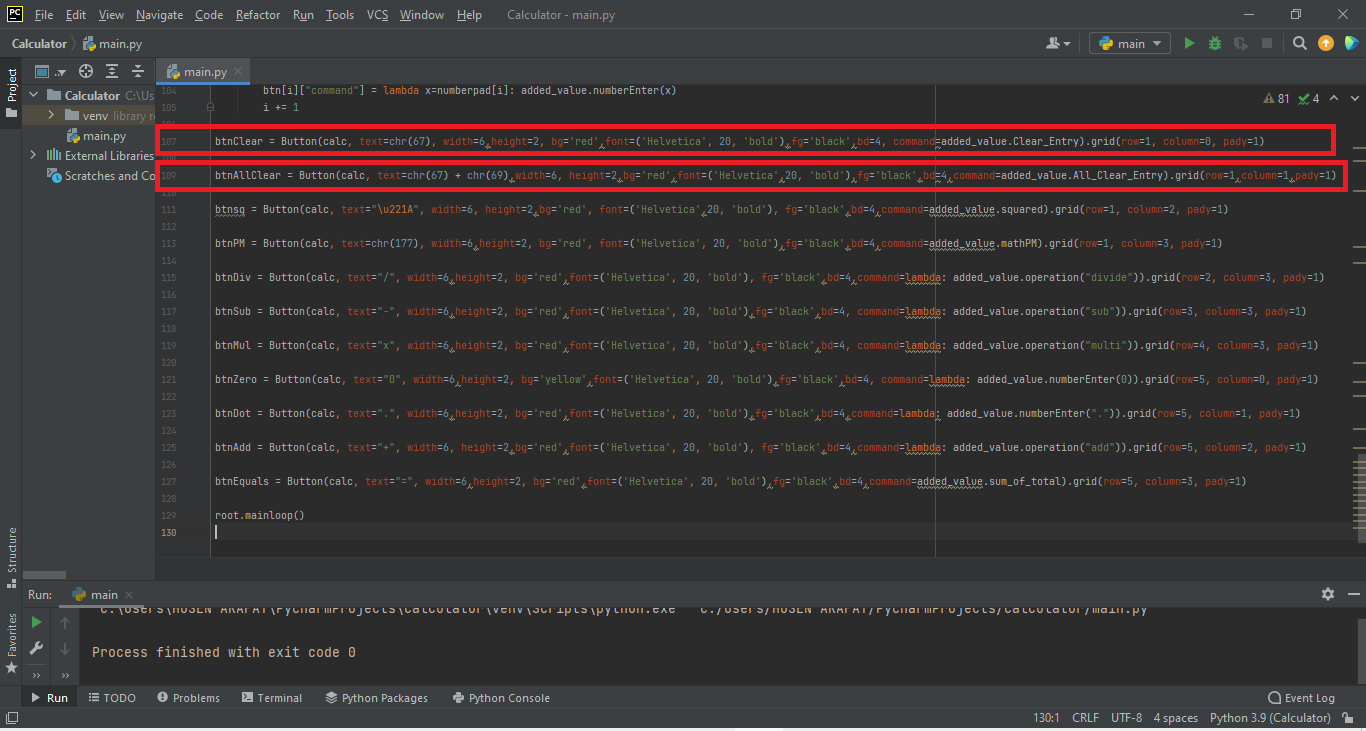


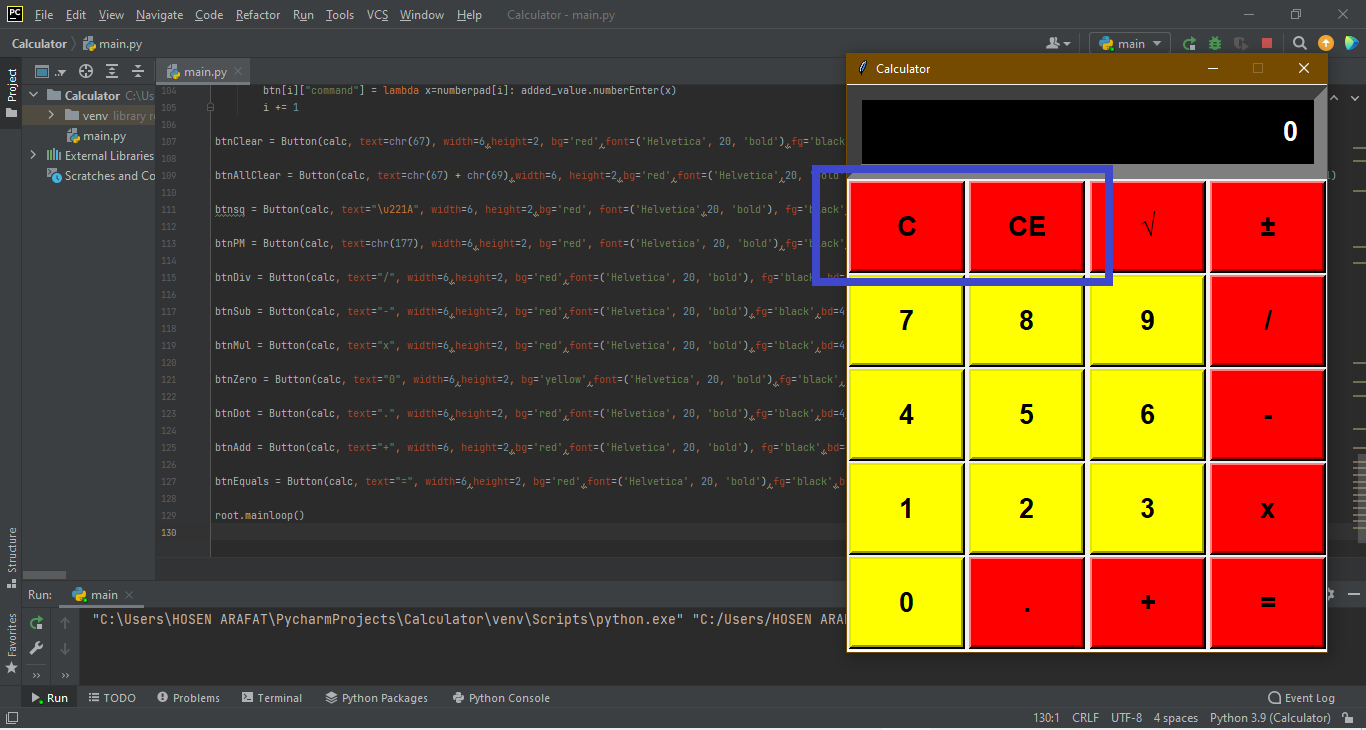


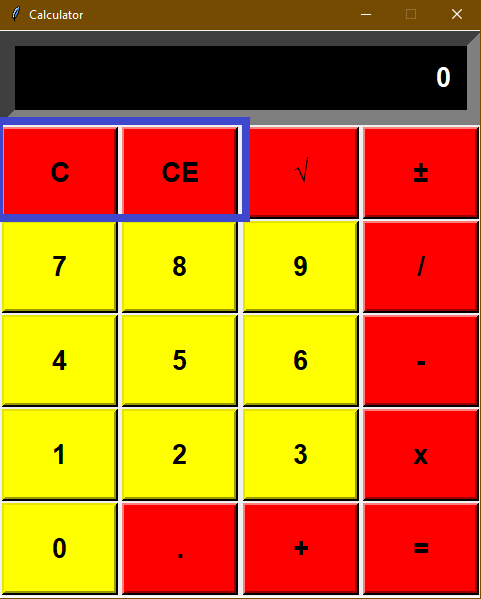


3.3. Write the Clear Entry and All Clear Entry function.

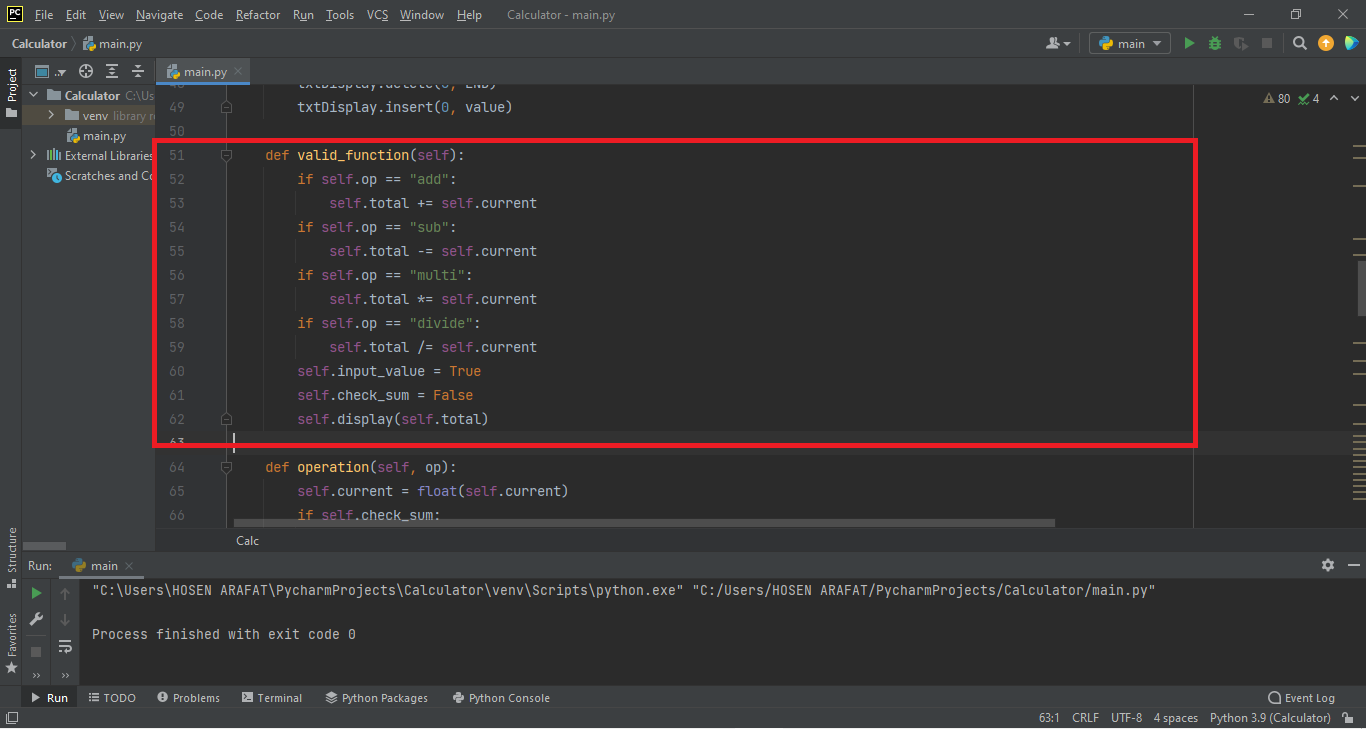


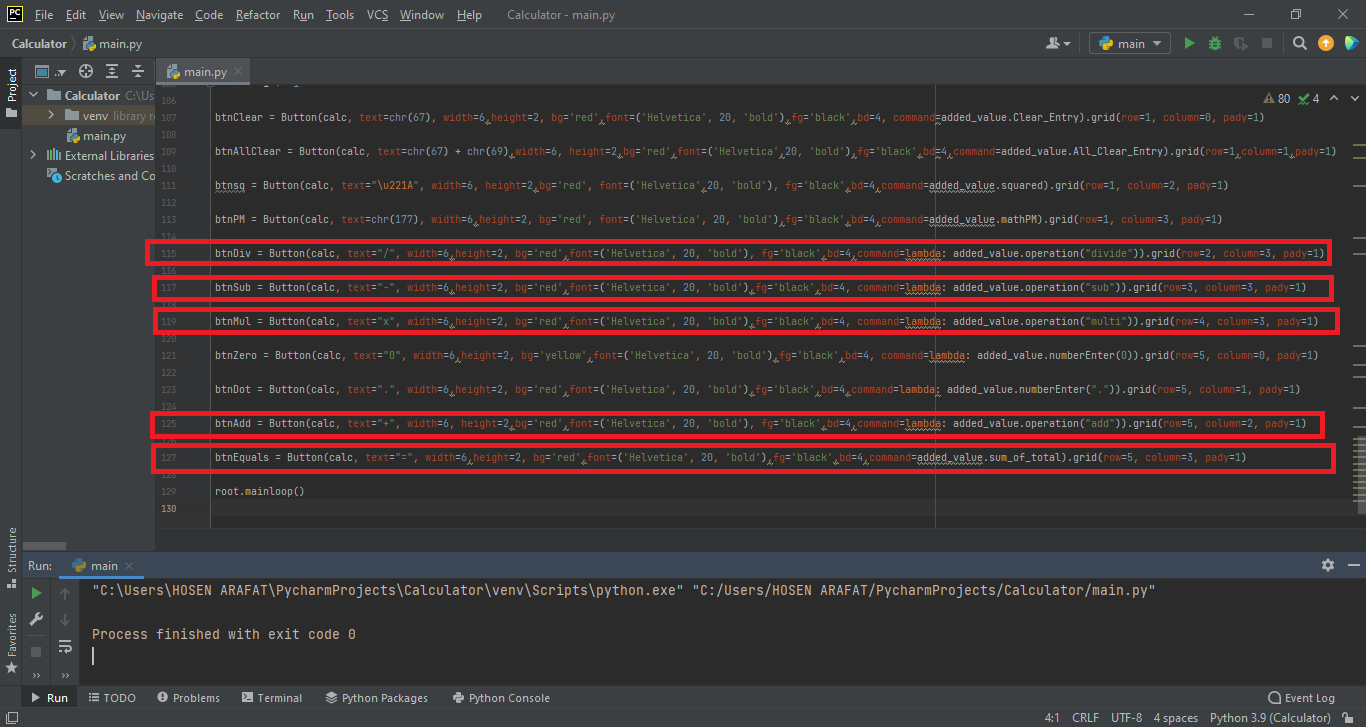


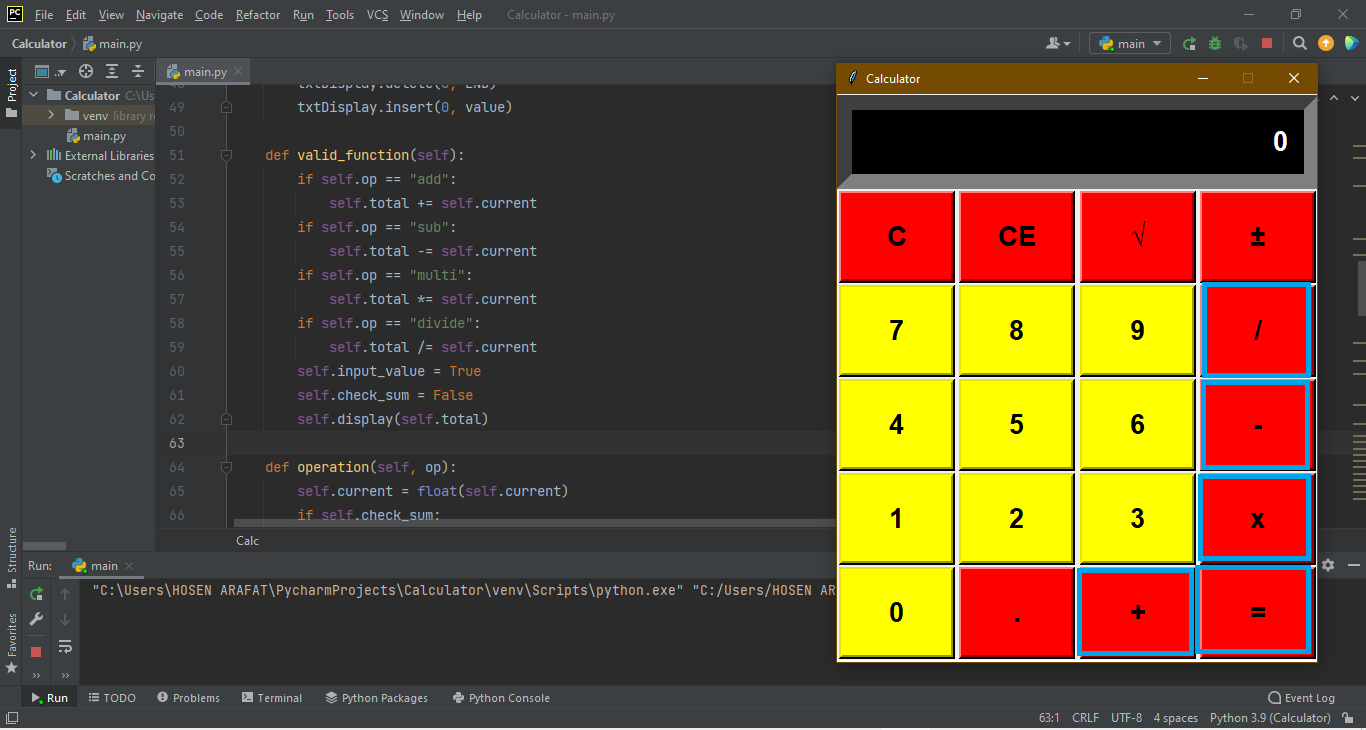


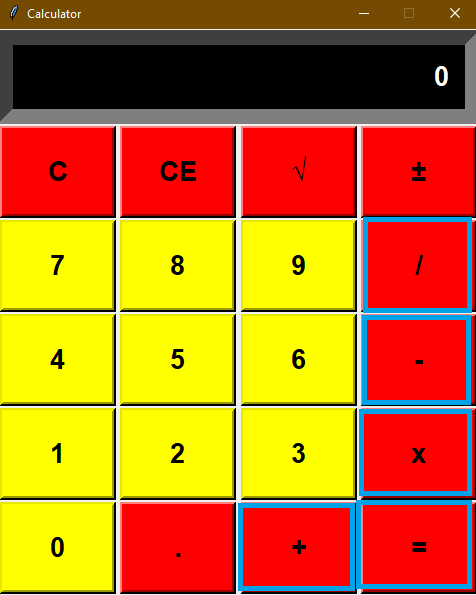


3.4 Compiling calculation function add, subtract, multiply and divide, and equal function, and write code, click the button to call the corresponding function

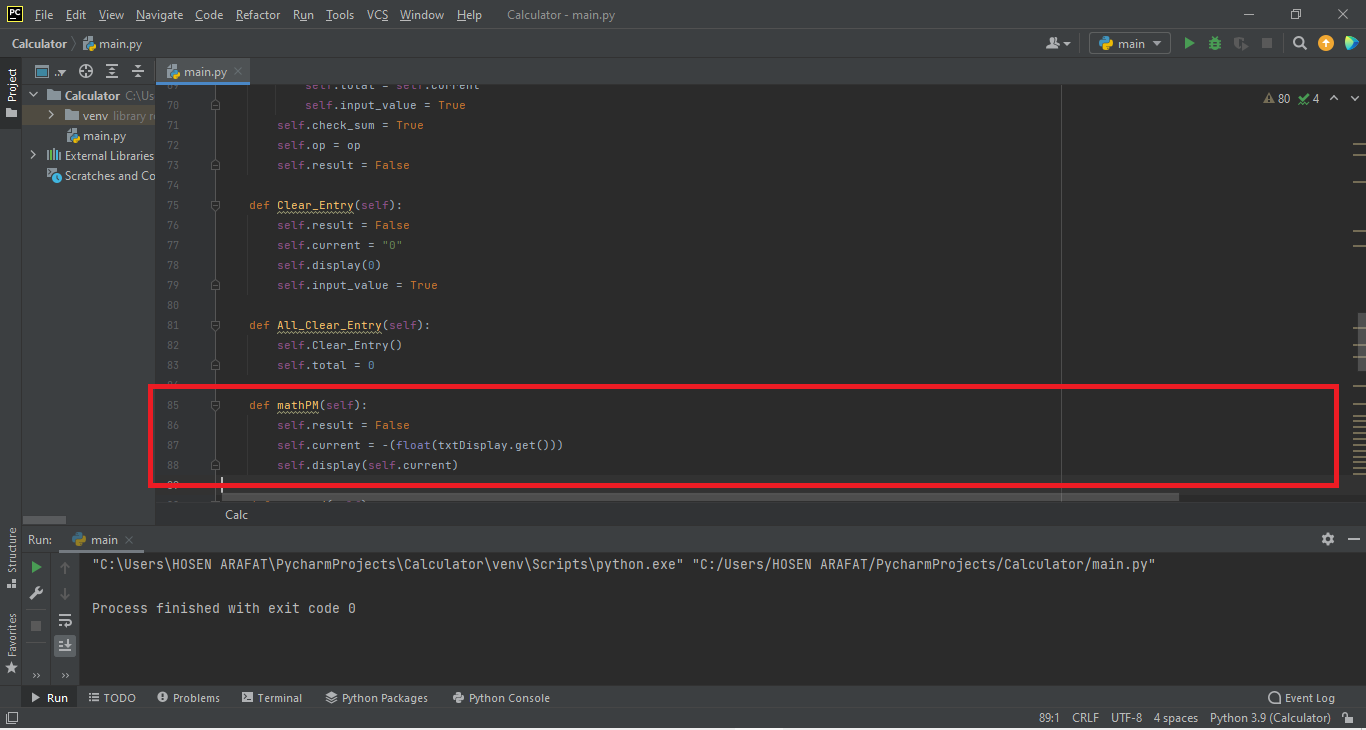


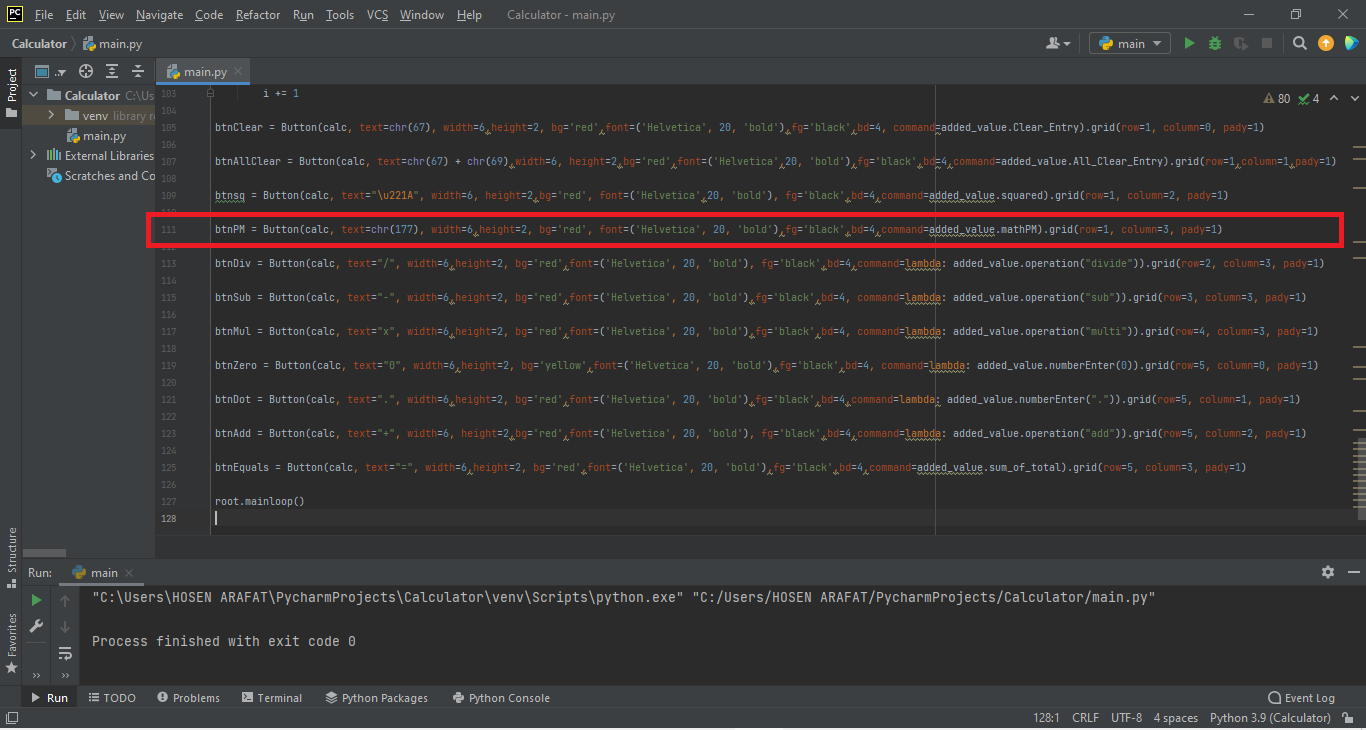


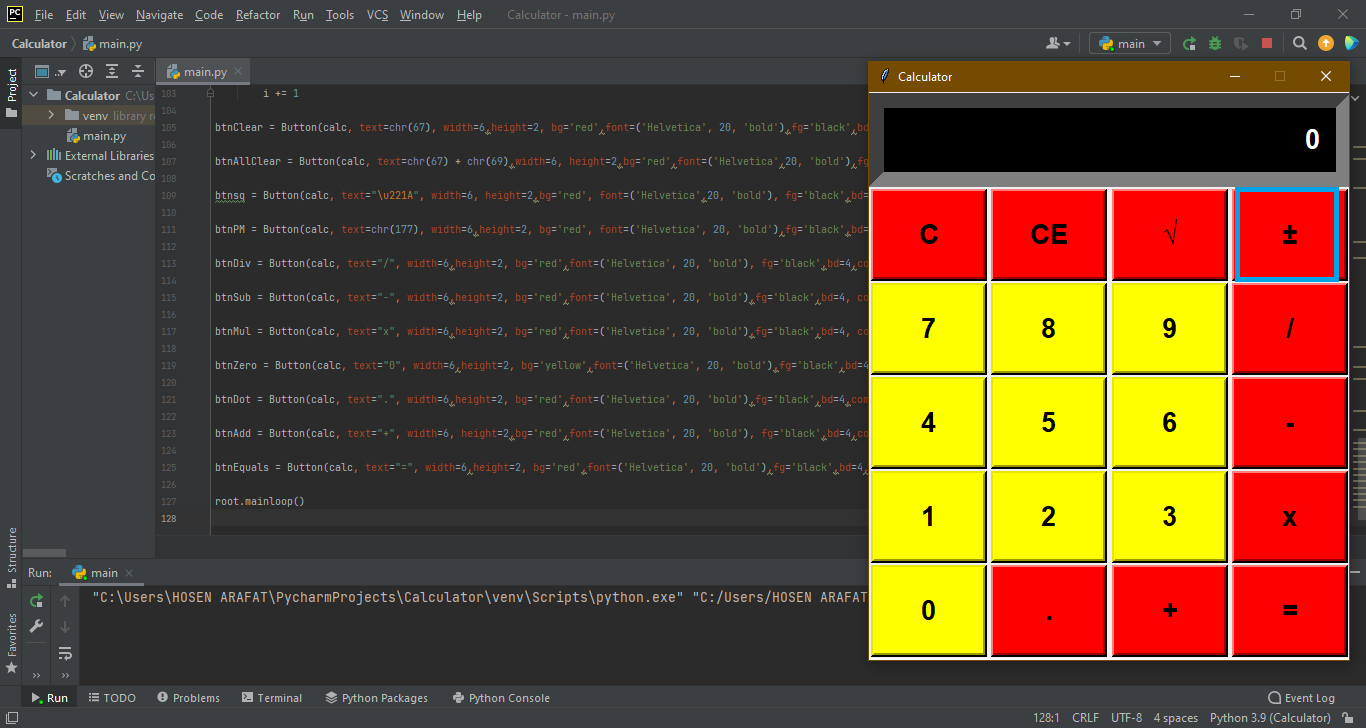


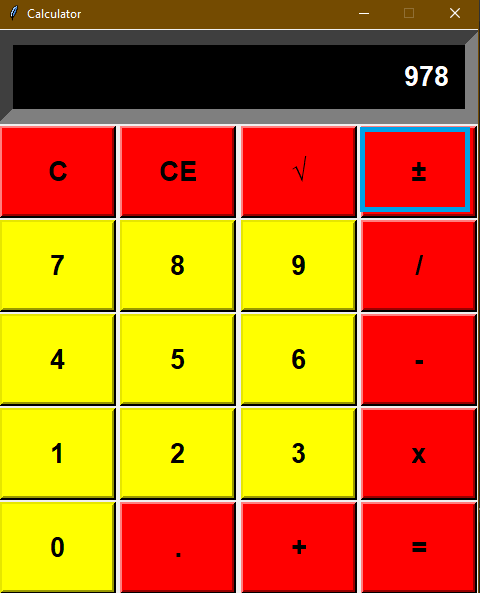


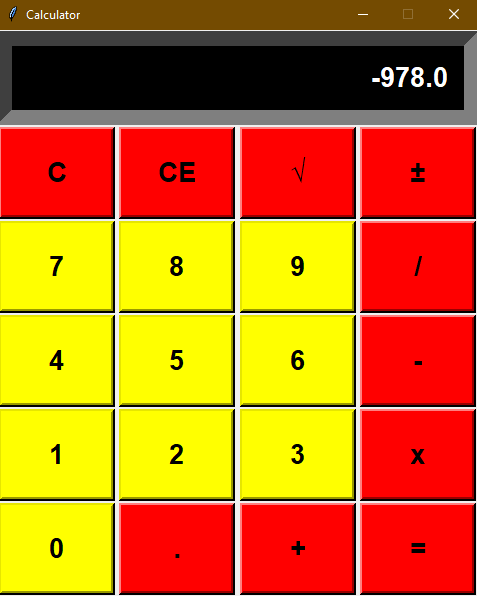
3.5. Write the mathPM function.



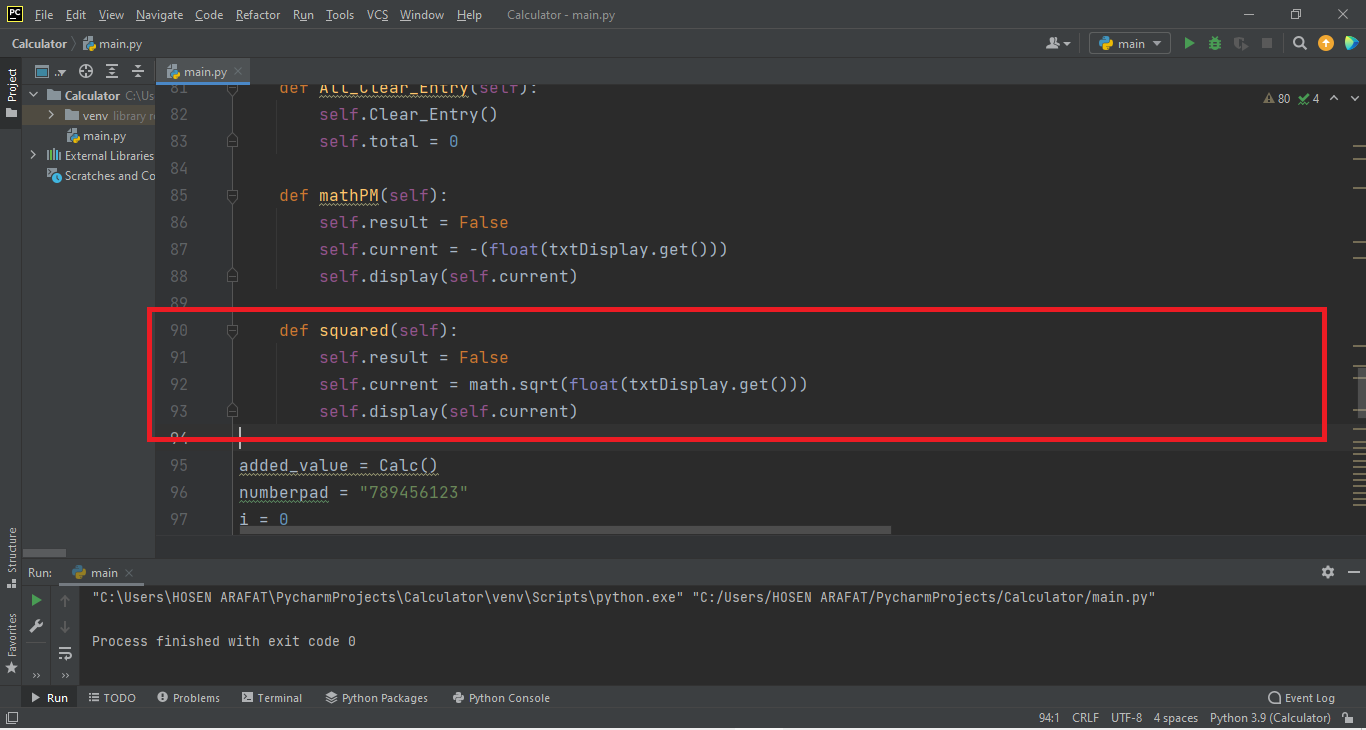


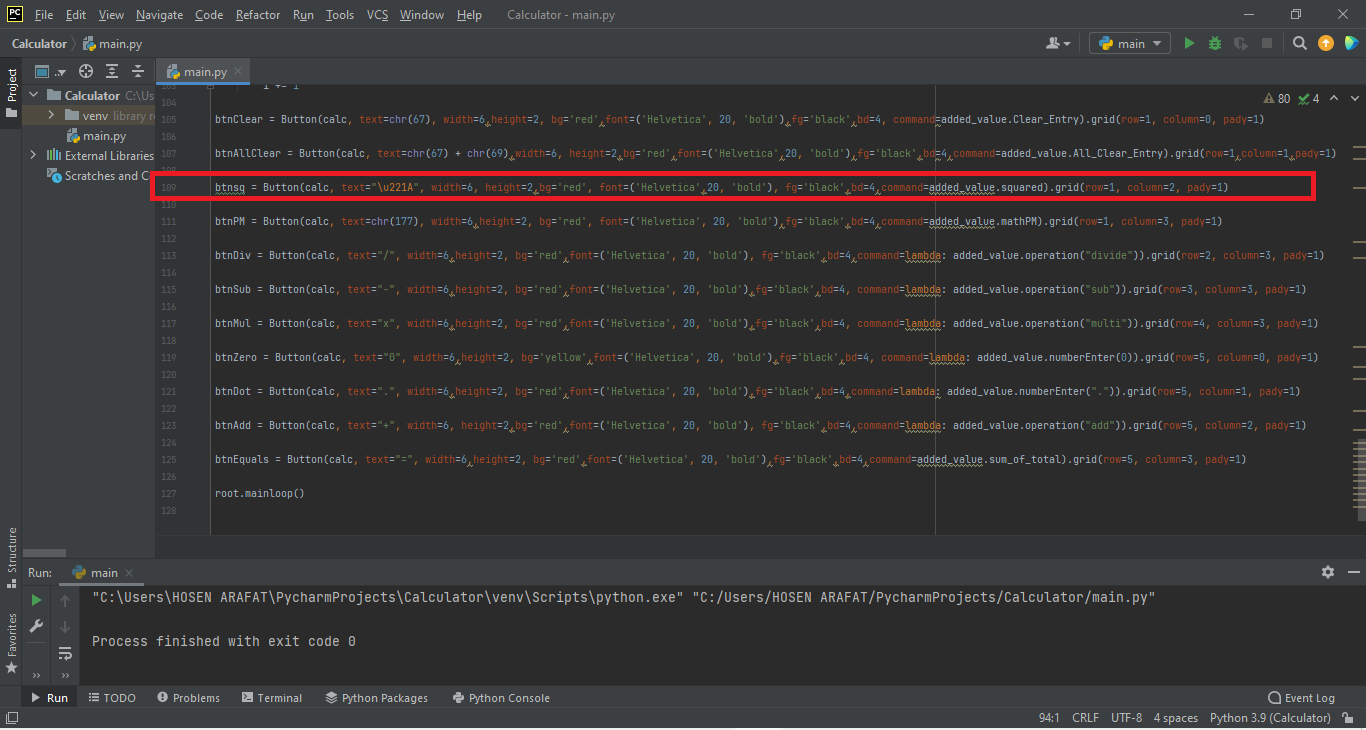


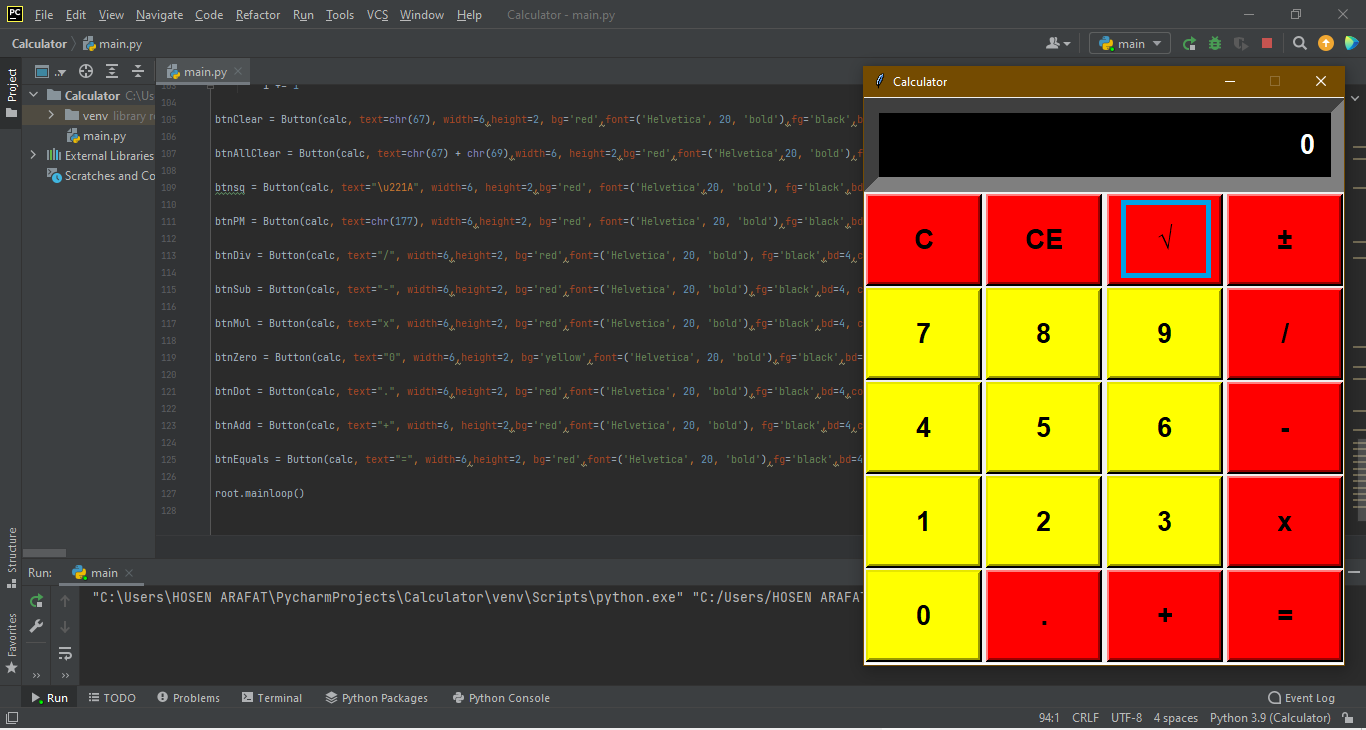


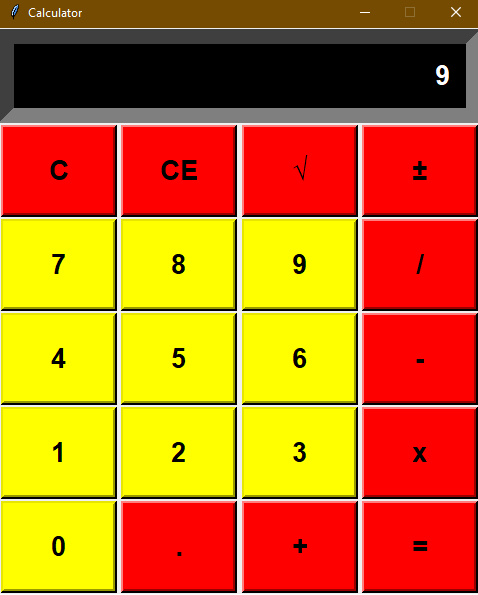


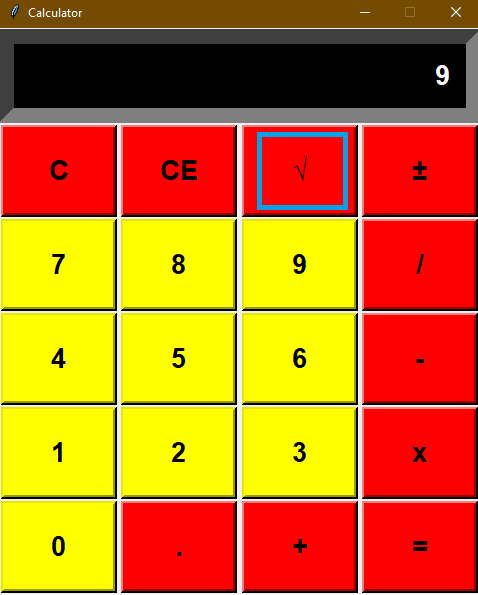
3.6. Write the Squared function.

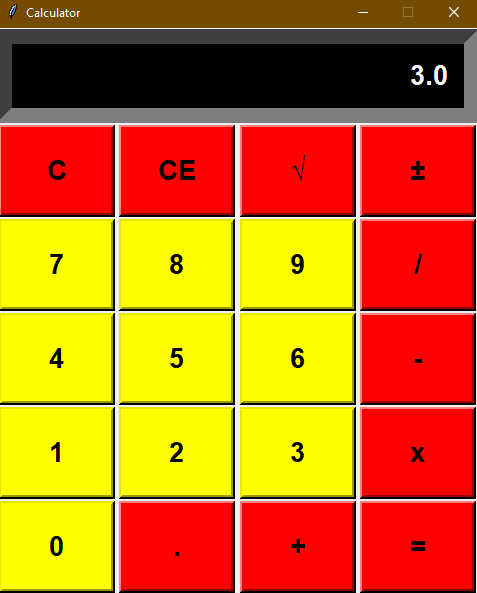






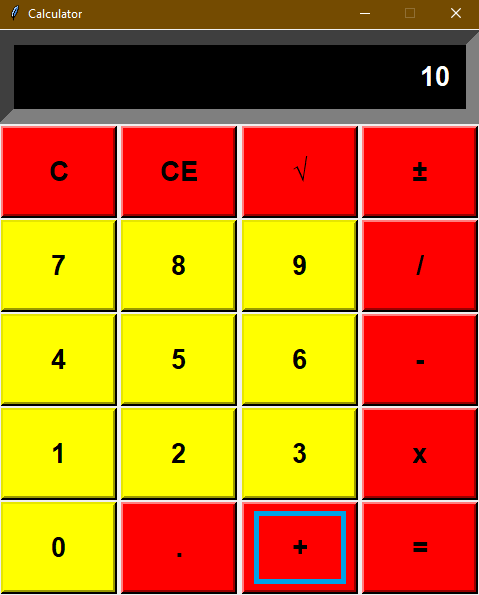


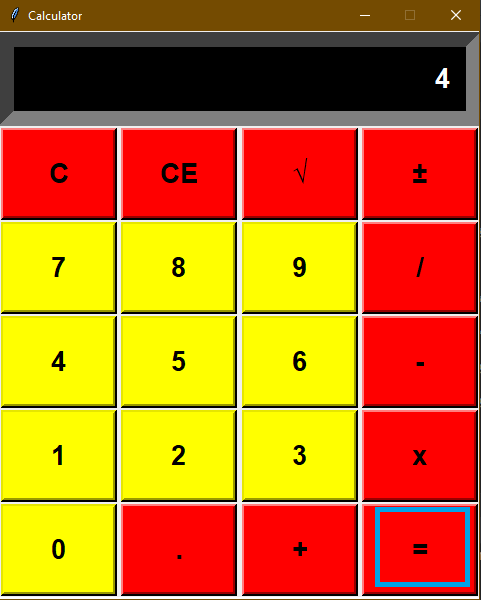


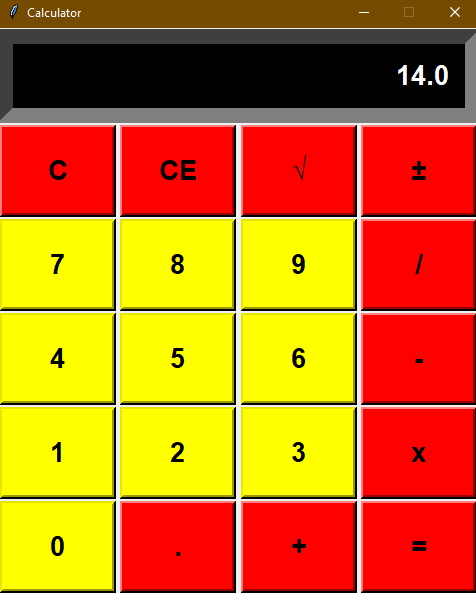


3.7 Run the program for calculator related operations

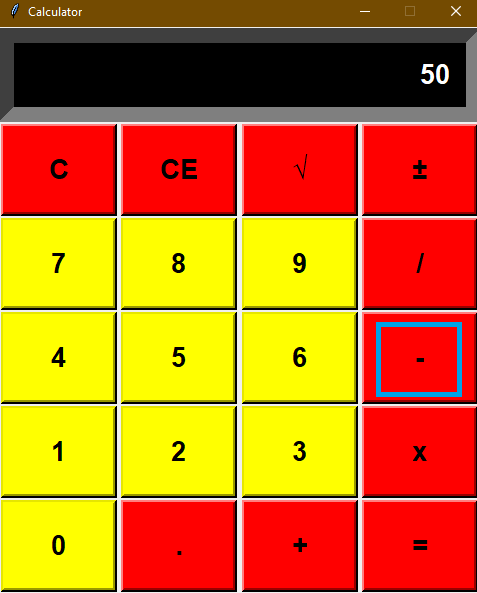
1) “add” 10+4=14

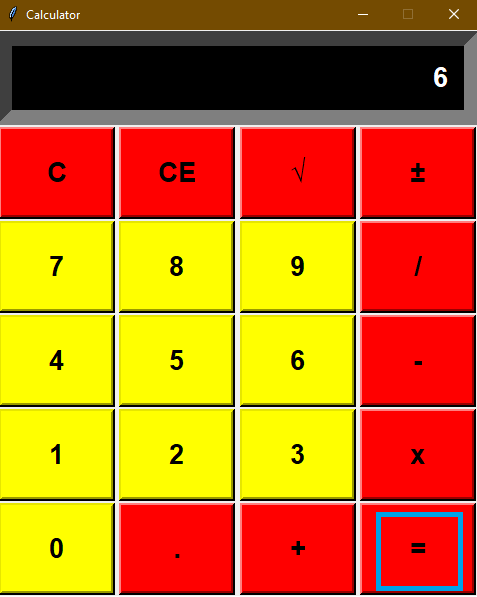




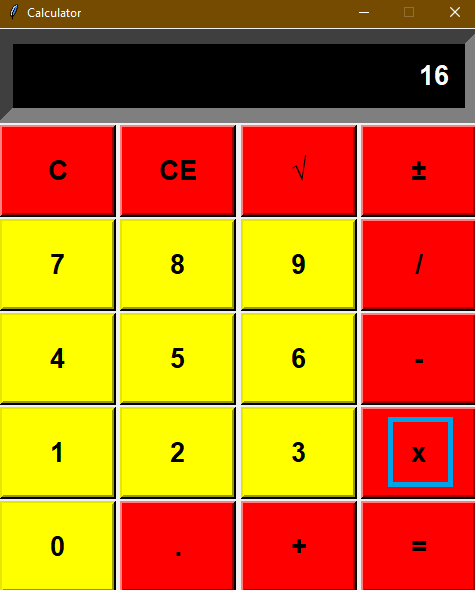


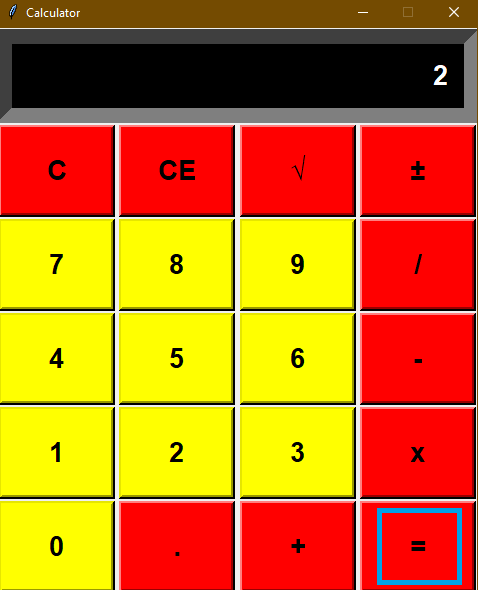
2) “sub” 50 – 6 = 44

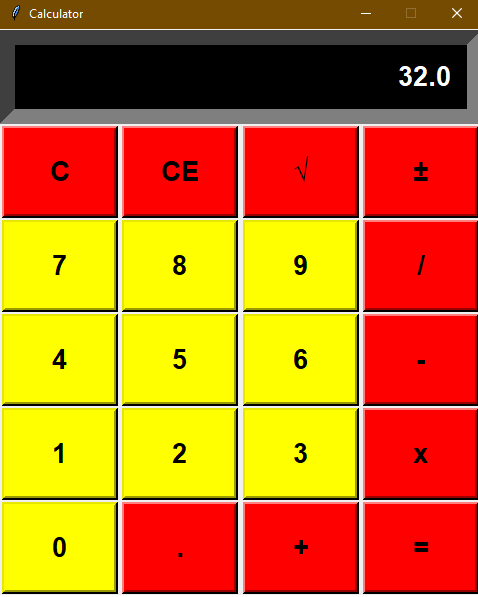




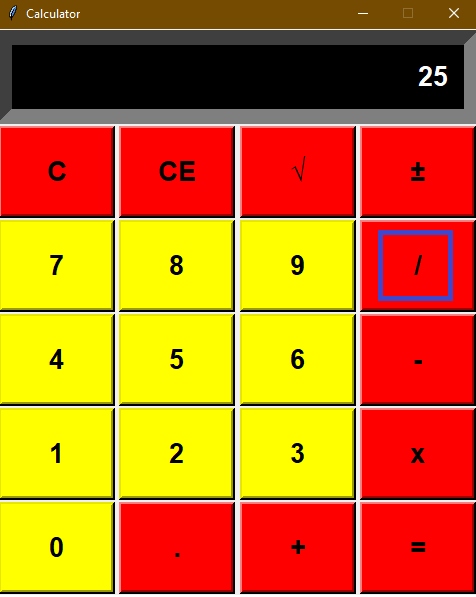
3) “multi” 16 \* 2 = 32

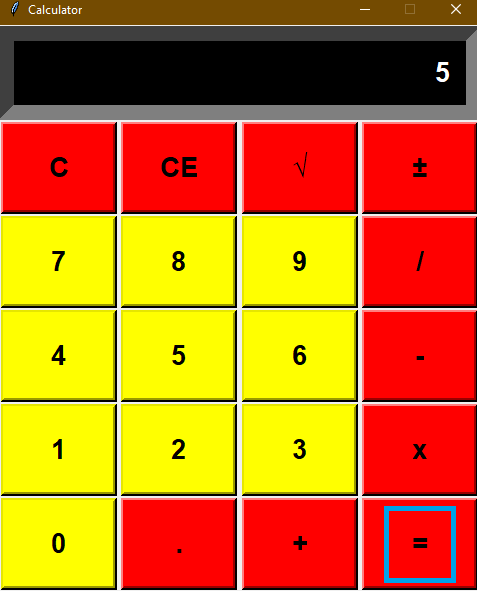


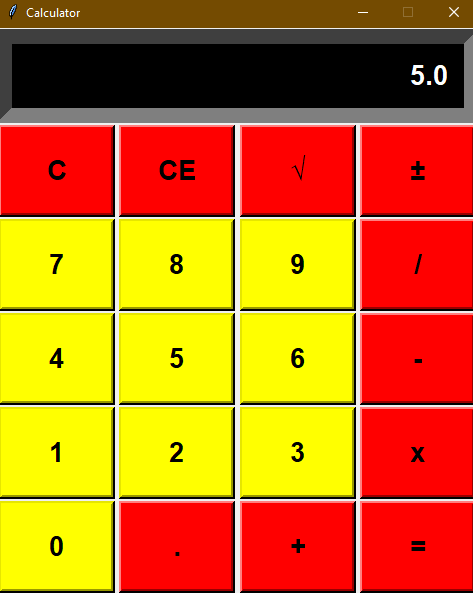




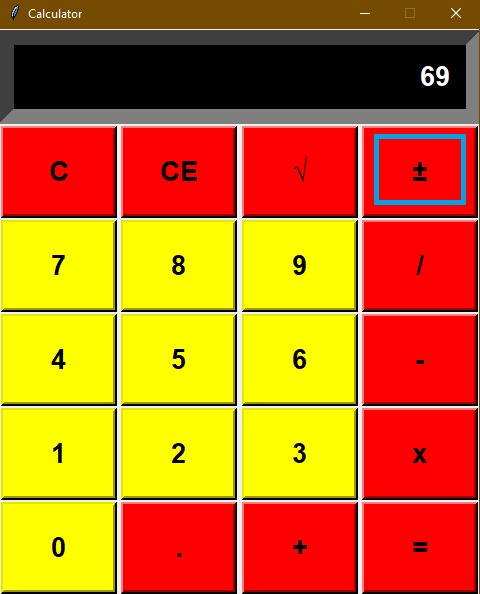
4) “divide” 25 / 5 = 5

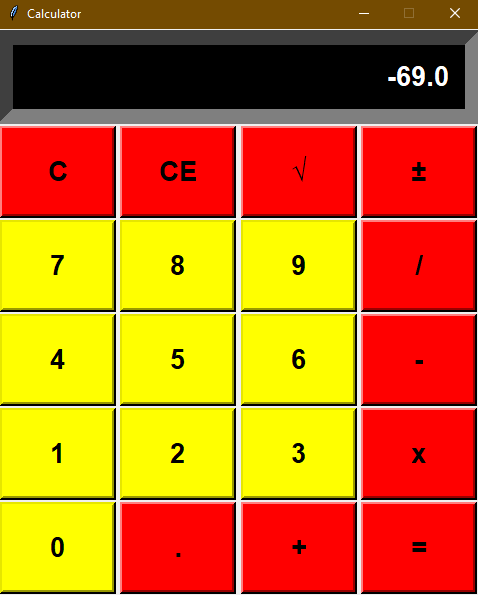




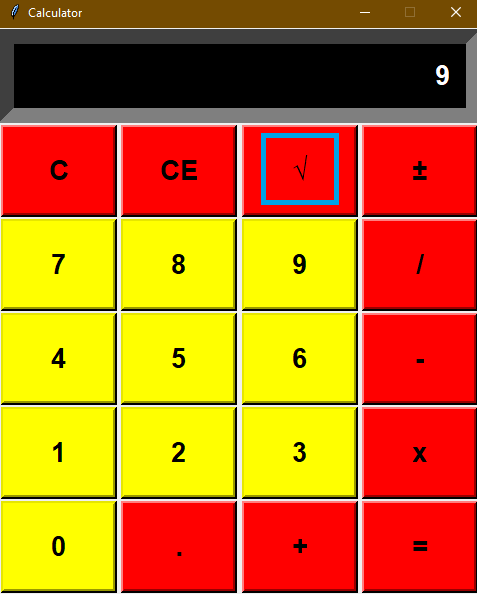


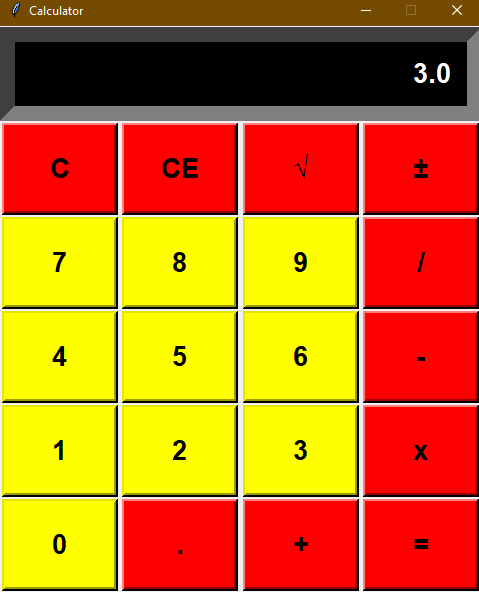
5) “mathPM” 69 ±





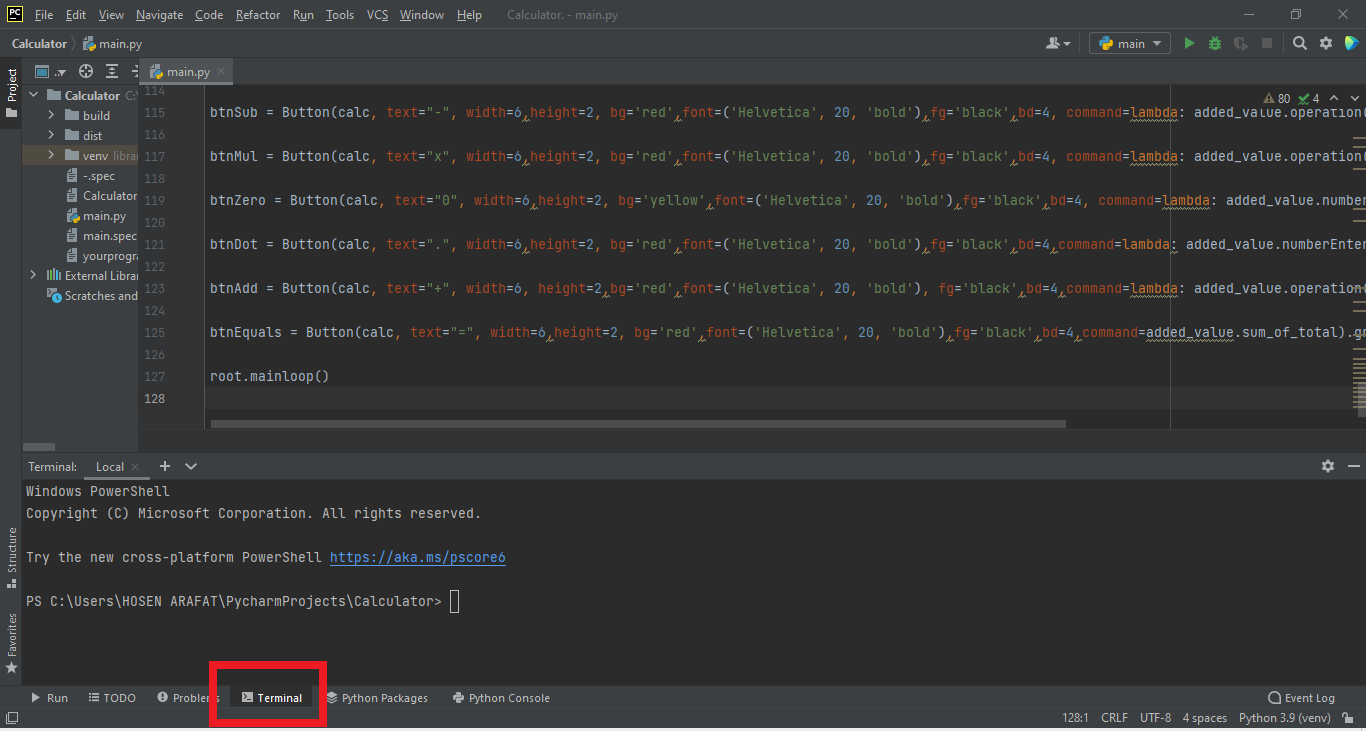
6) “Squared” 9√



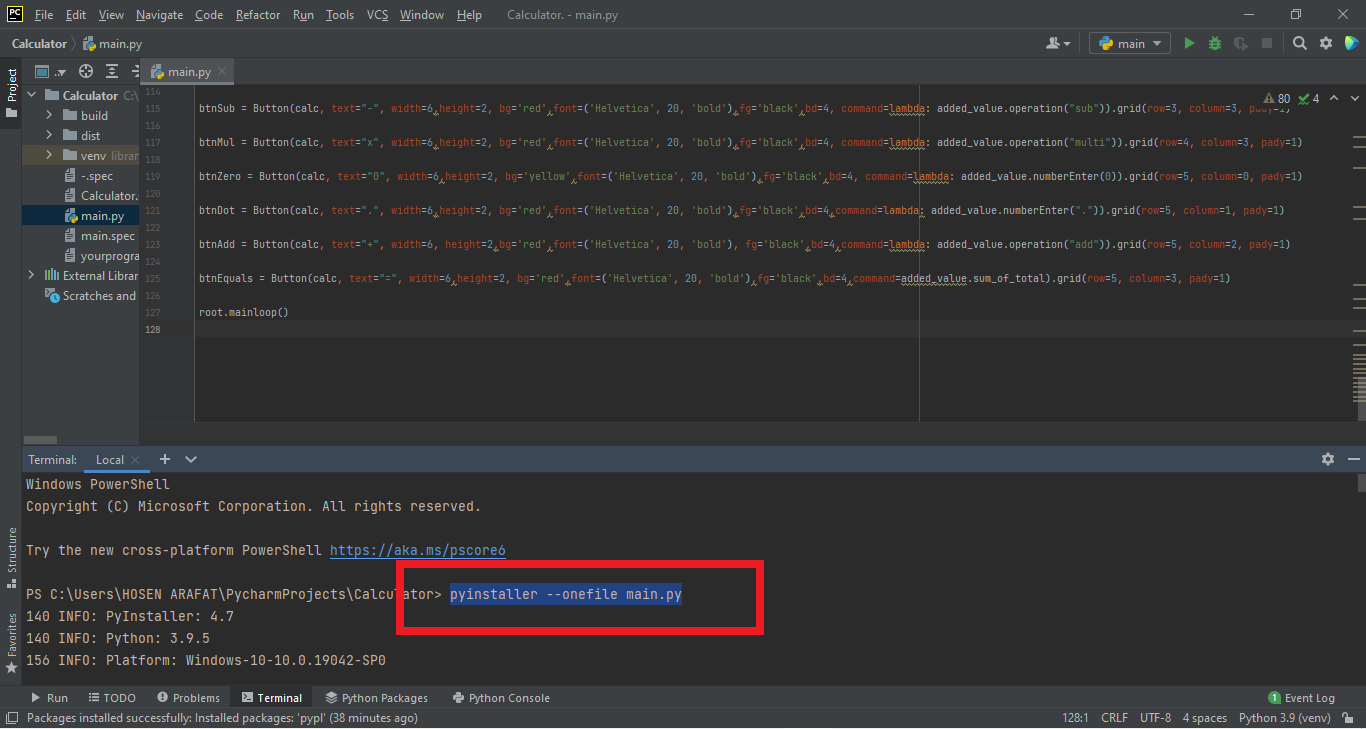


# **III. Packaged applications**

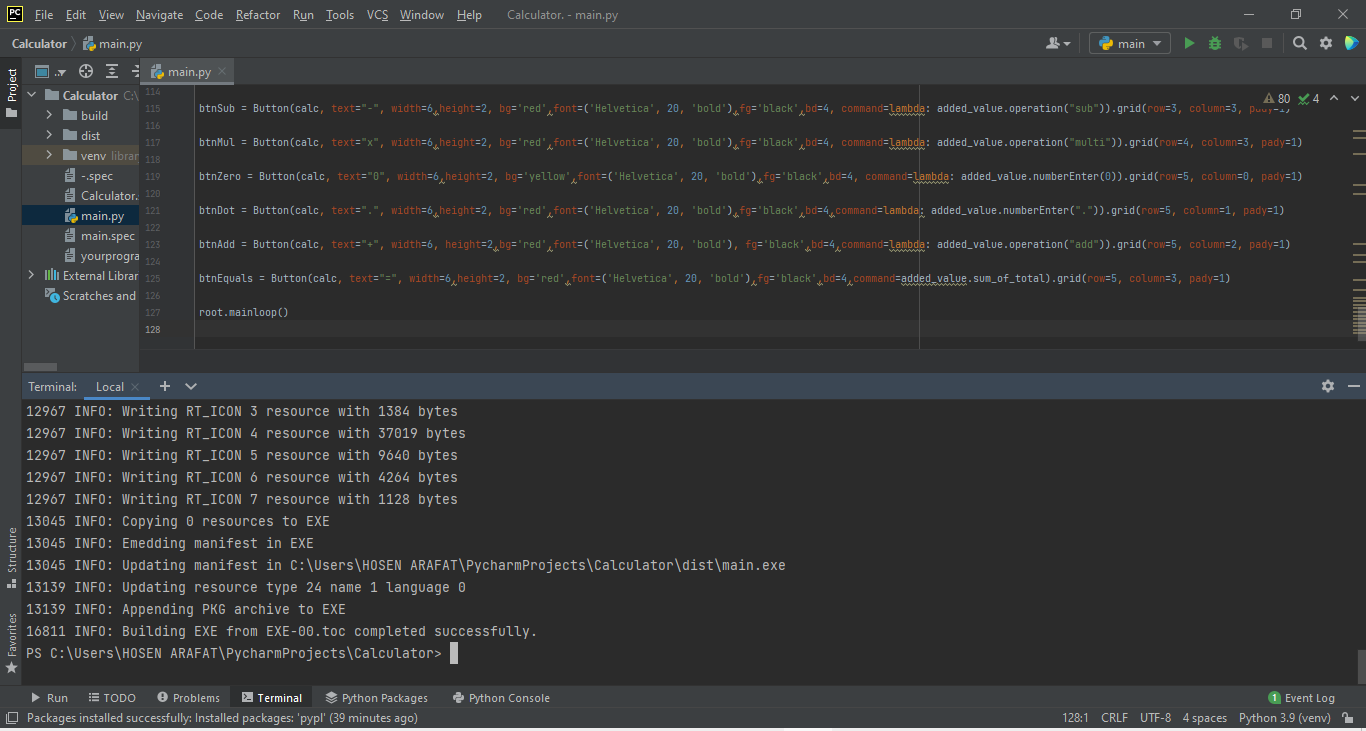
1. Click terminal to open the terminal



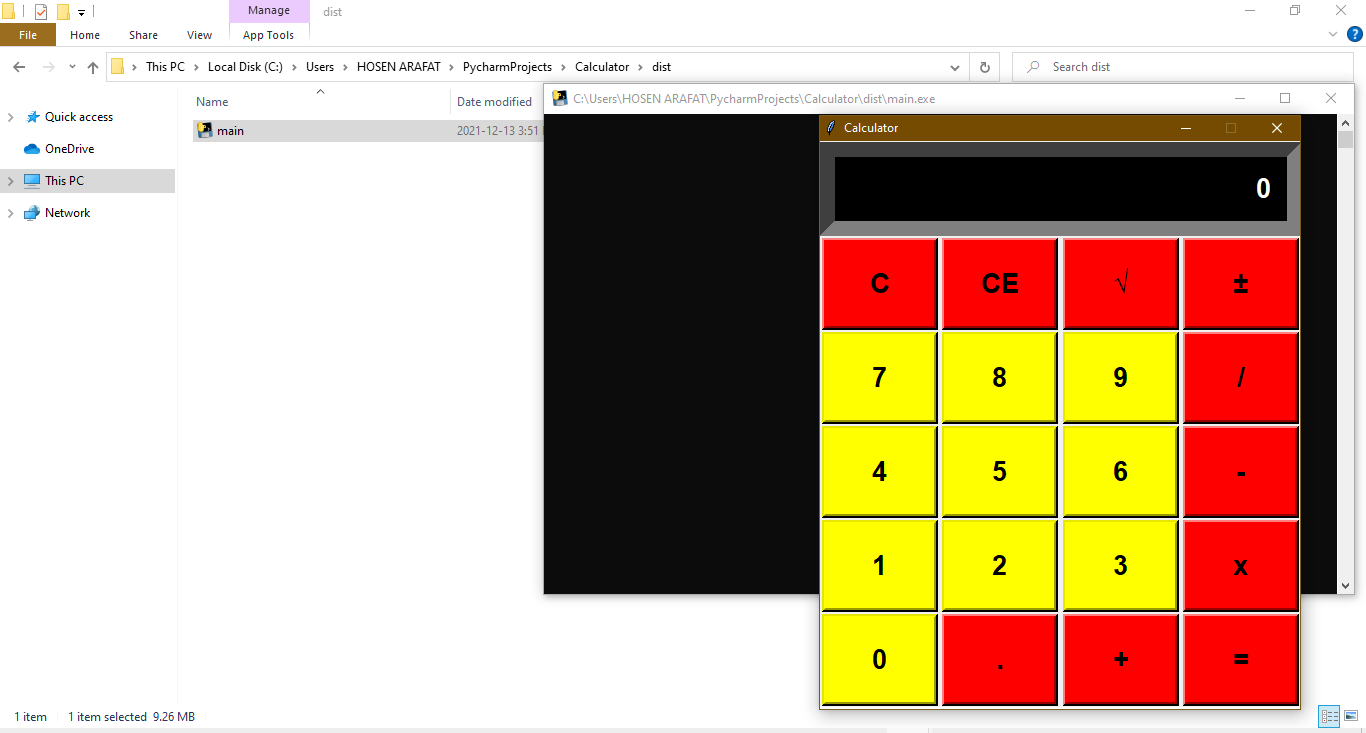
2. Enter the code "pyinstaller –onefile main.py " package "



### 3. Package succeeded



### 4.Open the main.exe file in the dist folder



**IV.Coedes**

from tkinter import \*

import math

root = Tk()

root.title("Calculator")

root.configure(background='white')

root.resizable(width=False, height=False)

root.geometry("480x568+450+90")

calc = Frame(root)

calc.grid()

txtDisplay = Entry(calc, font=('Helvetica', 20, 'bold'),bg='black', fg='white',bd=30, width=28, justify=RIGHT)

txtDisplay.grid(row=0, column=0, columnspan=4, pady=1)

txtDisplay.insert(0, "0")

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 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)

added\_value = Calc()

numberpad = "789456123"

i = 0

btn = []

for j in range(2, 5):

for k in range(3):

btn.append(Button(calc,width=6, height=2,bg='yellow', fg='black',font=('Helvetica', 20, 'bold'),bd=4,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

btnClear = Button(calc, text=chr(67), width=6,height=2, bg='red',font=('Helvetica', 20, 'bold'),fg='black',bd=4, command=added\_value.Clear\_Entry).grid(row=1, column=0, pady=1)

btnAllClear = Button(calc, text=chr(67) + chr(69),width=6, height=2,bg='red',font=('Helvetica',20, 'bold'),fg='black',bd=4,command=added\_value.All\_Clear\_Entry).grid(row=1,column=1,pady=1)

btnsq = Button(calc, text="\u221A", width=6, height=2,bg='red', font=('Helvetica',20, 'bold'), fg='black',bd=4,command=added\_value.squared).grid(row=1, column=2, pady=1)

btnPM = Button(calc, text=chr(177), width=6,height=2, bg='red', font=('Helvetica', 20, 'bold'),fg='black',bd=4,command=added\_value.mathPM).grid(row=1, column=3, pady=1)

btnDiv = Button(calc, text="/", width=6,height=2, bg='red',font=('Helvetica', 20, 'bold'), fg='black',bd=4,command=lambda: added\_value.operation("divide")).grid(row=2, column=3, pady=1)

btnSub = Button(calc, text="-", width=6,height=2, bg='red',font=('Helvetica', 20, 'bold'),fg='black',bd=4, command=lambda: added\_value.operation("sub")).grid(row=3, column=3, pady=1)

btnMul = Button(calc, text="x", width=6,height=2, bg='red',font=('Helvetica', 20, 'bold'),fg='black',bd=4, command=lambda: added\_value.operation("multi")).grid(row=4, column=3, pady=1)

btnZero = Button(calc, text="0", width=6,height=2, bg='yellow',font=('Helvetica', 20, 'bold'),fg='black',bd=4, command=lambda: added\_value.numberEnter(0)).grid(row=5, column=0, pady=1)

btnDot = Button(calc, text=".", width=6,height=2, bg='red',font=('Helvetica', 20, 'bold'),fg='black',bd=4,command=lambda: added\_value.numberEnter(".")).grid(row=5, column=1, pady=1)

btnAdd = Button(calc, text="+", width=6, height=2,bg='red',font=('Helvetica', 20, 'bold'), fg='black',bd=4,command=lambda: added\_value.operation("add")).grid(row=5, column=2, pady=1)

btnEquals = Button(calc, text="=", width=6,height=2, bg='red',font=('Helvetica', 20, 'bold'),fg='black',bd=4,command=added\_value.sum\_of\_total).grid(row=5, column=3, pady=1)

root.mainloop()