



# Python GUI 2022

by



Uncle Engineer  
ลุงวิศวกร สอนคำนวณ



# Python GUI

## 2022(EP.1)

by



Uncle Engineer

ลุงวิศวกร สอนคำนวณ

## GUI คืออะไร?

คือซอฟแวร์ที่มีเมนู มีปุ่มกด มีช่องกรอกข้อมูล มีdroptopดาว มีปุ่มให้เลือก มีเซ็คบีอกซ์ กราฟ และอื่นๆ หรือในภาษาทางด้านคอมพิวเตอร์ คือ "ส่วนติดต่อผู้ใช้งานแบบกราฟิก" (Graphical User Interface) ข้อดีของการเขียนด้วย Python คือ เขียนครั้งเดียว สามารถนำไปรันได้ทั้งใน Windows, Mac, Linux

# ติดตั้ง Python ดาวน์โหลด จากเว็บไซค์

<https://www.python.org/downloads/release/python-3101/> เลื่อนลงมาตรงส่วน File เลือก OS ที่ตัวเองใช้

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		91822157a97da16203877400c810d93e	25061625	SIG
XZ compressed source tarball	Source release		789210934745a65247a3ebf5da9adb64	18775460	SIG
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	d106e0f83028e9245397867148dc2323	39790346	SIG
Windows embeddable package (32-bit)	Windows		4f4d7a8f2118fc2e17a4ebdf5fe0aeef5	7527277	SIG
Windows embeddable package (64-bit)	Windows		864dcf204ba8b257809cc3f20beb1b9	8484449	SIG
Windows help file	Windows		993f5820d23ba72f9aee2e501f54980a	9573116	SIG
Windows installer (32-bit)	Windows		0b8c2b4677af4f47e534c7eee1c3cb03	27047408	SIG
Windows installer (64-bit)	Windows	Recommended	0e1c3a6ee3a05b5c4cd3d43fce8311a1	28179056	SIG

[About](#) [Downloads](#) [Documentation](#) [Community](#) [Success Stories](#) [News](#)

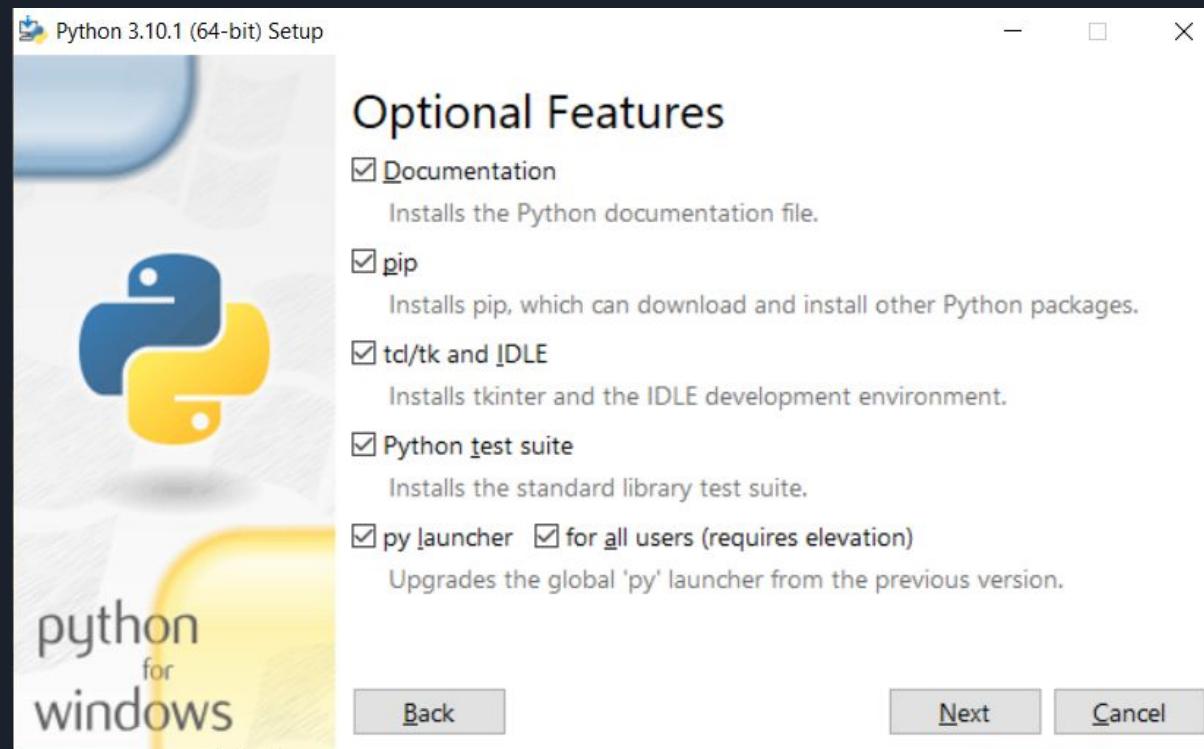
[Applications](#) [All releases](#) [Docs](#) [Community Survey](#) [Arts](#) [Python News](#)

[Django](#) [Source code](#) [Austin/Visual Talks](#) [Diversity](#) [Business](#) [PSF Newsletter](#)

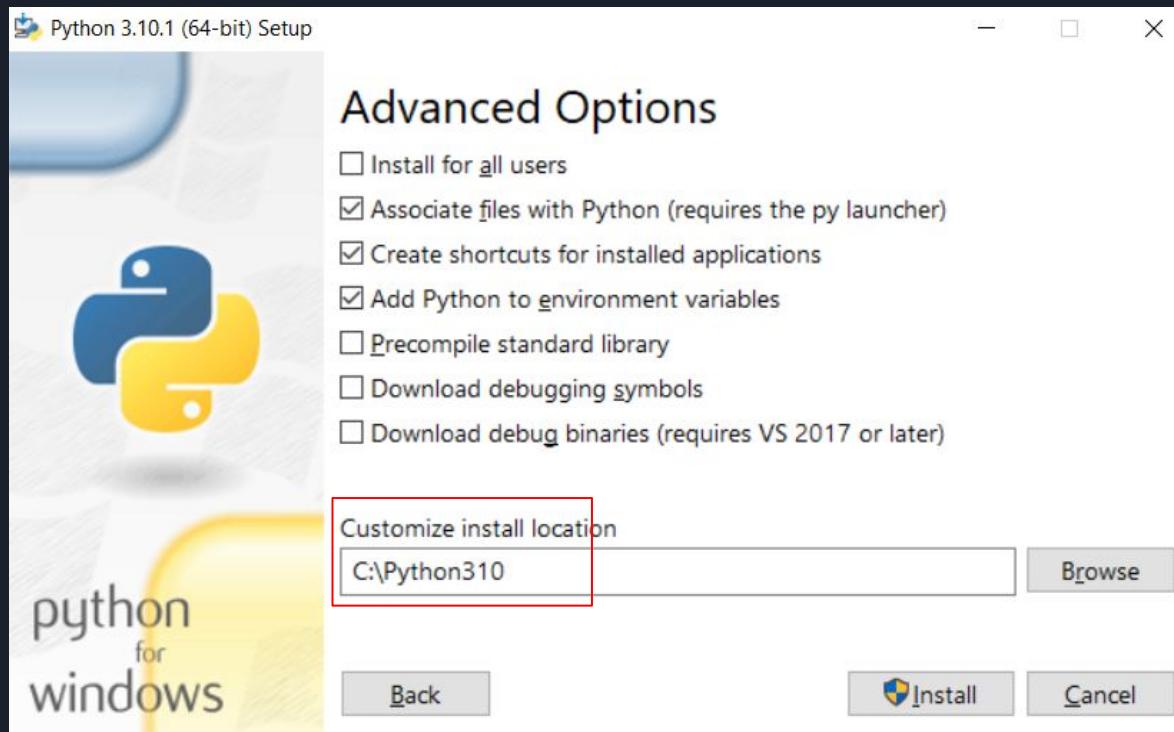
คลิกติดตั้งไฟล์ที่ได้มา ให้เลือก Add Python 3.10 to PATH เสร็จ  
แล้วเลือก Customize installation



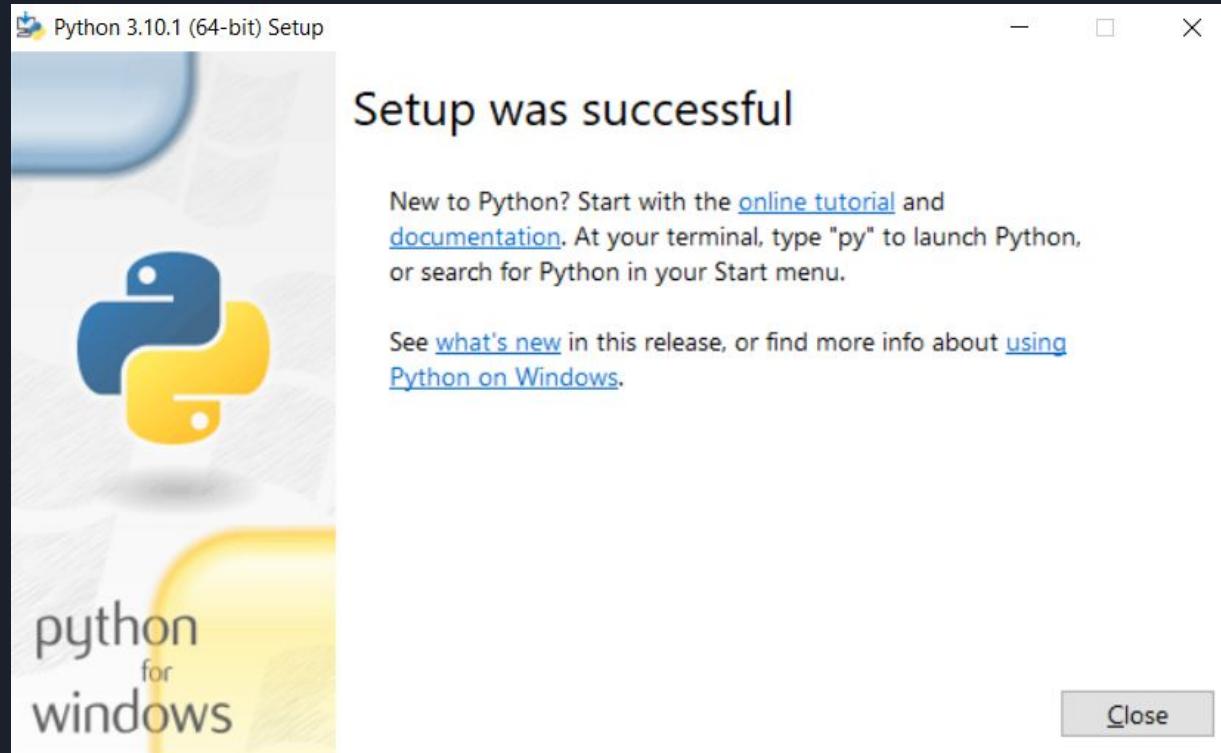
ກວດ Next



# เปลี่ยน Location ตามภาพแล้วกด Install เพื่อให้ง่ายต่อการ setting



# เสร็จแล้ว Close ได้เลย





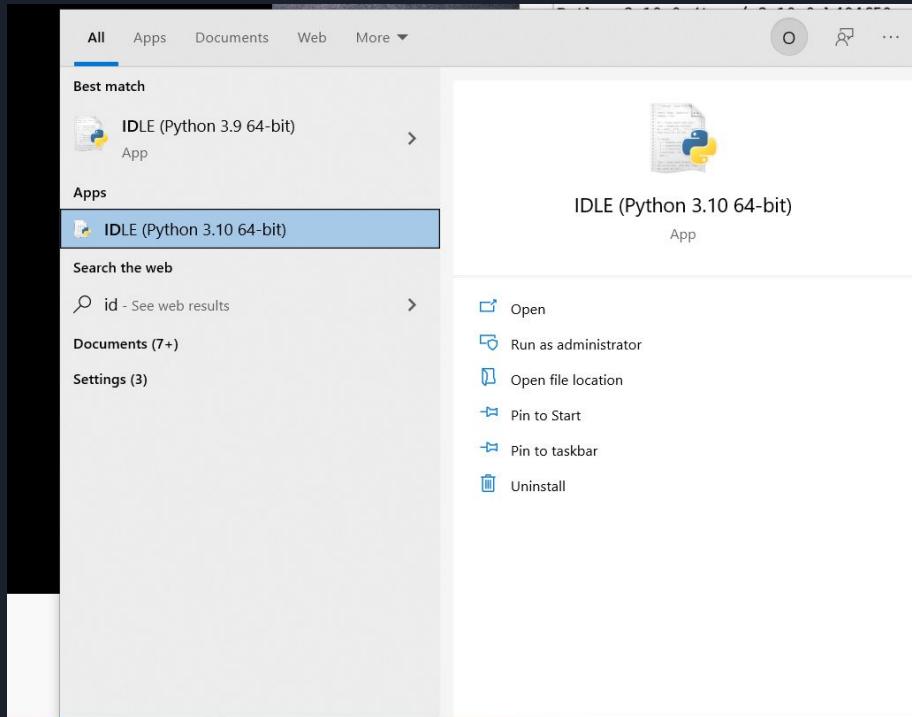
แนะนำ Python เวอร์ชันอื่นถ้าไม่มีปัญหาสำหรับเวอร์ชัน 3.10

windows7 - python 3.4.1

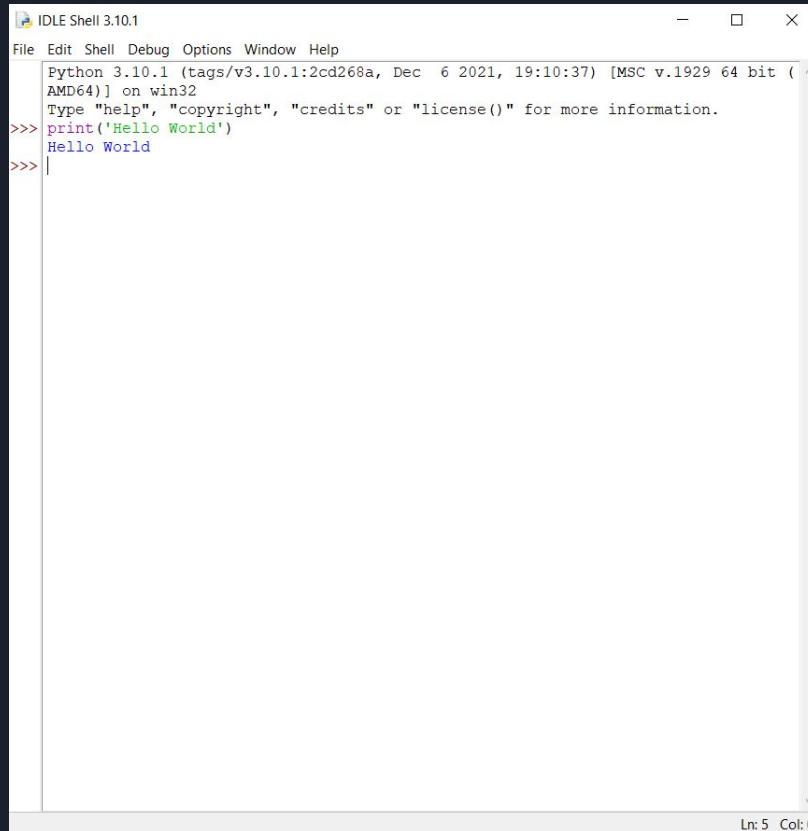
windows8 - python 3.6.3

windows10 - python 3.6-3.9

# เปิดโปรแกรม ไปที่ Start พิมพ์ IDLE



# ทดสอบ print Hello World

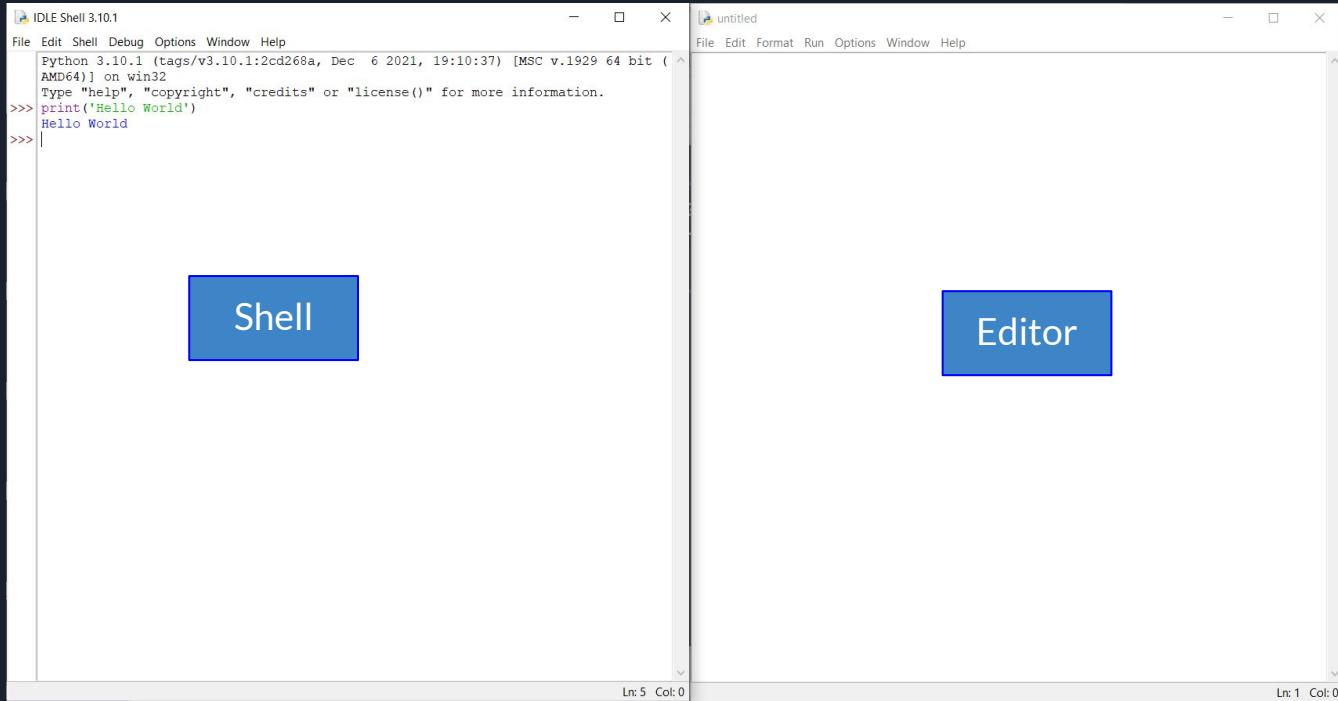


The screenshot shows a window titled "IDLE Shell 3.10.1". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The main area displays Python 3.10.1 code and its output. The code entered is:

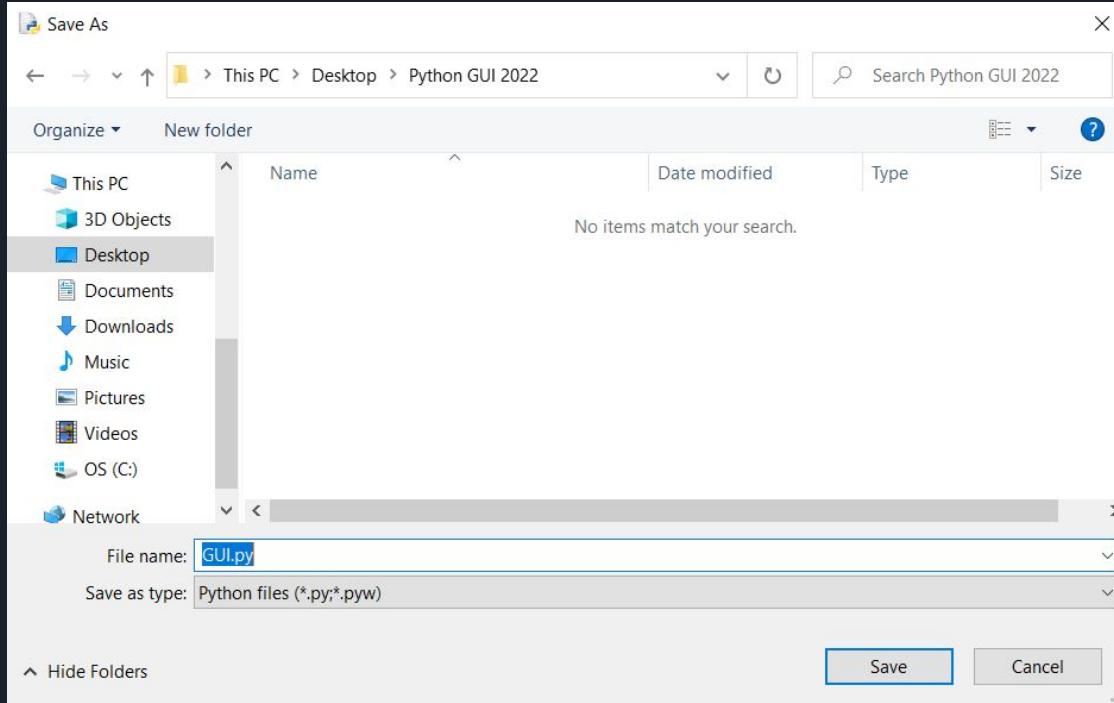
```
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print('Hello World')
Hello World
>>> |
```

The output "Hello World" is displayed in green. The status bar at the bottom right indicates "Ln: 5 Col: 0".

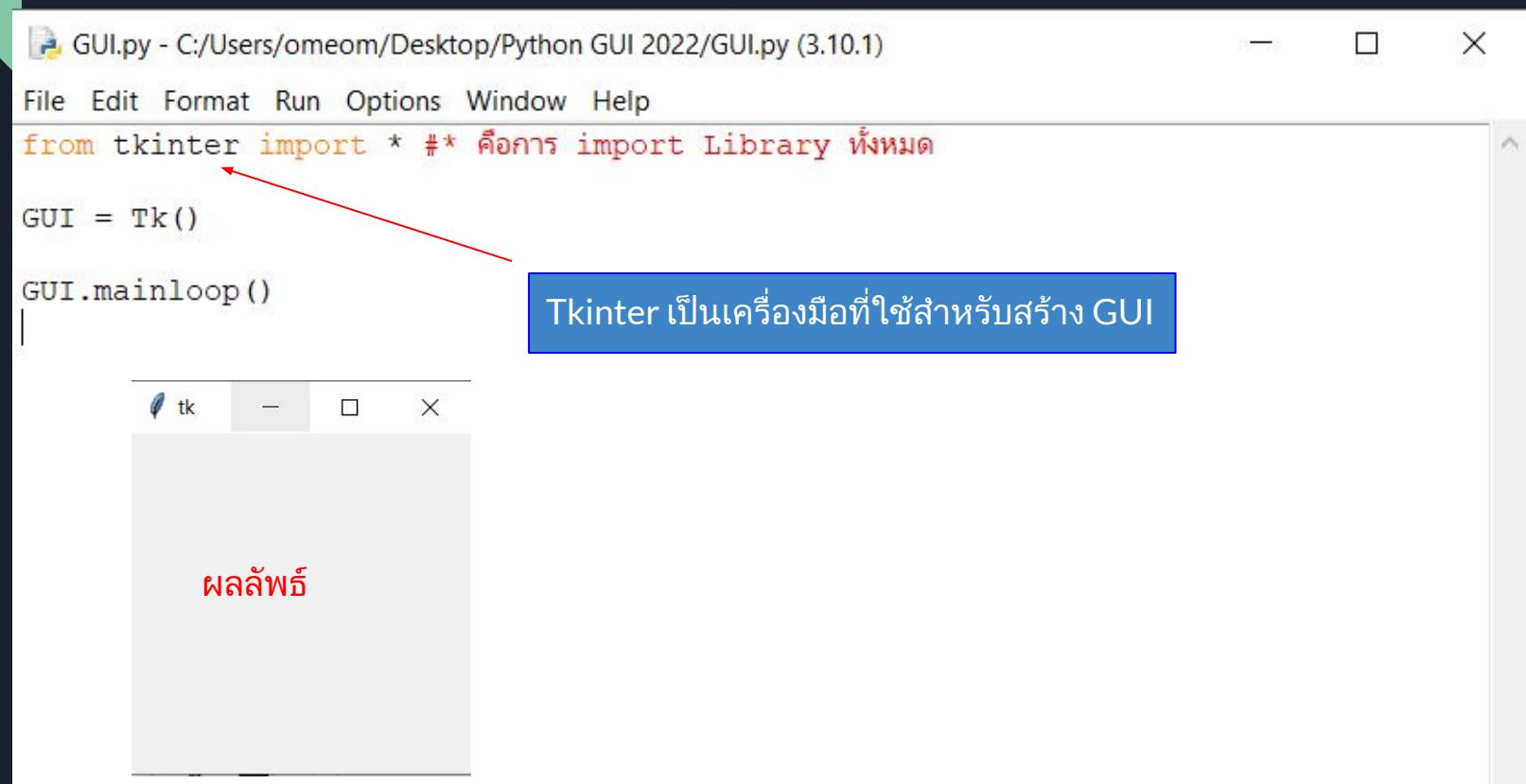
ความต่างระหว่าง Shell กับ Editor  
Shell จะไม่สามารถถกลับไปแก้ไขบรรทัดที่พิมพ์ไปแล้วได้ Editor  
จะสามารถแก้ไขได้(เปิด Editor โดยการไปที่ File > New File ใน  
หน้า Shell)



เปิด Editor เชฟไฟล์ สร้างโฟลเดอร์ใน Desktop ชื่อ Python GUI 2022 ตั้งชื่อไฟล์ GUI.py (ห้ามตั้งชื่อเหมือนกับชื่อแพ็คเกจ เช่น Tkinter)



# เริ่มเขียนโปรแกรม ลองทดสอบเรียก GUI ขึ้นมาโดยการพิมพ์



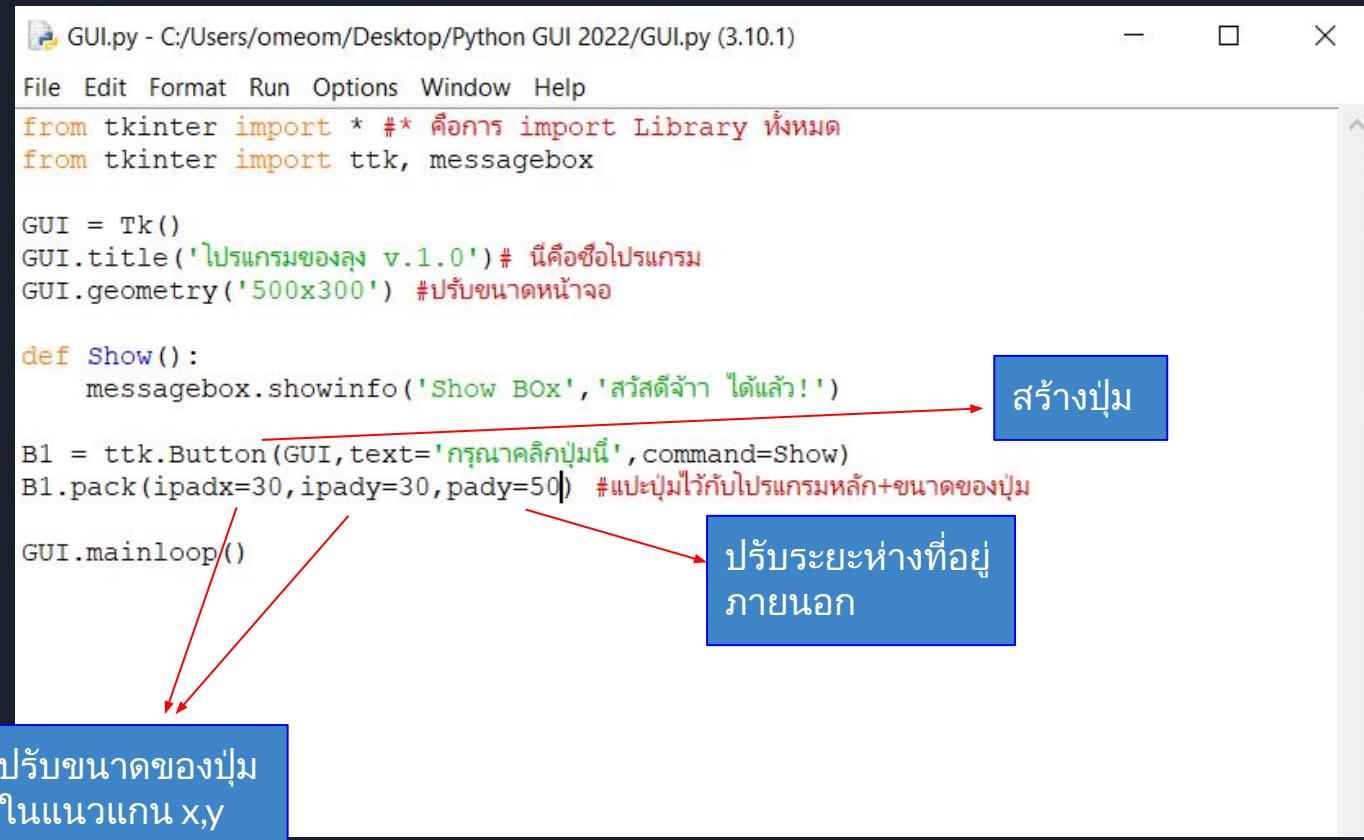
The screenshot shows a Windows-style code editor window titled "GUI.py - C:/Users/omeom/Desktop/Python GUI 2022/GUI.py (3.10.1)". The menu bar includes File, Edit, Format, Run, Options, Window, and Help. The code in the editor is:

```
from tkinter import * #* คือการ import Library ทั้งหมด
GUI = Tk()
GUI.mainloop()
```

A red arrow points from the explanatory text below to the line "from tkinter import \*". A blue callout box contains the text: "Tkinter เป็นเครื่องมือที่ใช้สำหรับสร้าง GUI".

ผลลัพธ์

# เขียน GUI เรอก



```
GUI.py - C:/Users/omeom/Desktop/Python GUI 2022/GUI.py (3.10.1)
File Edit Format Run Options Window Help
from tkinter import * #* คือการ import Library ทั้งหมด
from tkinter import ttk, messagebox

GUI = Tk()
GUI.title('โปรแกรมของคุณ v.1.0') # นี่คือชื่อโปรแกรม
GUI.geometry('500x300') #ปรับขนาดหน้าจอ

def Show():
    messagebox.showinfo('Show Box', 'สวัสดีจ้า ได้แล้ว!')

B1 = ttk.Button(GUI, text='กรุณาคลิกปุ่มนี่', command=Show)
B1.pack(ipadx=30, ipady=30, pady=50) #แบบปุ่มไว้กับโปรแกรมหลัก+ขนาดของปุ่ม

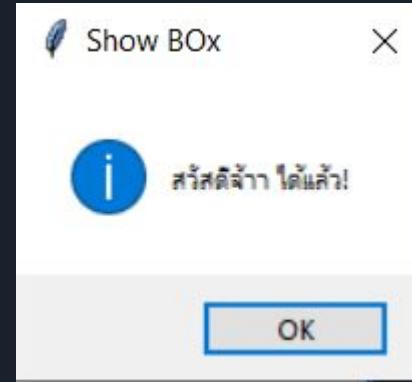
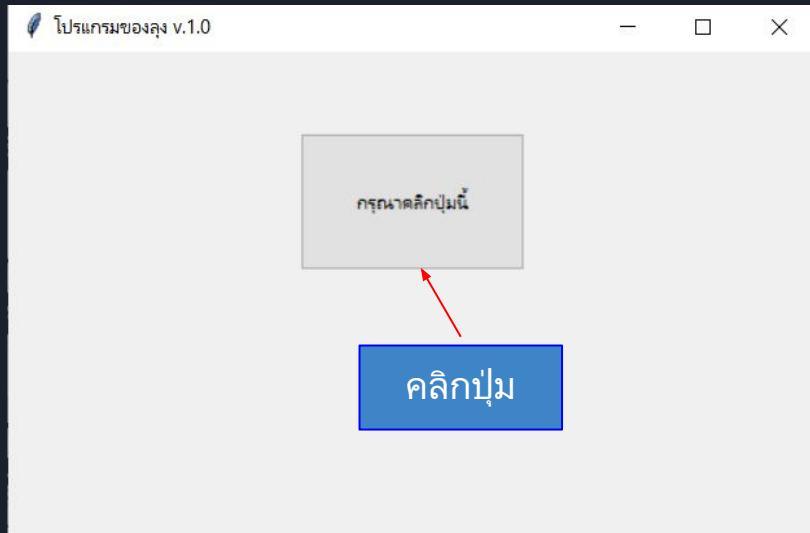
GUI.mainloop()
```

สร้างปุ่ม

ปรับระยะห่างที่อยู่ภายนอก

ปรับขนาดของปุ่มในแนวแกน x,y

# รันโดยการกด F5





# Python GUI 2022(EP.2)

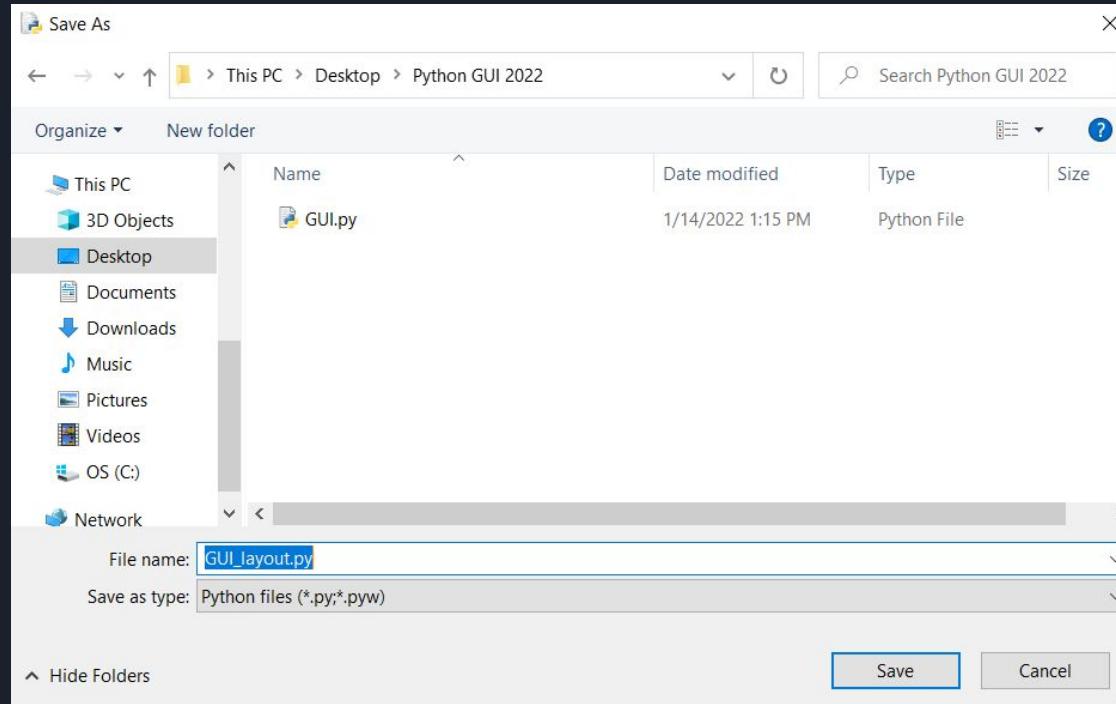
by



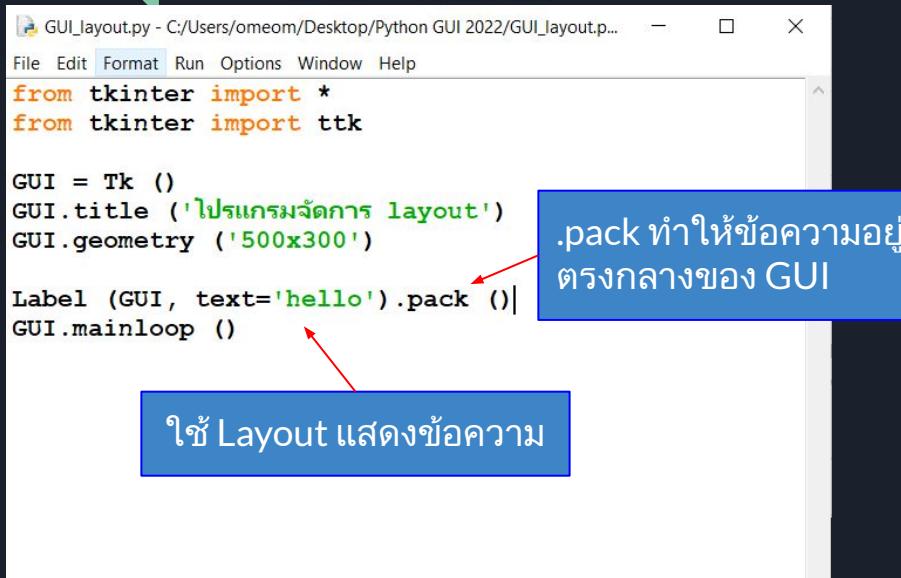
Uncle Engineer

ลุงวิศวกร สอนคำนวณ

# เปิด editor เชฟไฟล์ในโฟลเดอร์ Python GUI 2022 ตั้งชื่อ GUI\_layout.py



# การใช้ Layout โชว์ข้อความแบบบรรทัดเดียว



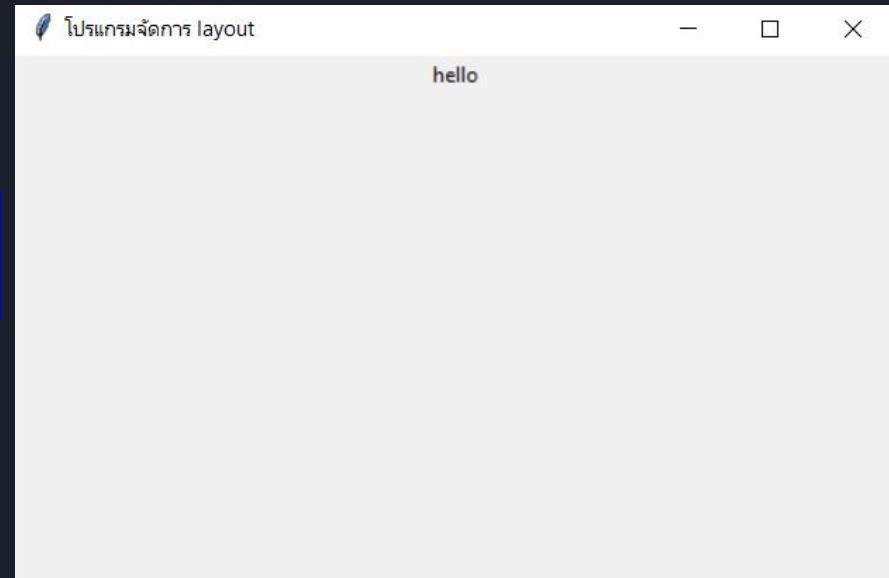
```
File Edit Format Run Options Window Help
from tkinter import *
from tkinter import ttk

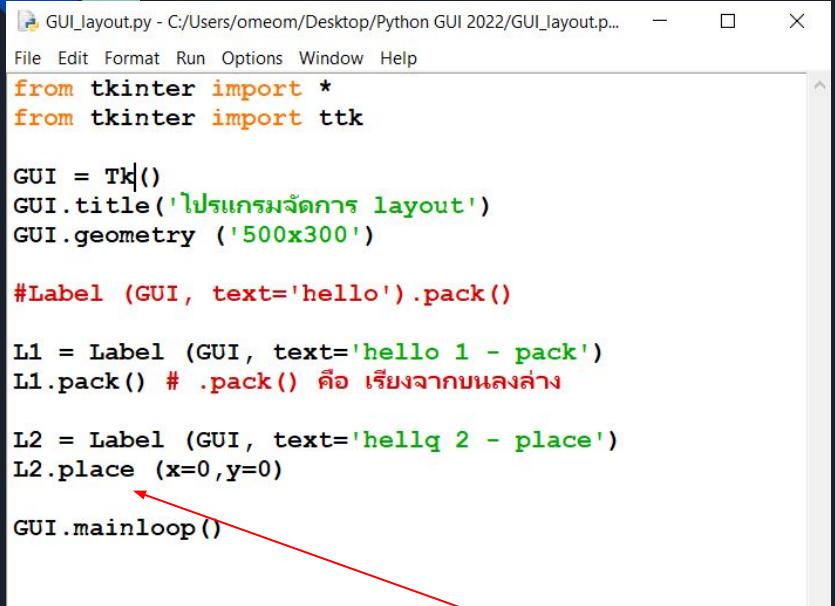
GUI = Tk ()
GUI.title ('โปรแกรมจัดการ layout')
GUI.geometry ('500x300')

Label (GUI, text='hello').pack ()
GUI.mainloop ()
```

.pack ทำให้ข้อความอยู่  
ตรงกลางของ GUI

ใช้ Layout และแสดงข้อความ





```
File Edit Format Run Options Window Help
from tkinter import *
from tkinter import ttk

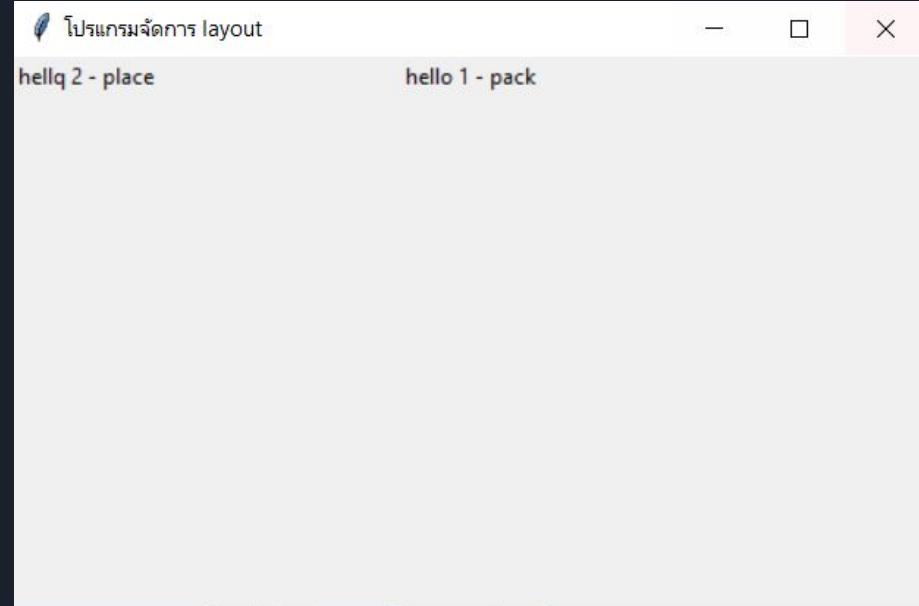
GUI = Tk()
GUI.title('โปรแกรมจัดการ layout')
GUI.geometry ('500x300')

#Label (GUI, text='hello').pack()

L1 = Label (GUI, text='hello 1 - pack')
L1.pack() # .pack() คือ เริ่มจากนั้งล่าง

L2 = Label (GUI, text='hello 2 - place')
L2.place (x=0,y=0)

GUI.mainloop()
```



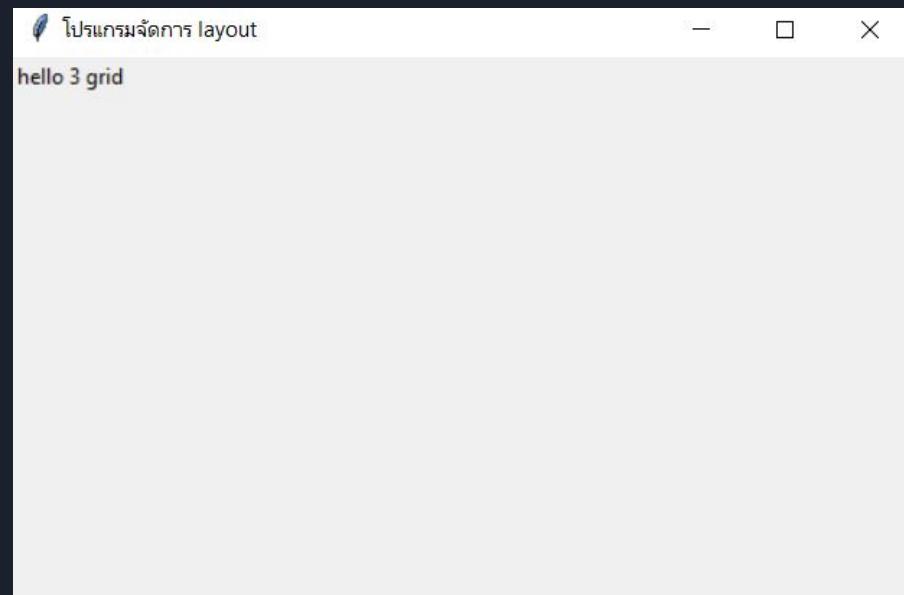
.place คือการคอนโทรลตำแหน่ง ใช้พิกัด x,y



```
*GUI_layout.py - C:/Users/omeom/Desktop/Python GUI 2022/GUI_layout.... - □
File Edit Format Run Options Window Help
from tkinter import *
from tkinter import ttk

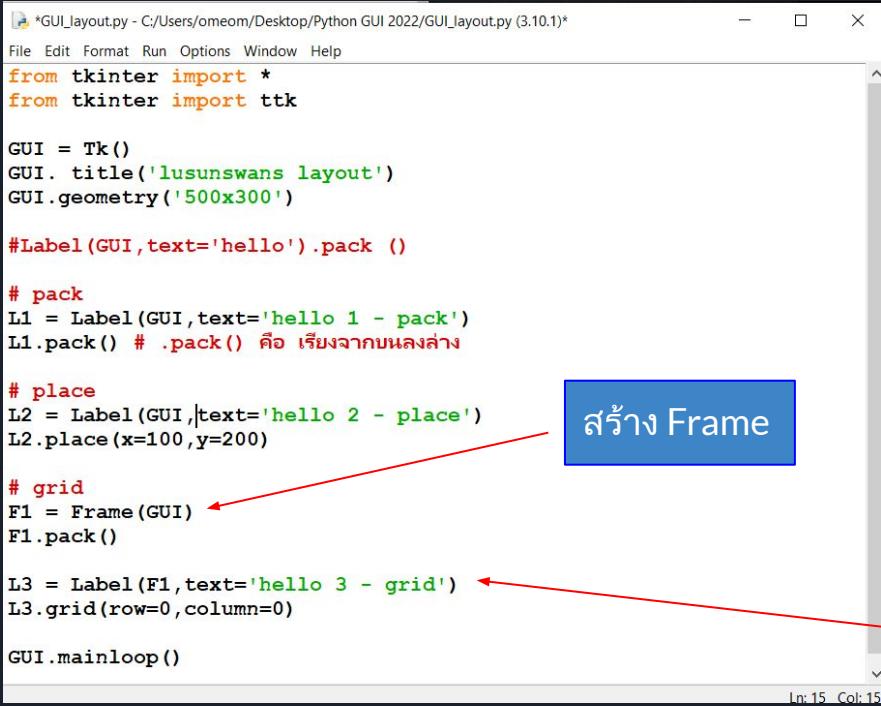
GUI = Tk()
GUI.title('โปรแกรมจัดการ layout')
GUI.geometry ('500x300')

L3 = Label(GUI, text='hello 3 grid')
L3.grid(row=1, column=1)
```



.grid คือการคอนโทรลตัว每逢 ใช้  
row, column ในการคอนโทรล

# .pack และ .grid ไม่สามารถใช้ร่วมกันได้ วิธีคือการสร้าง Frame และเอาเข้าไปไว้ด้านใน



```
*GUI_layout.py - C:/Users/omeom/Desktop/Python GUI 2022/GUI_layout.py (3.10.1)*
File Edit Format Run Options Window Help
from tkinter import *
from tkinter import ttk

GUI = Tk()
GUI.title('lusunwans layout')
GUI.geometry('500x300')

#Label(GUI, text='hello').pack ()

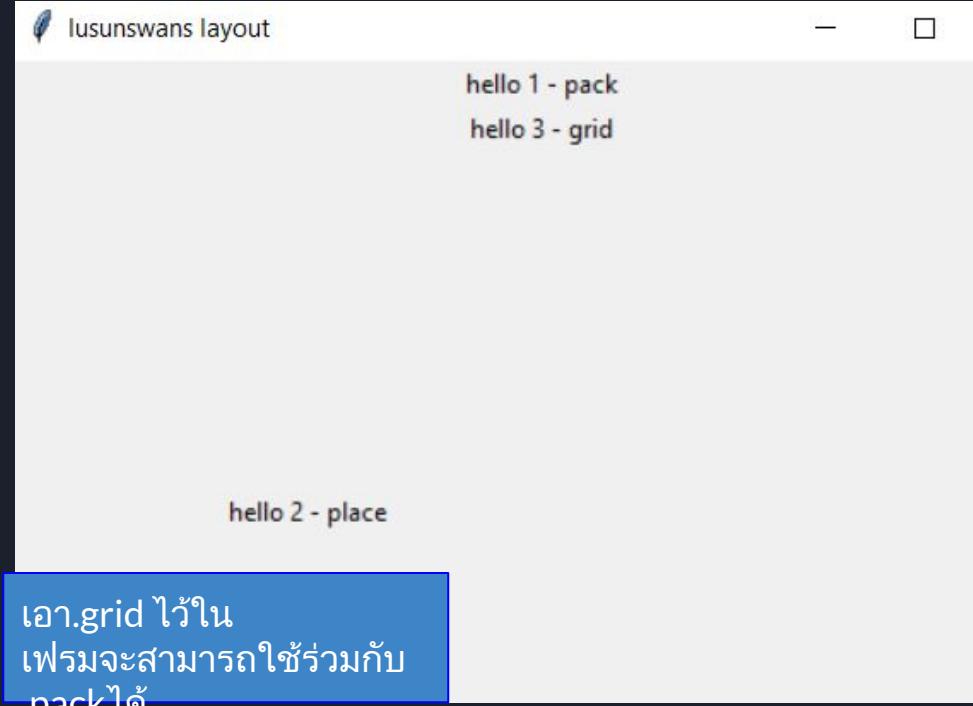
# pack
L1 = Label(GUI, text='hello 1 - pack')
L1.pack() # .pack() คือ เริ่มจากบล็อกล่าง

# place
L2 = Label(GUI, text='hello 2 - place')
L2.place(x=100, y=200)

# grid
F1 = Frame(GUI)
F1.pack()

L3 = Label(F1, text='hello 3 - grid')
L3.grid(row=0, column=0)

GUI.mainloop()
```



# วิธีการใส่สี

```
GUI_layout.py - C:/Users/omeom/Desktop/Python GUI 2022/GUI_layout.py (3.10.1)
File Edit Format Run Options Window Help
from tkinter import *
from tkinter import ttk

GUI = Tk()
GUI.title('โปรแกรมจัดการ layout')
GUI.geometry('500x300')

#Label(GUI,text='hello').pack ()

# pack
L1 = Label(GUI,text='hello 1 - pack')
L1.pack() # .pack() ต้อง เรียงจากบนลงล่าง

# place
L2 = Label(GUI,text='hello 2 - place')
L2.place(x=100,y=200)

# grid
F1 = Frame(GUI)
F1.pack()

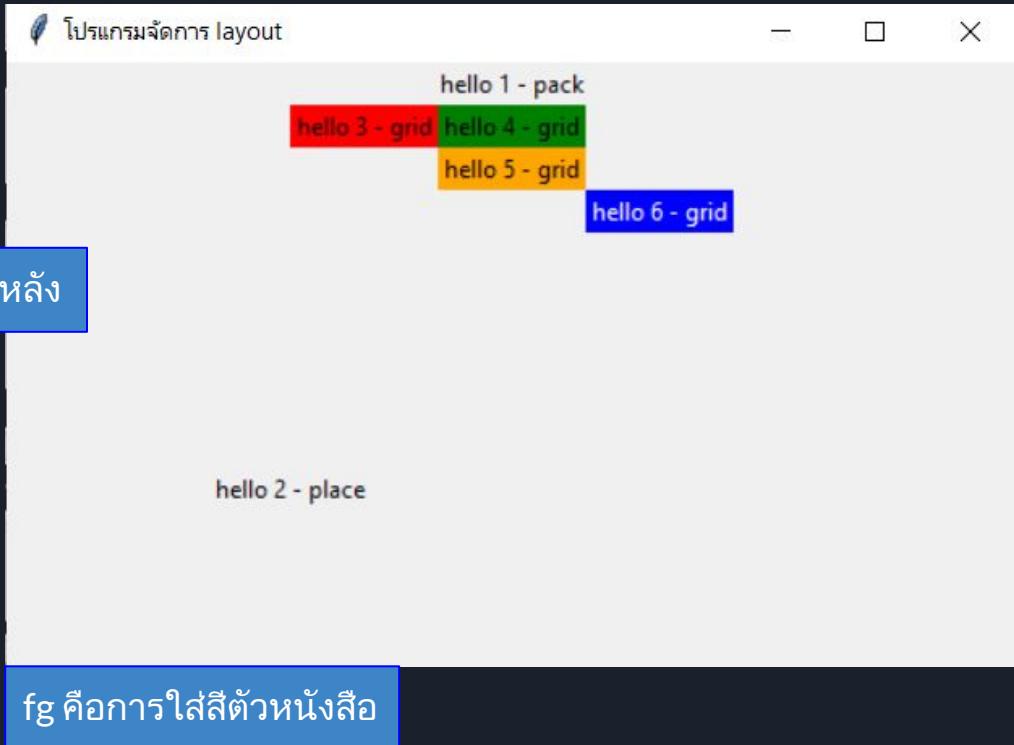
L3 = Label(F1,text='hello 3 - grid',bg='red')
L3.grid(row=0,column=0)

L4 = Label(F1,text='hello 4 - grid',bg='green')
L4.grid(row=0,column=1)

L5 = Label(F1,text='hello 5 - grid',bg='orange')
L5.grid(row=1,column=1)

L6 = Label(F1,text='hello 6 - grid',bg='blue', fg='white')
L6.grid(row=2, column=2)

GUI.mainloop ()
```



## ใส่เลี้นกรอบให้ Frame

GUI\_layout.py - C:/Users/omeom/Desktop/Python GUI 2022/GUI\_layout.py (3.10.1)

```
File Edit Format Run Options Window Help
from tkinter import *
from tkinter import ttk

GUI = Tk()
GUI.title('โปรแกรมจัดการ layout')
GUI.geometry('500x300')
|
#Label(GUI,text='hello').pack ()

# pack
L1 = Label(GUI, text='hello 1 - pack')
L1.pack() # .pack() คือ เรียงจากบนลงล่าง

# place
L2 = Label(GUI, text='hello 2 - place')
L2.place(x=100,y=200)

# grid
F1 = LabelFrame(GUI, text='grid color')
F1.place(x=150, y=200) _____

L3 = Label(F1, text='hello 3 - grid',bg='red')
L3.grid(row=0,column=0)

L4 = Label(F1, text='hello 4 - grid',bg='green')
L4.grid(row=0,column=1)

L5 = Label(F1, text='hello 5 - grid',bg='orange')
L5.grid(row=1,column=1)

L6 = Label(F1, text='hello 6 - grid',bg='blue', fg='white')
L6.grid(row=2, column=2)

GUI.mainloop ()
```

โปรแกรมจัดการ layout

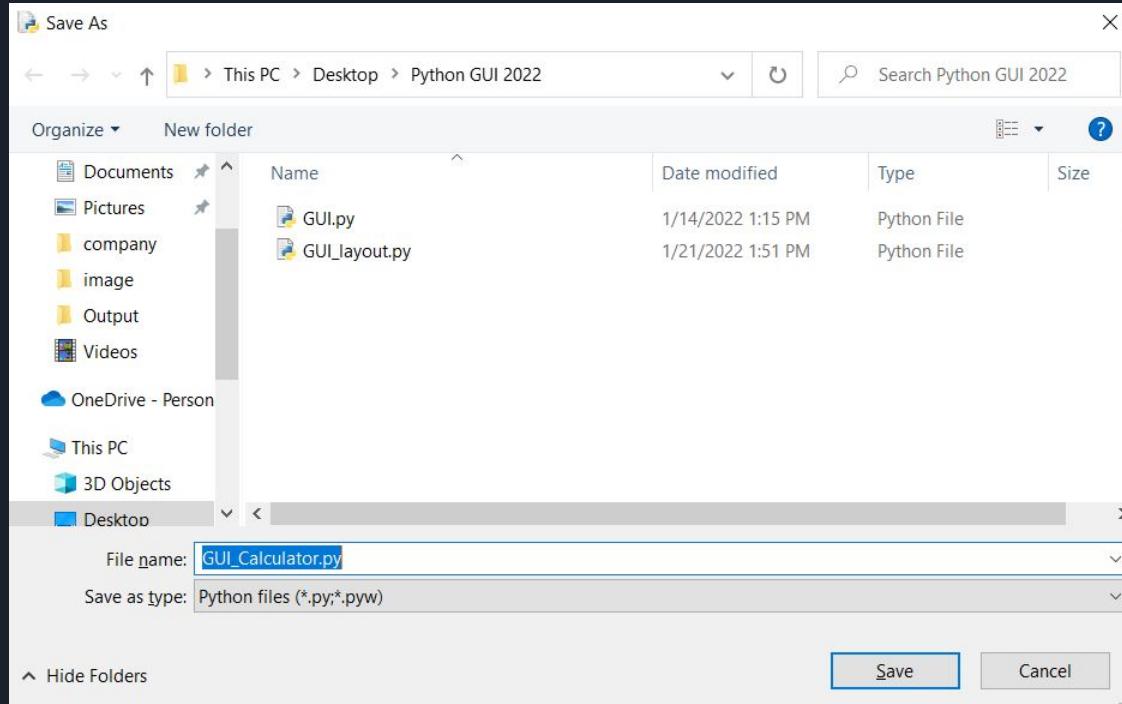
hello 1 - pack

ใช้ .place ถอนโโทร  
ลดตำแหน่ง Frame

hello 2 - grid color



# เปิด editor อันใหม่ เชฟไฟล์ในโฟลเดอร์ Python GUI 2022 ตั้งชื่อ GUI\_calculator.py



# โปรแกรมคำนวณราคากุ้ง

```
*GUI_Calculator.py - C:/Users/omeom/Desktop/Python GUI 2022/GUI_Calculator.py (3.10.1)*
File Edit Format Run Options Window Help
from tkinter import *
from tkinter import ttk, messagebox

GUI = Tk ()
GUI.title ('โปรแกรมจัดการ layout')
GUI.geometry ('500x300')

L1 = Label(GUI, text='กรอกจำนวนกุ้ง (กิโลกรัม)', font=('Angsana New', 25))
L1.pack ()

v_kilo = StringVar() #ตัวแปรพิเศษเอาไว้เก็บค่า

E1 = ttk.Entry(GUI, textvariable= v_kilo, width=10, justify='right', font=('impact', 30))
E1.pack (pady=20)

def Calc (event=None) :
    print('กำลังคำนวณ.. กรุณารอสักครู่')
    kilo = float (v_kilo.get ()) # .get () ดึงข้อมูลจากตัวแปรที่เปลี่ยนเป็น StringVar
    print (kilo * 10)
    calc_result = kilo * 299
    messagebox.showinfo('รวมราคากุ้ง', 'ลูกค้าต้องจ่ายตั้งศัทting หมด: {:.2f} บาท (กิโลกรัมละ 299 บาท)'.format(calc_result))

B1 = ttk.Button(GUI, text='คำนวนราคา', command=Calc)
B1.pack(ipadx=40, ipady=30)

E1.bind ('<Return>', Calc) # ต้องใส่คำว่า event=None ในฟังก์ชันเดียว

GUI.mainloop ()
```

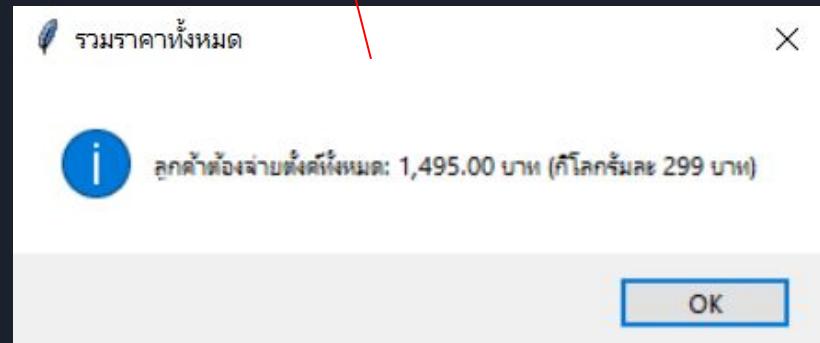
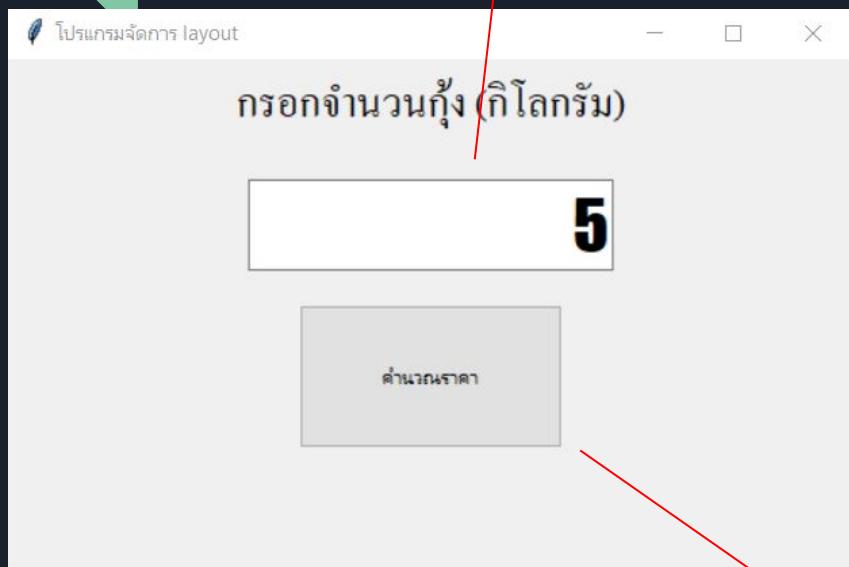
สร้างช่องกรอกข้อมูลโดยใช้ Entry ปรับขนาดของช่องโดยใช้ Width 1=1 ตัวอักษร , Justify คือการปรับตัวหนังสือให้อยู่ด้านขวา , font คือ ปรับฟอนต์

ลิ้ง StringVar กับ Entry โดยการใช้ textvariable

ทำให้สามารถกด Enter แทนการใช้เมาส์คลิกปุ่มคำนวณราคาได้เลย

ใส่ตัวเลขกิโลกรัมในช่อง

ผลลัพธ์



คลิกที่ปุ่มคำนวณราคารีกด  
ปุ่ม Enter บนแป้นพิมพ์เพื่อ  
คำนวณราคา



# Python GUI 2022(EP.3)

by



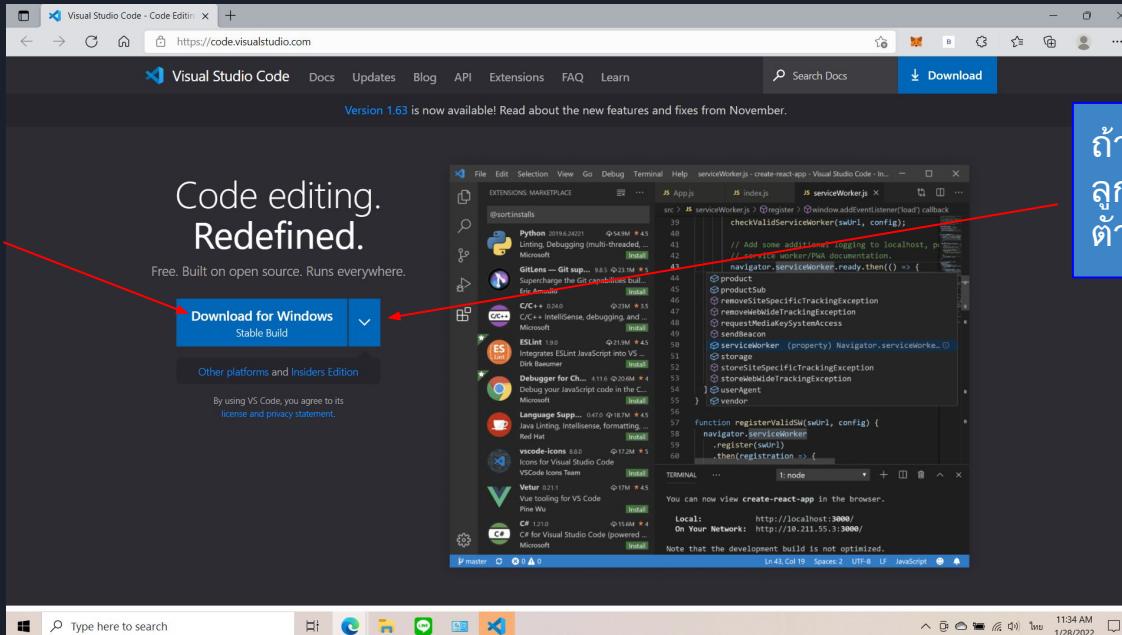
Uncle Engineer

ลุงวิศวกร สอนคำนวณ

# ติดตั้ง Editor

## 1. Visual Studio Code

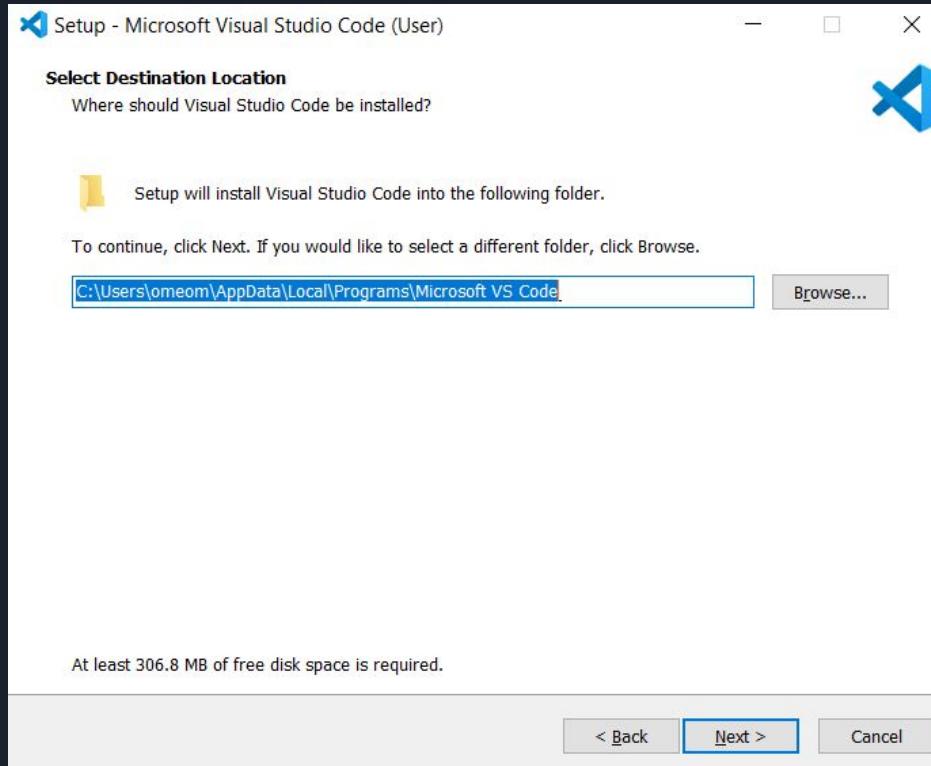
ดาวน์โหลดได้ที่ลิ้ง <https://code.visualstudio.com/> คลิก Download for Windows ได้เลยหรือถ้าใช้ OS อื่นให้คลิกที่ลูกศรลงปุ่ม



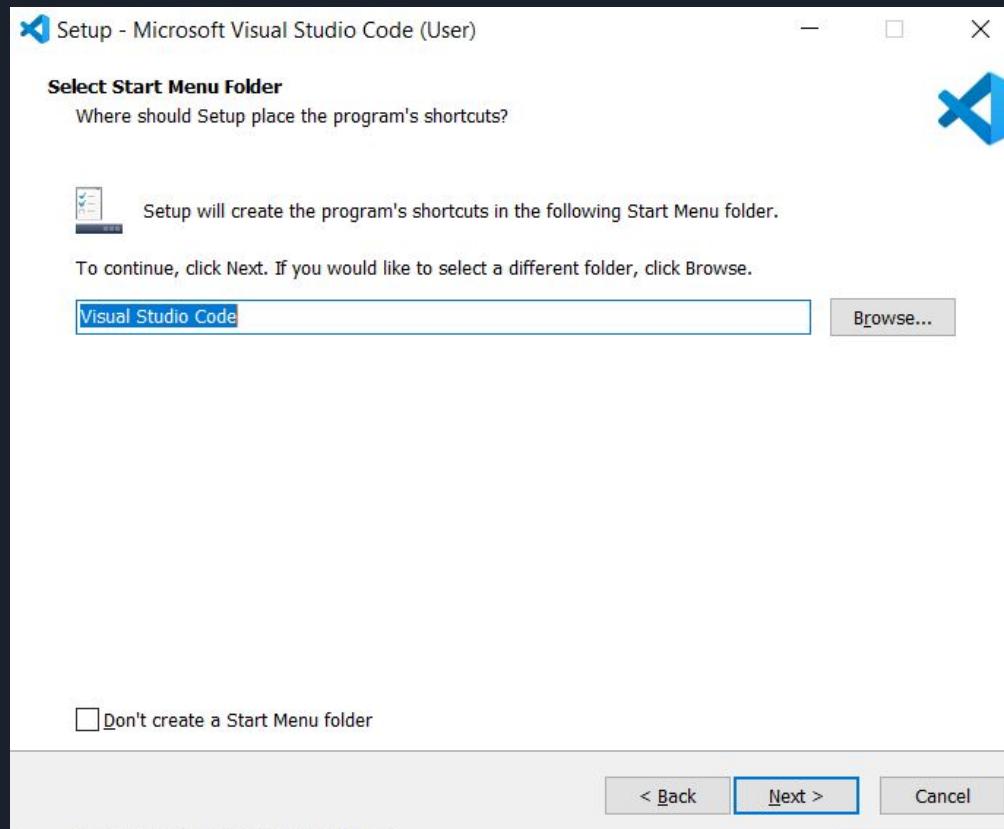
# ติดตั้งไฟล์ที่ดาวน์โหลดมา เลือกเป็น I accept the agreement และกด Next



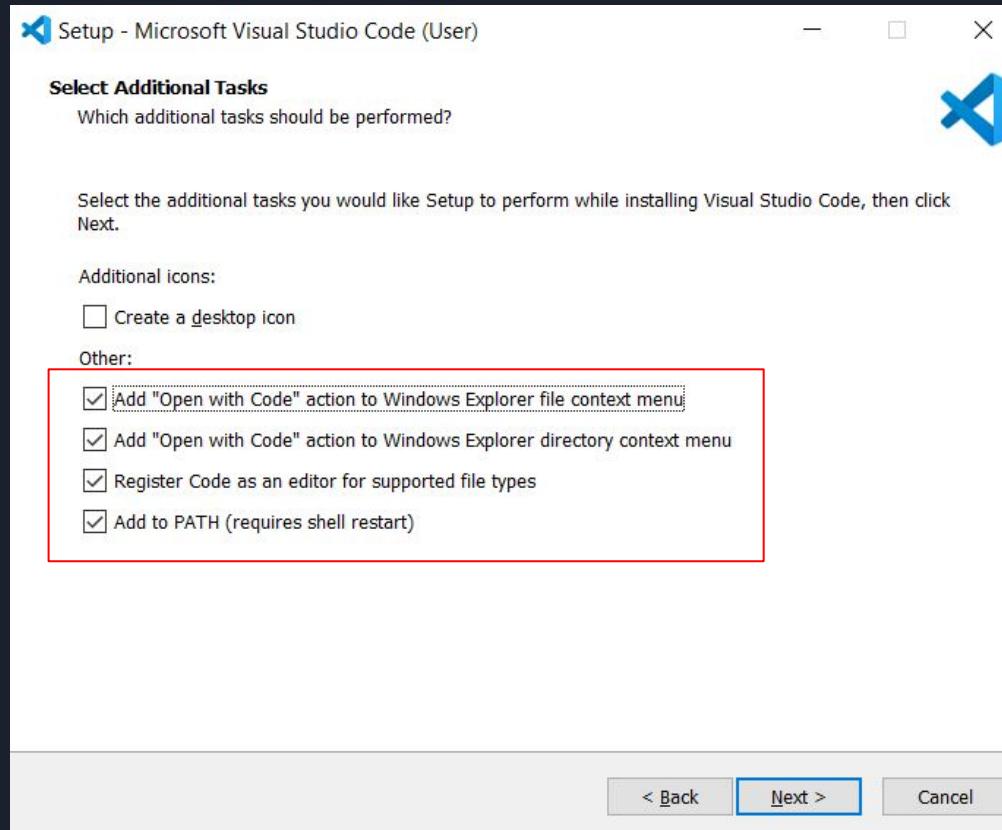
# เลือก Location สำหรับติดตั้งแล้วกด Next



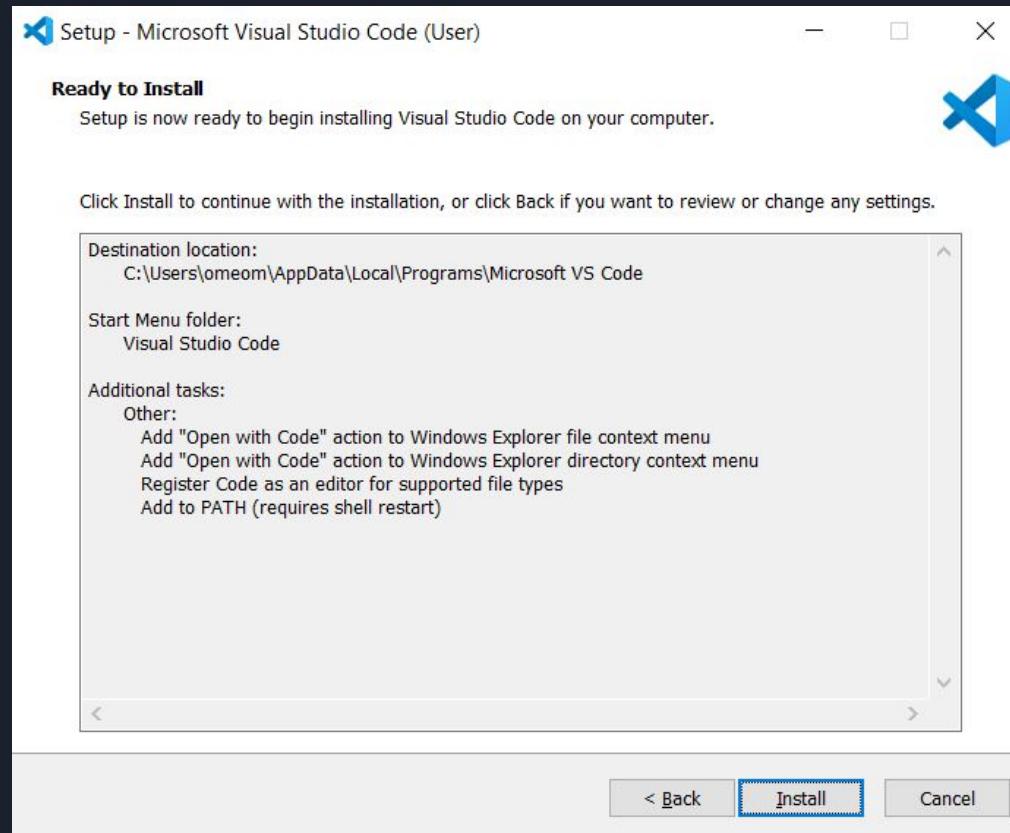
ନୀ Next



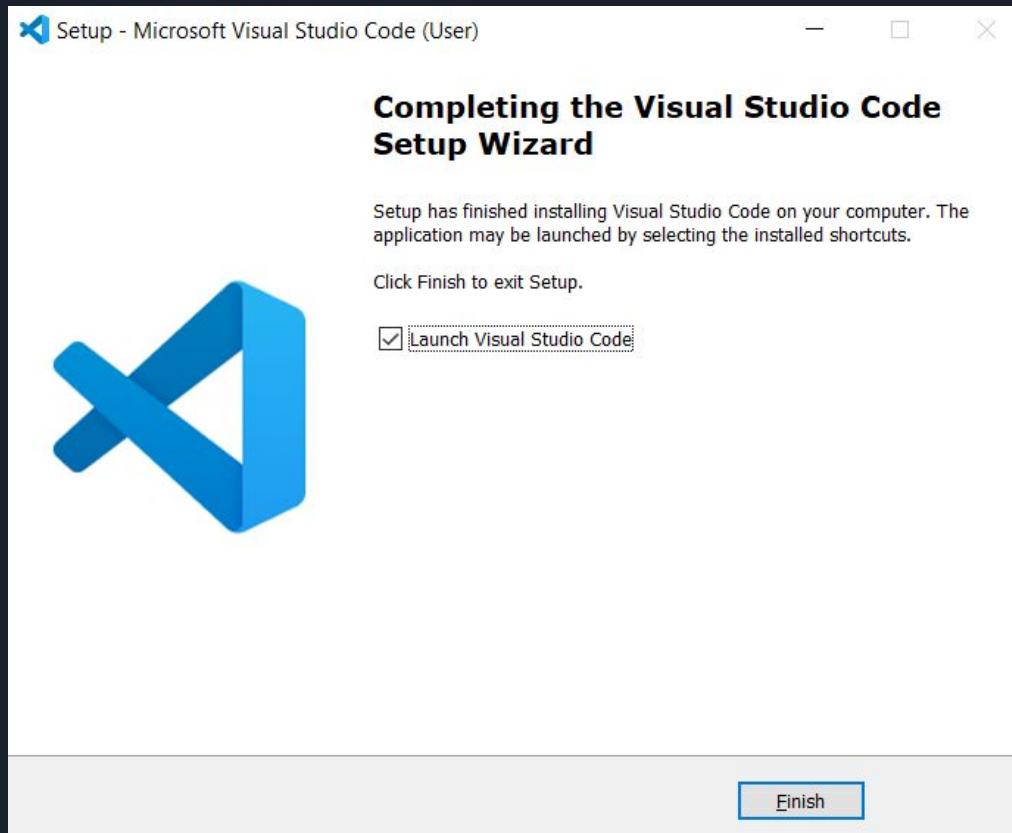
# ติดตั้งให้เป็นเครื่องหมายถูกตามภาพ



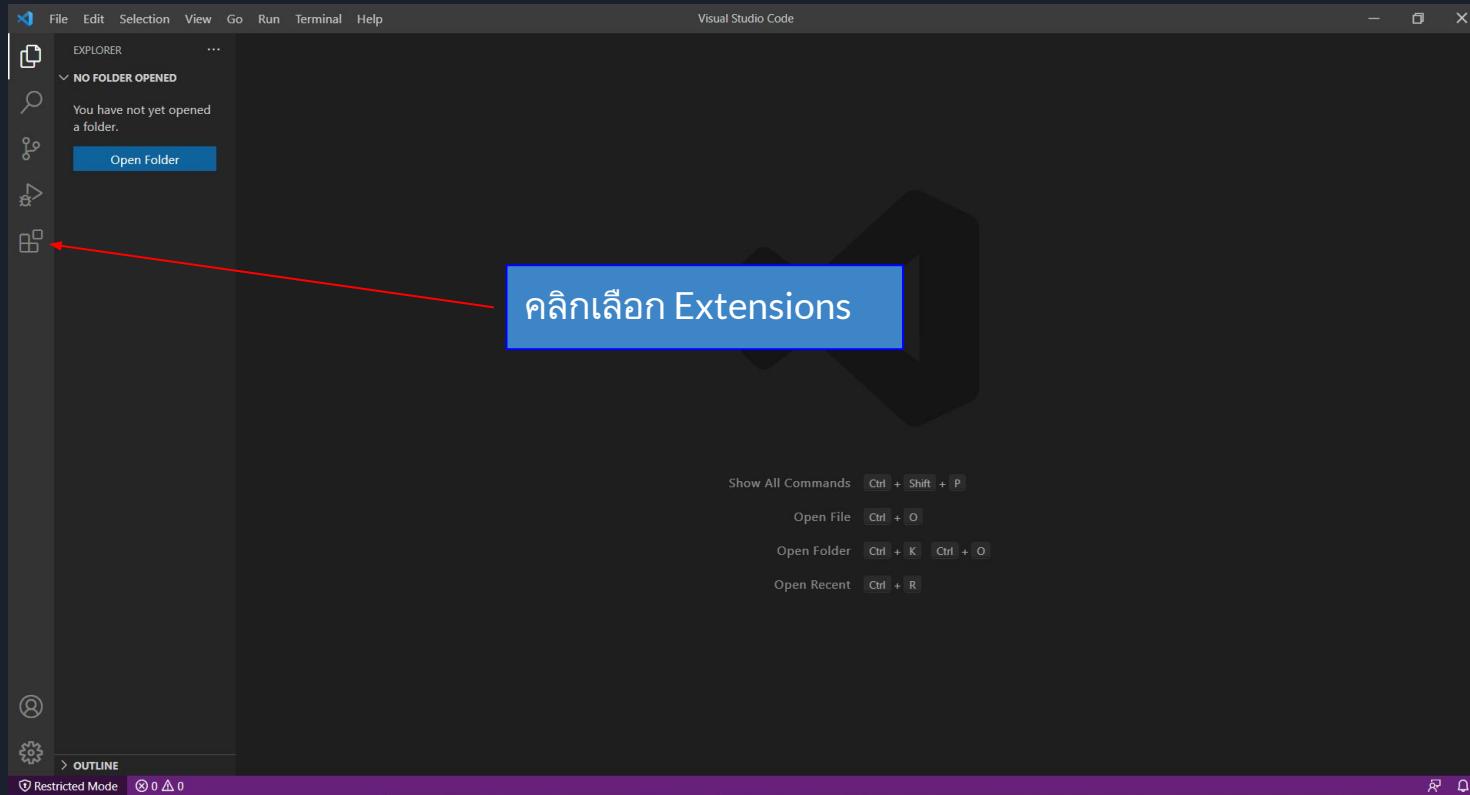
# ນັດ Install



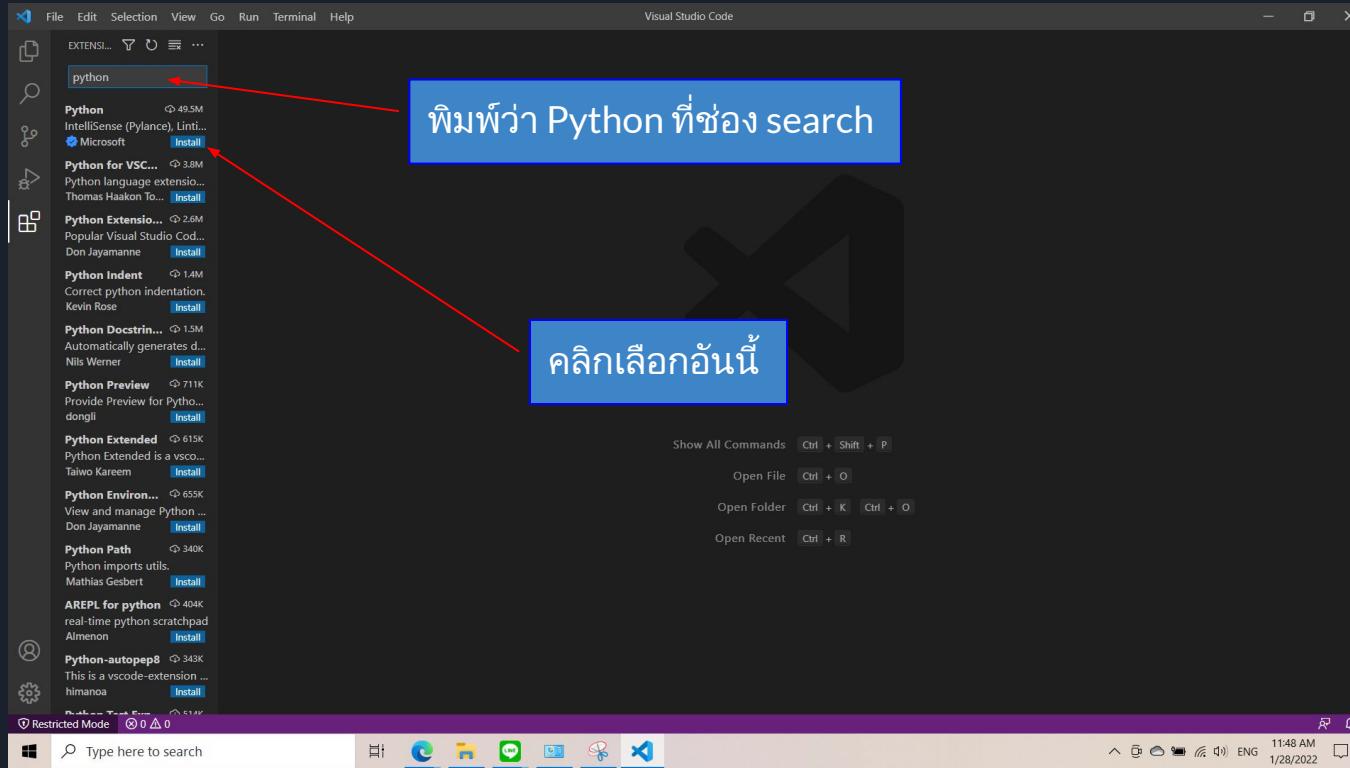
# ติดตั้งเรียบร้อย

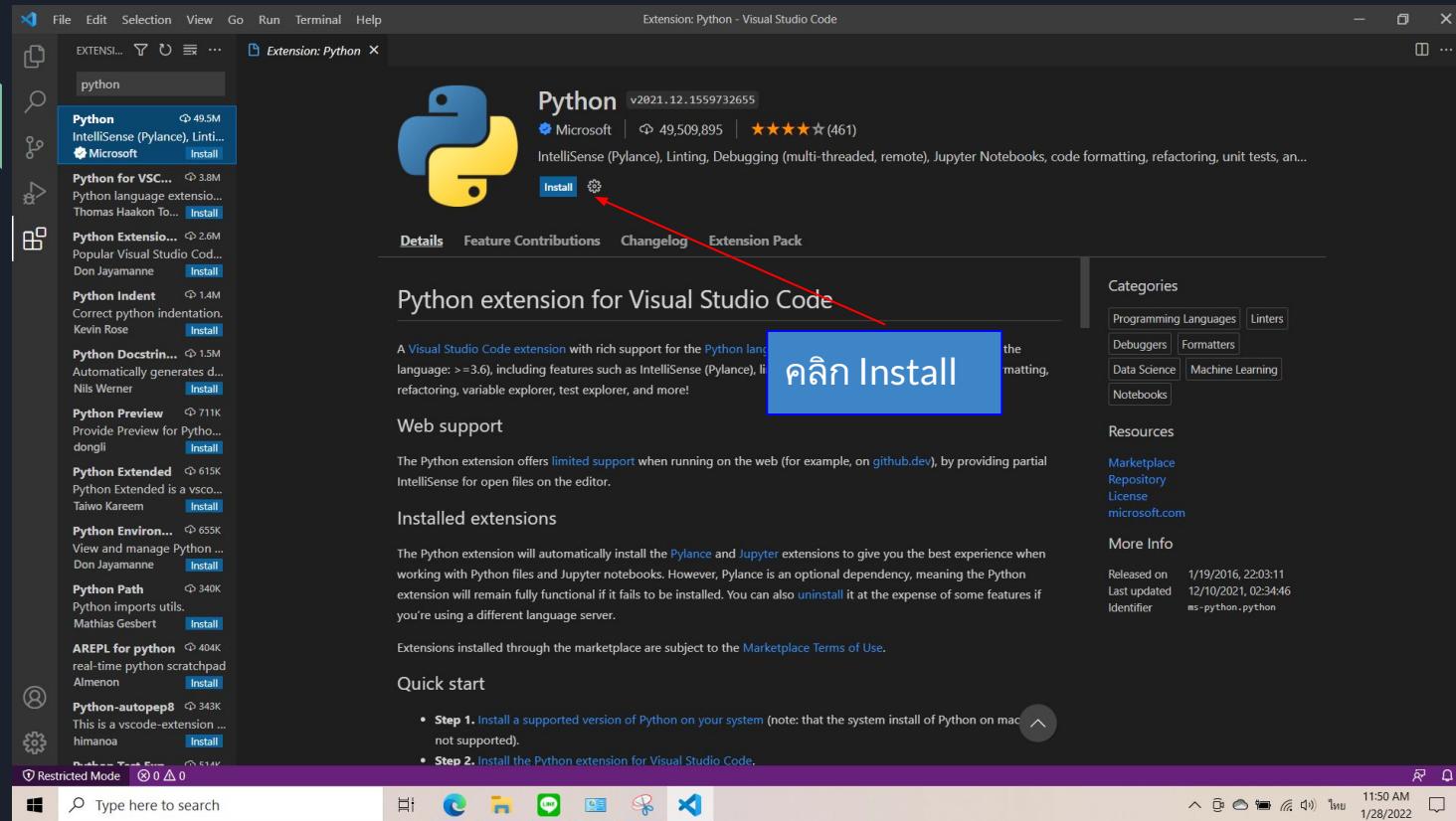


# ติดตั้ง Extensions

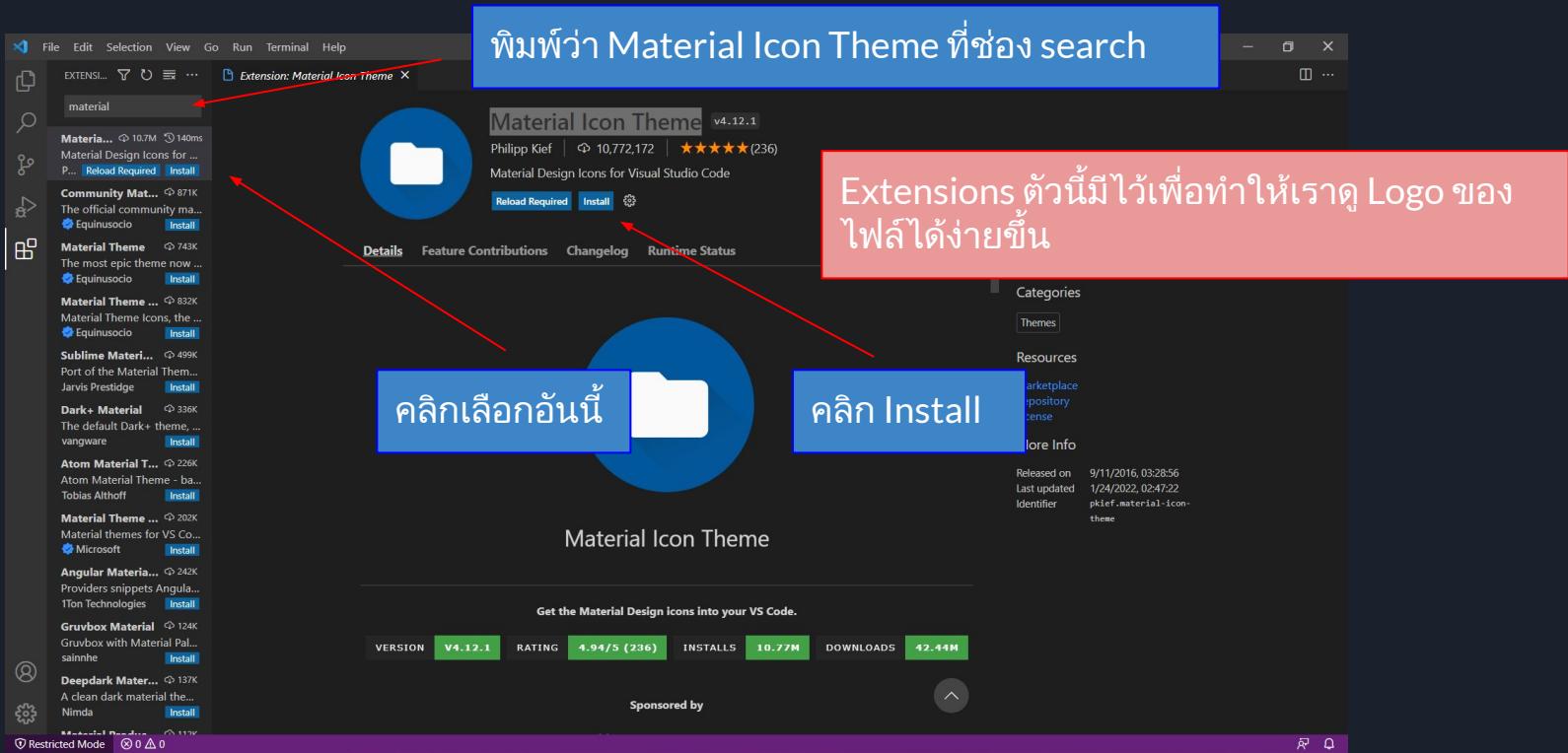


# พิมพ์ว่า Python คลิกเข้าที่ไปอันแรก

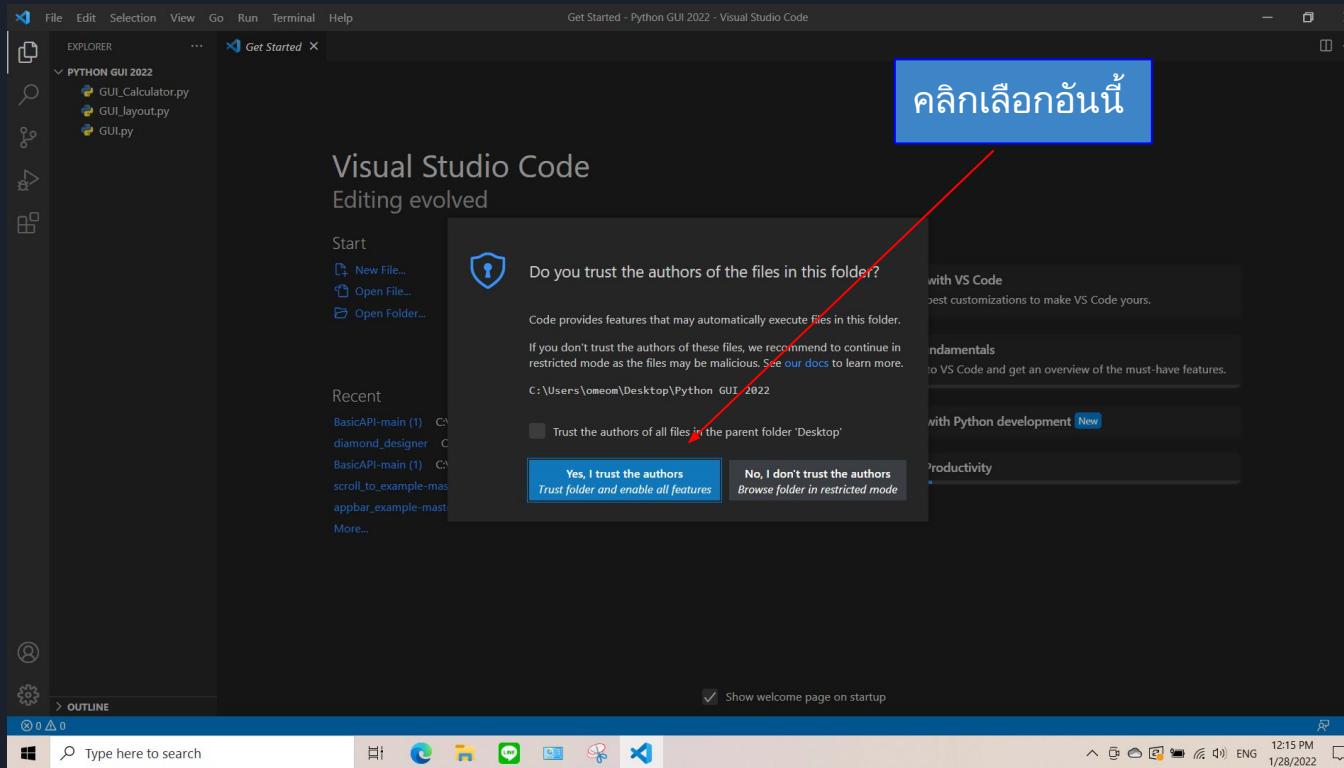




# พิมพ์ค้นหาในช่องค้นหาอีกครั้งนึงหาคำว่า Material Icon Theme



# เปิดไฟล์ที่ทำ Ep ที่แล้ว ด้วย VSCode จะมี Popup เด้งขึ้นมาให้คลิกเลือก Yes



เมื่อเราเปิดไฟล์ ที่ด้านล่างจะมีคำว่า Python ตามด้วยเวอร์ชัน ให้คลิกเลือก จะมีตัวเลือก Python เวอร์ชันที่เรามีในเครื่องขึ้นมา เราสามารถเลือกเวอร์ชัน Python ที่จะใช้ได้

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows files: `GUI_calculator.py`, `GUI_layout.py`, and `GUI.py`.
- Code Editor:** Displays Python code for a GUI application using Tkinter.
- Top Bar:** Shows "File", "Edit", "Selection", "View", "Go", "Run", "Terminal", and "Help".
- Bottom Status Bar:** Shows "Python 3.10.1 64-bit" and other status indicators.
- Bottom Right Corner:** Shows "In 1, Col 1", "Spaces: 4", "UTF-8", "CRLF", "Python", and a settings icon.

A red box highlights the "Select Interpreter" dialog, which lists available interpreters:

- Current: C:\Python310\python.exe
- + Enter interpreter path...
- ★ Python 3.10.1 64-bit Recommended (highlighted)
- C:\Python3.9\python.exe
- Python 3.9.6 64-bit
- C:\Python39\python.exe

A blue callout box with a red arrow points to the "Python 3.10.1 64-bit Recommended" entry, containing the text: "จะมีเวอร์ชัน Python ขึ้นมาให้เลือก".

A red callout box with a red arrow points to the bottom-left corner of the interface, containing the text: "คลิกตรงนี้".

```
from tkinter import *  
from tkinter import t  
GUI = Tk()  
GUI.title('โปรแกรมคำนวณ')  
GUI.geometry('500x300')  
  
#Label(GUI,text='hello').pack ()  
  
# pack  
L1 = Label(GUI,text='hello 1 - pack')  
L1.pack() # .pack() คือ รีจิกกานบลงถາ  
  
# place  
L2 = Label(GUI,text='hello 2 - place')  
L2.place(x=100,y=200)  
  
# grid  
F1 = LabelFrame(GUI,text='grid color')  
F1.place(x=150, y=200)  
  
L3 = Label(F1,text='hello 3 - grid',bg='red')  
L3.grid(row=0,column=0)  
  
L4 = Label(F1,text='hello 4 - grid',bg='green')  
L4.grid(row=0,column=1)  
  
L5 = Label(F1,text='hello 5 - grid',bg='orange')  
L5.grid(row=1,column=0)  
  
L6 = Label(F1,text='hello 6 - grid',bg='blue', fg='white')  
L6.grid(row=2, column=2)  
  
GUI.mainloop ()
```

รัน Code โดยการกดตรงนี้

```
File Edit Selection View Go Run Terminal Help  
GUI_Calculator.py - Python GUI 2022 - Visual Studio Code  
EXPLORER  
PYTHON GUI 2022  
GUI_Calculator.py > ...  
GUI_Calculator.py  
GUI_layout.py  
GUI.py  
1  from tkinter import *  
2  from tkinter import ttk, messagebox  
3  
4  GUI = Tk ()  
5  GUI.title ('โปรแกรมลักษณะ layout')  
6  GUI.geometry ('500x300')  
7  
8  L1 = Label(GUI, text='กรอกจำนวนกิโลกรัม', font=('Angsana New', 25))  
9  L1.pack ()  
10  
11 v_kilo = StringVar() #ตัวแปรที่เดียวเอาไว้เก็บค่า<...  
13 E1 = ttk.Entry(GUI, textvariable= v_kilo, width=10,justify='right',font=('impact',30))  
14 E1.pack(pady=20)  
15  
16 def Calc (event=None) :  
17     print('กำลังคำนวณ... กรุณารอสักครู่')  
18     kilo = float (v_kilo.get ()) # .get () ดึงข้อมูลจากลักษณะแบบ stringVar  
19     print (kilo * 10)  
20     calc_result = kilo * 299  
21     messagebox.showinfo('รวมราคารถใหม่', 'สูงค่าต้องจ่ายตั้งค่าทั้งหมด: {:.2f} บาท ( กิโลกรัมละ 299 บาท )'.format(calc_result))  
22  
23 B1 = ttk.Button(GUI, text='คำนวณราคา', command=Calc)  
24 B1.pack(ipadx=40, ipady=30)  
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE  
Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
Try the new cross-platform PowerShell https://aka.ms/pscore6  
PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/GUI_layout.py"  
PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/GUI_Calculator.py"  
PS C:\Users\omeom\Desktop\Python GUI 2022>  
Python 3.10.1 64-bit @ 0 △ 0  
Ln 8, Col 72 Spaces: 4 UTF-8 CRLF Python 🔍 ⚙
```

## ติดตั้ง Editor

### 2. Sublime Text

ดาวน์โหลดได้ที่ลิ้ง <https://www.sublimetext.com/download> คลิกดาวน์โหลดตาม OS ที่ตัวเองใช้

## Download

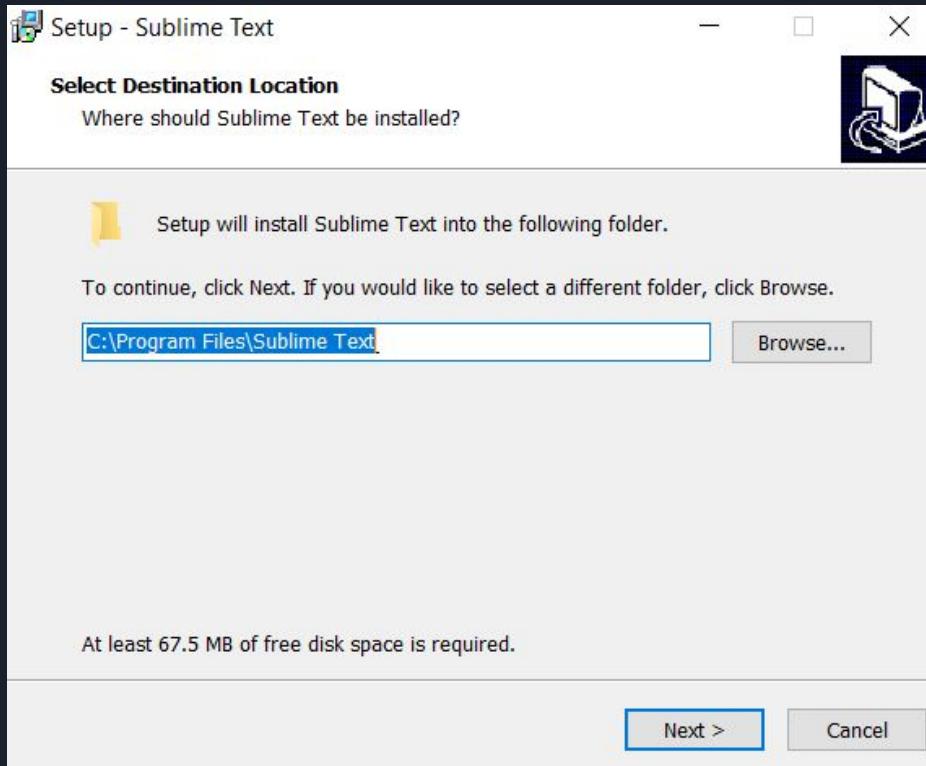
Sublime Text 4 is the current version of Sublime Text. For bleeding-edge releases, see the [dev builds](#).

Version: Build 4126

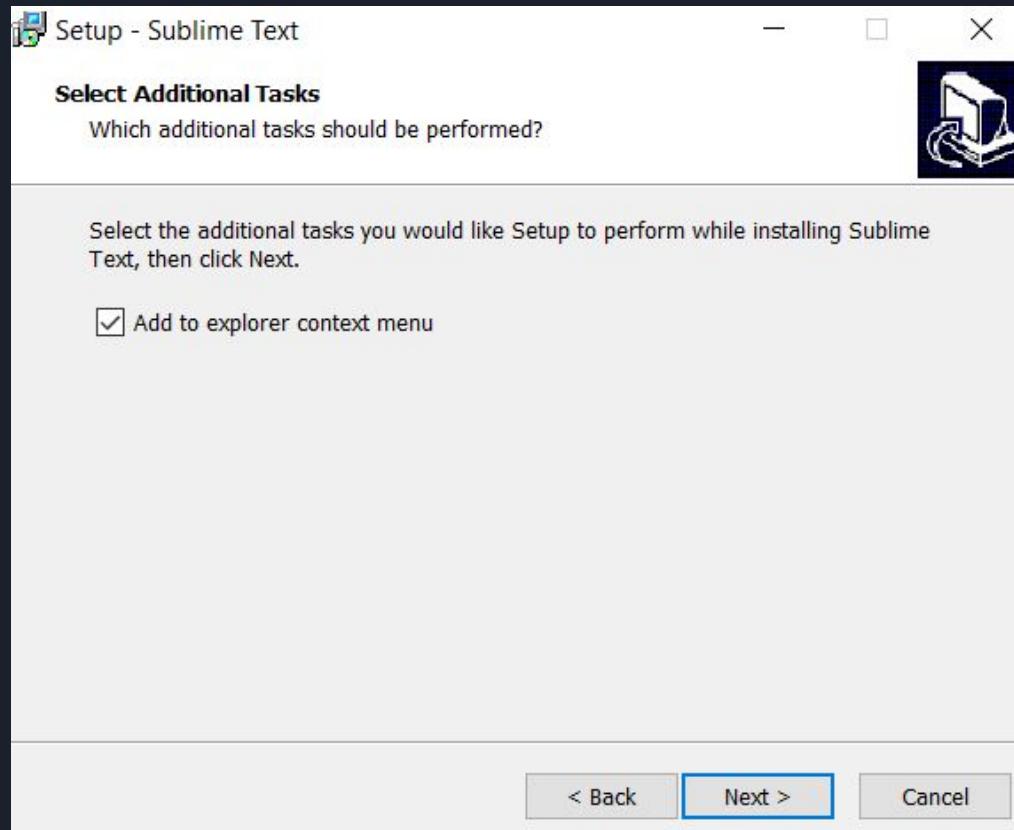
- [macOS](#) (10.9 or later is required)
- [Windows](#) - also available as a [portable version](#)
- [Linux repos](#) - also available as a [x86-64](#) or [ARM64](#) tarball

# ติดตั้งไฟล์ที่ดาวน์โหลดมา

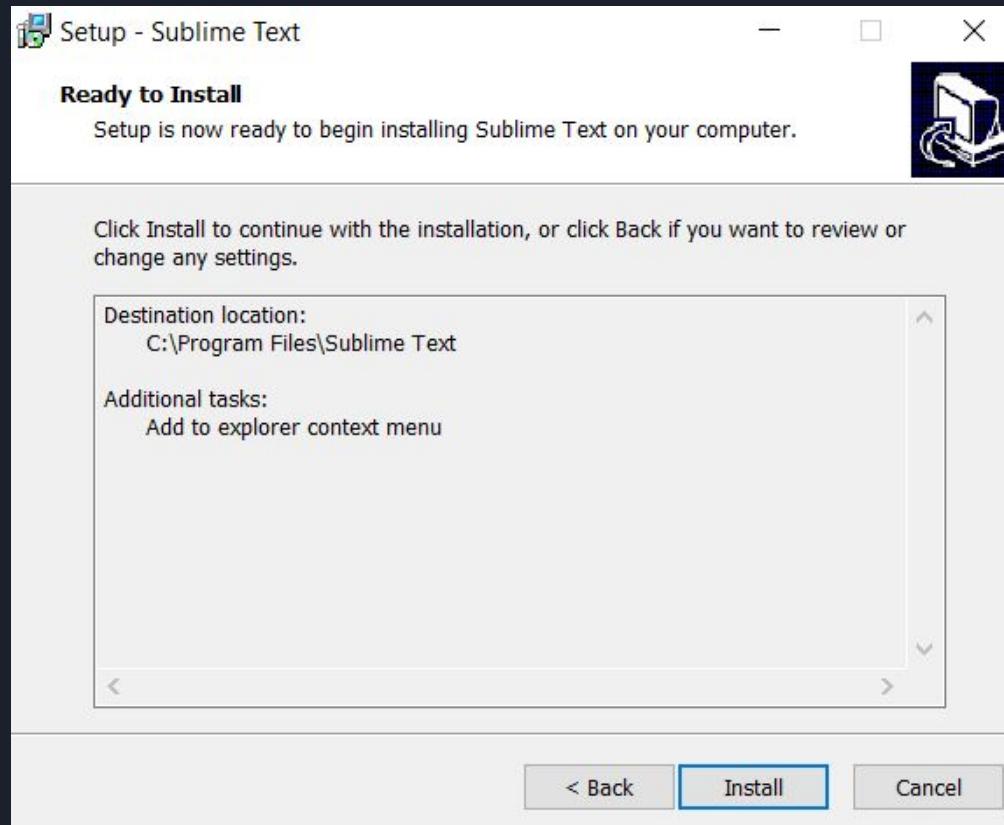
## เลือก Location สำหรับติดตั้งแล้วกด Next



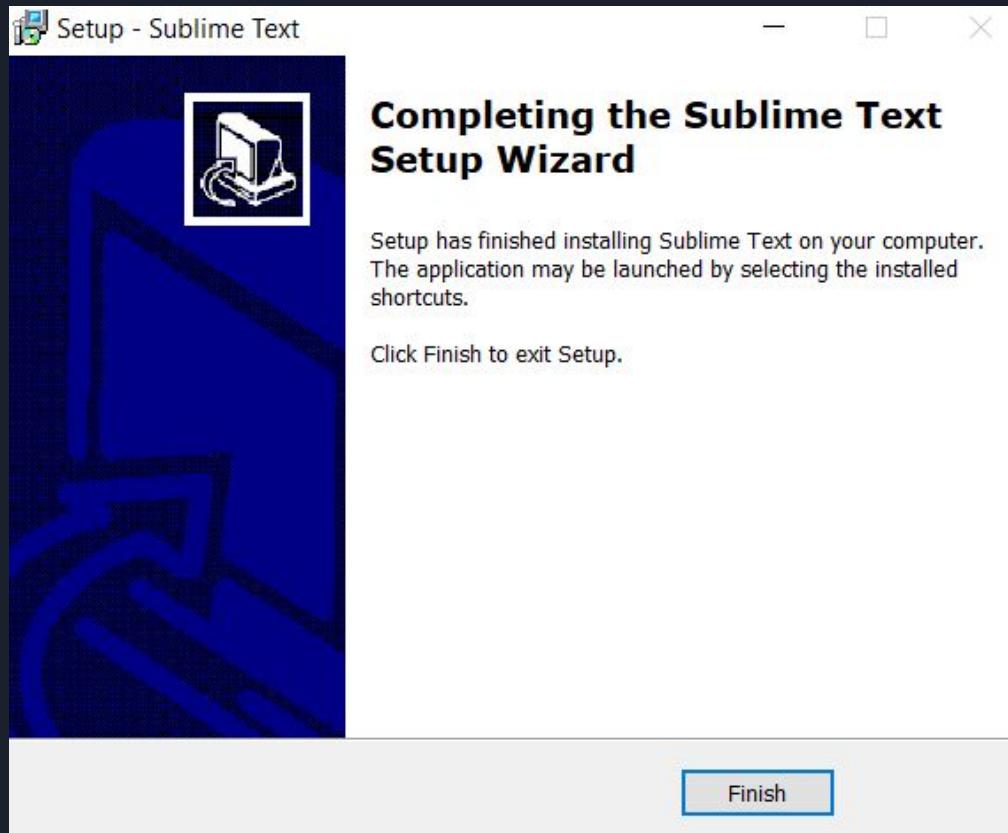
ติ๊กเลือก Add to explorer context menu และกด Next



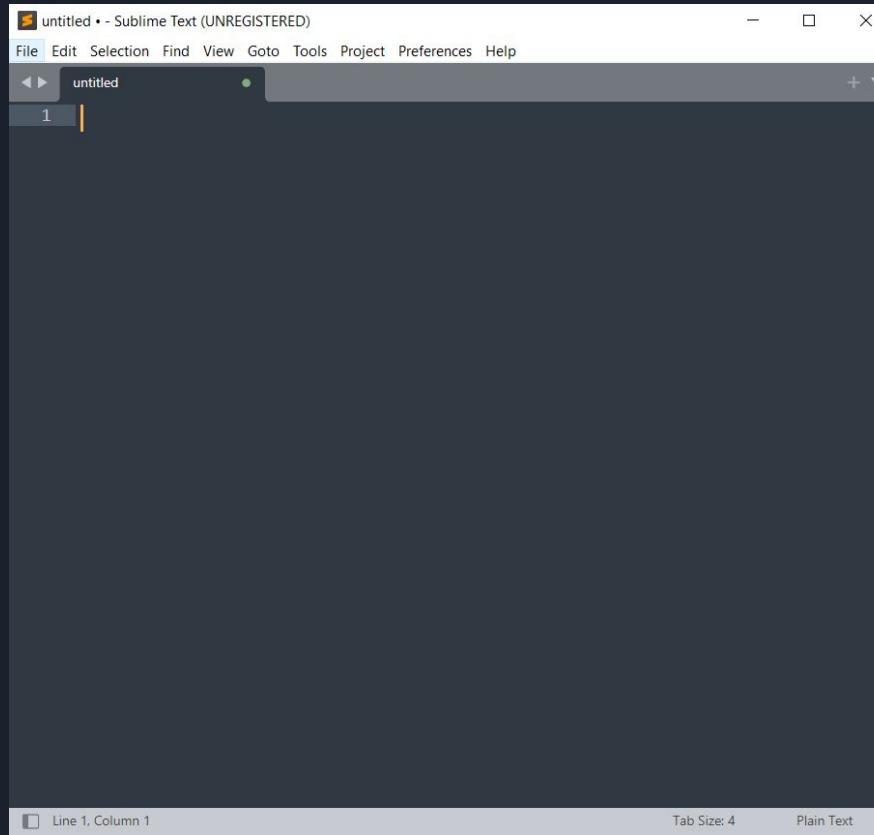
# ນັດ Install



# ติดตั้งเรียบร้อย

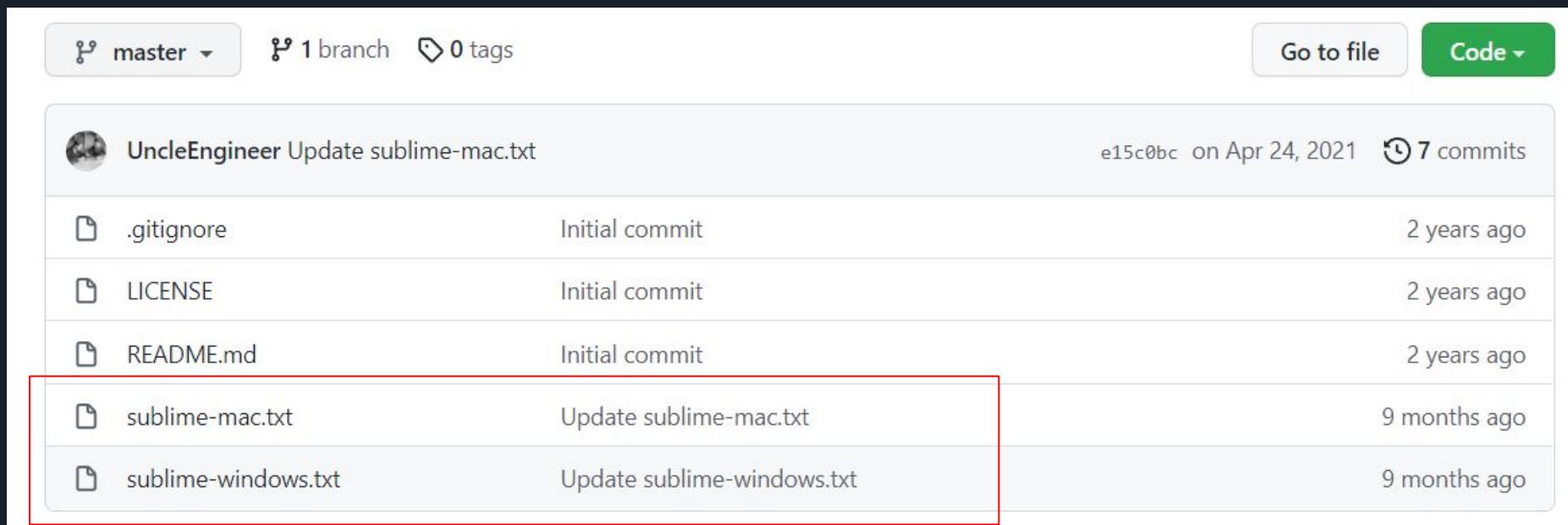


# หน้าตาโปรแกรม



# Setting Sublime ไปที่ลิ้ง

<https://github.com/UncleEngineer/sublime> คลิกเลือก OS ของตัวเอง



The screenshot shows a GitHub repository page for the user 'UncleEngineer' with the repository name 'sublime'. The 'master' branch is selected, indicated by a dropdown menu. There is 1 branch and 0 tags. On the right, there are 'Go to file' and 'Code' buttons. The main area displays a list of commits:

File	Commit Message	Date
.gitignore	Initial commit	2 years ago
LICENSE	Initial commit	2 years ago
README.md	Initial commit	2 years ago
sublime-mac.txt	Update sublime-mac.txt	9 months ago
sublime-windows.txt	Update sublime-windows.txt	9 months ago

A red box highlights the last two commits: 'sublime-mac.txt' and 'sublime-windows.txt'.

# Copy ข้อความที่ได้จากลิ้ง

UncleEngineer Update sublime-windows.txt

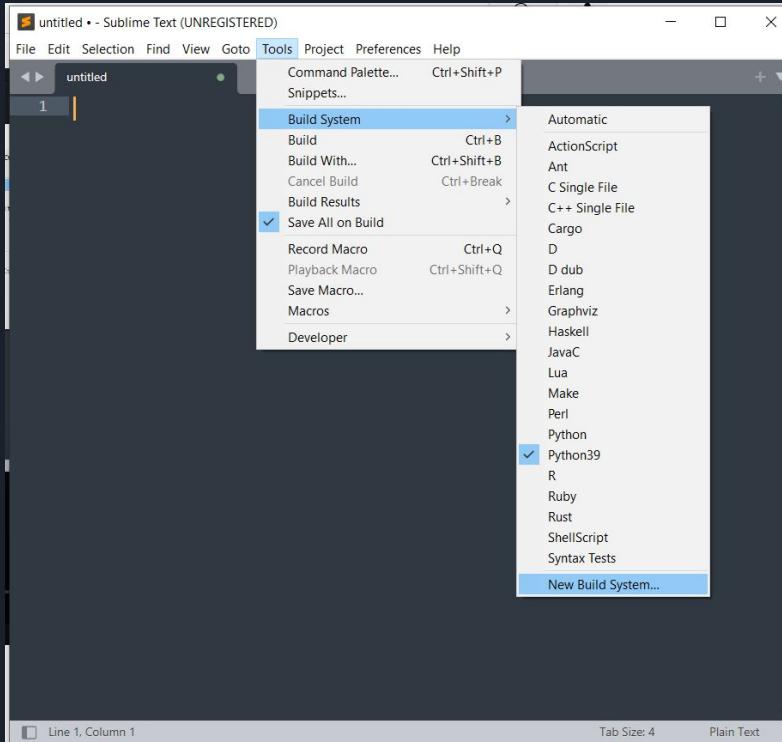
1 contributor

copy ข้อความในนี้

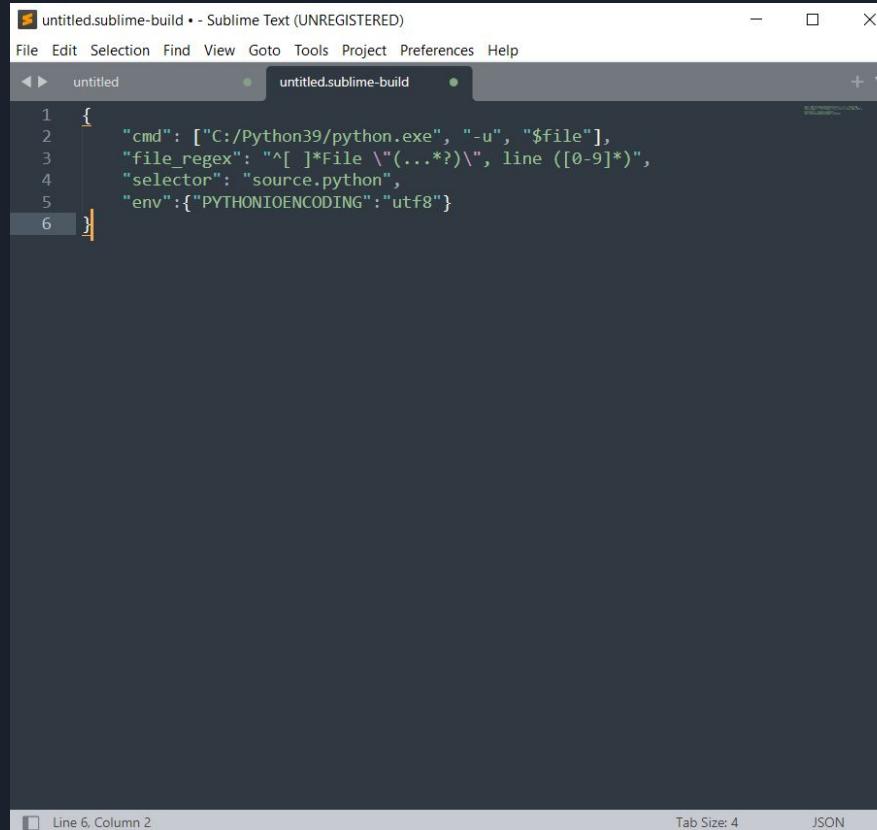
6 lines (6 sloc) | 187 Bytes

```
1 {  
2     "cmd": ["C:/Python39/python.exe", "-u", "$file"],  
3     "file_regex": "^[ ]*File \"(.*)\"", line ([0-9]*),  
4     "selector": "source.python",  
5     "env": {"PYTHONIOENCODING": "utf8"}  
6 }
```

มาที่โปรแกรม Sunlime ไปที่ Tools > Build System > New Build System

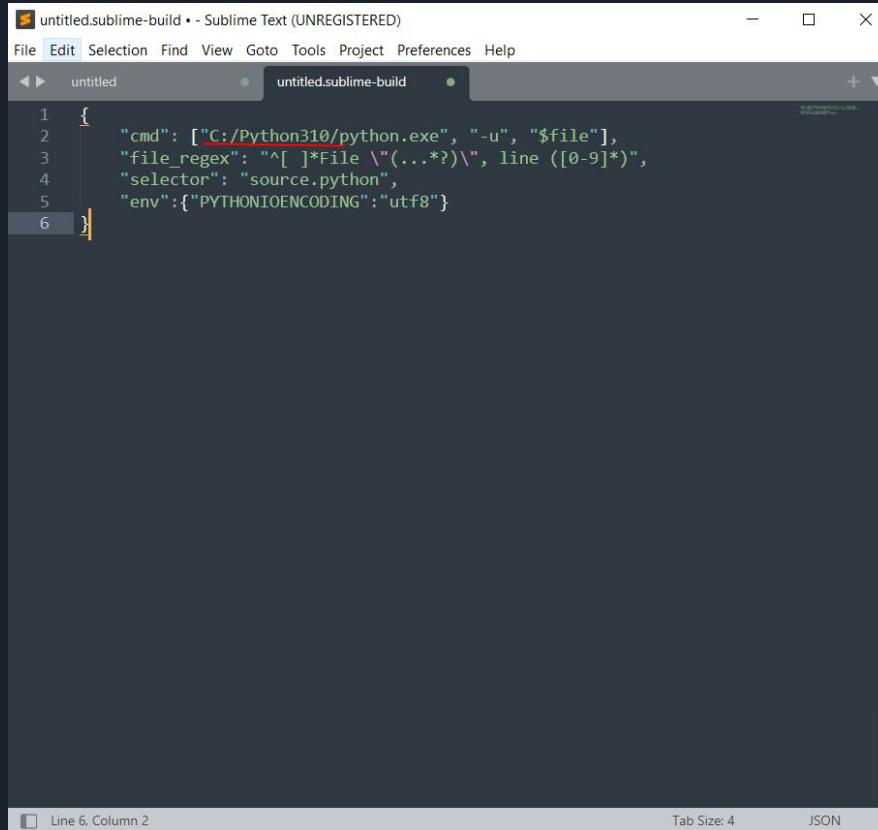


# เอาที่ Copy เมื่อสักครู่ มาใส่ทับทั้งหมด



```
1 {  
2     "cmd": ["C:/Python39/python.exe", "-u", "$file"],  
3     "file_regex": "[ ]*File \"(...*)\"", line ([0-9]*),  
4     "selector": "source.python",  
5     "env": {"PYTHONIOENCODING": "utf8"}  
6 }
```

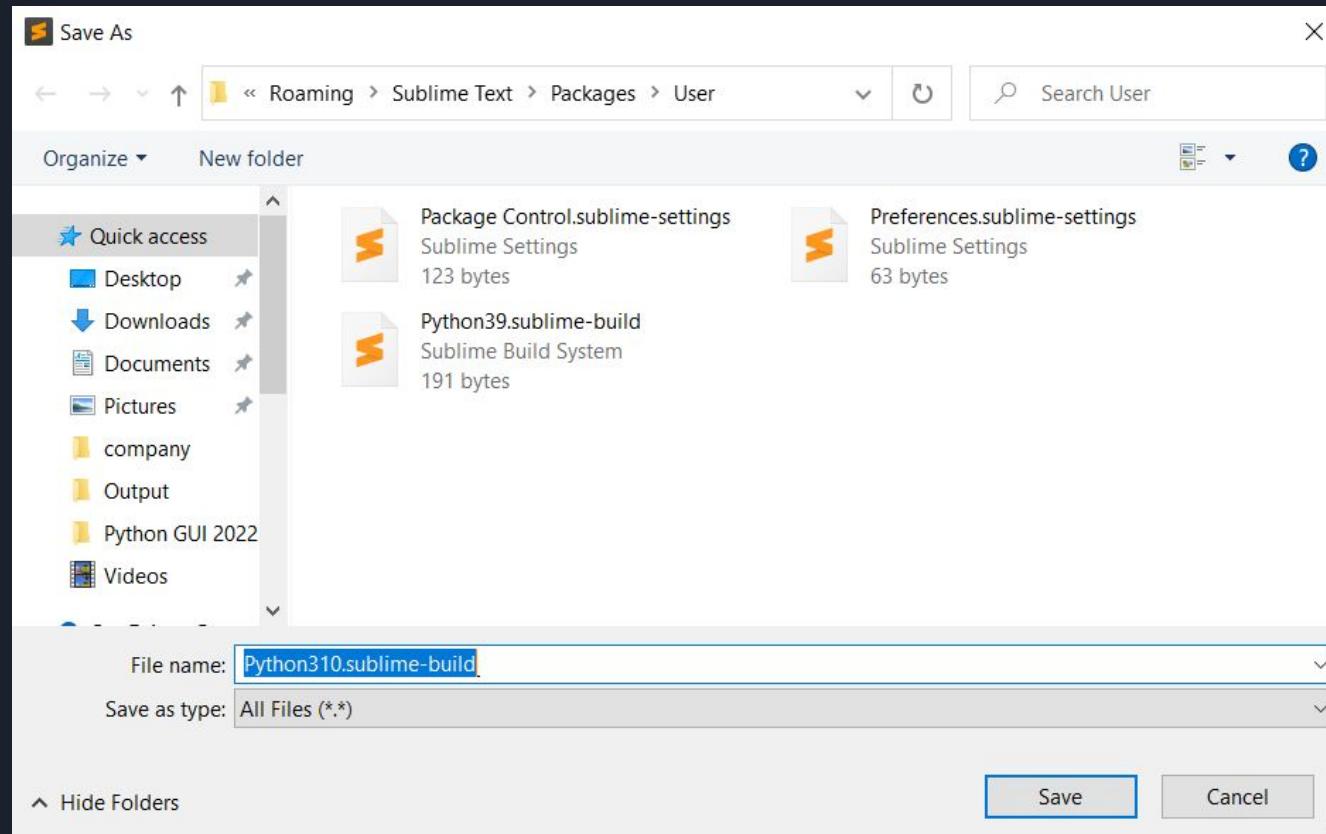
# เปลี่ยนเวอร์ชัน Python เป็นเวอร์ชันที่ตัวเองใช้ ในที่นี่เป็นเวอร์ชัน 3.10



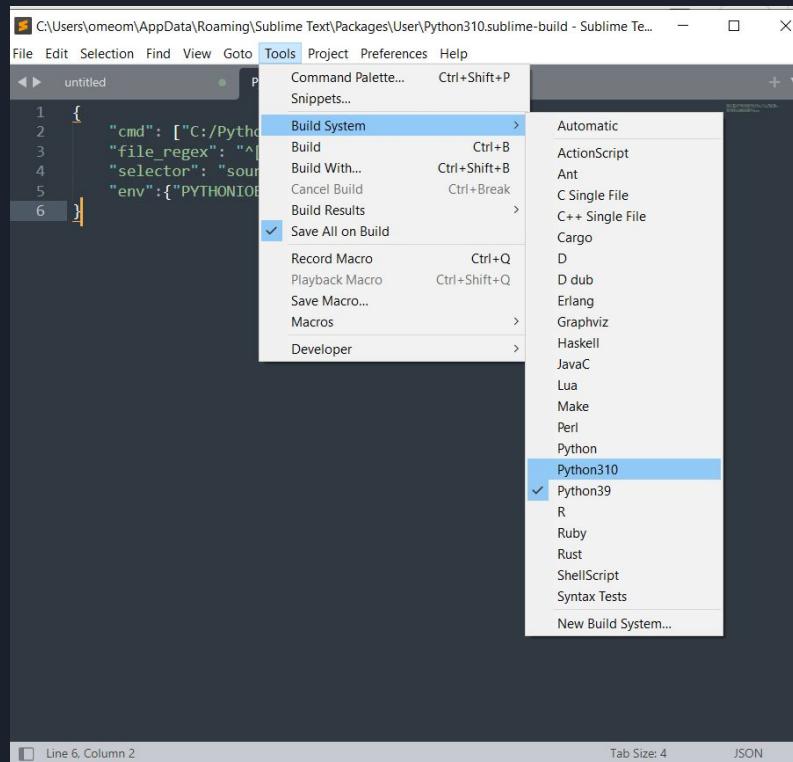
```
1 {
2     "cmd": ["C:/Python310/python.exe", "-u", "$file"],
3     "file_regex": "^[ ]*File \"(.*)\"", line ([0-9]*),
4     "selector": "source.python",
5     "env": {"PYTHONIOENCODING": "utf8"}
6 }
```

Line 6, Column 2      Tab Size: 4      JSON

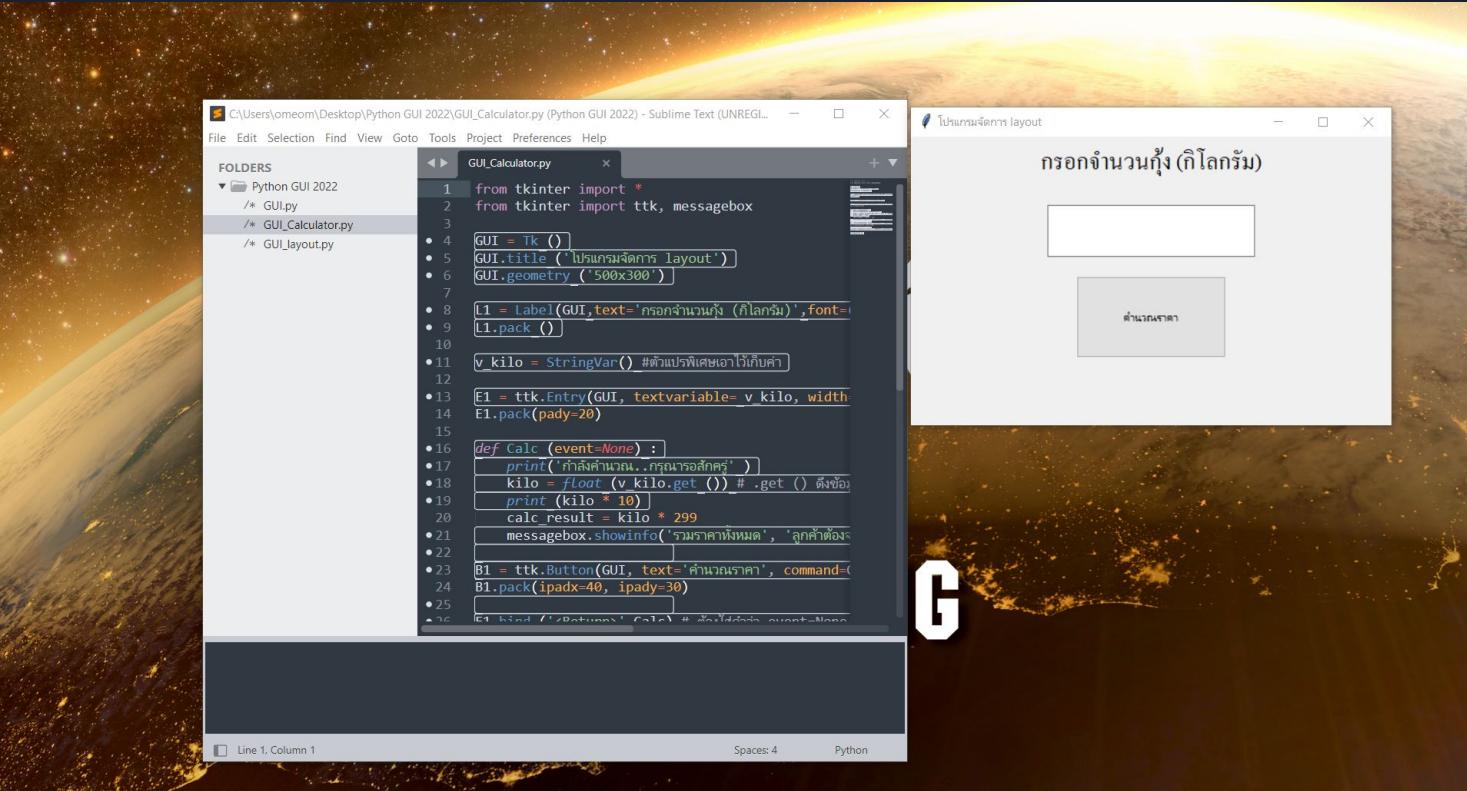
# กด Save ห้ามเปลี่ยน Locations ตั้งชื่อ Python310.sublime-build



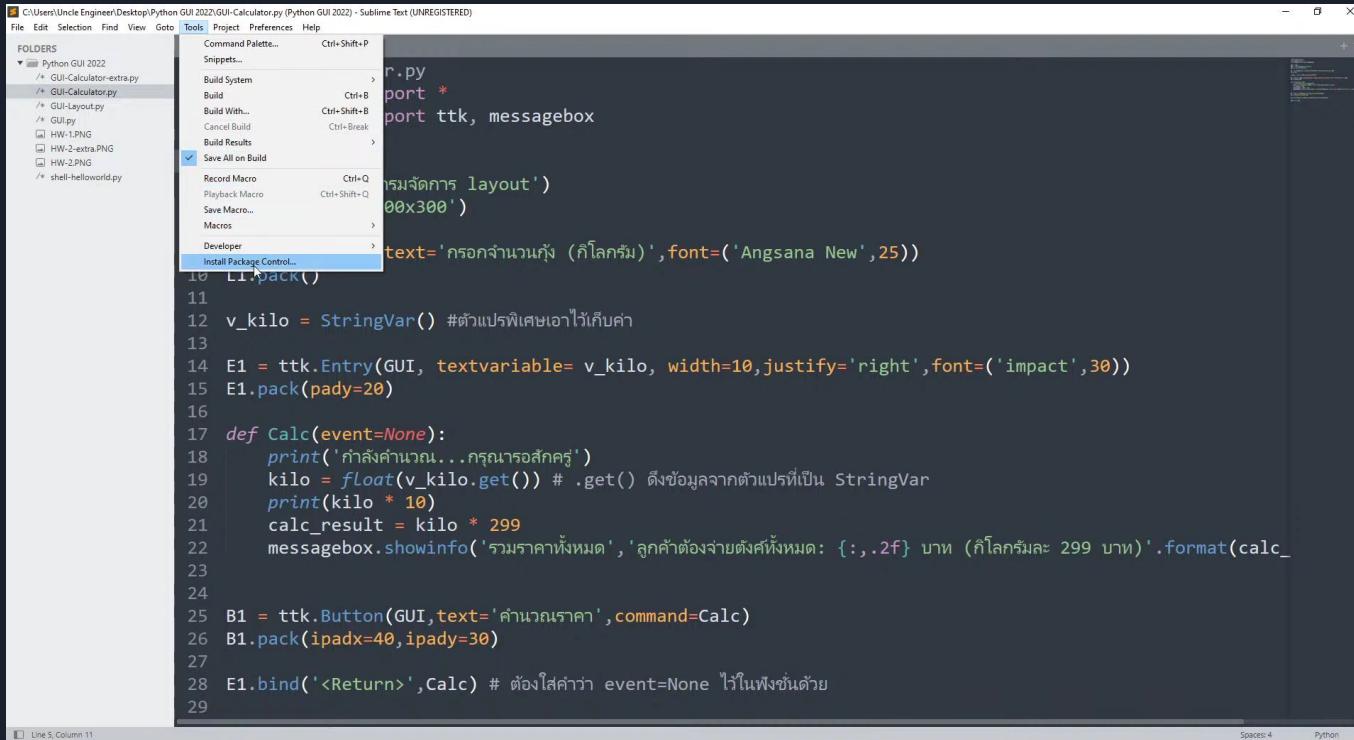
Open Folder ไฟล์ Python ที่เราเขียนมาแล้วไปที่ Tools > build System เลือก Python310



# ลองรันโปรแกรมด้วยการกด Ctrl+B

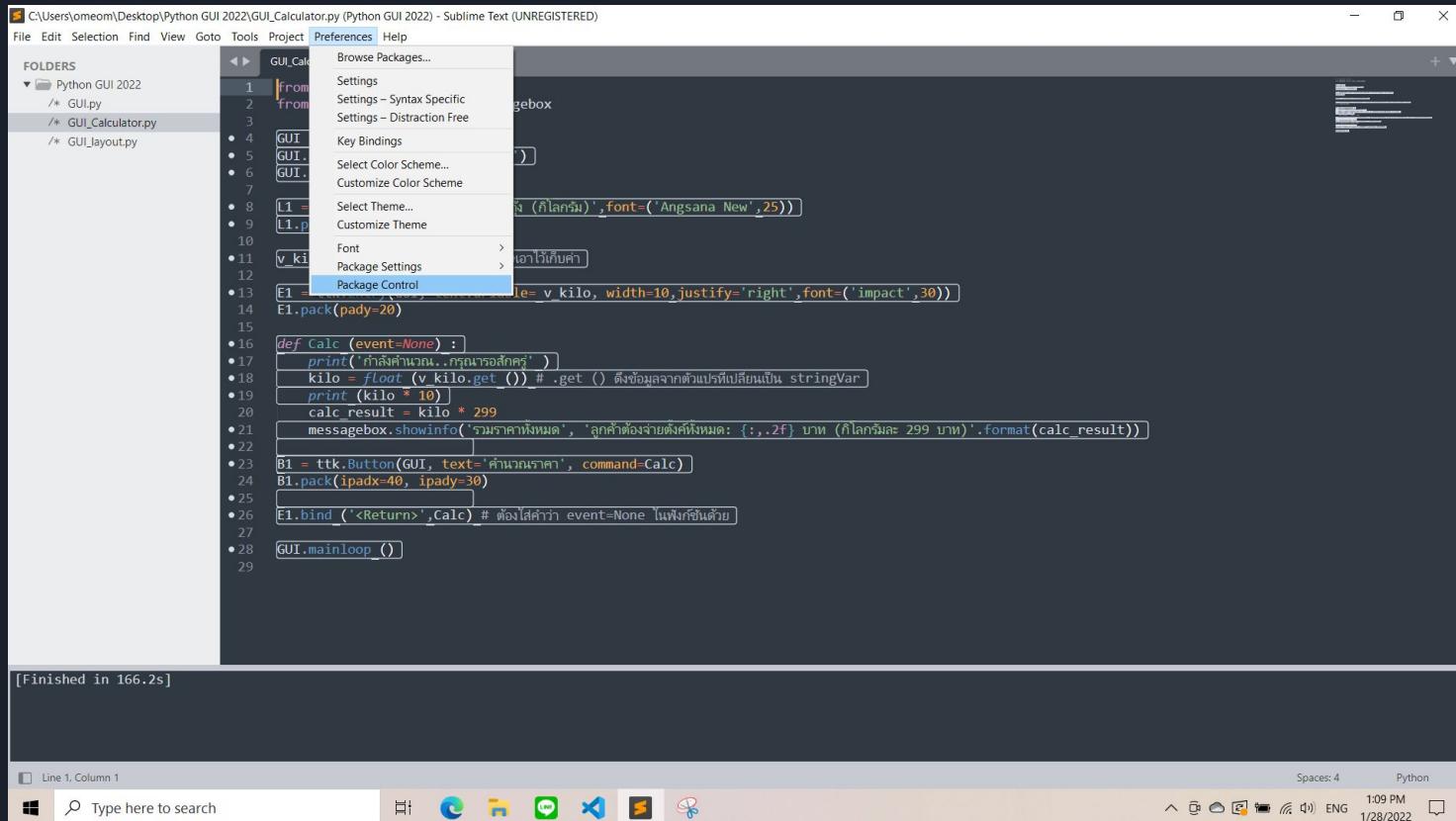


# ติดตั้ง Anaconda ให้ Sublime Text ไปที่ Tools เลือก Install Package Control



```
Line 5, Column 11
Spaces: 4 Python
C:\Users\Uncle Engineer\Desktop\Python GUI 2022.GUI-Calculator.py (Python GUI 2022) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
  Python GUI 2022
    GUI-Calculator-extra.py
    GUI-Calculator.py
    GUI-Layout.py
    HW-1.PNG
    HW-2.extra.PNG
    HW-2.PNG
    shell-helloworld.py
COMMANDS
  Command Palette... Ctrl+Shift+P
  Snippets...
  Build System
  Build Ctrl+B
  Build With...
  Cancel Build Ctrl+Break
  Build Results
  Save All on Build <checked>
  Record Macro Ctrl+Q
  Playback Macro Ctrl+Shift+Q
  Save Macro...
  Macros
  Developer
  Install Package Control...
  Help
r.py
port *
port ttk, messagebox
  0x300')
text='กรอกจำนวนเงิน (กิโลกรัม)', font=('Angsana New', 25))
  E1.pack()
11
12 v_kilo = StringVar() #ตัวแปรพิเศษเอาไว้เก็บค่า
13
14 E1 = ttk.Entry(GUI, textvariable= v_kilo, width=10,justify='right',font=('impact',30))
15 E1.pack(pady=20)
16
17 def Calc(event=None):
18     print('คำลั่งคำนวน...กรุณารอสักครู่')
19     kilo = float(v_kilo.get()) # .get() ดึงข้อมูลจากตัวแปรที่เป็น StringVar
20     print(kilo * 10)
21     calc_result = kilo * 299
22     messagebox.showinfo('รวมราคารถใหม่', 'ลูกค้าต้องจ่ายตังค์ทั้งหมด: {:.2f} บาท (กิโลกรัมละ 299 บาท)'.format(calc_
23
24
25 B1 = ttk.Button(GUI, text='คำนวนราคา', command=Calc)
26 B1.pack(ipadx=40, ipady=30)
27
28 E1.bind('<Return>',Calc) # ต้องใส่ค่าว่า event=None ไว้ในฟังชันด้วย
29
```

# ไปที่ Preferences เลือก Package Control



C:\Users\omeom\Desktop\Python GUI 2022\GUI\_Calculator.py (Python GUI 2022) - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS

- Python GUI 2022
- /\* GUI.py
- /\* GUI\_Calculator.py
- /\* GUI\_Layout.py

1 from tkinter import \*  
2 from tkinter import messagebox  
3  
4 GUI = Tk()  
5 GUI.title("GUI")  
6 GUI.geometry("300x200")  
7  
8 L1 = Label(GUI, text="กรอกจำนวน กิโลกรัม", font=('Angsana New', 25))  
9 L1.pack(ipady=10)  
10  
11 v\_kilo = StringVar()  
12 E1 = Entry(GUI, textvariable=v\_kilo, width=10, justify='right', font=('impact', 30))  
13 E1.pack(pady=20)  
14  
15  
16 def calc (event=None) :  
17 print('ค่าลักษณะ.. กรุณาใส่ค่า')  
18 kilo = float(v\_kilo.get()) # .get () ตั้งชื่อมาจากตัวแปรที่เปลี่ยนเป็น stringVar  
19 print (kilo \* 10)  
20 calc\_result = kilo \* 299  
21 messagebox.showinfo('รวมราคารถมด', 'ลูกค้าต้องจ่ายเงินค่าน้ำดื่ม: {:.2f} บาท ( กิโลกรัมละ 299 บาท )'.format(calc\_result))  
22  
23 B1 = ttk.Button(GUI, text="คำนวณค่า", command=calc)  
24 B1.pack(ipadx=40, ipady=30)  
25  
26 E1.bind('<Return>', calc) # ต้องใส่ค่าว่าง event=None ในพิมพ์ปุ่มกดด้วย  
27  
28 GUI.mainloop()

[Finished in 166.2s]

Line 1, Column 1

Type here to search

Spaces: 4 Python

1:09 PM 1/28/2022

# พิมพ์คำว่า Install Package เลือกอันแรก

The screenshot shows a Sublime Text window with the following details:

- File Path:** C:\Users\omeom\Desktop\Python GUI 2022\GUI\_Calculator.py (Python GUI 2022) - Sublime Text (UNREGISTERED)
- Code Content:** A Python script for a GUI calculator using Tkinter. It includes a label, an entry field, and a button to calculate the result.
- Search Bar:** The search bar at the top right contains the text "Package Control: install package".
- Autocomplete/Completion:** A dropdown menu is open, listing three suggestions:
  - Package Control: install package
  - Package Control: Install Local Dependency
  - Package Control: Advanced Install Package
- Status Bar:** At the bottom left, it says "[Finished in 166.2s]". At the bottom right, it shows "Spaces: 4" and "Python".

# พิมพ์คำว่า anaconda เลือกอันแรก

The screenshot shows a Sublime Text window with the file `GUI_Calculator.py` open. The code is a Python script for a GUI calculator using Tkinter. As the user types 'anaconda' in the editor, a completion dropdown appears, listing several options:

- Anaconda**: A Python development IDE.
- anaconda.go**: Auto-completion, linting, and IDE features for Go.
- anaconda.php**: PHP code standards checking and complexity/messing detector.
- anaconda.rust**: Auto completion, auto formatting, and linting for Rust.

The main code in `GUI_Calculator.py` is as follows:

```
from tkinter import *
from tkinter import ttk, messagebox

GUI = Tk()
GUI.title('โปรแกรมจัดการ layout')
GUI.geometry('500x300')

L1 = Label(GUI, text='กรอกจำนวนตั้ง ค่า')
L1.pack()

v_kilo = StringVar() #ตัวแปรเพื่อเก็บค่าที่ใช้ใน Entry
E1 = ttk.Entry(GUI, textvariable=v_kilo)
E1.pack(pady=20)

def Calc(event=None):
    print('กดลูกศรซ้าย..กรอกตัวเลขครับ')
    kilo = float(v_kilo.get()) # .get() ดึงข้อมูลจากตัวแปรที่เป็น stringVar
    print(kilo * 10)
    calc_result = kilo * 299
    messagebox.showinfo('รวมราคาทั้งหมด', 'ลูกค้าต้องจ่ายเงินทั้งหมด: {:.2f} บาท (ค่ากิโลละ 299 บาท)'.format(calc_result))

B1 = ttk.Button(GUI, text='คำนวนราคา', command=Calc)
B1.pack(ipadx=40, ipady=30)

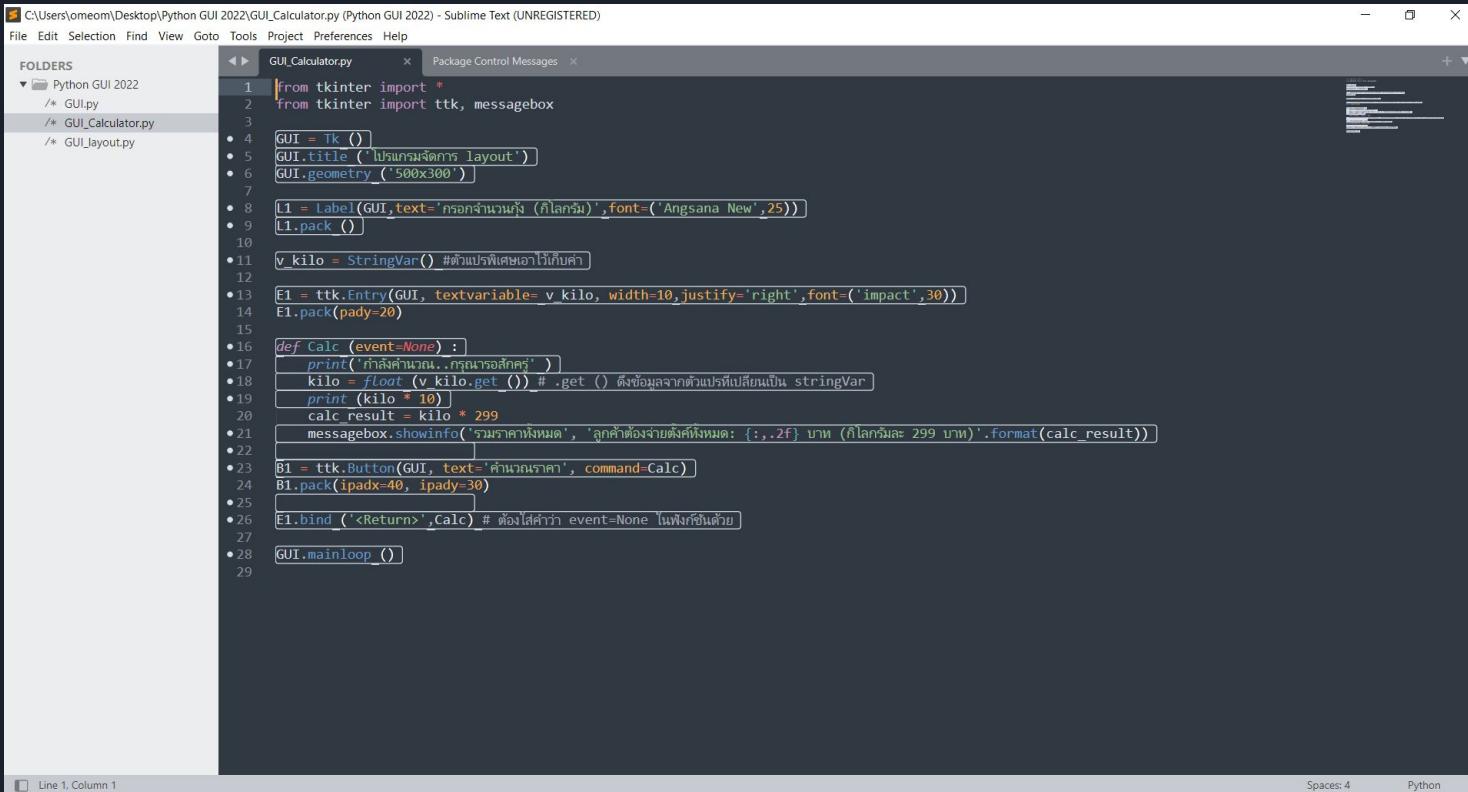
E1.bind('<Return>', Calc) # ต้องใส่คำว่า event=None ให้ฟังก์ชันทำงาน

GUI.mainloop()
```

[Finished in 166.2s]

Sublime Text status bar: Line 1, Column 1, Spaces: 4, Python, 1:11 PM, ENG, 1/28/2022.

# วิธีเอาตัวกรอบสีขาวที่ตัวอักษรออก

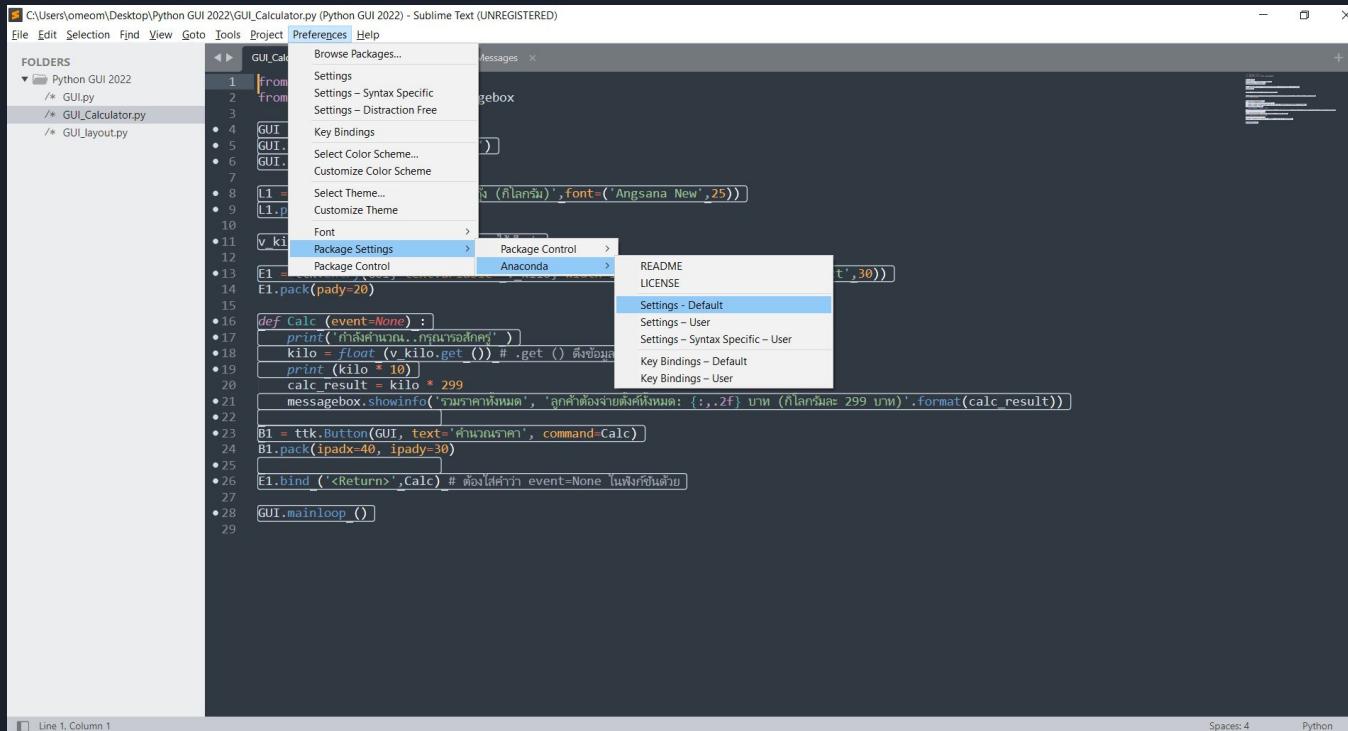


The screenshot shows a Sublime Text window with the following details:

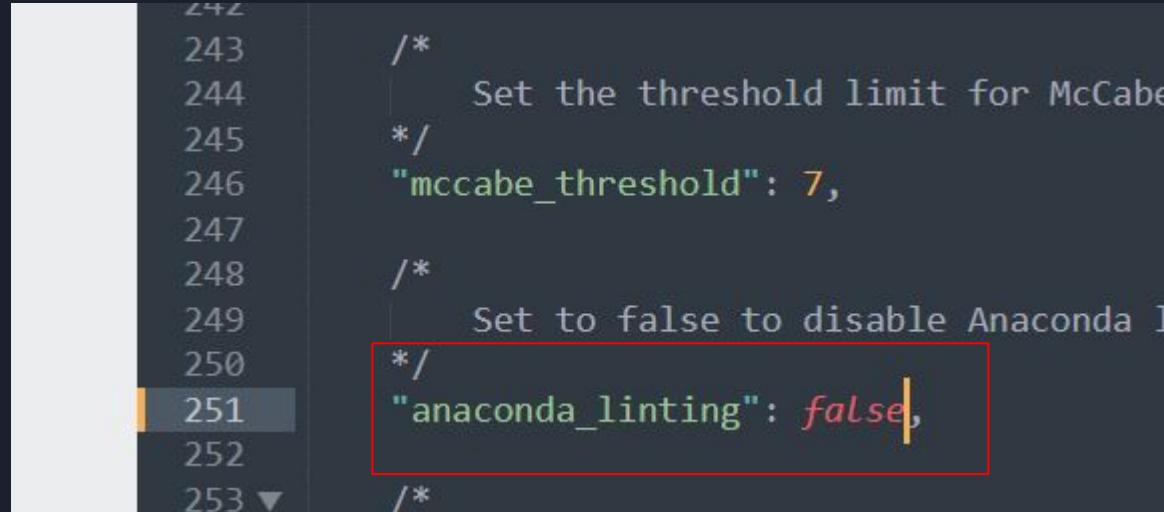
- Title Bar:** C:\Users\omeom\Desktop\Python GUI 2022\GUI\_Calculator.py (Python GUI 2022) - Sublime Text (UNREGISTERED)
- Menu Bar:** File Edit Selection Find View Goto Tools Project Preferences Help
- Folders:** Python GUI 2022
- Code Area:** The code is written in Python using the Tkinter library. It creates a window titled "โปรแกรมหาผลการ layout" with a geometry of 500x300. It contains a label L1 with text "กรอกจำนวนหนั่ง (กิโลกรัม)" and a font of "Angsana New", size 25. A StringVar v\_kilo is defined. An Entry E1 is created with width 10, justify right, and font "impact" size 30. A calculate function Calc is defined to print the input value and calculate its equivalent in pounds (kilo \* 299). A message box shows the result. A button B1 is created with text "คำนวณผล" and command Calc. The entry E1 is bound to the <Return> key to trigger the calculate function. Finally, the main loop is entered.
- Status Bar:** Line 1, Column 1, Spaces: 4, Python

```
from tkinter import *
from tkinter import ttk, messagebox
GUI = Tk()
GUI.title ('โปรแกรมหาผลการ layout')
GUI.geometry ('500x300')
L1 = Label(GUI, text='กรอกจำนวนหนั่ง (กิโลกรัม)', font=('Angsana New', 25))
L1.pack ()
v_kilo = StringVar() #ลักษณะของตัวแปรเป็นค่า
E1 = ttk.Entry(GUI, textvariable= v_kilo, width=10, justify='right', font=('impact', 30))
E1.pack(pady=20)
def Calc (event=None) :
    print('ค่าที่คุณพิมพ์ . คือ %.2f กิโลกรัม' % v_kilo.get())
    kilo = float(v_kilo.get()) # .get () ต้องมุ่งมาจาก Entry ที่ประกาศเป็น stringVar
    print (kilo * 299)
    calc_result = kilo * 299
    messagebox.showinfo('รวมราคารถจักรยานยนต์', 'ลูกค้าต้องจ่ายเงินค่าจักรยานยนต์: {:.2f} บาท (กิโลกรัมละ 299 บาท)'.format(calc_result))
B1 = ttk.Button(GUI, text='คำนวณผล', command=Calc)
B1.pack(ipadx=40, ipady=30)
E1.bind ('<Return>', Calc) # ต้องใส่ค่าว่าง event=None ในไฟล์ชื่อเดียวกัน
GUI.mainloop()
```

# ไปที่ Preferences > Package Setting > Anaconda เลือก Setting Default

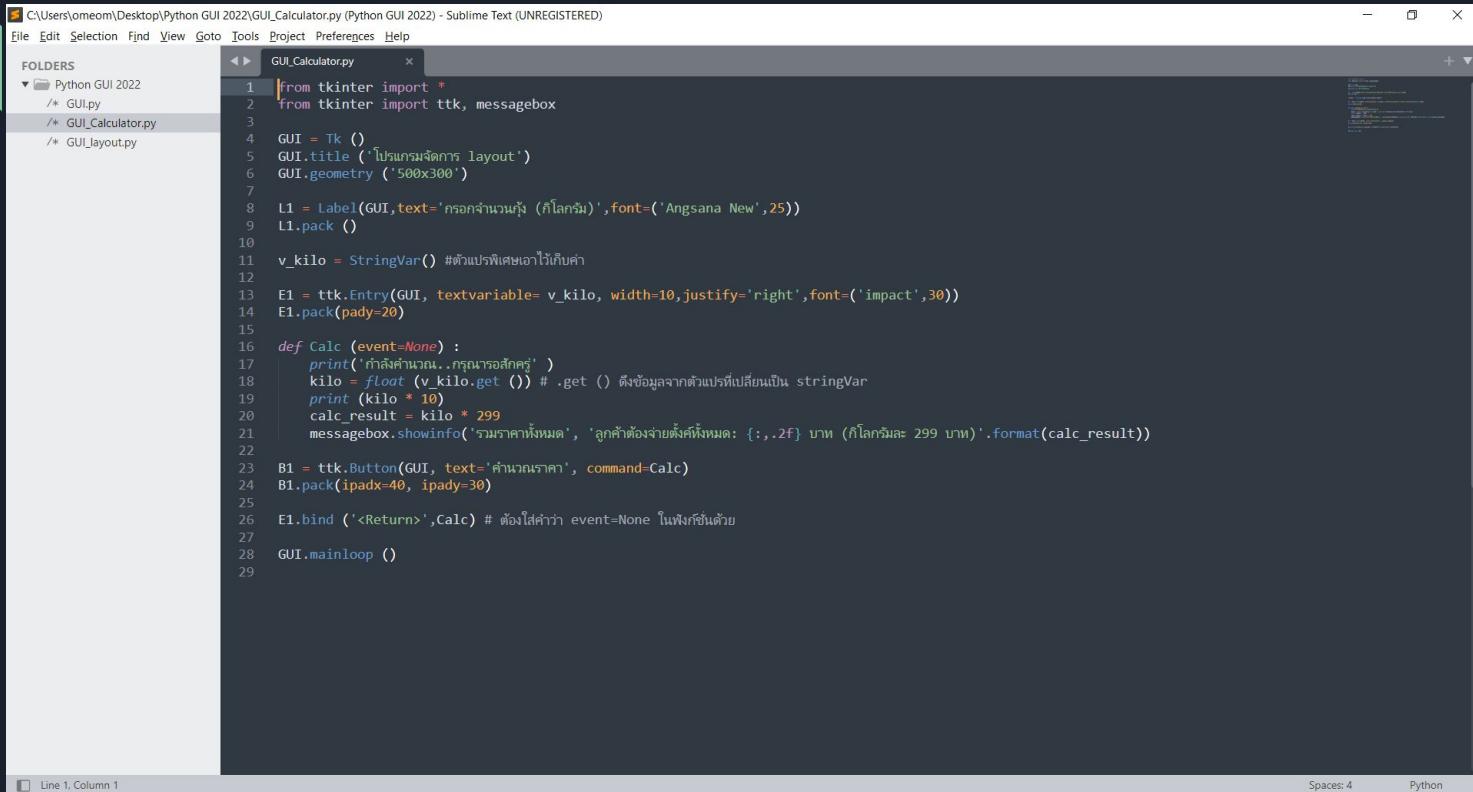


ไปที่บรรทัด 251 ตรงส่วน `anaconda_linting": true`, ให้เราเปลี่ยนจาก `true` เป็น `false` และกด Save



```
243     /*  
244      Set the threshold limit for McCabe  
245      */  
246      "mccabe_threshold": 7,  
247  
248      /*  
249      Set to false to disable Anaconda l  
250      */  
251      "anaconda_linting": false,  
252      /*  
253 ▼
```

# ตัวกรอบลีข้าวจะหายไป



The screenshot shows a Sublime Text window with the following details:

- Title Bar:** C:\Users\omeom\Desktop\Python GUI 2022\GUI\_Calculator.py (Python GUI 2022) - Sublime Text (UNREGISTERED)
- Menu Bar:** File Edit Selection Find View Goto Tools Project Preferences Help
- Folders:** Python GUI 2022 (containing GUI.py, GUI\_Calculator.py, and GUI\_Layout.py)
- Code Editor:** The file GUI\_Calculator.py contains the following Python code:

```
1  from tkinter import *
2  from tkinter import ttk, messagebox
3
4  GUI = Tk()
5  GUI.title('โปรแกรมคำนวณ (กิโลกรัม)')
6  GUI.geometry('500x300')
7
8  L1 = Label(GUI, text='กรอกจำนวน (กิโลกรัม)', font=('Angsana New', 25))
9  L1.pack()
10
11 v_kilo = StringVar() #ตัวแปรพิเศษอ้างอิงกับค่า
12
13 E1 = ttk.Entry(GUI, textvariable=v_kilo, width=10, justify='right', font=('impact', 30))
14 E1.pack(pady=20)
15
16 def Calc(event=None):
17     print('กำลังคำนวณ..กรุณารอสักครู่')
18     kilo = float(v_kilo.get()) # .get() ถึงข้อมูลจากตัวแปรที่เปลี่ยนเป็น stringVar
19     print(kilo * 10)
20     calc_result = kilo * 299
21     messagebox.showinfo('รวมราคาทั้งหมด', 'ลูกค้าต้องจ่ายทั้งค่าห้อง: {:.2f} บาท (กิโลกรัมละ 299 บาท)'.format(calc_result))
22
23 B1 = ttk.Button(GUI, text='คำนวณราคา', command=Calc)
24 B1.pack(ipadx=40, ipady=30)
25
26 E1.bind('<Return>', Calc) # ต้องใส่ค่าว่า event=None ไม่งั้นกดแล้วย
27
28 GUI.mainloop()
29
```

**Status Bar:** Line 1, Column 1 | Spaces: 4 | Python

## List

List ทำหน้าที่เก็บ item หรือว่าเก็บอะไรบางอย่าง ในลักษณะการเก็บจะเป็นการเรียงลำดับแบบชัดเจน ใช้ Index ในการเข้าถึง Index คือลำดับที่อยู่ใน List ตัวนั้นๆ



IDLE Shell 3.10.1

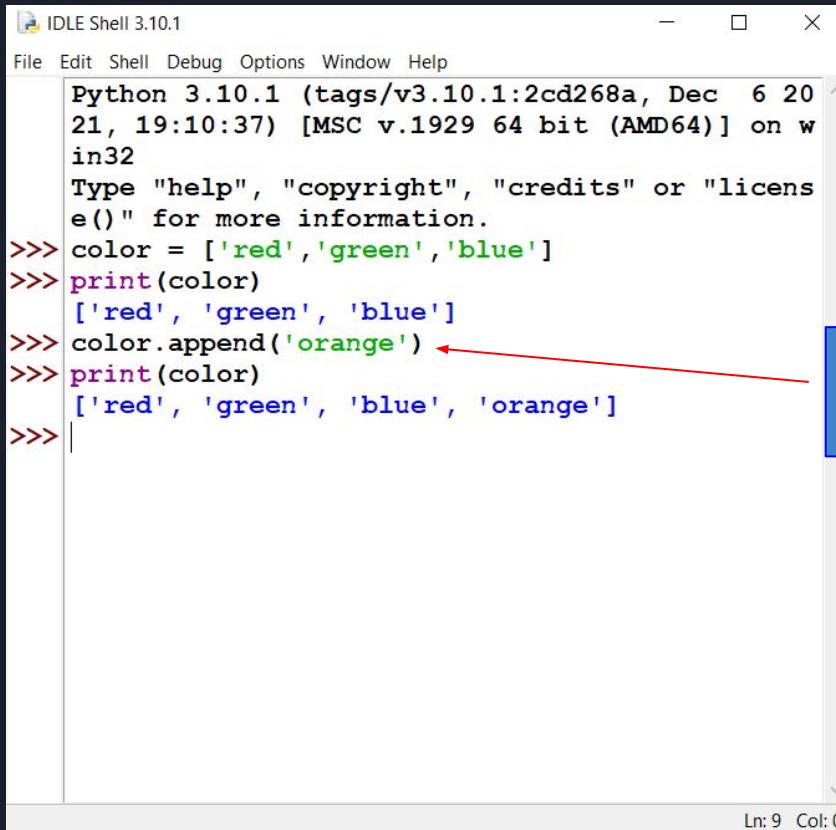
File Edit Shell Debug Options Window Help

```
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec 6 2021, 19:11:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for
more information.

>>> color = ['red', 'green', 'blue']
>>> print(color[1])
green
>>>
```

ใช้ Index ในการเข้าถึง Index เริ่มจาก 0, 1, 2 ... ในที่นี้ Index 1 คือ green หรือ อีก 1 วิธีคือการเรียกแบบ negative คือ -1, -2, -3 ... โดยที่ -1 จะเป็นตัวหลังสุดในที่นี้คือ blue

# เพิ่มค่าบางอย่างเข้าไปใน List โดยใช้ .append

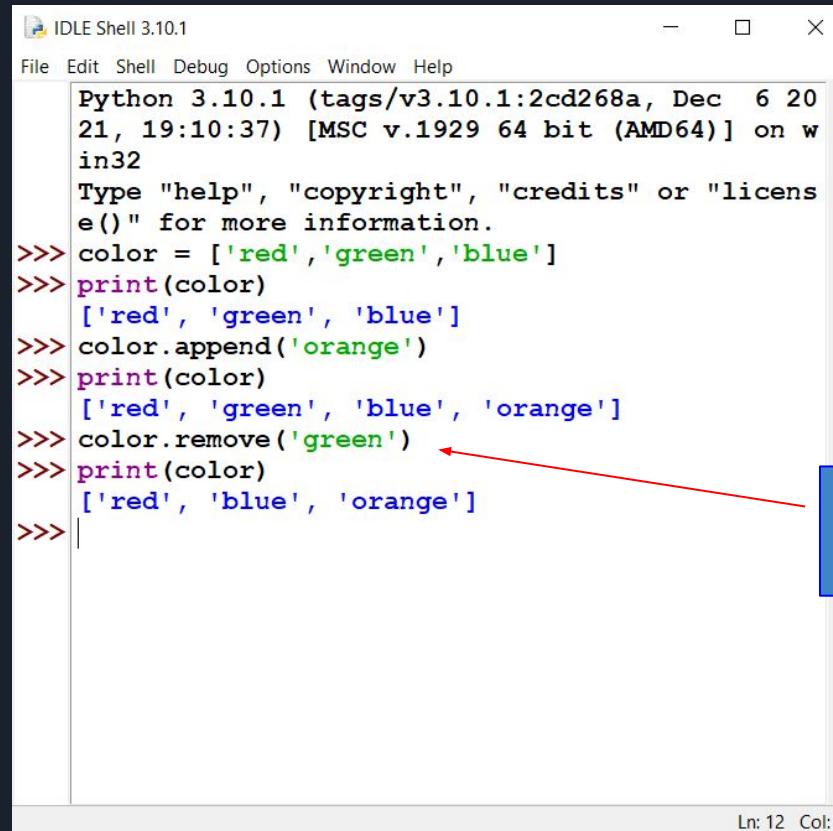


```
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> color = ['red', 'green', 'blue']
>>> print(color)
['red', 'green', 'blue']
>>> color.append('orange') ←
>>> print(color)
['red', 'green', 'blue', 'orange']
>>>
```

The screenshot shows a Python shell window titled "IDLE Shell 3.10.1". The code demonstrates the use of the `append` method to add an element to a list. A red arrow points from the explanatory text to the `append` call in the code.

ใช้ .append เพิ่ม orange  
เข้าไปใน list

# ลบค่าบางอย่างใน List มีหลายวิธี วิธีแรกใช้ .remove ระบุชื่อใน list



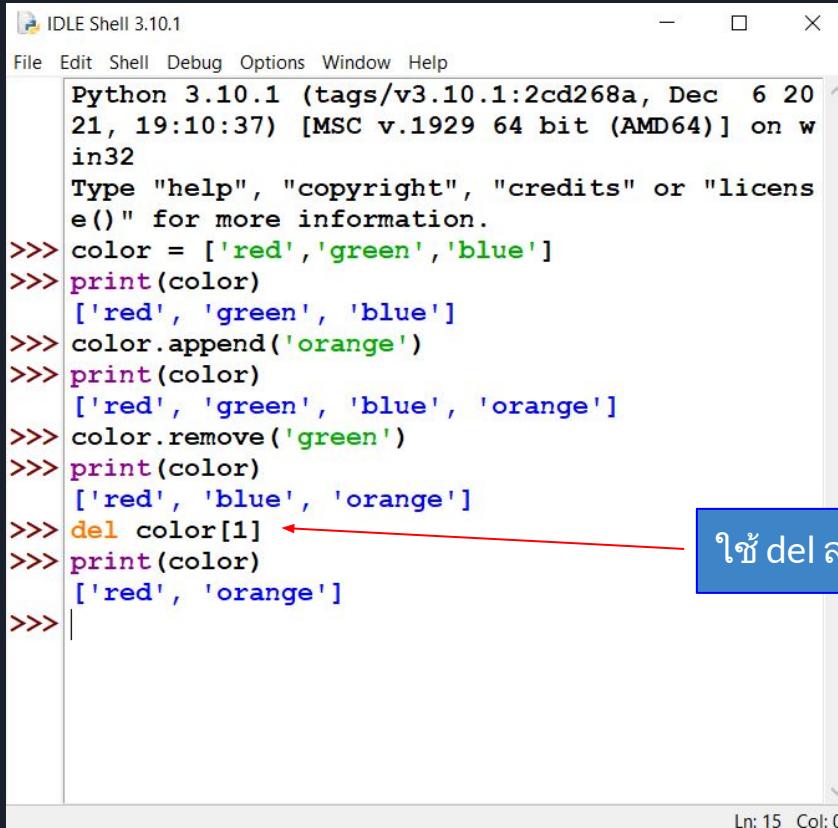
```
IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> color = ['red','green','blue']
>>> print(color)
['red', 'green', 'blue']
>>> color.append('orange')
>>> print(color)
['red', 'green', 'blue', 'orange']
>>> color.remove('green') ←
>>> print(color)
['red', 'blue', 'orange']
>>>
```

Ln: 12 Col: 0

ข้อเสียคือถ้ามีข้อมูลที่ซื้อเดียวกันใน list อาจจะลบข้อมูลผิดอันได้

ใช้ .remove ลบ green ใน list โดยระบุชื่อ

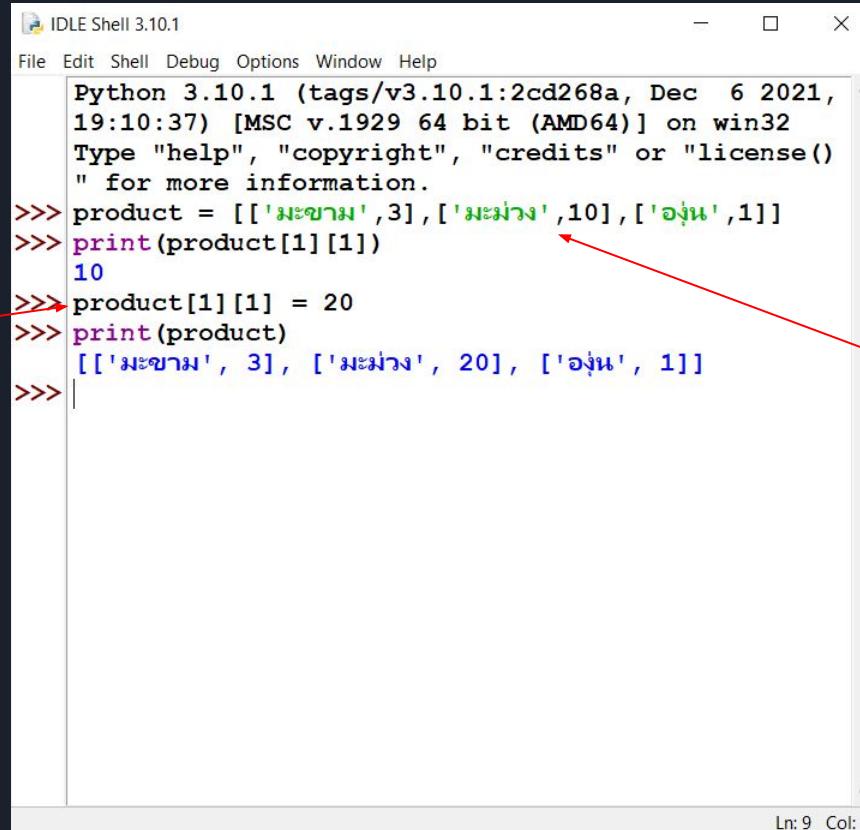
# ลบค่าบางอย่างใน List วิธีที่สองคือการ ระบุ Index โดยใช้ del



```
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> color = ['red', 'green', 'blue']
>>> print(color)
['red', 'green', 'blue']
>>> color.append('orange')
>>> print(color)
['red', 'green', 'blue', 'orange']
>>> color.remove('green')
>>> print(color)
['red', 'blue', 'orange']
>>> del color[1] ←
>>> print(color)
['red', 'orange']
>>>
```

ใช้ del ลบ Index ที่ 1 ในที่นี่คือ blue

# การเปลี่ยนข้อมูลใน List



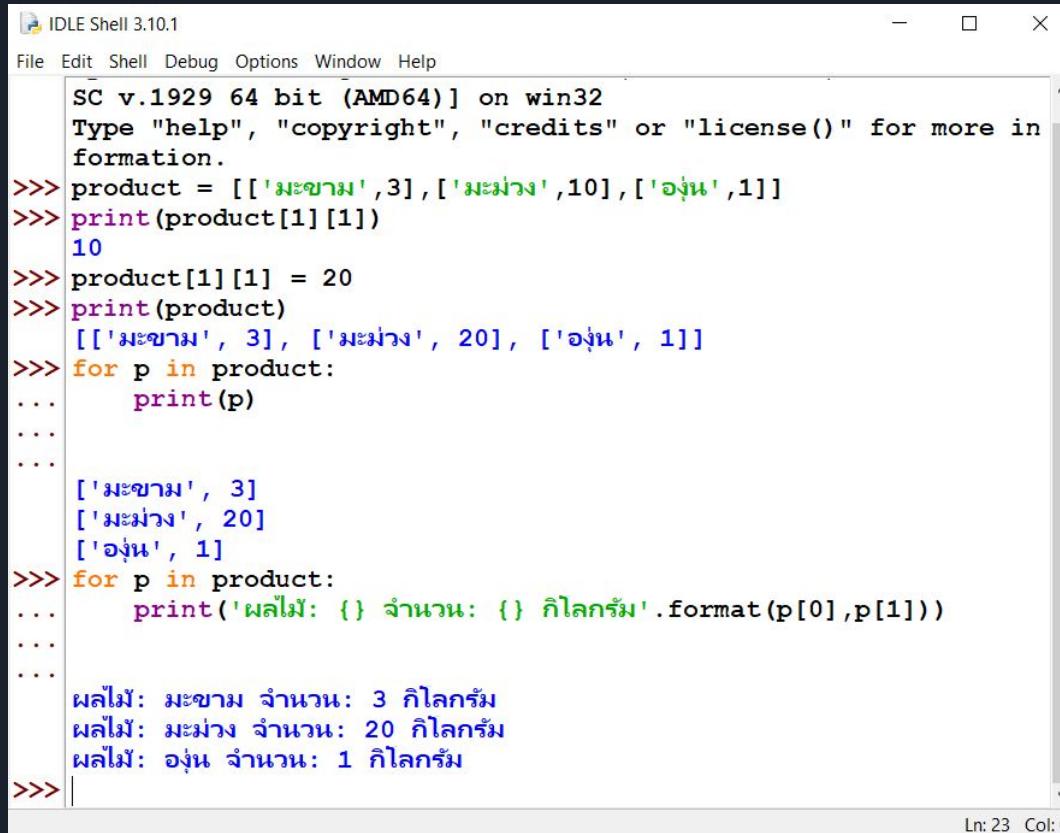
```
IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021,
19:10:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()"
" for more information.
>>> product = [['มะเข้าม',3],['มะม่วง',10],['อุ่น',1]]
>>> print(product[1][1])
10
>>> product[1][1] = 20
>>> print(product)
[['มะเข้าม', 3], ['มะม่วง', 20], ['อุ่น', 1]]
>>>
```

The screenshot shows a Python shell window titled "IDLE Shell 3.10.1". The code demonstrates how to change a value in a nested list. It first defines a list "product" with three items: ['มะเข้าม', 3], ['มะม่วง', 10], and ['อุ่น', 1]. It then prints the second item's second value (Index 1 of the list at index 1), which is 10. Finally, it changes the value at index 1 of the list at index 1 to 20, and prints the list again, showing the updated value.

ทำการเปลี่ยนข้อมูล  
Index 1 ใน list  
product และ Index 1  
ใน list มะม่วง ก็คือ 10  
เปลี่ยนเป็น20

ตัวนี้คือ List ซ่อน List  
มะม่วงคือ Index1 และเลข  
10 ก็คือ Index 1 ใน List  
มะม่วงอีกที

# การใช้ for loop



```
IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
SC v.1929 64 bit (AMD64) ] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> product = [[['มะขาม',3],['มะม่วง',10],['อุ่น',1]]
>>> print(product[1][1])
10
>>> product[1][1] = 20
>>> print(product)
[['มะขาม', 3], ['มะม่วง', 20], ['อุ่น', 1]]
>>> for p in product:
...     print(p)
...
...
...
['มะขาม', 3]
['มะม่วง', 20]
['อุ่น', 1]
>>> for p in product:
...     print('ผลไม้: {} จำนวน: {} กิโลกรัม'.format(p[0],p[1]))
...
...
...
ผลไม้: มะขาม จำนวน: 3 กิโลกรัม
ผลไม้: มะม่วง จำนวน: 20 กิโลกรัม
ผลไม้: อุ่น จำนวน: 1 กิโลกรัม
>>> |
```

The screenshot shows the Python IDLE Shell interface with the following session:

- System information: SC v.1929 64 bit (AMD64) ] on win32
- Type "help", "copyright", "credits" or "license()" for more information.
- Product list assignment:

```
>>> product = [[['มะขาม',3],['มะม่วง',10],['อุ่น',1]]
```
- Print first item's quantity:

```
>>> print(product[1][1])  
10
```
- Change second item's quantity:

```
>>> product[1][1] = 20
```
- Print updated product list:

```
>>> print(product)  
[['มะขาม', 3], ['มะม่วง', 20], ['อุ่น', 1]]
```
- Loop through the product list and print each item:

```
>>> for p in product:  
...     print(p)  
...  
...  
...  
['มะขาม', 3]  
['มะม่วง', 20]  
['อุ่น', 1]
```
- Loop through the product list and print fruit name and quantity using string.format:

```
>>> for p in product:  
...     print('ผลไม้: {} จำนวน: {} กิโลกรัม'.format(p[0],p[1]))  
...  
...  
...  
ผลไม้: มะขาม จำนวน: 3 กิโลกรัม  
ผลไม้: มะม่วง จำนวน: 20 กิโลกรัม  
ผลไม้: อุ่น จำนวน: 1 กิโลกรัม
```

## Function

ส่วนของโค้ดหรือโปรแกรมที่ทำงานเพื่อวัตถุประสงค์บางอย่าง ในภาษา Python คุณสามารถสร้างฟังก์ชันของคุณเองเพื่อให้ทำงานที่ต้องการ ในการเขียนโปรแกรมเรามักจะแยกโค้ดที่มีการทำงานเหมือนๆ กันเป็นฟังก์ชันเอาไว้ และเรียกใช้ฟังก์ชันนั้นๆ ซึ่งเป็นแนวคิดของการนำโค้ดกลับมาใช้ใหม่ (Code reuse) นี้เป็นรูปแบบของการประกาศฟังก์ชันในภาษา Python

```
>>> def Hello(name):
...     print('สวัสดี ()  สบายดีไหม?' .format(name))
...
...
...
>>> Hello('สมชาย')
  สวัสดี สมชาย  สบายดีไหม?
>>> |
```

Ln: 17 Col: 0

# Datetime



```
IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:1
0:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for
more information.
>>> from datetime import datetime
>>> datetime.now()
datetime.datetime(2022, 1, 28, 15, 44, 32, 226707)
>>> datetime.now().strftime('%Y-%m-%d %H:%M:%S')
'2022-01-28 15:45:47'
>>> |
```

Ln: 8 Col: 0

# CSV

## ให้สร้างไฟล์ชื่อ csv.py

The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** testcsv.py - Python GUI 2022 - Visual Studio Code.
- Explorer Panel:** Shows a file tree under PYTHON GUI 2022:
  - GUI\_Calculator.py
  - GUI\_layout.py
  - GUI.py
  - testcsv.py
- Code Editor:** The testcsv.py file is open, showing the number "1".
- Bottom Status Bar:** Python 3.10.1 64-bit, 0 0 ▲ 0, Ln 1, Col 1, Spaces: 4, UTF-8, CRLF, Python, search icon, refresh icon.

File Edit Selection View Go Run Terminal Help testcsv.py - Python GUI 2022 - Visual Studio Code

a คือ append คือการเพิ่ม

เขียนโค้ดตามภาพแล้วกด RUN

ทำให้เพิ่มภาษาไทยได้

```
1 # testcsv.py
2 import csv
3 data = ['Apple', 20]
4 with open('data.csv', 'a', newline='',encoding='utf-8') as file:
5     fw = csv.writer(file) # fw = file writer
6     fw.writerow(data)
7
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

```
PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/GUI_layout.py"
PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/GUI_Calculator.py"
PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/GUI_Calculator.py"
PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/testcsv.py"
PS C:\Users\omeom\Desktop\Python GUI 2022>
```

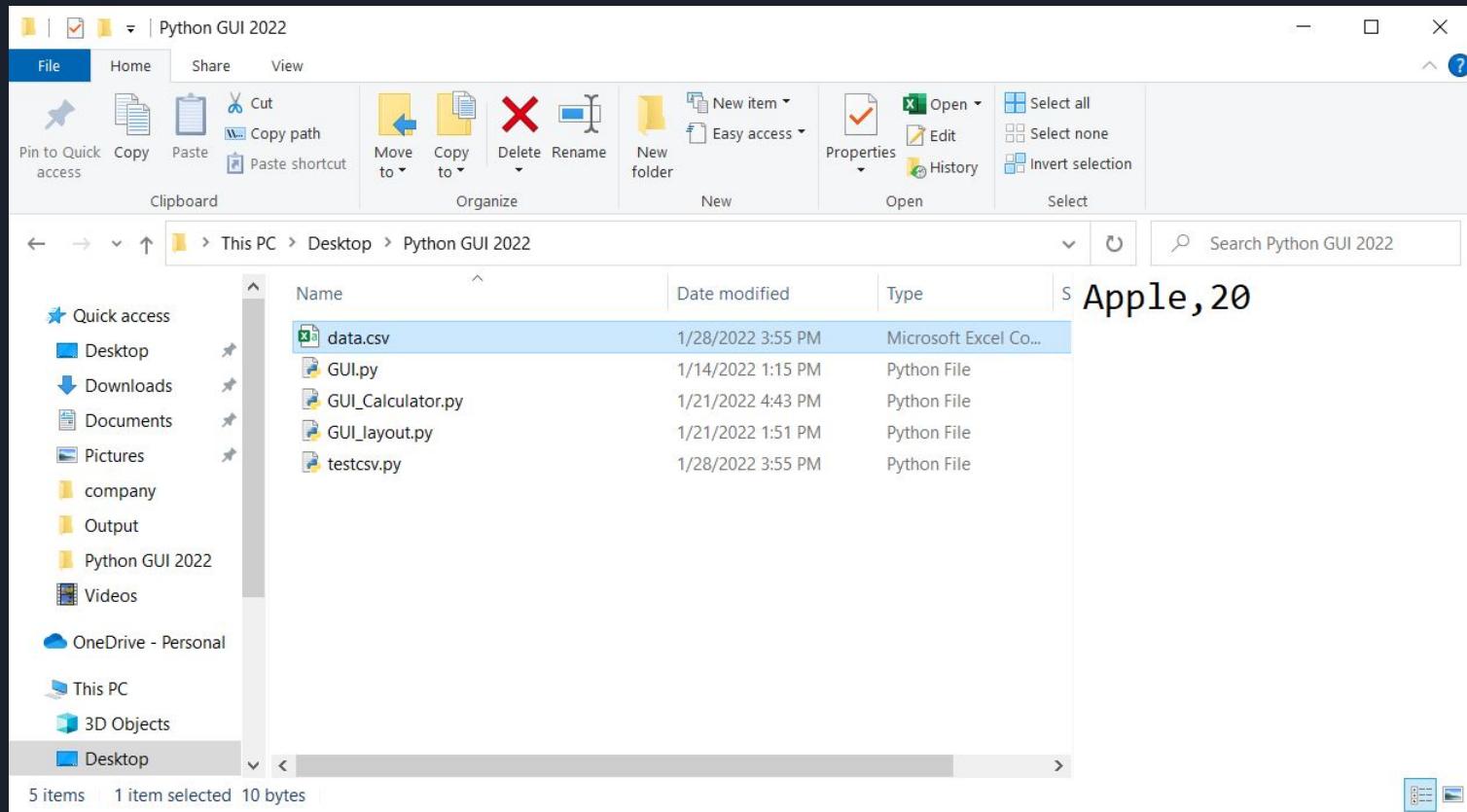
Python 3.10.1 64-bit 0 △ 0

Type here to search

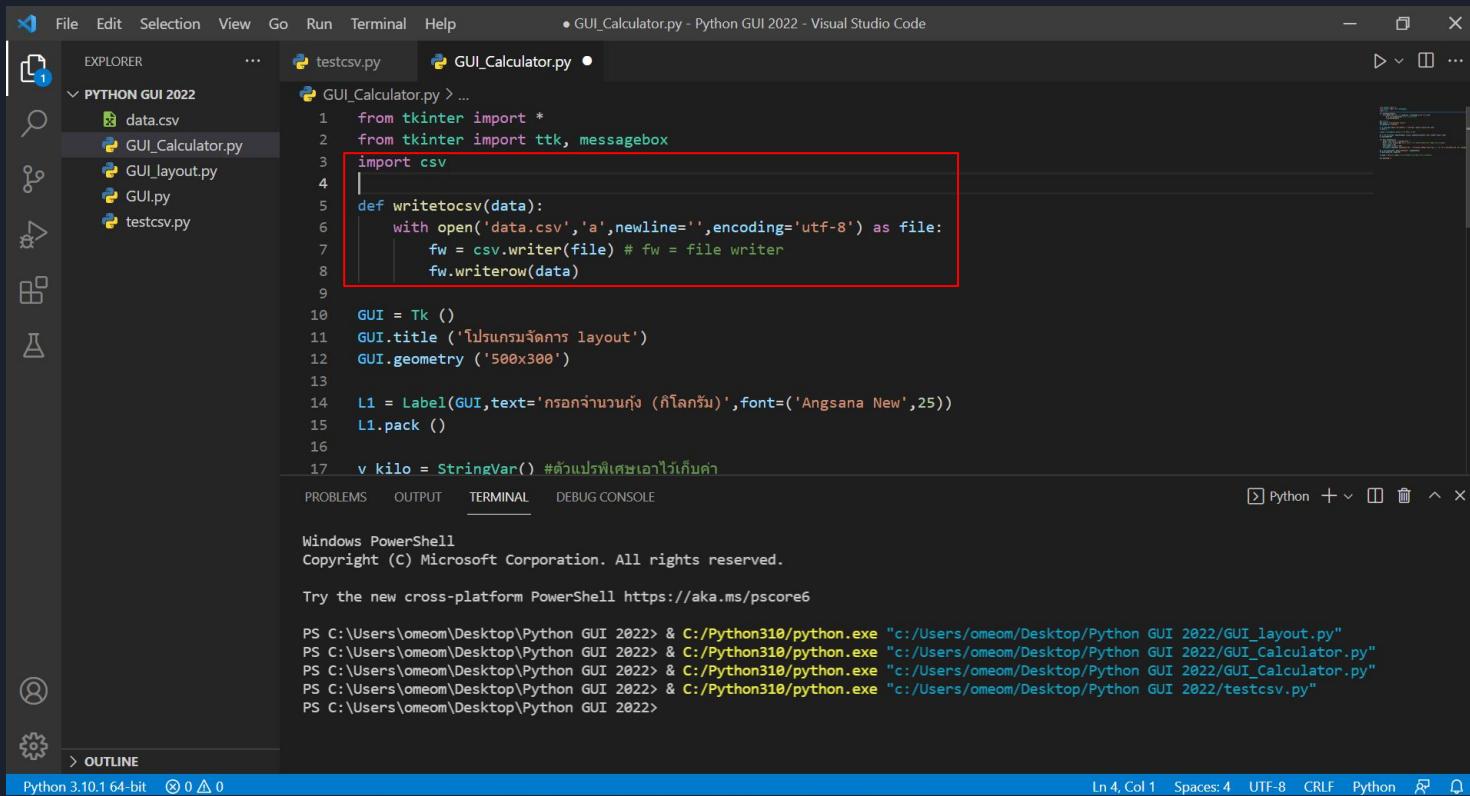
Ln 1, Col 13 Spaces: 4 UTF-8 CRLF Python ⚙️ 🔍

3:56 PM 1/28/2022

# จะได้ไฟล์ .CSV ตามภาพ



# ເອົາໂຄດມາໃສ່ໃນໂຄດ calcuator ດ້ວຍບັນຕາມກາພ



The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows files in the "PYTHON GUI 2022" folder: testcsv.py, GUI\_Calculator.py, GUI\_Layout.py, GUI.py, and testcsv.py. GUI\_Calculator.py is the active file.
- Code Editor:** The code for `GUI_Calculator.py` is displayed. A red box highlights the following code block:

```
import csv
|
def writetocsv(data):
    with open('data.csv','a',newline='',encoding='utf-8') as file:
        fw = csv.writer(file) # fw = file writer
        fw.writerow(data)
```
- Terminal:** Displays command-line output from a Windows PowerShell session:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/GUI_Layout.py"
PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/GUI_Calculator.py"
PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/GUI_Calculator.py"
PS C:\Users\omeom\Desktop\Python GUI 2022> & C:/Python310/python.exe "c:/Users/omeom/Desktop/Python GUI 2022/testcsv.py"
PS C:\Users\omeom\Desktop\Python GUI 2022>
```
- Status Bar:** Shows "Python 3.10.1 64-bit" and other status indicators.
- Bottom Status Bar:** Shows "In 4, Col 1 Spaces: 4 UFT-8 CRLF Python ⚙️ 📌".

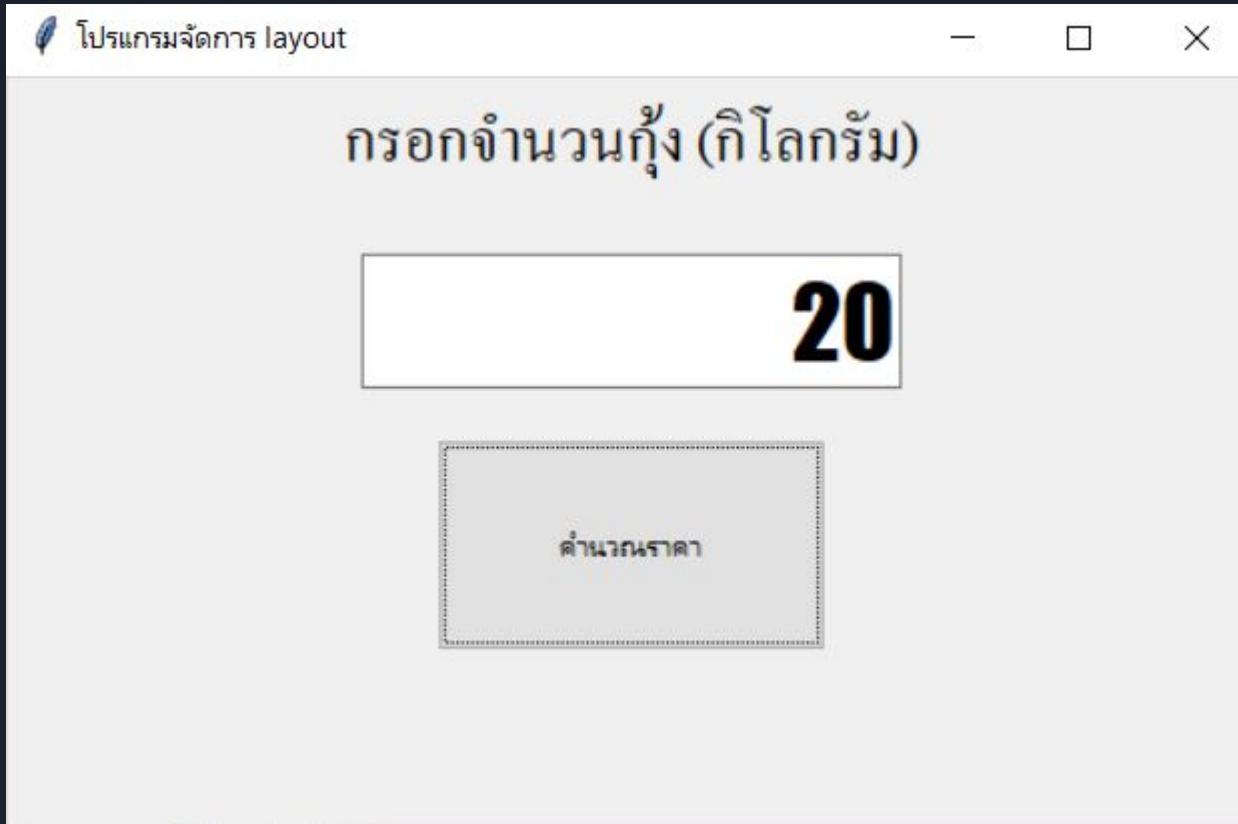
## ໃນຝຶກສ້າງ Calc ເພີ່ມ Code ຕາມການ

```
File Edit Selection View Go Run Terminal Help  
GUI_Calculator.py - Python GUI 2022 - Visual Studio Code  
EXPLORER testcsv.py GUI_Calculator.py X data.csv  
PYTHON GUI 2022 data.csv GUI_Calculator.py X GUI_layout.py GUI.py testcsv.py  
18  
19 E1 = ttk.Entry(GUI, textvariable= v_kilo, width=10,justify='right',font=('impact',30))  
20 E1.pack(pady=20)  
21  
22 def Calc (event=None) :  
23     print('ກໍລັງຄ່ານາວໂລກຈະອໍາລັດ')  
24     kilo = float (v_kilo.get ()) # .get () ດຶງຂໍ້ມູນຈາກຕົວແປຣທີ່ເປົ້າຍືນເປັນ stringVar  
25     print (kilo * 10)  
26     calc_result = kilo * 299  
27     data = ['ຖິ່ງ', '{:.2f}'.format(calc_result)]  
28     writetocsv(data)  
29     messagebox.showinfo('ຮ້າມຄາທັງໝາດ', 'ລູກຄ້າດັ່ງຕ້ອງຈ່າຍຕັ້ງຄ່າທັງໝາດ: {:.2f} ນາທ (ກົໂລກຮັນລະ 299 ນາທ)  
30  
31 B1 = ttk.Button(GUI, text='ຄ່ານາວຄາ', command=Calc)  
32 B1.pack(ipadx=40, ipady=30)  
33  
34 E1.bind ('<Return>',Calc) # ຕ້ອງໃສ່ຄ່າວ່າ event=None ໃນຝຶກຂຶ້ນດ້ວຍ  
35  
36 GUI.mainloop ()  
37
```

Python 3.10.1 64-bit ⚡ 0 ▲ 0

Ln 36, Col 16 Spaces: 4 UTF-8 CRLF Python ⚡ ⚡

# ทดสอบรัน และใส่ข้อมูลกิจิโล



# จะได้ข้อมูลใน CSV ตามภาพ

The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** data.csv - Python GUI 2022 - Visual Studio Code.
- Explorer Panel:** Shows a tree view with a folder named "PYTHON GUI 2022" containing files: data.csv, GUI\_Calculator.py, GUI\_layout.py, GUI.py, and testcsv.py.
- Code Editor:** The "data.csv" file is open, displaying the following content:

```
1 Apple,20
2 ,5980.0
3 กก, "5,980.00"
4
```
- Bottom Status Bar:** Python 3.10.1 64-bit, 0 △ 0, Ln 1, Col 1, Spaces: 4, UTF-8, CRLF, Plain Text, and icons for search, replace, and notifications.



# Python GUI 2022(EP.4)

by



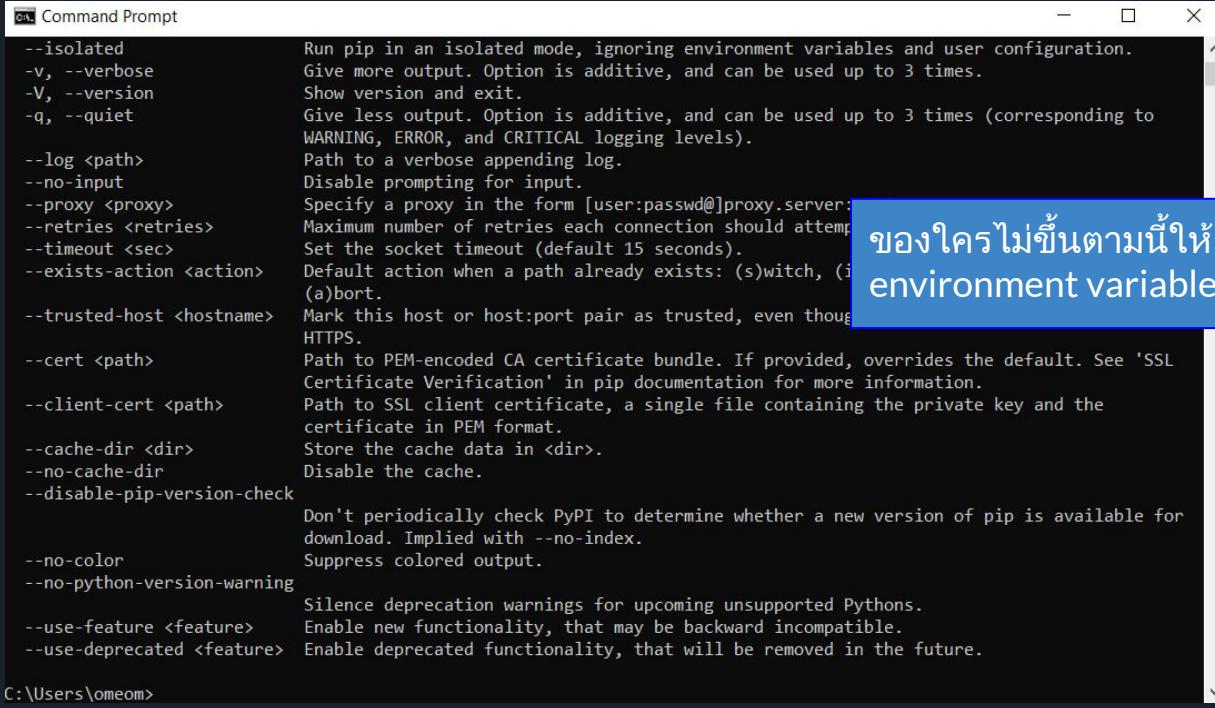
Uncle Engineer

ลุงวิศวกร สอนคำนวณ

# Pip

คือตัวติดตั้ง packages เสริมใน python ซึ่ง packages เสริมพากนี้จะมีคุณสร้างไว้อยู่แล้วใน <https://pypi.org/> เราสามารถนำมาใช้กับโปรเจคของเราได้โดยติดตั้ง packages ต่าง ๆ ผ่าน pip

# วิธีการใช้งาน เปิด cmd ทดลองพิมพ์ว่า pip ถ้าเขียนตามภาพแสดงว่าใช้งานได้



```
Command Prompt
--isolated           Run pip in an isolated mode, ignoring environment variables and user configuration.
-v, --verbose        Give more output. Option is additive, and can be used up to 3 times.
-V, --version        Show version and exit.
-q, --quiet          Give less output. Option is additive, and can be used up to 3 times (corresponding to
                     WARNING, ERROR, and CRITICAL logging levels).
--log <path>         Path to a verbose appending log.
--no-input           Disable prompting for input.
--proxy <proxy>      Specify a proxy in the form [user:passwd@]proxy.server:port
--retries <retries>   Maximum number of retries each connection should attempt.
--timeout <sec>       Set the socket timeout (default 15 seconds).
--exists-action <action> Default action when a path already exists: (i)gnore, (s)witch, (a)bort.
--trusted-host <hostname> Mark this host or host:port pair as trusted, even though it is not in the system's
                           certificate store. This option is useful for HTTPS.
--cert <path>          Path to PEM-encoded CA certificate bundle. If provided, overrides the default. See 'SSL
                           Certificate Verification' in pip documentation for more information.
--client-cert <path>  Path to SSL client certificate, a single file containing the private key and the
                           certificate in PEM format.
--cache-dir <dir>     Store the cache data in <dir>.
--no-cache-dir        Disable the cache.
--disable-pip-version-check
                      Don't periodically check PyPI to determine whether a new version of pip is available for
                      download. Implied with --no-index.
--no-color             Suppress colored output.
--no-python-version-warning
                      Silence deprecation warnings for upcoming unsupported Pythons.
--use-feature <feature> Enable new functionality, that may be backward incompatible.
--use-deprecated <feature> Enable deprecated functionality, that will be removed in the future.

C:\Users\omeom>
```

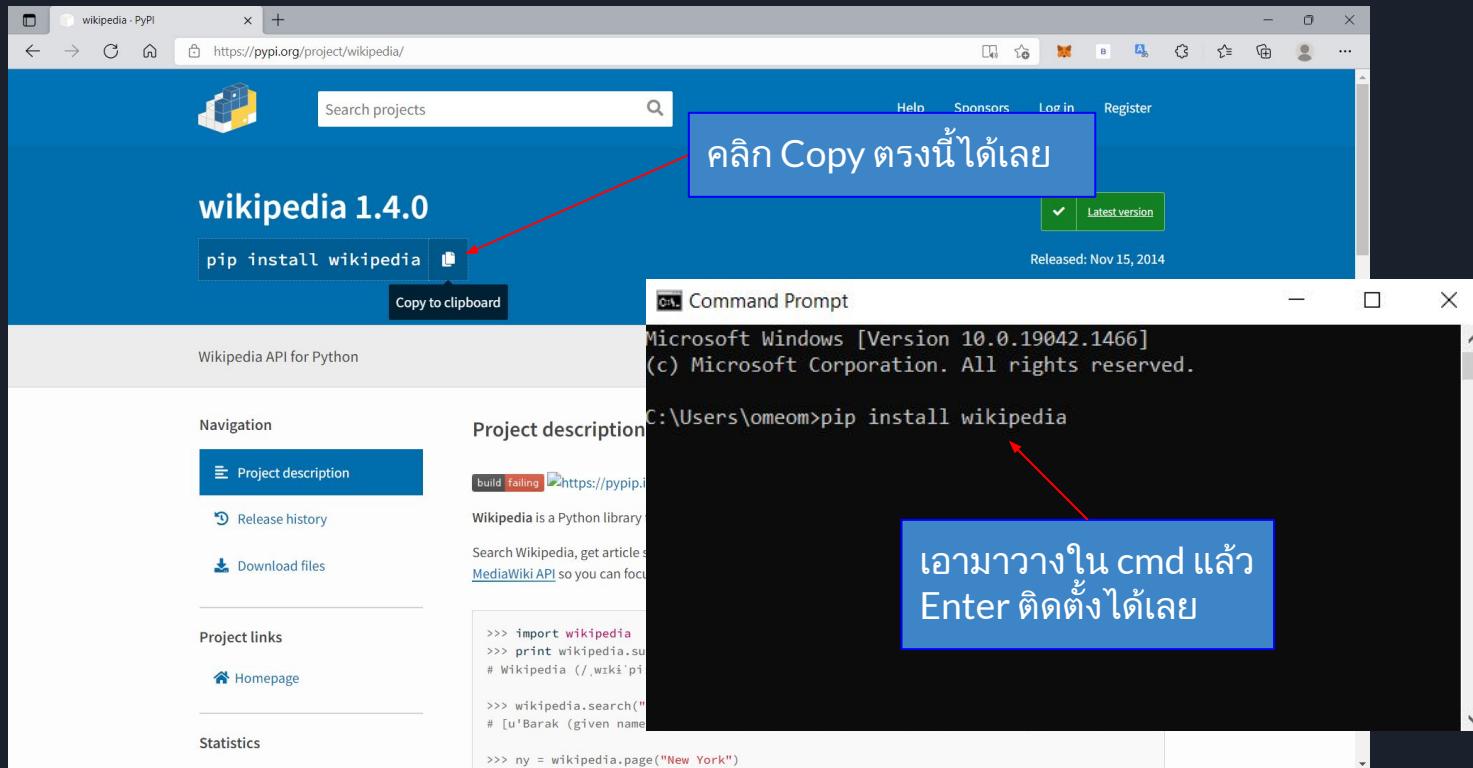
ของใครไม่เขียนตามนี้ให้ set path  
environment variable python

เข้าไปที่เว็บไซต์ <https://pypi.org/> พิมพ์ในช่องค้นหาว่า wikipedia

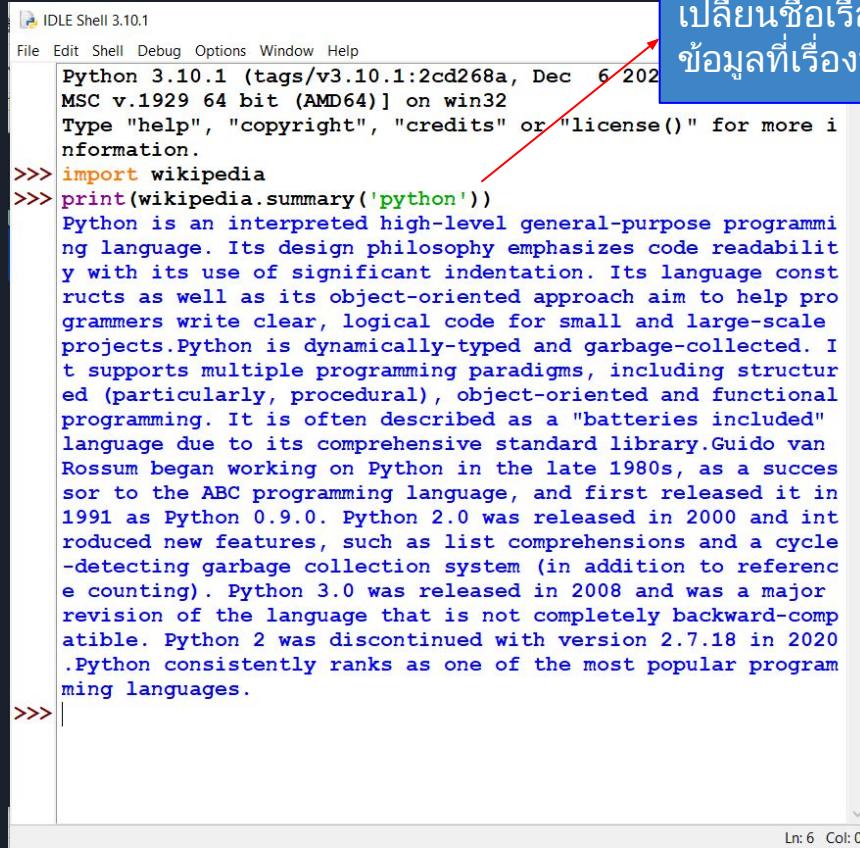
The screenshot shows a web browser window displaying the PyPI search results for the query "wikipedia". The search bar at the top contains "wikipedia". On the left, there is a sidebar titled "Filter by classifier" with several checkboxes: Framework, Topic, Development Status, License, Programming Language, Operating System, Environment, Intended Audience, Natural Language, and Typing. The "Topic" checkbox is checked. The main content area shows a list of 1,456 projects. A red arrow points from the text "คลิกเลือกอันแรก" (Click to select the first one) in a blue banner at the top right towards the first project in the list. The first project listed is "wikipedia 1.4.0", which is described as "Wikipedia API for Python". It was published on Nov 15, 2014. Below it are other projects like "strippers.wikipedia 0.3", "wikipedia2vecSM002583 0.1", "wikipedia-downloader 0.2", "wikipedia-histories 1.0.0", and "wikipedia-sections 2.0.0".

Project Name	Description	Last Published
wikipedia 1.4.0	Wikipedia API for Python	Nov 15, 2014
strippers.wikipedia 0.3	Wikipedia(ja) article parser	Jan 18, 2011
wikipedia2vecSM002583 0.1	A tool for learning vector representations of words and entities from Wikipedia	Apr 16, 2019
wikipedia-downloader 0.2	Download Wikipedia data dumps	Jan 4, 2019
wikipedia-histories 1.0.0	A simple package designed to collect the edit histories of Wikipedia pages	Apr 8, 2021
wikipedia-sections 2.0.0	Wikipedia API for Python extracting sections	Sep 5, 2020

# จะเจอคำสั่ง ให้คลิก Copy และเอาไปใส่ใน cmd



# วิธีใช้



```
IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2022
MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> import wikipedia
>>> print(wikipedia.summary('python'))
Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library. Guido van Rossum began working on Python in the late 1980s, as a successor to the ABC programming language, and first released it in 1991 as Python 0.9.0. Python 2.0 was released in 2000 and introduced new features, such as list comprehensions and a cycle-detecting garbage collection system (in addition to reference counting). Python 3.0 was released in 2008 and was a major revision of the language that is not completely backward-compatible. Python 2 was discontinued with version 2.7.18 in 2020. Python consistently ranks as one of the most popular programming languages.

>>> |
```

เปลี่ยนชื่อเรื่องตรงส่วนนี้เพื่อดึงข้อมูลที่เรื่องที่ต้องการมาแสดง

# วิธีเปลี่ยนให้เป็นภาษาไทย

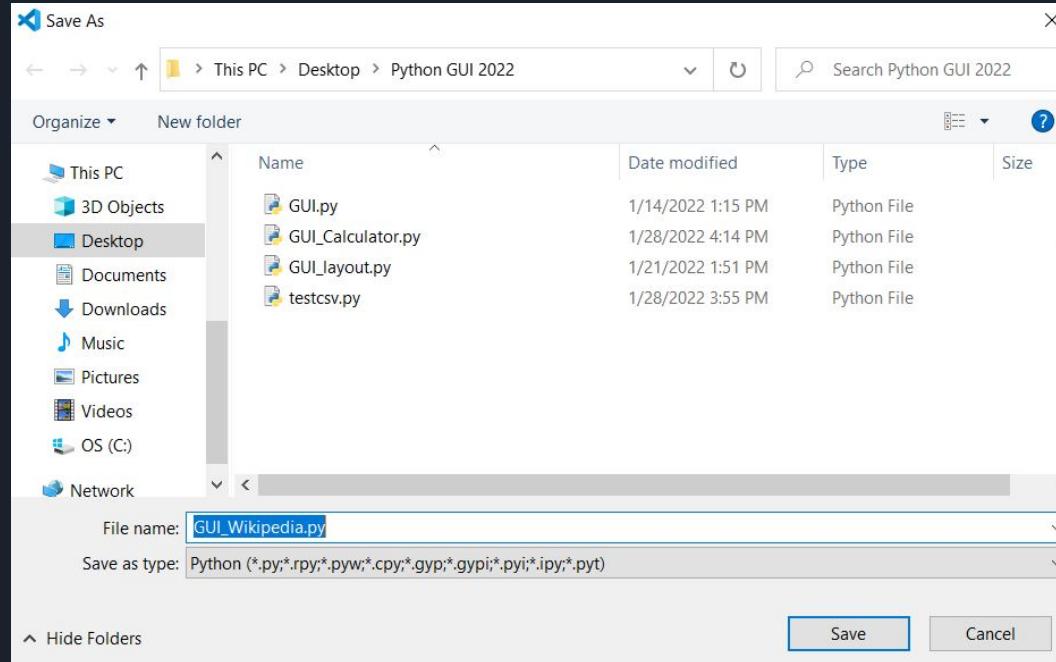
```
IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:10:37) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> import wikipedia
>>> wikipedia.set_lang('th')
>>> wikipedia.summary('กรุงเทพ')
'กรุงเทพมหานคร เป็นเมืองหลวงและนครที่มีประชากรมากที่สุดของประเทศไทย เป็นศูนย์กลางการปกครอง การศึกษา การคมนาคมขนส่ง การเงิน การธุรกิจ การพาณิชย์ การสื่อสาร และความเจริญของประเทศ ตั้งอยู่บนสามเหลี่ยมปากแม่น้ำเจ้าพระยา มีแม่น้ำเจ้าพระยาไหลผ่านและแบ่งเมืองออกเป็น 2 ฝั่ง คือ ฝั่งพระนครและฝั่งธนบุรี กรุงเทพมหานครมีพื้นที่ทั้งหมด 1,568.737 ตร.กม. มีประชากรตามทะเบียนราษฎรกว่า 6 ล้านคน ทำให้กรุงเทพมหานครจัดเป็นเอกนคร (Primate City) มีผู้กล่าวว่า กรุงเทพมหานครเป็น "เอกนครที่สุดในโลก" เพราะมีประชากรมากกว่านครที่มีประชากรมากเป็นอันดับ 2 ถึง 40 เท่ากรุงเทพมหานครเป็นเมืองที่มีตึกระฟ้ามากที่สุดเป็นอันดับที่ 11 ของโลกในปี พ.ศ. 2563 มีสถานที่ท่องเที่ยวหลากหลายหลายในหลายรูปแบบทั้งด้านศาสนา ด้านการจับจ่ายซื้อของ ศูนย์การค้าต่าง ๆ หรือสถานบริการกลางคืน เช่น พระบรมมหาราชวัง สยามสแควร์ ถนนข้าวสาร ที่ดึงดูดนักท่องเที่ยวต่างชาติด้วยความงาม ได้แก่ ในปี พ.ศ. 2562 นิตยสารฟอบส์ นิตยสารเกี่ยวกับธุรกิจและการเงินชื่อดัง สกู๊ปชาร์ตเมริกา ได้จัดอันดับกรุงเทพมหานครเป็นเมืองที่มีนักท่องเที่ยวต่างชาติมากเป็นลำดับที่ 1 ของโลก โดยมีนักเดินทางเข้ามาเกิน 22.78 ล้านคนและทำรายได้จากการท่องเที่ยวต่างชาติมากกว่า 2,003 ล้านดอลลาร์ สหรัฐฯ มากเป็นลำดับที่ 3 ของโลกรองจาก อุรุวาน และ เม็กกัน ตามลำดับก...
```

Ln: 7 Col: 0

ใช้คำสั่งนี้เพื่อเปลี่ยนเป็นภาษาไทย

# สร้าง Tab ใน GUI ให้ Save as ไฟล์ GUI\_Calculator.py ตั้งชื่อเป็น GUI\_Wikipedia.py



File Edit Selection View Go Run Terminal Help

GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code

EXPLORER testcsv.py GUI-Wikipedia.py data.csv

PYTHON GUI 2022

- data.csv
- GUI\_Calculator.py
- GUI\_layout.py
- GUI-Wikipedia.py
- GUI.py
- testcsv.py

```
4 import wikipedia
5 #####CSV#####
6 import csv
7 from datetime import datetime
8 def writetocsv(data):
9     with open('data.csv','a',newline='',encoding='utf-8') as file:
10         fw = csv.writer(file) # fw = file writer
11         fw.writerow(data)
12
13 #####GUI#####
14 GUI = Tk()
15 GUI.title('โปรแกรมจัดการ layout')
16 GUI.geometry('800x600')
17
#####TAB SETTING#####
19 Tab = ttk.Notebook(GUI)
20 Tab.pack(fill=BOTH,expand=1)
21
22 T1 = Frame(Tab)
23 T2 = Frame(Tab)
24
25 icon_tab1 = PhotoImage(file='tab1.png')
26 icon_tab2 = PhotoImage(file='tab2.png')
27
28 Tab.add(T1, text='กุ้ง',image=icon_tab1,compound='left')
29 Tab.add(T2, text='wiki',image=icon_tab2,compound='left')
30
#####TAB 1 - กุ้ง#####
31
32 L1 = Label(T1,text='กรอกจำนวนกุ้ง (กิโลกรัม)',font=('Angsana New',25))
33 L1.pack()
```

สร้าง Tab

กำหนดตำแหน่ง Tab

สร้างเฟรมซึ่งอตามภาพตามจำนวน Tab ที่จะสร้าง

เพิ่มเฟรมที่สร้างไปอยู่ใน Tab

ใส่ Icon หาได้จากเว็บ <https://iconarchive.com/>

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Python ⚙️ ⚙️

# เปลี่ยนให้ตัวโปรแกรมคำนวนกุ้งของเราอยู่ Table จะต้องเปลี่ยนจาก GUI มาเป็น TUI ตามตำแหน่ง ตามภาพ

```
L1 = Label(T1, text='กรอกจำนวนกุ้ง (กิโลกรัม)', font=('Angsana New', 25))
L1.pack()

v_kilo = StringVar() #ตัวแปรพิเศษเอาไว้เก็บค่า

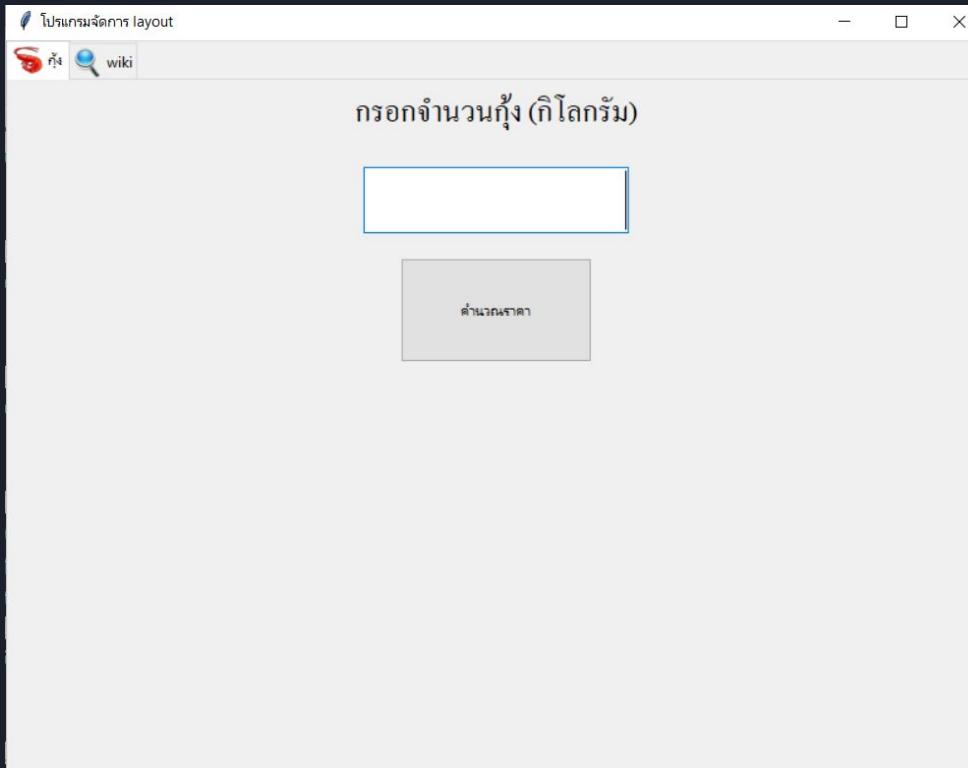
E1 = ttk.Entry(T1, textvariable= v_kilo, width=10, justify='right', font=('impact', 30))
E1.pack(pady=20)

def Calc(event=None):
    print('กำลังคำนวณ...กรุณารอสักครู่')
    kilo = float(v_kilo.get()) # .get() ดึงข้อมูลจากตัวแปรที่เป็น StringVar
    print(kilo * 299)
    calc_result = kilo * 299
    date = datetime.now()
    year = date.year + 543
    stamp = date.strftime('{}-%m-%d %H:%M:%S'.format(year)) #Thai Year
    data = [stamp, 'กุ้ง', '{:.2f}'.format(calc_result)]
    writetocsv(data)
    messagebox.showinfo('รวมราคาห้างหมด', 'ลูกค้าต้องจ่ายตังค์ทั้งหมด: {:.2f} บาท (กิโลกรัมละ 299 บาท)'.format(calc_result))

B1 = ttk.Button(T1, text='คำนวณราคา', command=Calc)
B1.pack(ipadx=40, ipady=30)

E1.bind('<Return>', Calc) # ต้องใส่ค่าว่า event=None ไว้ในฟังชันค่าย
```

# ผลลัพธ์



# ใส่โปรแกรมค้นหาใน Wikipedia ใน GUI ของเราใน Tab2

```
File Edit Selection View Go Run Terminal Help
...
testcsv.py
GUI_calculator.py
GUI_layout.py
GUI-Wikipedia.py >
GUI.py
tab1.png
tab2.png
testcsv.py

...
58 ######TAB 2 - Wiki #####
59
60 FONT1 = ('Angsana New',25)
61
62 L2 = Label(T2,text='ค้นหาข้อมูล wikipedia',font=('Angsana New',25))
63 L2.pack()
64
65 v_search = StringVar() # .get()=ถึงช่อง .set('hello') เซ็ตชื่อความให้เป็นแบบนั้น
66
67 E2 = ttk.Entry(T2, textvariable=v_search, font=FONT1)
68 E2.pack(pady=10)
69
70 wikipedia.set_lang('th') #ที่ให้เป็นภาษาไทย
71
72 def Search():
73     try:
74         search = v_search.get() #ถึงมีความจากช่องกรอกมา
75         text = wikipedia.summary(search)
76         print(text)
77         v_result.set(text[:1500])
78     except:
79         v_result.set('ไม่มีข้อมูล กรุณาค้นหาใหม่')
80
81     # เพิ่มฟังชันสำหรับดึงไปอ่านบทความฉบับเต็มในรูปแบบภาษาอื่น
82
83 B2 = ttk.Button(T2,text='Search',image=icon_tab2,compound='left',command=Search)
84 B2.pack()
85
86
87 v_result = StringVar()
88 v_result.set('-----Result-----')
89 result = Label(T2,textvariable=v_result,wraplength=550, font=(None,15))
90 result.pack()
91
92 GUI.mainloop()
```

คำแนะนำ font

ทำให้เป็นภาษาไทย

สร้างปุ่ม ใส่ icon ให้ปุ่ม

ด้านบนสุดให้ Import wikipedia ด้วย

ตัวแปรพิเศษไว้เก็บค่า

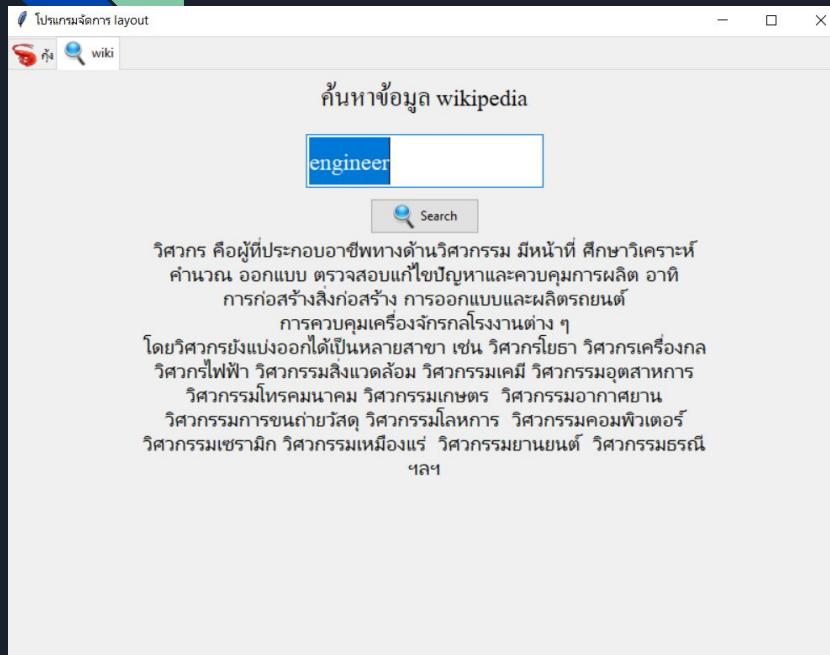
สร้างช่องกรอก

ลิงตัวแปรพิเศษกับช่องกรอกด้วย textvariable

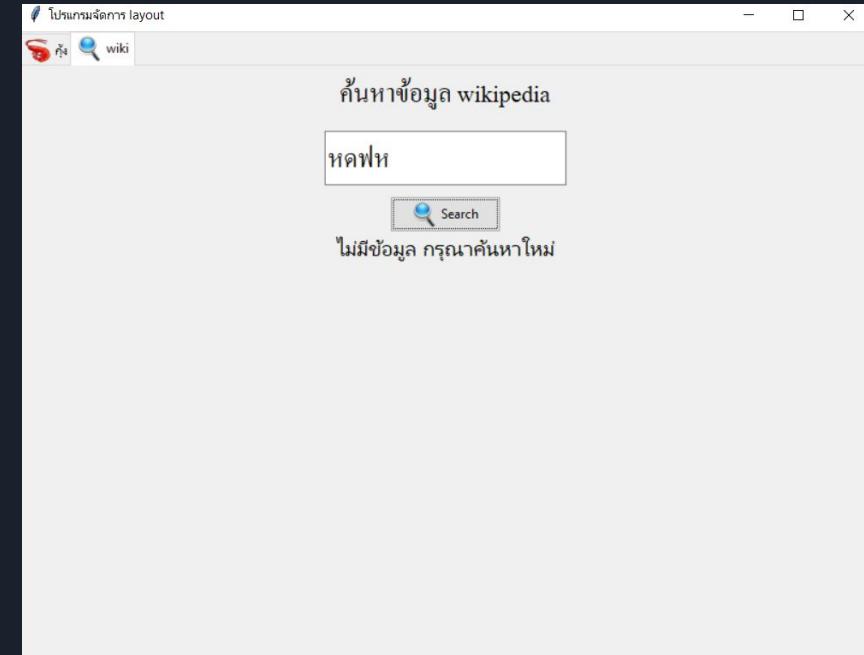
ฟังก์ชันของ wikipedia เขียน try / except ดัก error ถ้าพิมพ์ตัวไหนที่ไม่มีข้อมูล จะโชว์ text ตามที่ตั้ง

ดึงผลลัพธ์ที่ได้มาแสดง

# ผลลัพธ์



ปกติ



Error



# Python GUI 2022(EP.5)

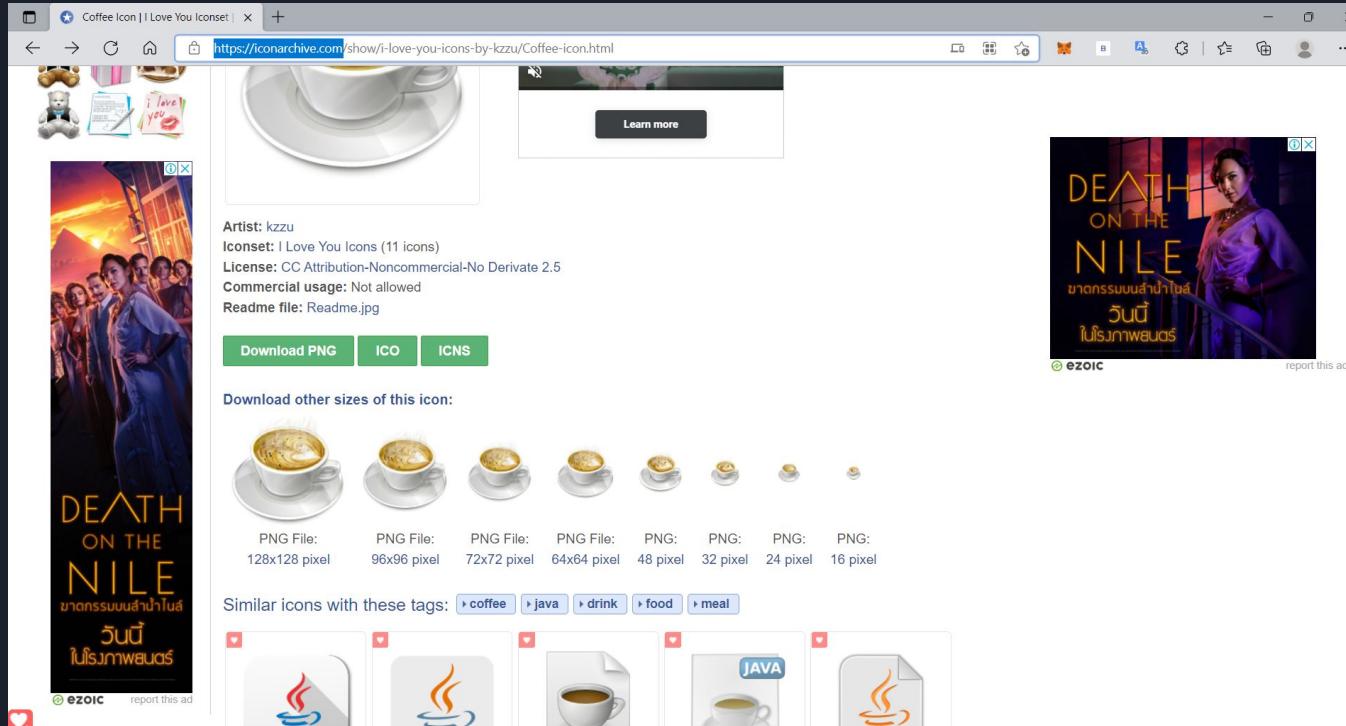
by



Uncle Engineer

ลุงวิศวกร สอนคำนวณ

ดาวน์โหลดภาพก้าแฟเจก เว็บ <https://iconarchive.com/>



# เพิ่ม Tab ที่ 3

The image shows a development environment with two windows. On the left is Visual Studio Code (VS Code) displaying Python code for a GUI application named 'GUI-Wikipedia.py'. The code uses the Tkinter library to create a window titled 'โปรแกรมจัดการ layout' with a size of 800x600 pixels. It creates a notebook tab control with three tabs labeled 'กุ้ง', 'wiki', and 'CAFE', each associated with a specific icon. On the right is a screenshot of the application window titled 'โปรแกรมจัดการ layout', which shows the three tabs with their respective icons.

เพิ่ม Tab 3 โดยการเพิ่ม 3 บรรทัดนี้

```
fw = csv.writer(file) # fw = file writer
fw.writerow(data)

#####
GUI = Tk()
GUI.title('โปรแกรมจัดการ layout')
GUI.geometry('800x600')

#####
Tab = ttk.Notebook(GUI)
Tab.pack(fill=BOTH,expand=1)

T1 = Frame(Tab)
T2 = Frame(Tab)
T3 = Frame(Tab)

icon_tab1 = PhotoImage(file='tab1.png')
icon_tab2 = PhotoImage(file='tab2.png')
icon_tab3 = PhotoImage(file='tab3.png')

Tab.add(T1, text='กุ้ง',image=icon_tab1,compound='left')
Tab.add(T2, text='wiki',image=icon_tab2,compound='left')
Tab.add(T3, text='CAFE',image=icon_tab3,compound='left')
|  
#####
#TAB 1 - กุ้ง#####
L1 = Label(T1,text='กรอกจำนวนกุ้ง (กิโลกรัม)',font=('Angsana New',25))
L1.pack()

v_kilo = StringVar() #ตัวแปรพิเศษเอาไว้เก็บค่า
```

File Edit Selection View Go Run Terminal Help GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code

testcsv.py GUI-Wikipedia.py

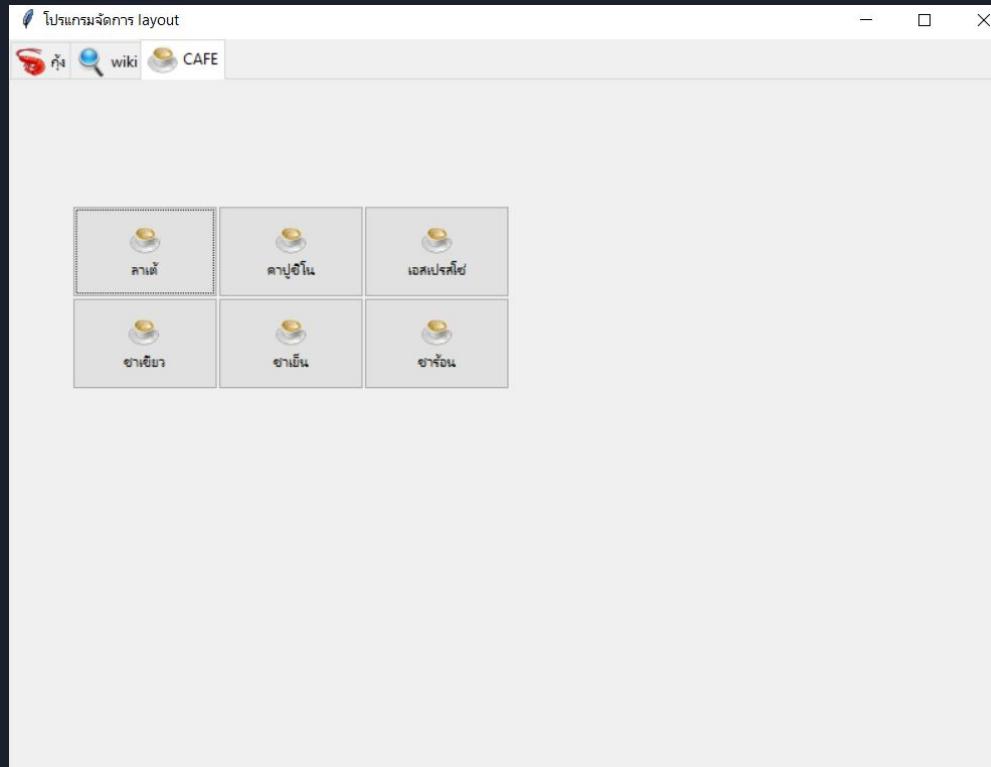
```
108 #######TAB 4 - Coffee #####
109
110 CF1 = Frame(T3)
111 CF1.place(x=50,y=100)
112
113 B = ttk.Button(CF1,text='ລາຍເຕີ',image=icon_tab3,compound='top')
114 B.grid(row=0,column=0,ipadx=20,ipady=10)
115 B = ttk.Button(CF1,text='ຄາມຫຼິກໃນ',image=icon_tab3,compound='top')
116 B.grid(row=0,column=1,ipadx=20,ipady=10)
117 B = ttk.Button(CF1,text='ລາສປະລໂອຊີ',image=icon_tab3,compound='top')
118 B.grid(row=0,column=2,ipadx=20,ipady=10)
119
120 B = ttk.Button(CF1,text='ໜ້າເພື່ອຍາ',image=icon_tab3,compound='top')
121 B.grid(row=1,column=0,ipadx=20,ipady=10)
122 B = ttk.Button(CF1,text='ໜ້າເກື່ອນ',image=icon_tab3,compound='top')
123 B.grid(row=1,column=1,ipadx=20,ipady=10)
124 B = ttk.Button(CF1,text='ໜ້າຮອນ',image=icon_tab3,compound='top')
125 B.grid(row=1,column=2,ipadx=20,ipady=10)
126
127 GUI.mainloop()
128
```

ใส่ เฟรมใน tab

ใส่ปุ่มใน เฟรม

Python 3.10.1 64-bit ⊗ 0 △ 0 Ln 126, Col 41 Spaces: 4 UTF-8 CRLF Python ⚙️ 🔍

# ผลลัพธ์



# วิธีเปลี่ยนฟอนต์ในปุ่ม

```
File Edit Selection View Go Run Terminal Help  
GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code  
testcsv.py GUI-Wikipedia.py ...  
101     B3.place(x=600,y=50)  
102  
103  
104     v_result = StringVar()  
105     v_result.set('-----Result-----')  
106     result = Label(T2,textvariable=v_result,wraplength=550, font=(None,15))  
107     result.pack()  
108  
109 #####TAB 4 - Coffee #####  
110  
111 Bfont = ttk.Style()  
112 Bfont.configure('TButton',font=['Angsana New',15])  
113  
114 CF1 = Frame(T3)  
115 CF1.place(x=50,y=100)  
116  
117 B = ttk.Button(CF1,text='ลาเต',image=icon_tab3,compound='top')  
118 B.grid(row=0,column=0,ipadx=20,ipady=10)  
119 B = ttk.Button(CF1,text='คามป์ชิโน',image=icon_tab3,compound='top')  
120 B.grid(row=0,column=1,ipadx=20,ipady=10)  
121 B = ttk.Button(CF1,text='เอสเพรสโซ',image=icon_tab3,compound='top')  
122 B.grid(row=0,column=2,ipadx=20,ipady=10)  
123  
124 B = ttk.Button(CF1,text='ชาเขียว',image=icon_tab3,compound='top')  
Python 3.10.1 64-bit ⑧ 0 △ 0 Ln 112, Col 49 Spaces: 4 UTF-8 CRLF Python ⚙ ⚙
```

ใส่ชื่อฟอนต์ที่ต้องการ

ขนาดของฟอนต์

## สร้างเฟรม

## สร้างตาราง

The screenshot shows a Visual Studio Code interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar indicates the file is "GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code". The left sidebar contains icons for file operations like Open, Save, Find, and others. The main editor area displays the following Python code:

```
File Edit Selection View Go Run Terminal Help
GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code
testcsv.py GUI-Wikipedia.py

128 b = L1K.BUTTON(CF1, text="  ", image=ICON_CADS, compound="top")
129 b.grid(row=1, column=2, ipadx=20, ipady=10)
130
131 ######TABLE#####
132 CF2 = Frame(T3)
133 CF2.place(x=500,y=100)
134
135 header = ['No.', 'title', 'price','quantity','total']
136 hwidth = [50,200,100,100,100]
137 table = ttk.Treeview(CF2,columns=header, show='headings',height=15)
138 table.pack()
139
140 for hd,hw in zip(header,hwidth):
141     table.column(hd,width=hw)
142     table.heading(hd,text=hd)
143
144 GUI.mainloop()

Python 3.10.1 64-bit ⊗ 0 △ 0
```

Annotations in blue boxes explain parts of the code:

- A box labeled "header ของตาราง" points to the line `header = ['No.', 'title', 'price','quantity','total']`.
- A box labeled "ขนาดของ header" points to the line `hwidth = [50,200,100,100,100]`.
- A box labeled "แสดง header , ขนาดของตาราง" points to the line `table = ttk.Treeview(CF2,columns=header, show='headings',height=15)`.
- A box labeled "for loop ขนาดของ header" points to the line `for hd,hw in zip(header,hwidth):`.

header ของตาราง

ขนาดของ header

แสดง header , ขนาดของตาราง

for loop ขนาดของ header

# ผลลัพธ์

ໂປ່ນເກມຈົດການ layout

ກໍ່າ wiki CAFE

 ລາວຕີ	 ຄາງຈິໂນ	 ເອສປະກໄກ
 ชาເຂົ້າວ	 ชาເຢືນ	 ชาຮັອນ

No.	title	price	quantity	total

# เพิ่มข้อมูลจากปุ่มการແພັນໃນຕາரາງ

```
File Edit Selection View Go Run Terminal Help
GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code
testcsv.py GUI-Wikipedia.py ...
110
111
112
113
114
115
116
117 allmenu = {}
118
119 product = {'latte':{'name':'ลาเต','price':30},
120           'cappuccino':{'name':'คาปูชิโน','price':35},
121           'espresso':{'name':'เอสเพรสโซ','price':40}}
122
123 def UpdateTable():
124     table.delete(*table.get_children()) # ແກ້ໄຂຂໍ້ມູນສາກາໃນຕາරາງ
125     for i,m in enumerate(allmenu.values(),start=1):
126         # m = ['ลาเต', 30, 1, 30]
127         table.insert('end',value=[ i,m[0],m[1],m[2],m[3] ])
128
129
130 def AddMenu(name='latte'):
131     # name = 'latte'
132     if name not in allmenu:
133         allmenu[name] = [product[name]['name'],product[name]['price'],1,product[name]['price']]
134
135     else:
136         # {'latte': ['ลาเต', 30, 1, 30]}
137         quan = allmenu[name][2] + 1
138         total = quan * product[name]['price']
139         allmenu[name] = [product[name]['name'],product[name]['price'],quan ,total]
140     print(allmenu)
141
142 def Menu1():
143     AddMenu('latte')
144     UpdateTable()
145
146 def Menu2():
147     AddMenu('cappuccino')
148     UpdateTable()
149
150 def Menu3():
151     AddMenu('espresso')
152     UpdateTable()
153
```

ເກີນຂໍ້ມູນໄວ້ໃນ list

ເຄີຍວ່າຂໍ້ມູນເກົ່າໃນຕາරາງ

ໃສ່ຂໍ້ມູນໃນຕາරາງ

ເຂີຍ if condition ເພື່ອໃຫ້  
ເພີ່ມຈຳນວນ ແລະ ຮາຄາ ຕອນ  
ເພີ່ມເມນຸໃສ່ໃນຕາරາງ

ຝຶກໜັນຂອງແຕ່ລະເມນຸ

reference เมนูในปุ่ม

```
File Edit Selection View Go Run Terminal Help
GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code
testcsv.py  GUI-Wikipedia.py < ...
140     print(allmenu)
141
142     def Menu1():
143         AddMenu('latte')
144         UpdateTable()
145
146     def Menu2():
147         AddMenu('cappuccino')
148         UpdateTable()
149
150     def Menu3():
151         AddMenu('espresso')
152         UpdateTable()
153
154     B = ttk.Button(CF1,text='ลาเต้',image=icon_tab3,compound='top',command=Menu1)
155     B.grid(row=0,column=0,ipadx=20,ipady=10)
156     B = ttk.Button(CF1,text='คัปปuccino',image=icon_tab3,compound='top',command=Menu2)
157     B.grid(row=0,column=1,ipadx=20,ipady=10)
158     B = ttk.Button(CF1,text='เอสเพรสโซ่',image=icon_tab3,compound='top',command=Menu3)
159     B.grid(row=0,column=2,ipadx=20,ipady=10)
160
161     B = ttk.Button(CF1,text='ชาเย็น',image=icon_tab3,compound='top')
162     B.grid(row=1,column=0,ipadx=20,ipady=10)
163     B = ttk.Button(CF1,text='ชาเขียว',image=icon_tab3,compound='top')
164     B.grid(row=1,column=1,ipadx=20,ipady=10)
165     B = ttk.Button(CF1,text='ชาเย็น',image=icon_tab3,compound='top')
166     B.grid(row=1,column=2,ipadx=20,ipady=10)
167
168
169 #####TABLE#####
```

# ผลลัพธ์

โปรแกรมจัดการ layout

กู้ง wiki CAFE

	ลาเต้
	คาปูชิโน
	เอสเปรสโซ่
	ชาเขียว
	ชาเย็น
	ชาอ่อน

No.	title	price	quantity	total
1	ลาเต้	30	3	90
2	คาปูชิโน	35	4	140
3	เอสเปรสโซ่	40	8	320



# Python GUI 2022(EP.6)

by



Uncle Engineer

ลุงวิศวกร สอนคำนวณ

# Add ข้อมูลเก็บไว้ใน CSV

```
File Edit Selection View Go Run Terminal Help
GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code
testcsv.py  GUI-Wikipedia.py X
GUI-Wikipedia.py > ...
207     B = ttk.Button(T3,text='Clear',command=Reset).place(x=600,y=500)
208
209
210 # Transaction ID
211 v_transaction = StringVar()
212 trstamp = datetime.now().strftime('%Y-%m-%d %H:%M:%S') #GEN Transaction
213 v_transaction.set(trstamp)
214 LTR = Label(T3, textvariable=v_transaction, font=(None,10)).place(x=950,y=70)
215
216 # Save Button
217 FB = Frame(T3)
218 FB.place(x=890,y=450)
219
220 def AddTransaction():
221     # writetocsv('transaction.csv')
222     stamp = datetime.now().strftime('%Y-%m-%d %H:%M:%S')
223     transaction = v_transaction.get()
224     print(transaction, stamp, allmenu.values())
225     for m in allmenu.values():
226         # before: m = [ 'ค่าใช้จ่าย', 35, 1, 35]
227         # after: m = ['12341234', '2022-02-17 21:04:19', 'ค่าใช้จ่าย', 35, 1, 35]
228         m.insert(0,transaction)
229         m.insert(1,stamp)
230         writetocsv(m,'transaction.csv')
231     Reset() #clear data
232
233
234 B = ttk.Button(FB,text='บันทึก',command=AddTransaction)
235 B.pack(ipadx=30,ipady=20)
236
237
238 GUI.mainloop()
```

Generate Transaction เพื่อเตรียมใช้งาน

บันทึก transaction ที่ใส่เข้าไปใน list รายการในตาราง

ปุ่มบันทึก

```
File Edit Selection View Go Run Terminal Help
GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code
testcsv.py GUI-Wikipedia.py ...
191 L = Label(T3,text='Total:', font=(None,15)).place(x=500,y=450)
192
193 v_total = StringVar()
194 v_total.set('0.0')
195
196 LT = Label(T3,textvariable=v_total, font=(None,15))
197 LT.place(x=600,y=450)
198
199 def Reset():
200     global allmenu
201     allmenu = {}
202     table.delete(*table.get_children())
203     v_total.set('0.0')
204     trstamp = datetime.now().strftime('%Y-%m-%d %H:%M:%S') #GEN Transaction
205     v_transaction.set(trstamp)
206
207 B = ttk.Button(T3,text='Clear',command=Reset).place(x=600,y=500)
208
209 # Transaction ID
210 v_transaction = StringVar()
211 trstamp = datetime.now().strftime('%Y-%m-%d %H:%M:%S') #GEN Transaction
212 v_transaction.set(trstamp)
213 LTR = Label(T3,textvariable=v_transaction,font=(None,10)).place(x=950,y=70)
214
215 # Save Button
216 FB = Frame(T3)
217 FB.place(x=890,y=450)
218
219 def AddTransaction():
220     # writetocsv('transaction.csv')
221     stamp = datetime.now().strftime('%Y-%m-%d %H:%M:%S')
222     transaction = v_transaction.get()
223     print(transaction, stamp, allmenu.values())
224     for m in allmenu.values():
225         # before: m = ['ค่าใช้สิน', 35, 1, 35]
226         # after: m = ['12341234', '2022-02-17 21:04:19', 'ค่าใช้สิน', 35, 1, 35]
```

Generate Transaction เอามาใส่ในฟังก์ชัน Reset ของการบ้าน Ep.5 เพื่อให้ Gen ใหม่ทุกครั้งที่กดปุ่ม Reset

File Edit Selection View Go Run Terminal Help

GUI-Wikipedia.py - Python GUI 2022 - Visual Studio Code

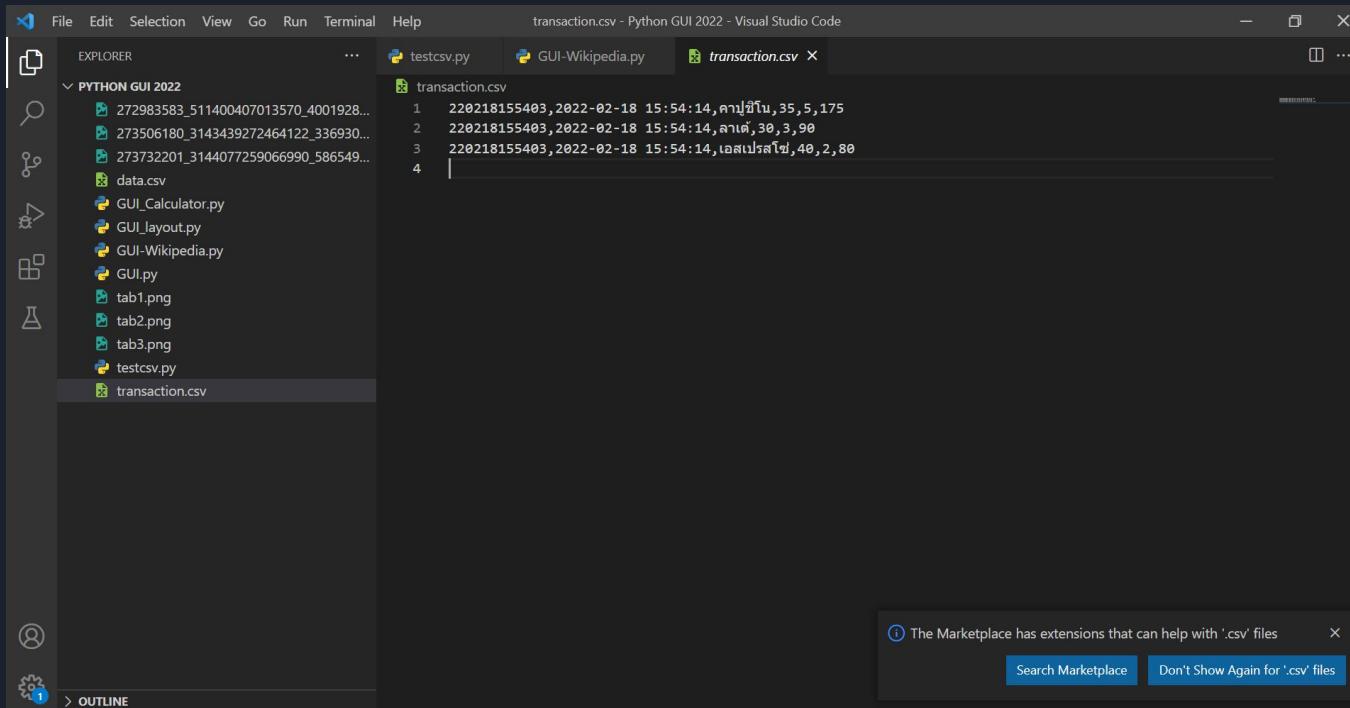
testcsv.py    GUI-Wikipedia.py    transaction.csv

GUI-Wikipedia.py > ...

```
1 # GUI-Calculator.py
2 from tkinter import *
3 from tkinter import ttk, messagebox
4 import wikipedia
5 import webbrowser
6 #####CSV#####
7 import csv
8 from datetime import datetime
9 def writetocsv(data, filename='data.csv'):
10     with open(filename, 'a', newline='', encoding='utf-8') as file:
11         fw = csv.writer(file) # fw = file writer
12         fw.writerow(data)
13
14 #####GUI#####
15 GUI = Tk()
16 GUI.title('โปรแกรมจัดการ layout')
17 GUI.geometry('1200x600')
18
19 #####TAB SETTING#####
20 Tab = ttk.Notebook(GUI)
21 Tab.pack(fill=BOTH,expand=1)
22
23 T1 = Frame(Tab)
24 T2 = Frame(Tab)
```

เปลี่ยนฟังก์ชัน writetocsv  
ของกุ้งให้สามารถใช้ร่วมกันได้

# ผลลัพธ์



The screenshot shows a Visual Studio Code interface with a dark theme. The left sidebar contains a tree view labeled 'EXPLORER' under 'PYTHON GUI 2022'. It lists several Python files (testcsv.py, GUI-Wikipedia.py, data.csv, GUI\_Calculator.py, GUI\_layout.py, GUI-Wikipedia.py, GUI.py, tab1.png, tab2.png, tab3.png) and two CSV files (transaction.csv). The right pane displays the contents of the 'transaction.csv' file:

```
transaction.csv - Python GUI 2022 - Visual Studio Code
transaction.csv
1 220218155403,2022-02-18 15:54:14,คาปูชิโน,35,5,175
2 220218155403,2022-02-18 15:54:14,ลาเต้,30,3,90
3 220218155403,2022-02-18 15:54:14,เอสเปรสโซ,40,2,80
4 |
```

A notification bar at the bottom right says: 'The Marketplace has extensions that can help with '.csv' files' with buttons for 'Search Marketplace' and 'Don't Show Again for '.csv' files'.

# ดึงข้อมูลจาก CSV นำมาโชว์ในตารางประวัติออร์เดอร์

```
File Edit Selection View Go Run Terminal Help
testcsv.py  GUI-Wikipedia.py x transaction.csv
238
239 def HistoryWindow(event):
240     HIS = Toplevel() # คลายกับ GUI = Tk()
241     HIS.geometry('750x500')
242
243     L = Label(HIS,text='ประวัติการสั่งซื้อ', font=(None,15)).pack()
244
245     # History
246     header = ['ts-id','datetime', 'title', 'price','quantity','total']
247     hwidth = [100,100,200,100,100,100]
248
249     table_history = ttk.Treeview(HIS,columns=header, show='headings',height=15)
250     table_history.pack()
251
252     for hd,hw in zip(header,hwidth):
253         table_history.column(hd,width=hw)
254         table_history.heading(hd,text=hd)
255
256     # Update from CSV
257     with open('transaction.csv',newline='',encoding='utf-8') as file:
258         fr = csv.reader(file) # file reader
259         for row in fr:
260             table_history.insert('',0,value=row)
261
262     HIS.mainloop()
263
264     GUI.bind('<F1>',HistoryWindow)
265
266     GUI.mainloop()
```

สร้าง windows ใหม่ใช้สำหรับแสดงตาราง

สร้างตาราง

ดึงข้อมูลจาก csv มาแสดงในตาราง

# ผลลัพธ์

โปรแกรมจัดการ layout

โปรแกรมจัดการ layout

ประวัติการสั่งซื้อ

ts-id	datetime	title	price	quantity	total
220218155414	2022-02-18 16:11:	เอสเปรสโซ่	40	3	120
220218155414	2022-02-18 16:11:	ลาเต้	30	3	90
220218155414	2022-02-18 16:11:	คาปูชิโน	35	3	105
220218155403	2022-02-18 15:54:	เอสเปรสโซ่	40	2	80
220218155403	2022-02-18 15:54:	ลาเต้	30	3	90
220218155403	2022-02-18 15:54:	คาปูชิโน	35	5	175



# Python GUI 2022(EP.7)

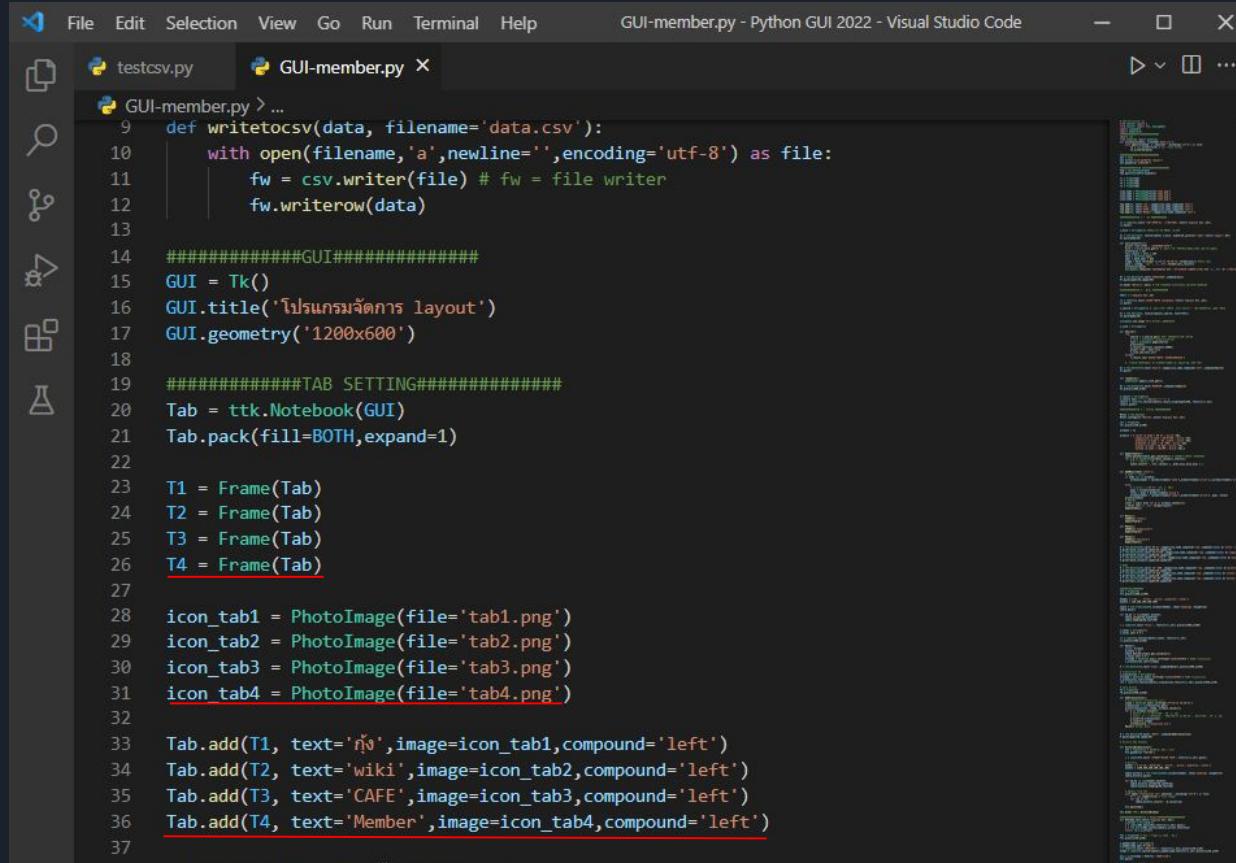
by



Uncle Engineer

ลุงวิศวกร สอนคำนวณ

# เพิ่ม Tab4 สำหรับทำหน้าสมาชิก



```
File Edit Selection View Go Run Terminal Help GUI-member.py - Python GUI 2022 - Visual Studio Code
testcsv.py GUI-member.py

GUI-member.py > ...
9     def writetocsv(data, filename='data.csv'):
10         with open(filename,'a',newline='',encoding='utf-8') as file:
11             fw = csv.writer(file) # fw = file writer
12             fw.writerow(data)
13
14 #####GUI#####
15 GUI = Tk()
16 GUI.title('โปรแกรมจัดการ layout')
17 GUI.geometry('1200x600')
18
19 #####TAB SETTING#####
20 Tab = ttk.Notebook(GUI)
21 Tab.pack(fill=BOTH,expand=1)
22
23 T1 = Frame(Tab)
24 T2 = Frame(Tab)
25 T3 = Frame(Tab)
26 T4 = Frame(Tab)
27
28 icon_tab1 = PhotoImage(file='tab1.png')
29 icon_tab2 = PhotoImage(file='tab2.png')
30 icon_tab3 = PhotoImage(file='tab3.png')
31 icon_tab4 = PhotoImage(file='tab4.png')
32
33 Tab.add(T1, text='หน้า',image=icon_tab1,compound='left')
34 Tab.add(T2, text='wiki',image=icon_tab2,compound='left')
35 Tab.add(T3, text='CAFE',image=icon_tab3,compound='left')
36 Tab.add(T4, text='Member',image=icon_tab4,compound='left')
37
```

# ช่องสำหรับกรอกรายละเอียดของสมาชิก

```
269 #####TAB 4 Member#####
270 def ET3(GUI,text,font=('Angsana New',20)):
271     v_strvar = StringVar()
272     T = Label(GUI,text=text,font=(None,15)).pack()
273     E = ttk.Entry(GUI,textvariable=v_strvar,font=font)
274     return (E,T,v_strvar)
275
276 F41 = Frame(T4) # F41 = Frame in Tab4 , No.1
277 F41.place(x=50,y=50)
278
279 v_membercode = StringVar()
280 v_membercode.set('M-1001')
281 L = Label(T4,text='รหัสสมาชิก:',font=(None,13)).place(x=50,y=20)
282 LCode = Label([T4, textvariable=v_membercode, font=(None,13)]).place(x=150,y=20)
283
284 E41,L,v_fullname = ET3(F41, 'ชื่อ-สกุล')
285 E41.pack()
286
287 E42,L,v_tel = ET3(F41, 'เบอร์โทร')
288 E42.pack()
289
290 E43,L,v_usertype = ET3(F41, 'ประเภทสมาชิก')
291 E43.pack()
292 v_usertype.set('general')
293
294 E44,L,v_point = ET3(F41, 'คะแนนสะสม')
295 E44.pack()
296 v_point.set('0') # ใส่ค่า default ของ point
```

# บันทึกสมาชิกใหม่

```
299  
300 def SaveMember():  
301     code = v_membercode.get()  
302     fullname = v_fullname.get()  
303     tel = v_tel.get()  
304     usertype = v_usertype.get()  
305     point = v_point.get()  
306     print(fullname, tel, usertype, point)  
307     writetocsv([code, fullname, tel, usertype, point], 'member.csv') #บันทึกสมาชิกใหม่  
308     table_member.insert('', 0, value=[code, fullname, tel, usertype, point])  
309     UpdateTable_Member()  
310  
311     v_fullname.set('')  
312     v_tel.set('')  
313     v_usertype.set('general')  
314     v_point.set('0')  
315  
316  
317     BSave = ttk.Button(F41, text='บันทึก', command=SaveMember)  
318     BSave.pack()  
319
```

ดึงค่า V จากช่องกรอก

# ตารางโคว์สมานาชิก

```
356 F42 = Frame(T4)
357 F42.place(x=500,y=100)
358
359 header = ['Code', 'ชื่อ-สกุล', 'เบอร์โทร', 'ประเภทสมาชิก', 'คะแนนสะสม']
360 hwidth = [50,200,100,100,100]
361
362 table_member = ttk.Treeview(F42,columns=header, show='headings',height=15)
363 table_member.pack()
364
365 for hd,hw in zip(header,hwidth):
366     table_member.column(hd,width=hw)
367     table_member.heading(hd,text=hd)
368
```

# ฟังก์ชันสำหรับ update csv ไว้สำหรับให้อัพเดท csv เมื่อสร้าง/ลบ/แก้ไข สมาชิก

```
370 ✓ def UpdateCSV(data, filename='data.csv'):
371     # data = [[a,b],[a,b]]
372 ✓     with open(filename,'w',newline='',encoding='utf-8') as file:
373         fw = csv.writer(file) # fw = file writer
374         fw.writerows(data) # writerows = replace with list
375
376
```

# ลบข้อมูลในตารางที่เลือก

เมื่อมีการลบให้  
update csv

เมื่อมีการลบให้  
update table

```
377 # Delete ข้อมูลในตารางที่เลือก
378 def DeleteMember(event):
379     select = table_member.selection() #เลือก item
380     data = table_member.item(select)['values']
381     print(data)
382     del allmember[data[0]]
383     UpdateCSV(list(allmember.values()), 'member.csv')
384     UpdateTable_Member()
385
386 table_member.bind('<Delete>', DeleteMember)
```

เมื่อมีการกดปุ่ม Delete บน  
คีย์บอร์ดให้ทำฟังก์ชัน  
DeleteMember

# Update Table อัพเดตตารางโขว์สมาชิก

```
408     # Update Table
409     last_member = ''
410     allmember = {}
● 411
412     def UpdateTable_Member():
413         global last_member
414         with open('member.csv',newline='',encoding='utf-8') as file:
415             fr = csv.reader(file) # file reader
416             table_member.delete(*table_member.get_children()) #clear table
417             for row in fr:
418                 table_member.insert('',0,value=row)
419                 code = row[0] # ดึงรหัสมา
420                 allmember[code] = row
421
422             print('Last ROW:',row)
423             last_member = row[0]
424             # M-1001
425             # [ 'M',1001+1]
426             next_member = int(last_member.split('-')[1]) + 1
427             v_membercode.set('M-{}'.format(next_member))
428             print(allmember)
429
430
431             BEdit.state(['disabled'])
432             UpdateTable_Member()
433             GUI.mainloop()
```

ให้อ่านค่าเลขสมาชิก  
จากคนล่าสุดแล้วเพิ่ม 1  
หน่วยสำหรับสมาชิก  
คนต่อไปเรื่อยๆ

## Update ข้อมูลสมาชิกให้เมื่อตับเบิลคลิกจะมาแสดงในช่องกรอก

```
389 # Update ข้อมูลสมาชิก
390 def UpdateMemberInfo(event):
391
392     select = table_member.selection() #เลือก item
393     code = table_member.item(select)['values'][0]
394     print(allmember[code])
395     memberinfo = allmember[code]
396
397     v_membercode.set(memberinfo[0])
398     v_fullname.set(memberinfo[1])
399     v_tel.set(memberinfo[2])
400     v_usertype.set(memberinfo[3])
401     v_point.set(memberinfo[4])
402
403     BEdit.state(['!disabled']) # เปิดปุ่มแก้
404     BSave.state(['disabled']) # ปิดปุ่มบันทึก
405
406     table_member.bind('<Double-1>',UpdateMemberInfo)
407
408 # Update Table
```

# EditMember ໄວ້ສໍາหารັບແກ້ໄຂຂໍ້ມູນລສມາຊີກ

```
320 def EditMember():
321     code = v_membercode.get()
322     allmember[code][1] = v_fullname.get()
323     allmember[code][2] = v_tel.get()
324     allmember[code][3] = v_usertype.get()
325     allmember[code][4] = v_point.get()
326     UpdateCSV(list(allmember.values()), 'member.csv')
327     UpdateTable_Member()
328
329     BEedit.state(['disabled']) # ປຶບປຸມແກ້
330     BSave.state(['!disabled']) # ເປີດປຸມບັນທຶກ
331     # set default
332     v_fullname.set('')
333     v_tel.set('')
334     v_usertype.set('general')
335     v_point.set('0')
336
337     BEedit = ttk.Button(F41, text='ແກ້ໄຂ', command=EditMember)
338     BEedit.pack()
---
```

## NewMember สำหรับ Reset ค่าในช่องกรอก

```
341     def NewMember():
342         UpdateTable_Member()
343         BEdit.state(['disabled']) # ปิดปุ่มแก้
344         BSave.state(['!disabled']) # เปิดปุ่มบันทึก
345         # set default
346         v_fullname.set('')
347         v_tel.set('')
348         v_usertype.set('general')
349         v_point.set('0')
350
351
352         BNew = ttk.Button(F41,text='New',command=NewMember)
353         BNew.pack()
354
```

set ข้อมูลในช่องกรอกเป็น  
ค่า default



# Python GUI 2022(EP.8)

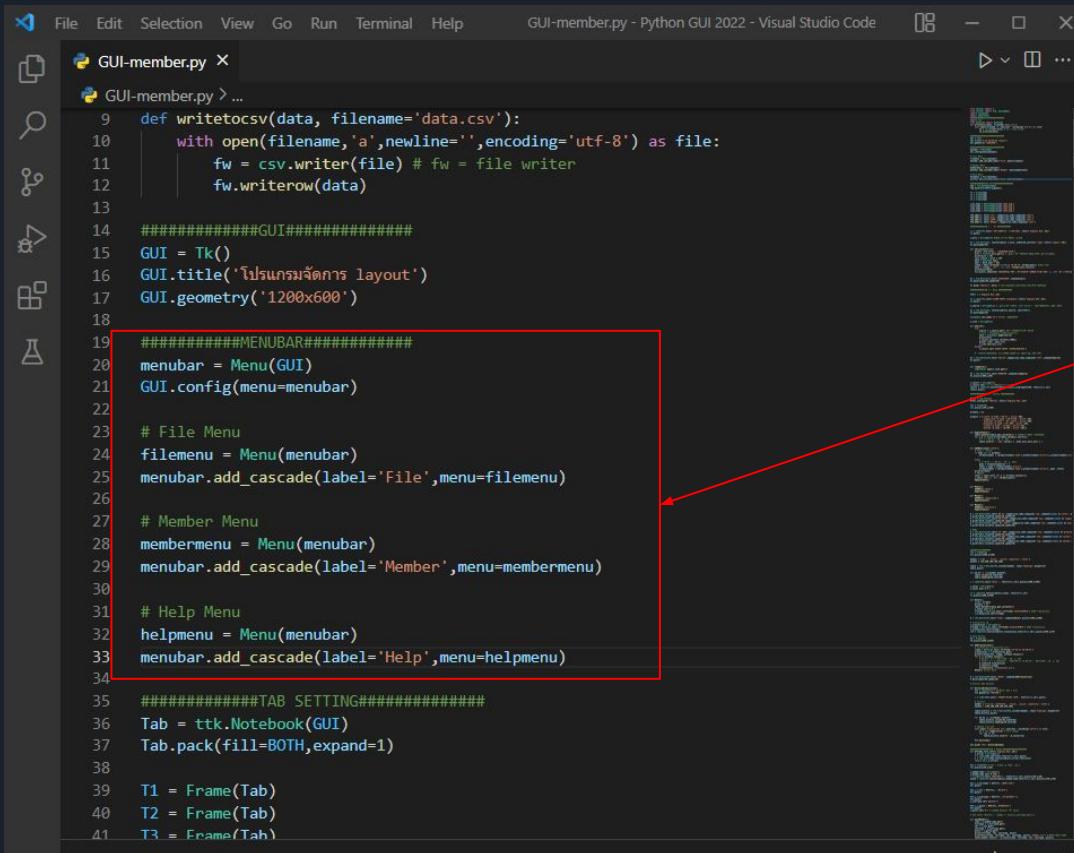
by



Uncle Engineer

ลุงวิศวกร สอนคำนวณ

# สร้างเมนูของโปรแกรม

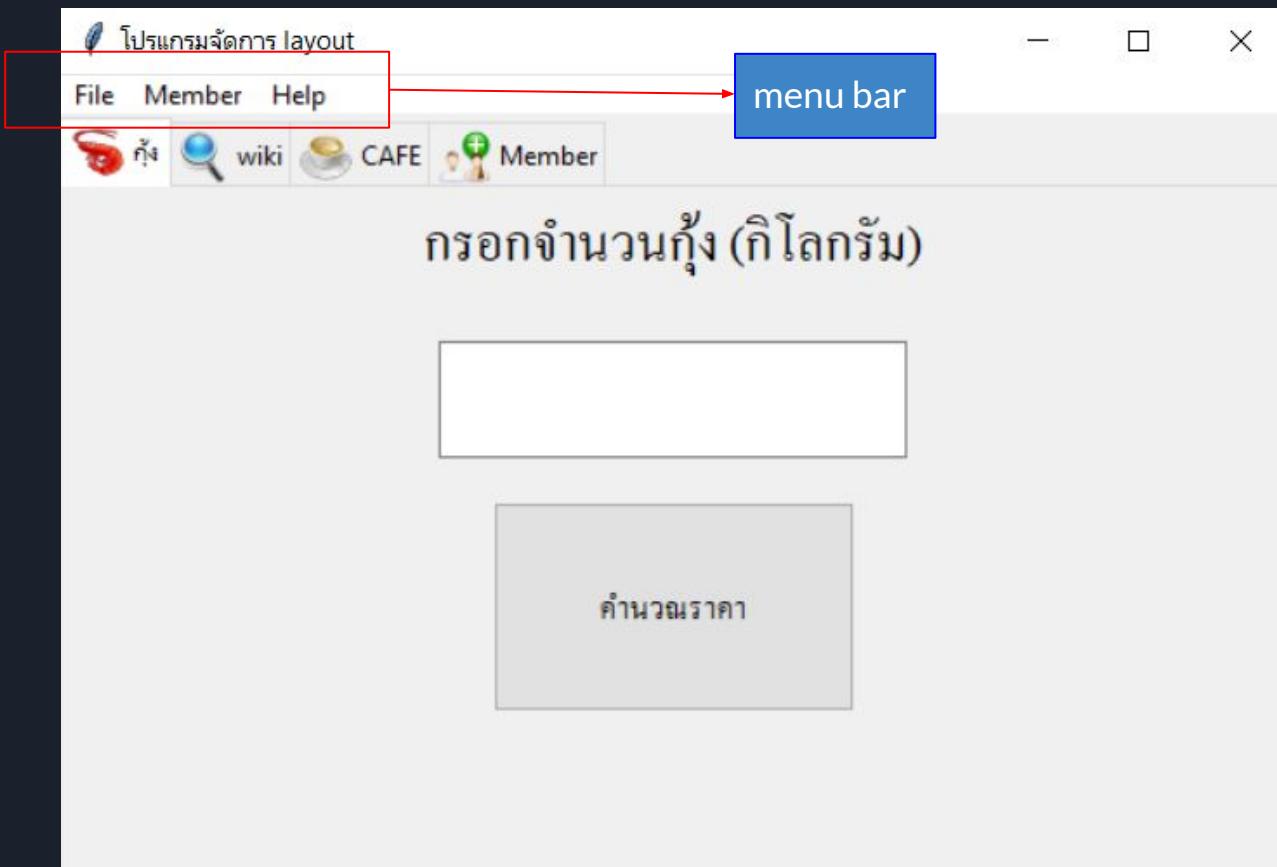


The screenshot shows a Visual Studio Code window with a dark theme. The title bar reads "GUI-member.py - Python GUI 2022 - Visual Studio Code". The left sidebar has icons for file operations. The main editor area contains Python code for a Tkinter application. A red box highlights the menu creation section from line 19 to line 74. An arrow points from this highlighted area to a blue callout box on the right containing the text "สร้าง Menu bar".

```
File Edit Selection View Go Run Terminal Help GUI-member.py - Python GUI 2022 - Visual Studio Code
GUI-member.py X
GUI-member.py > ...
9  def writetocsv(data, filename='data.csv'):
10     with open(filename,'a',newline='',encoding='utf-8') as file:
11         fw = csv.writer(file) # fw = file writer
12         fw.writerow(data)
13
14 #####GUI#####
15 GUI = Tk()
16 GUI.title('โปรแกรมจัดการ layout')
17 GUI.geometry('1200x600')
18
19 #####MENUBAR#####
20 menubar = Menu(GUI)
21 GUI.config(menu=menubar)
22
23 # File Menu
24 filemenu = Menu(menubar)
25 menubar.add_cascade(label='File',menu=filemenu)
26
27 # Member Menu
28 membermenu = Menu(menubar)
29 menubar.add_cascade(label='Member',menu=membermenu)
30
31 # Help Menu
32 helpmenu = Menu(menubar)
33 menubar.add_cascade(label='Help',menu=helpmenu)
34
35 #####TAB SETTING#####
36 Tab = ttk.Notebook(GUI)
37 Tab.pack(fill=BOTH,expand=1)
38
39 T1 = Frame(Tab)
40 T2 = Frame(Tab)
41 T3 = Frame(Tab)
```

สร้าง Menu bar

# ผลลัพธ์



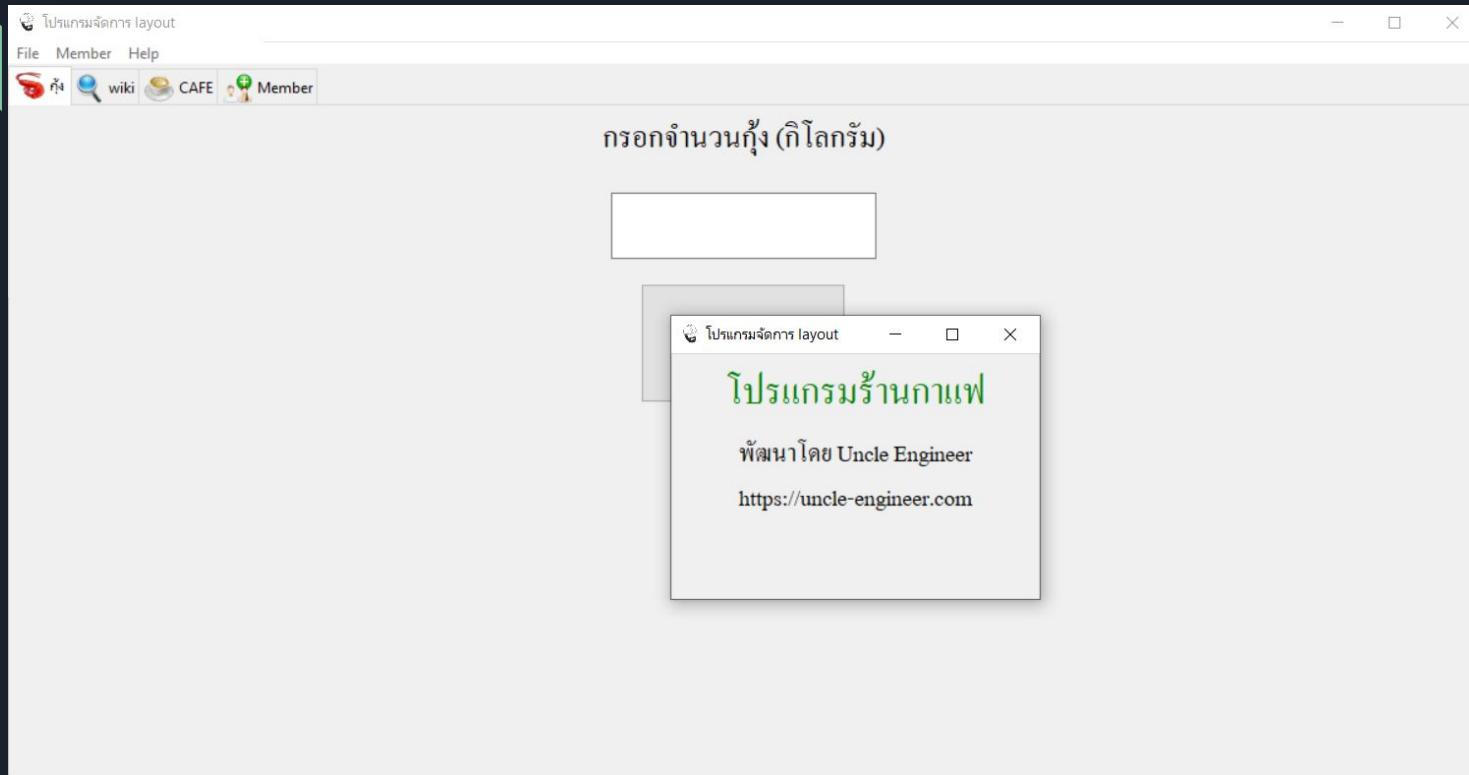
## วิธีทำให้เวลารัน GUI ให้ GUI อยู่กลางหน้าจอ

```
#####GUI#####
GUI = Tk()
GUI.title('โปรแกรมจัดการ layout')

W = 1200
H = 600
MW = GUI.winfo_screenwidth()
MH = GUI.winfo_screenheight()
SX = (MW/2) - (W/2)
SY = (MH/2) - (H/2)

GUI.geometry('{0}x{1}+{2:.0f}+{3:.0f}'.format(W,H,SX,SY))
```

# วิธีทำหน้าต่างเล็กๆ กลางหน้าจอเมื่อ用 sublime



# Code

```
File Edit Selection View Go Run Terminal Help
GUI-member.py - Python GUI 2022 - Visual Studio Code
GUI-member.py X
File Explorer Search Home Insert Editor Split Horizontal Split Vertical Terminal
GUI-member.py > ...
43 menubar.add_cascade(label='Member',menu=membermenu)
44
45
46 # Help Menu
47 import webbrowser
48
49 helpmenu = Menu(menubar,tearoff=0)
50 menubar.add_cascade(label='Help',menu=helpmenu)
51 contact_url = 'https://uncle-engineer.com'
52 helpmenu.add_command(label='Contact US',command=lambda: webbrowser.open(contact_url))
53
54 def About():
55     ABGUI = Toplevel()
56     ABGUI.iconbitmap('loong.ico')
57     W = 300
58     H = 200
59     MW = GUI.winfo_screenwidth() # Monitor Width
60     MH = GUI.winfo_screenheight() # Monitor Height
61     SX = (MW/2) - (W/2) # Start X
62     SY = (MH/2) - (H/2) # Start Y
63     ABGUI.geometry('{W}x{H}+{SX}+{SY}'.format(W,H,SX,SY))
64     L = Label(ABGUI,text='โปรดทราบว่าผม',fg='green',font=('Angsana New',30)).pack()
65     L = Label(ABGUI,text='พัฒนาโดย Uncle Engineer\nhttps://uncle-engineer.com',font=('Angsana New',20)).pack()
66     ABGUI.mainloop()
67
68 helpmenu.add_command(label='About',command=About)
69 #####TAB SETTING#####
70 Tab = ttk.Notebook(GUI)
71 Tab.pack(fill=BOTH,expand=1)
72 T1 = Frame(Tab)
```

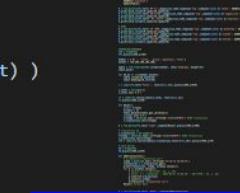
เพิ่ม command ใน menu bar

ใส icon ให้ GUI

ทำให้หน้าต่างอยู่กลางจอ

# เมนู Popup(คลิกขวา)

```
480 # POP UP Menu
481 member_rcmenu = Menu(GUI,tearoff=0) # rcmenu = right click menu
482 table_member.bind('<Button-3>',lambda event: member_rcmenu.post( event.x_root , event.y_root ) )
483 member_rcmenu.add_command(label='Delete',command=DeleteMember)
484 member_rcmenu.add_command(label='Update',command=UpdateMemberInfo)
485
486 def SearchName():
487     select = table_member.selection()
488     name = table_member.item(select)['values'][1]
489     print([name])
490     url = 'https://www.google.com/search?q={}'.format(name)
491     webbrowser.open(url)
492
493 member_rcmenu.add_command(label='Search Name',command=SearchName)
494
495 def SearchBCC():
496     select = table_member.selection()
497     name = table_member.item(select)['values'][1]
498     print(name)
499     url = 'https://www.bbc.co.uk/search?q={}'.format(name)
500     webbrowser.open(url)
501
502 member_rcmenu.add_command(label='Search BCC',command=SearchBCC)
503 # https://www.bbc.co.uk/search?q=putin
504
505
506
507 BEdit.state(['disabled'])
508 UpdateTable_Member()
509 GUI.mainloop()
```



search ชื่อใน google



search ชื่อใน BCC



ใส่ event=None

```
# Delete ข้อมูลในตารางที่เลือก
def DeleteMember(event=None):
    select = table_member.selection() #เลือก item
    print(select)
    if len(select) != 0:
        data = table_member.item(select)[‘values’]
        print(data)
        del allmember[data[0]]
        UpdateCSV(list(allmember.values()),‘member.csv’)
        UpdateTable_Member()
    else:
        messagebox.showwarning(‘ไม่ได้เลือกรายการ’, ‘กรุณาเลือกรายการก่อนลบข้อมูล’)

table_member.bind(‘<Delete>’,DeleteMember)
```

ใส่ event=None

```
# Update ข้อมูลสมาชิก
def UpdateMemberInfo(event=None):
    select = table_member.selection() #เลือก item
    if len(select) != 0:
        code = table_member.item(select)[‘values’][0]
        print(allmember[code])
        memberinfo = allmember[code]

        v_membercode.set(memberinfo[0])
        v_fullname.set(memberinfo[1])
        v_tel.set(memberinfo[2])
        v_usertype.set(memberinfo[3])
        v_point.set(memberinfo[4])

        BEdit.state([‘!disabled’]) # ปิดปุ่มแก้
        BSave.state([‘disabled’]) # ปิดปุ่มบันทึก
    else:
        messagebox.showwarning([‘ไม่ได้เลือกรายการ’, ‘กรุณาเลือกรายการก่อนแก้ไขข้อมูล’])

table_member.bind(‘<Double-1>’,UpdateMemberInfo)
```

ใช้ if len เพื่อเช็คว่ามี การเลือกซึ่งก่อนคลิก ขวาหรือไม่ ถ้าไม่ให้มี ข้อความแจ้งเตือน

# ผลลัพธ์

โปรแกรมจัดการ layout

File Member Help

กู้ง wiki CAFE Member

รหัสสมาชิก: M-1002

ชื่อ-สกุล: ชื่อ-สกุล

เบอร์โทร: เบอร์โทร

ประเภทสมาชิก: ประเภทสมาชิก

general

คะแนนสะสม: คะแนนสะสม

0

บันทึก

แก้ไข

New

Code: M-1001 ชื่อ-สกุล: ชื่อ-สกุล เบอร์โทร: 0812345678 ประเภทสมาชิก: general คะแนนสะสม: 0

Delete  
Update  
Search Name  
Search BCC



# Python GUI 2022(EP.9)

by



Uncle Engineer

ลุงวิศวกร สอนคำนวณ

# basic sqlite3

## สร้างไฟล์ ชื่อ basic-sql.py

○ Python GUI 2022 > 🐍 basic-sql.py > ...

สร้างตารางใน  
การจัดเก็บ

```
1 # basic-sql.py
2
3 import sqlite3
4
5 conn = sqlite3.connect('basicdb.sqlite3') #สร้างไฟล์ฐานข้อมูล
6 c = conn.cursor()
7
8
9 # สร้างตารางในการจัดเก็บ
10 c.execute("""CREATE TABLE IF NOT EXISTS member (
11     ID INTEGER PRIMARY KEY AUTOINCREMENT,
12     membercode TEXT,
13     fullname TEXT,
14     tel TEXT,
15     usertype TEXT,
16     points INTEGER ) """)
17
```

สร้างไฟล์ฐานข้อมูล

เก็บอะไร

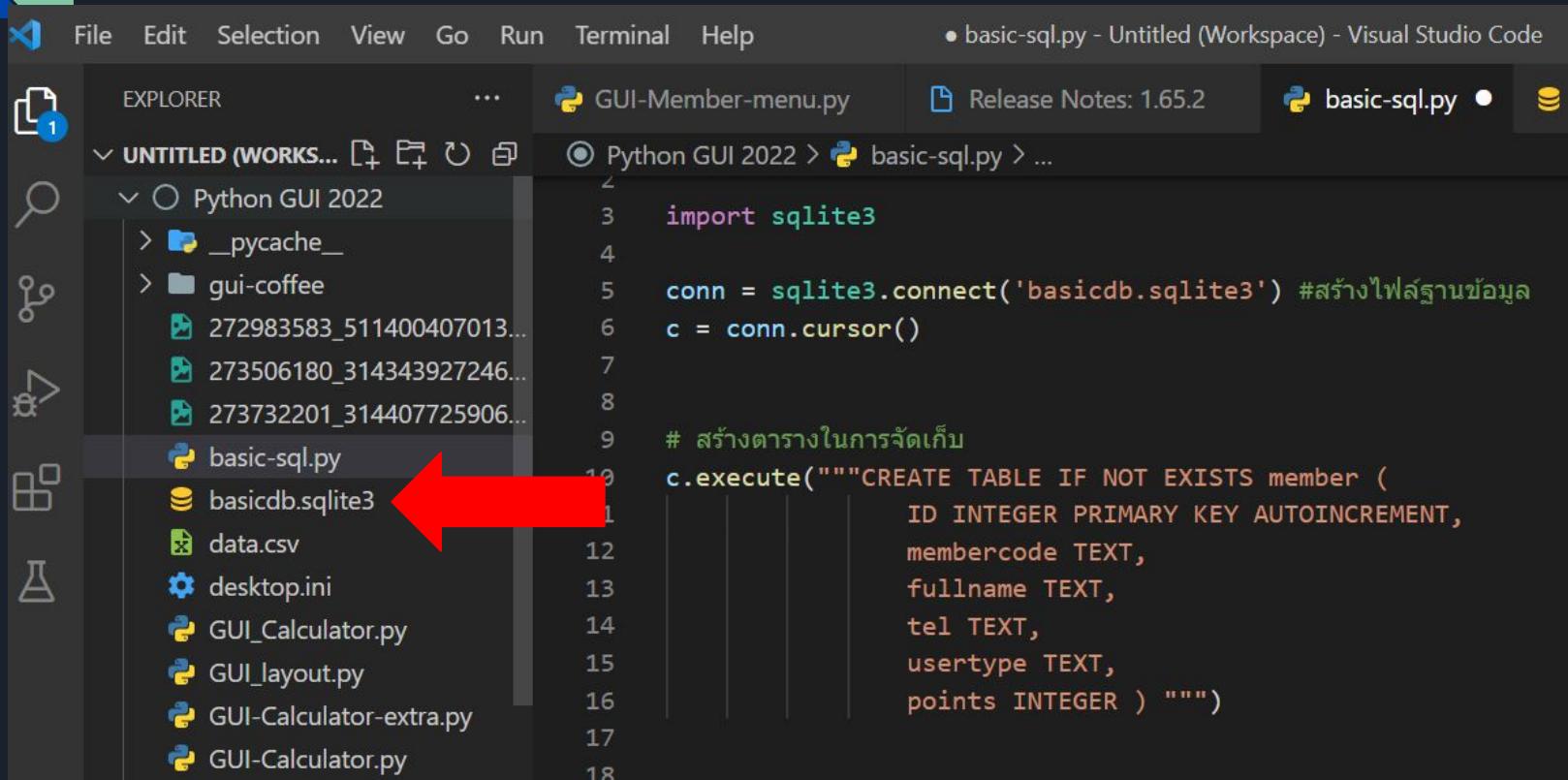
สร้าง field

สั่งให้มีการ  
เพิ่มขึ้นเรื่อยๆ

เก็บเป็นอะไร

# basic sqlite3

## เสร็จแล้วให้กด Build จะมีไฟล์ basicdb.sqlite3 ขึ้นมา



File Edit Selection View Go Run Terminal Help

basic-sql.py - Untitled (Workspace) - Visual Studio Code

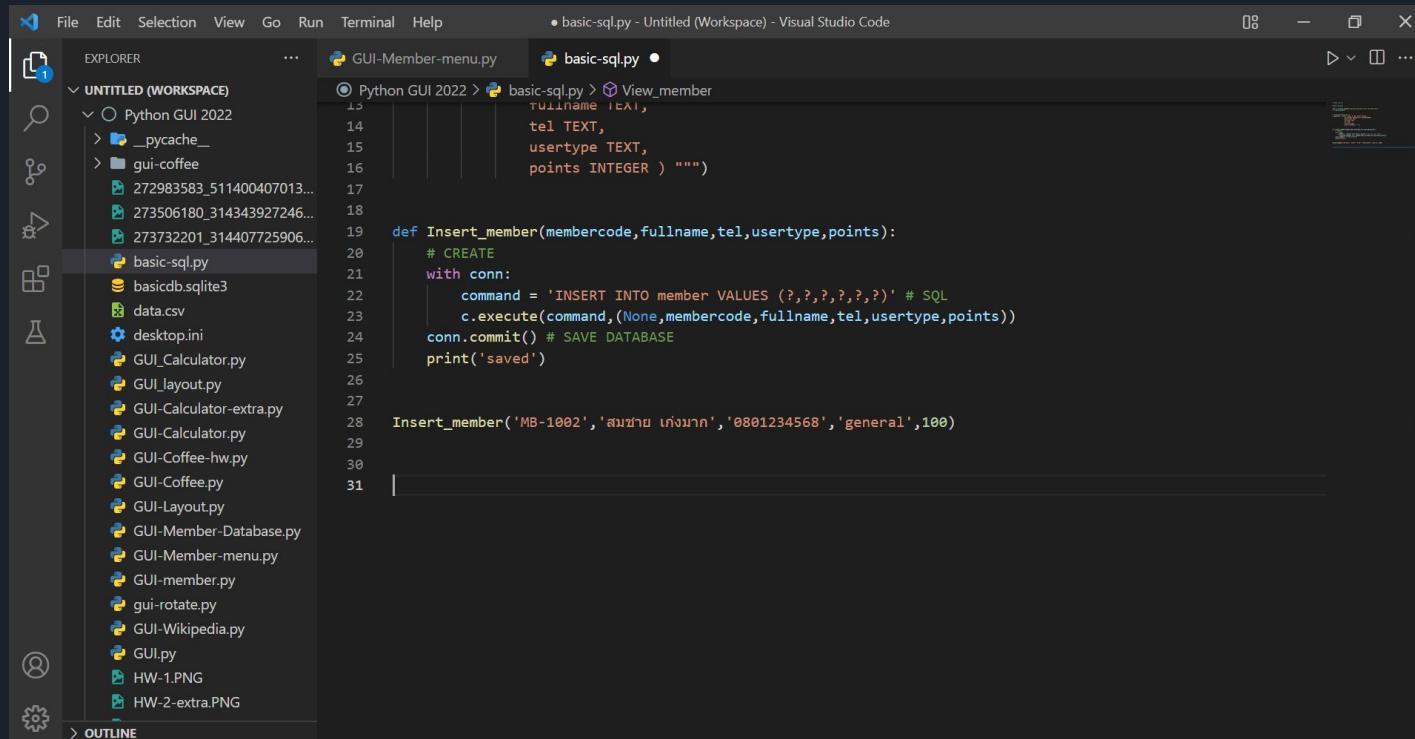
EXPLORER

UNTITLED (WORKS...) Python GUI 2022

- \_\_pycache\_\_
- gui-coffee
- 272983583\_511400407013...
- 273506180\_314343927246...
- 273732201\_314407725906...
- basic-sql.py
- basicdb.sqlite3
- data.csv
- desktop.ini
- GUI\_Calculator.py
- GUI\_layout.py
- GUI-Calculator-extra.py
- GUI-Calculator.py

```
3 import sqlite3
4
5 conn = sqlite3.connect('basicdb.sqlite3') #สร้างไฟล์ฐานข้อมูล
6 c = conn.cursor()
7
8
9 # สร้างตารางในการจัดเก็บ
10 c.execute("""CREATE TABLE IF NOT EXISTS member (
11         ID INTEGER PRIMARY KEY AUTOINCREMENT,
12         membercode TEXT,
13         fullname TEXT,
14         tel TEXT,
15         usertype TEXT,
16         points INTEGER ) """)
17
18
```

# Insert ค่าไปใน sqlite

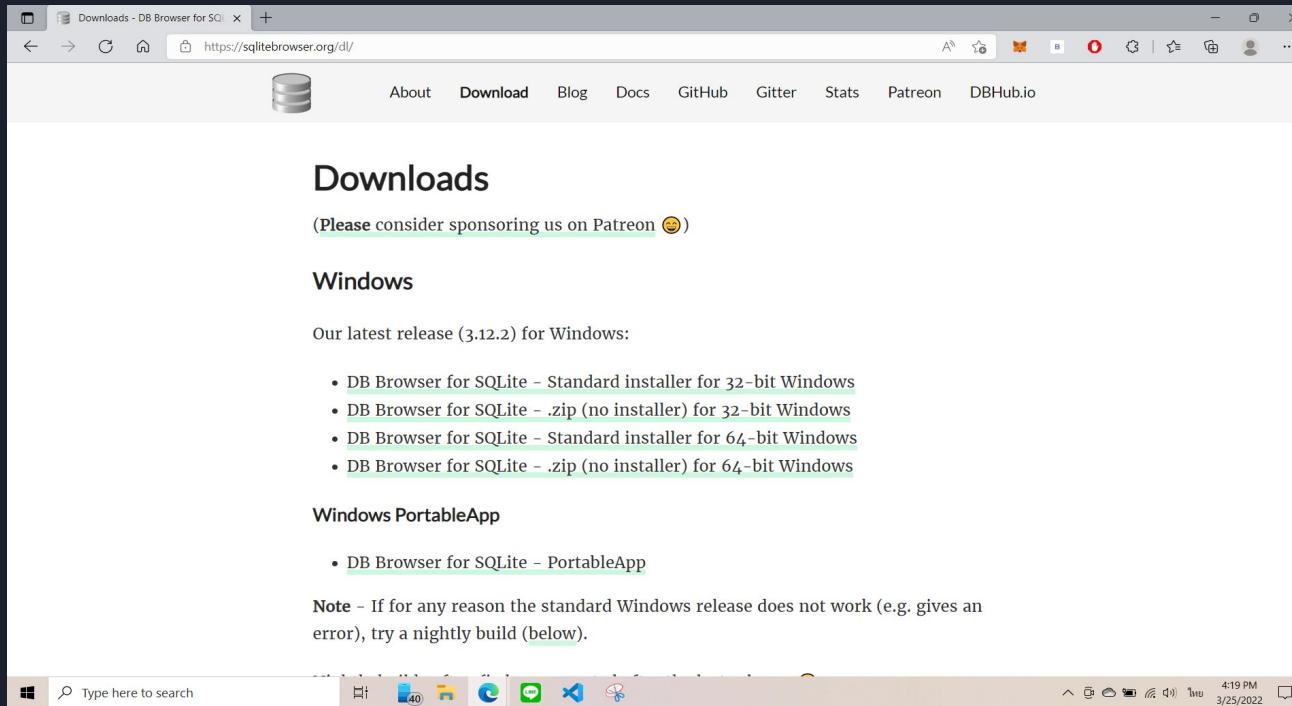


The screenshot shows a Visual Studio Code interface with a dark theme. On the left is the Explorer sidebar, which lists several Python files and other files like 'data.csv' and 'desktop.ini'. The main editor area displays a Python script named 'basic-sql.py'. The code is as follows:

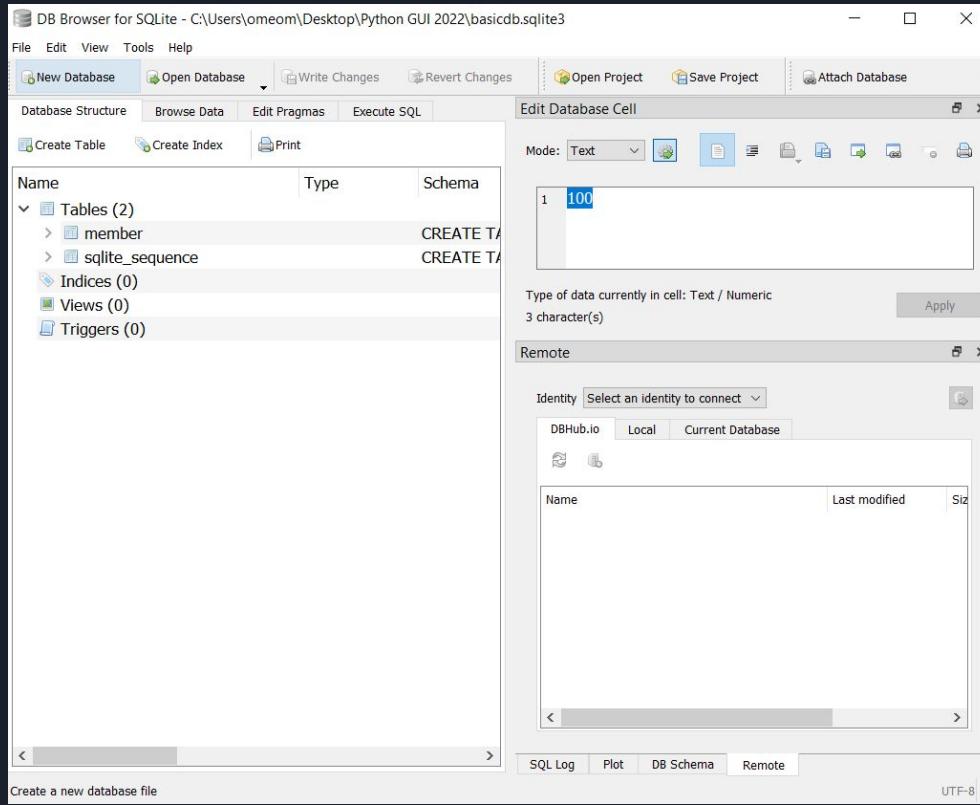
```
Python GUI 2022 > basic-sql.py > View_member
12     fullname TEXT,
13     tel TEXT,
14     usertype TEXT,
15     points INTEGER ) """)
16
17
18
19 def Insert_member(membercode,fullname,tel,usertype,points):
20     # CREATE
21     with conn:
22         command = 'INSERT INTO member VALUES (?,?,?,?,?,?)' # SQL
23         c.execute(command,(None,membercode,fullname,tel,usertype,points))
24     conn.commit() # SAVE DATABASE
25     print('saved')
26
27
28 Insert_member('MB-1002','สมชาย เก่งมาก','0801234568','general',100)
29
30
31
```

The code defines a function 'Insert\_member' that takes five parameters: membercode, fullname, tel, usertype, and points. It uses a context manager 'with conn:' to handle the database connection. Inside the context manager, it executes an 'INSERT INTO' SQL statement with question mark placeholders. The 'c.execute' method is used to run the command with the provided parameters. Finally, it calls 'conn.commit()' to save the changes to the database and prints 'saved' to the console.

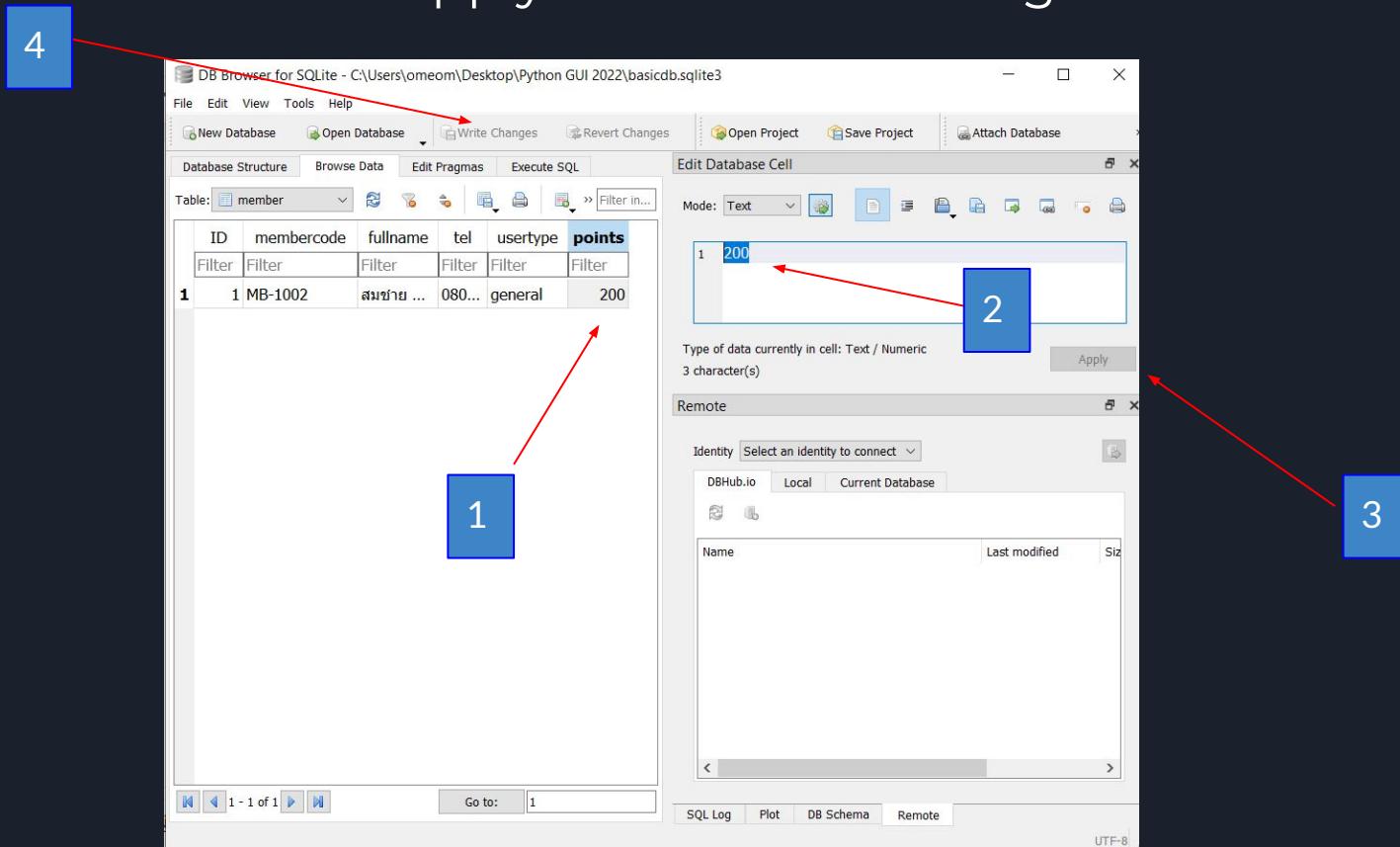
# การเปิดไฟล์ นามสกุล db ให้ไปโหลดโปรแกรมจาก <https://sqlitebrowser.org/dl/> เลือก OS ที่ตัว เองใช้



# จะเลือ hnāต่างโปรแกรมให้ไปที่ Browse Data



# วิธีแก้ข้อมูลผ่าน DB Browser ให้คลิกช่องที่ต้องการแก้ เลยวไปแก้ที่ด้าน ขวา กด Apply และกด Write Changes



## ฟังก์ชัน view

```
def View_member():
    # READ
    with conn:
        command = 'SELECT * FROM member'
        c.execute(command)
        result = c.fetchall()
        print(result)
    return result
```

## ฟังก์ชัน Update\_member

```
def Update_member(ID,field,newvalue):
    # UPDATE
    with conn:
        command = 'UPDATE member SET {} = (?) WHERE ID=(?)'.format(field)
        c.execute(command,([newvalue,ID]))
    conn.commit()
    print('updated')
```

## ฟังก์ชัน Delete\_member

```
def Delete_member(ID):
    # DELETE
    with conn:
        command = 'DELETE FROM member WHERE ID=(?)'
        c.execute(command,([ID]))
    conn.commit()
    print('deleted')
```



# Python GUI 2022(EP.10)

by



Uncle Engineer

ลุงวิศวกร สอนคำนวณ

# สร้างฐานข้อมูลสำหรับสินค้า สร้างไฟล์ productdb.py

```
productdb.py
File Edit Selection View Go Run Terminal Help
productdb.py - Python GUI 2022 - Visual Studio Code

1 import sqlite3
2
3 conn = sqlite3.connect('productdb.sqlite3') #สร้างไฟล์ฐานข้อมูล
4 c = conn.cursor()
5
6 c.execute("""CREATE TABLE IF NOT EXISTS product (
7     ID INTEGER PRIMARY KEY AUTOINCREMENT,
8     productid TEXT,
9     title TEXT,
10    price REAL,
11    image TEXT )""")
12
13
14 def Insert_product(productid,title,price,image):
15     # CREATE
16     with conn:
17         command = 'INSERT INTO product VALUES (?, ?, ?, ?, ?)' # SQL
18         c.execute(command,(None,productid,title,price,image))
19         conn.commit() # SAVE DATABASE
20         print('saved')
21
22
23 def View_product():
24     # READ
25     with conn:
26         command = 'SELECT * FROM product'
27         c.execute(command)
28         result = c.fetchall()
29         print(result)
30         return result
31
32 def View_product_single(productid):
33     # READ
34     with conn:
35         command = 'SELECT * FROM product WHERE productid=(?)'
36         c.execute(command,[productid])
37         result = c.fetchone()
38         print(result)
39         return result
40
41
42 if __name__ == '__main__':
43     # ต้องนำไฟล์มาต่อหน้าไฟล์ที่กำลังรันนี้อยู่ในไฟล์เดียวกัน
44     #Insert_product('CF-1002','เจสเปอร์',45, r'C:\Image\latte.png')
45     # r'Users\uncleengineering\Desktop\GUI\Image\b.a.png'
46     View_product()
```

REAL คือจุดทศนิยม

พึงชั่นนี้เอาไว้เช็คว่าตอนนี้ไฟล์ที่กำลังรันนี้อยู่  
ในไฟล์จริงหรือไม่?

เพิ่มส่วนนี้ใน GUI

```
File Edit Selection View Go Run Terminal Help GUI-Member-Product.py - Python GUI 2022 - Visual Studio Code
```

```
1 #!/usr/bin/python  
2  
3 # Import required libraries  
4 import tkinter  
5 from tkinter import *  
6  
7 # Create the main window  
8 ABGUI = Tk()  
9 ABGUI.iconbitmap('loong.ico')  
10  
11 # Create a menu bar  
12 menubar = Menu(ABGUI)  
13  
14 # File Menu  
15 filemenu = Menu(menubar, tearoff=0)  
16 filemenu.add_command(label='Exit', command=ABGUI.destroy)  
17  
18 # Member Menu  
19 membermenu = Menu(menubar, tearoff=0)  
20 menubar.add_cascade(label='Member', menu=membermenu)  
21  
22 # Product Menu  
23 productmenu = Menu(menubar, tearoff=0)  
24 menubar.add_cascade(label='Product', menu=productmenu)  
25  
26 productmenu.add_command(label='Add Product', command=addmember.command)  
27  
28 # Help Menu  
29 helpmenu = Menu(menubar, tearoff=0)  
30 menubar.add_cascade(label='Help', menu=helpmenu)  
31 contact_url = 'https://uncle-engineer.com'  
32 helpmenu.add_command(label='Contact US', command=lambda: webbrowser.open(contact_url))  
33  
34 def About():  
35     ABGUI = Toplevel()  
36     ABGUI.iconbitmap('loong.ico')  
37  
38 # Import required database module  
39 #import productdb  
40  
41 # View_member()
```

import productdb ด้วย

# สร้างเมนูสำหรับเพิ่มรายการสินค้า

## สร้าง menufunction.py

เครื่องหมายที่ต้องการจะดำเนินการไปใช้ในฟังก์ชันได้

```
menufunction.py
1  from tkinter import *
2  from tkinter import ttk
3  from tkinter import filedialog
4  from productdb import *
5
6  class AddMember:
7
8      def __init__(self):
9          self.v_productid = None
10         self.v_title = None
11         self.v_price = None
12         self.v_imagepath = None
13
14     def popup(self):
15         MGUI = Toplevel()
16         MGUI.geometry('500x600')
17         MGUI.title('Add Product')
18
19         self.v_productid = StringVar()
20         self.v_title = StringVar()
21         self.v_price = StringVar()
22         self.v_imagepath = StringVar()
23
24         L = Label(MGUI, text='เพิ่มรายการสินค้า', font=(None, 30))
25         L.pack(pady=20)
26
27         # -----
28         L = Label(MGUI, text='รหัสสินค้า', font=(None, 20)).pack()
29         E1 = ttk.Entry(MGUI, textvariable= self.v_productid, font=(None, 20))
30         E1.pack(pady=10)
```

# code(ต่อ)

```
32
33     # -----
34     L = Label(MGUI,text='ชื่อสินค้า',font=(None,20)).pack()
35     E2 = ttk.Entry(MGUI,textvariable= self.v_title,font=(None,20))
36     E2.pack(pady=10)
37
38     L = Label(MGUI,text='ราคา',font=(None,20)).pack()
39     E3 = ttk.Entry(MGUI,textvariable= self.v_price,font=(None,20))
40     E3.pack(pady=10)
41
42     L = Label(MGUI,textvariable=self.v_imagepath).pack()
43
44     Bselect = ttk.Button(MGUI, text='เลือกรูปสินค้า ( 50 x 50 px )',command=self.selectfile)
45     Bselect.pack(pady=10)
46
47     Bsave = ttk.Button(MGUI, text='บันทึก',command=self.saveproduct)
48     Bsave.pack(pady=10,ipadx=20,ipady=10)
49
50
51     MGUI.mainloop()
52
52     def selectfile(self):
53         filetypes = (
54             ('PNG', '*.png'),
55             ('All files', '*.*')
56         )
57         select = filedialog.askopenfilename(title='เลือกไฟล์ภาพ',initialdir='/',
58         filetypes=filetypes)
59         self.v_imagepath.set(select)
60         ...
61
62         # focus on top level (next time)
63         self.lift()
64         self.focus_force()
65         self.grab_set()
66         self.grab_release()
67         ...
```

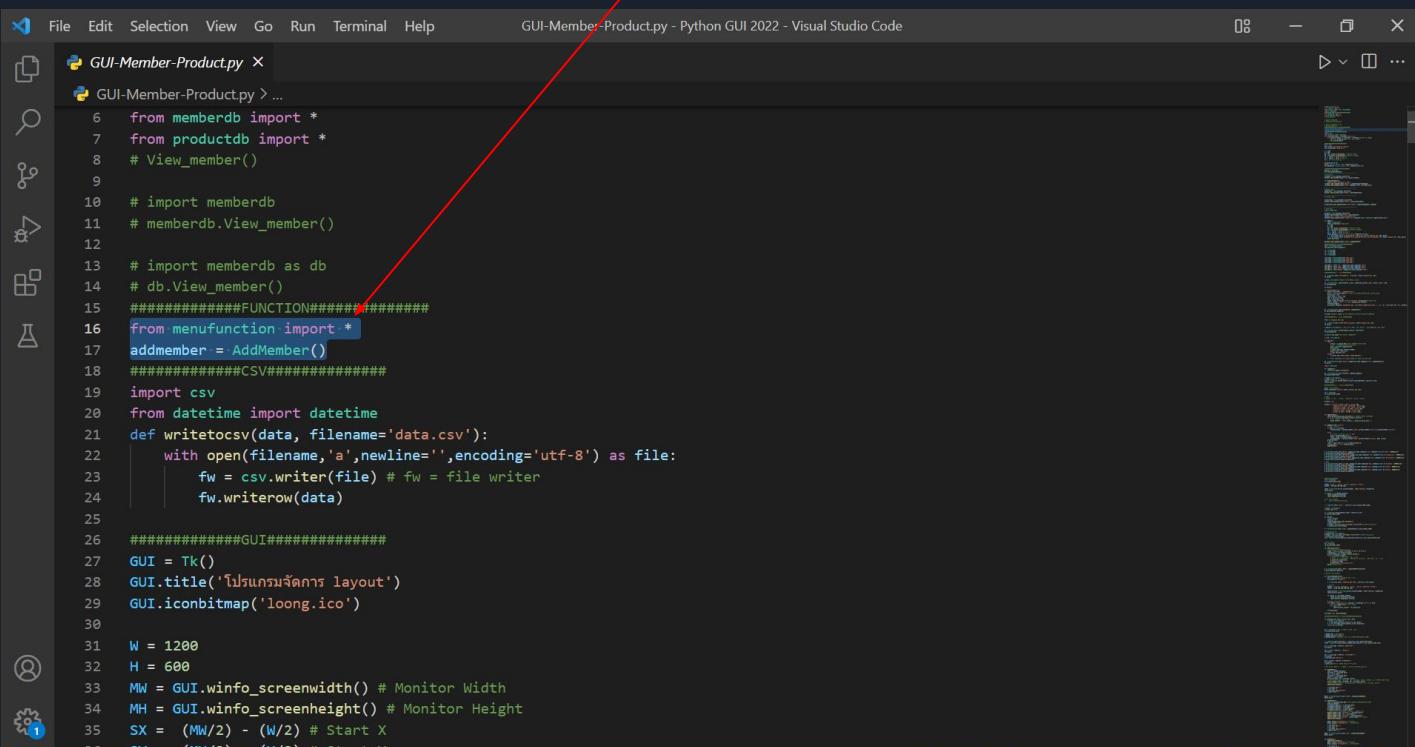
ฟังก์ชันเลือกรูปภาพ

# code(ពីរ)

```
66  
67  
68  
69     def saveproduct(self):  
70         v1 = self.v_productid.get()  
71         v2 = self.v_title.get()  
72         v3 = float(self.v_price.get())  
73         v4 = self.v_imagepath.get()  
74         Insert_product(v1,v2,v3,v4)  
75         self.v_productid.set('')  
76         self.v_title.set([''])  
77         self.v_price.set('')  
78         self.v_imagepath.set('')  
79         View_product()  
80  
81  
82     def command(self):  
83         self.popup()  
84  
85  
86 if __name__ == '__main__':  
87     test = AddMember()
```

ឱ្យ SAVE

# ใส่ code ตามภาพใน GUI



File Edit Selection View Go Run Terminal Help

GUI-Member-Product.py - Python GUI 2022 - Visual Studio Code

```
GUI-Member-Product.py x
GUI-Member-Product.py > ...
6  from memberdb import *
7  from productdb import *
8  # View_member()
9
10 # import memberdb
11 # memberdb.View_member()
12
13 # import memberdb as db
14 # db.View_member()
15 #####FUNCTION#####
16 from menufunction import *
17 addmember = AddMember()
18 #####CSV#####
19 import csv
20 from datetime import datetime
21 def writetocsv(data, filename='data.csv'):
22     with open(filename,'a',newline='',encoding='utf-8') as file:
23         fw = csv.writer(file) # fw = file writer
24         fw.writerow(data)
25
26 ######GUI#####
27 GUI = Tk()
28 GUI.title('โปรแกรมจัดการ layout')
29 GUI.iconbitmap('loong.ico')
30
31 W = 1200
32 H = 600
33 MW = GUI.winfo_screenwidth() # Monitor Width
34 MH = GUI.winfo_screenheight() # Monitor Height
35 SX = (MW/2) - (W/2) # Start X
```



# Python GUI

## 2022(EP.11)

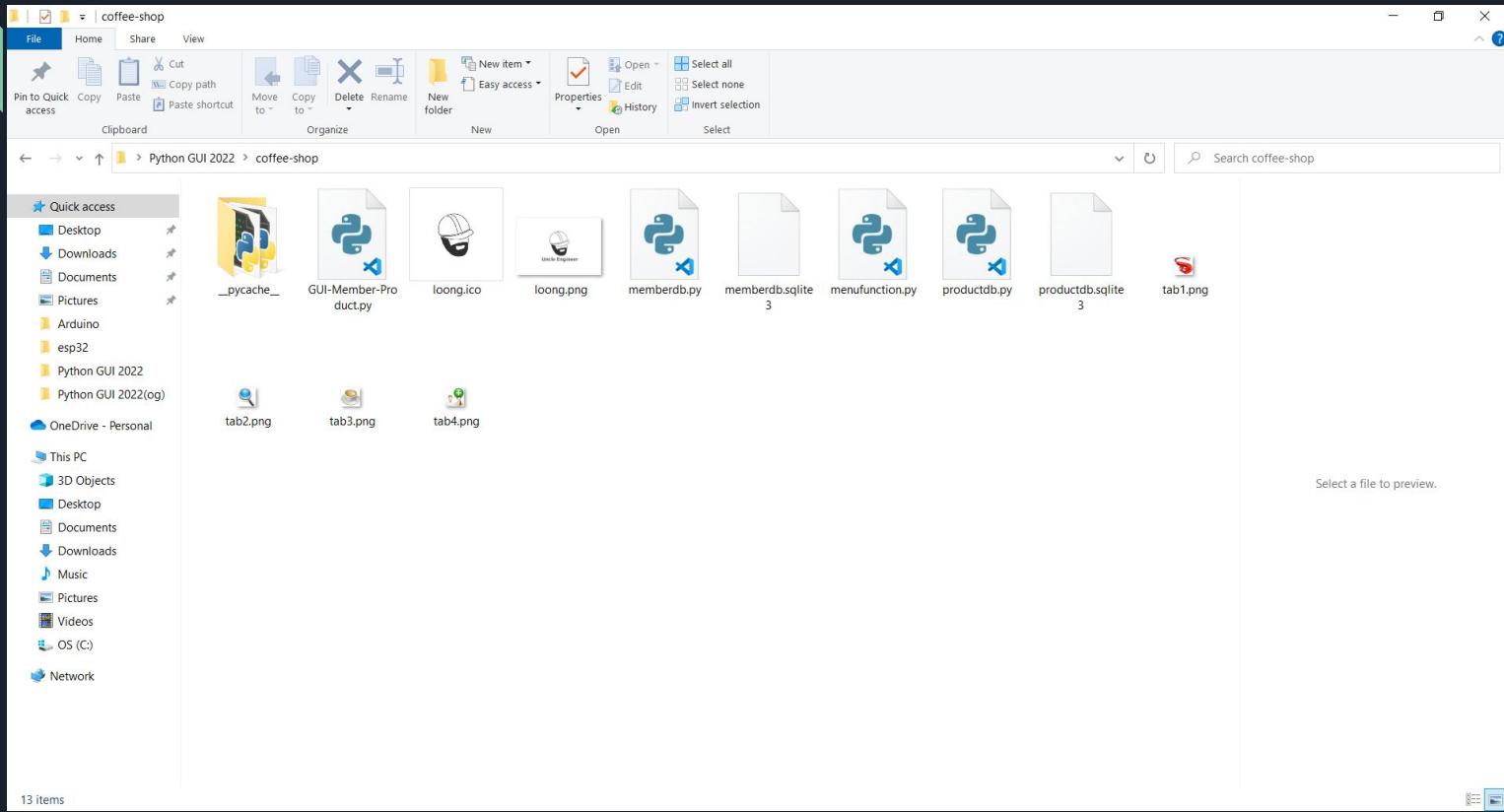
by



Uncle Engineer

ลุงวิศวกร สอนคำนวณ

# สร้าง folder coffee-shop และcopy ไฟล์ที่เกี่ยวข้องมาใส่



ເອົາ tab ກຸ້ງແລະwikiອອກ ລບT1ແລະT2

The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** GUI-Member-Product.py - coffee-shop - Visual Studio Code
- Left Sidebar (Explorer):** COFFEE-SHOP folder containing files: \_pycache\_, GUI-Member-Product.py, loong.ico, memberdb.py, memberdb.sqlite3, memfunction.py, productdb.py, productdb.sqlite3, tab1.png, tab2.png, tab3.png, tab4.png.
- Code Editor:** The main area displays Python code for a Tkinter-based GUI application. The code includes imports for `tk`, `ttk`, and `PhotoImage`. It defines a class `ABGUI` with methods for creating frames, adding tabs, and setting up a menu bar. The code also includes a dictionary `product` mapping drink names to their details (name, price).
- Bottom Status Bar:** Line 126, Col 56, Spaces: 4, UTF-8, CR/LF, Python, 3.10.1 64-bit.
- Bottom Right:** A message box with a blue exclamation icon: "Restart Visual Studio Code to apply the latest update." Below it are three buttons: "Update Now" (blue), "Later" (light blue), and "Release Notes" (white).

# เพิ่ม setting menu ในหน้า GUI

```
69  
70 # -----  
71 # Setting  
72  
73 settingmenu = Menu(menubar,tearoff=0)  
74 menubar.add_cascade(label='Setting',menu=settingmenu)  
75  
76 settingmenu.add_command(label='Product Icon', command=product_icon.command)  
77  
78  
79 # -----  
80 # Help Menu  
81 import webbrowser  
82
```

โปรแกรมจัดการ layout

File Member Product Setting Help



Member

Product Icon



กาแฟ



กาแฟใน



เมล็ดกาแฟ



ชาเขียว



ชาเขียว



ชาเย็น

220422155308

No.	title	price	quantity	total

Total: 0.0

Clear

บันทึก

# เพิ่ม class ProductIcon ใน menufunction.py

ใส่ตาราง

```
class ProductIcon:  
    def __init__(self):  
        self.quantity = None  
        self.table_product = None  
        self.v_radio = None  
  
    def popup(self):  
        # PGUI = Product GUI  
        PGUI = Toplevel()  
        PGUI.geometry('500x500')  
        PGUI.title('สั่งค่า -> ใช้ไอคอนรายการสินค้า')  
  
        # ตารางสินค้า  
        header = ['ID', 'รหัสสินค้า', 'ชื่อสินค้า', 'แสดงไอคอน']  
        hwidth = [50,50,200,70]  
  
        self.table_product = ttk.Treeview(PGUI,columns=header, show='headings',height=15)  
        self.table_product.pack()  
  
        for hd,hw in zip(header,hwidth):  
            self.table_product.column(hd,width=hw)  
            self.table_product.heading(hd,text=hd)  
  
        self.table_product.bind('<Double-1>', self.change_status)  
        self.insert_table()  
        PGUI.mainloop()  
  
    def insert_table(self):  
        data = View_product_table_icon()  
        print(data)  
        for d in data:  
            row = list(d) #convert tuple to list  
            row.append('✓')  
            self.table_product.insert('', 'end', value=row)
```

นำรายการสินค้ามาใช้วเป็น  
icon

# code(ຕົວ)

```
def change_status(self,event=None):  
    select = self.table_product.selection()  
    pid = self.table_product.item(select)['values'][0]  
    print('PID:',pid)  
  
    SGUI = Toplevel() # SGUI = Status GUI  
    SGUI.geometry('400x400')  
  
    self.v_radio = StringVar()  
  
    # Radio  
    RB1 = ttk.Radiobutton(SGUI,text='ໃຫຍ່ໄວຄອນ', variable=self.v_radio, value='show',command=lambda x=None:insert_product_status(int(pid),'show'))  
    RB2 = ttk.Radiobutton(SGUI,text='ນີ້ໃຫຍ່ໄວຄອນ', variable=self.v_radio, value='',command=lambda x=None:insert_product_status(int(pid),''))  
    RB1.pack()  
    RB2.pack()  
    RB1.invoke() #ຕັ້ງຄ່າ default ຂອງ radio  
  
    # Dropdown  
  
    dropdown = ttk.Combobox(SGUI, values=['ໃຫຍ່ໄວຄອນ','ນີ້ໃຫຍ່ໄວຄອນ'])  
    dropdown.pack()  
    dropdown.set('ໃຫຍ່ໄວຄອນ')  
  
    dropdown.bind('<<ComboboxSelected>>',lambda x=None: print(dropdown.get()))  
  
    SGUI.mainloop()  
  
def command(self):  
    self.popup()
```

Radio Button

Dropdown

## เพิ่มฟังก์ชัน View\_product\_table\_icon ใน productdb.py

```
def View_product_table_icon():
    # READ
    with conn:
        command = 'SELECT ID, productid, title FROM product'
        c.execute(command)
        result = c.fetchall()
    print(result)
    return result
```

# สร้าง table อีกชุด ใน productdb เพื่ออ้างอิง table

ฟังก์ชันที่ไว้ใส่ค่า

```
c.execute("""CREATE TABLE IF NOT EXISTS product_status (
    ID INTEGER PRIMARY KEY AUTOINCREMENT,
    product_id INTEGER,
    status TEXT) """)

def insert_product_status(pid,status):
    #pid = product id
    check = view_product_status(pid)
    if check == None:
        with conn:
            command = 'INSERT INTO product_status VALUES (?,?,?,?)'
            c.execute(command,(None,pid,status))
            conn.commit()
            print('status saved')
    else:
        print('pid exist!')
        print(check)
        update_product_status(pid,status)

def view_product_status(pid):
    # READ
    with conn:
        command = 'SELECT * FROM product_status WHERE product_id=(?)'
        c.execute(command,([pid]))
        result = c.fetchone()
    return result

def update_product_status(pid,status):
    # UPDATE
    with conn:
        command = 'UPDATE product_status SET status = (?) WHERE ID=(?)'
        c.execute(command,([status,pid]))
        conn.commit()
    print('updated:',(pid,status))
```

ໂປຣແກຣມຈົດກາ layout

File Member Product Setting Help

CAFE Member

ຕັ້ງຄ່າ -> ໂອງເວັບຄອນນາຍກາສີນຄ້າ

ID	ລະຫັດສິນຄ້າ	ຊື່ສິນຄ້າ	ແສດຖານຂອບຂອນ
1	A-1001	ເມສທ່ຽງເກົ່າ	✓
2	A-3333	ອມເຮົາໄຟ	✓

220422155308

quantity	total
----------	-------

Clear

ບັນທຶກ

โปรแกรมจัดการ layout

File Member Product Setting Help

CAFE Member

รายการ -> รายการคำนวณการสั่งค่า

ID	รหัสสินค้า	ชื่อสินค้า	แสดงเงื่อนไข
1	A-1001	เอสเพรสโซ่	✓
2	A-3333	อเมริกาโน่	✓

220422155308

quantity	total

โปรแกรมจัดการ layout

ใช้รีเฟรช

ไม่ใช้รีเฟรช

ใช้รีเฟรช

ไม่ใช้รีเฟรช

ไม่ใช้รีเฟรช

บันทึก



# Python GUI

## 2022(EP.12)

by



Uncle Engineer

ลุงวิศวกร สอนคำนวณ

- ดึงสถานะติ๊กถูกมาโซ้ว์ในตาราง
- อัพเดตตารางเมื่อมีการปิดหน้าต่าง
- นำรายการสินค้ามาโซ้วเป็น icon
- กรอกข้อมูลคันหาสินค้า (สินค้าที่ไม่มีในปุ่ม)

## ไปที่ไฟล์ menufunction.py เพิ่มโค้ดในฟังก์ชัน change\_status

```
SGUI = Toplevel() # SGUI = Status GUI
SGUI.geometry('400x200')

self.v_radio = StringVar()

# Radio
RB1 = ttk.Radiobutton(SGUI, text='ใช้ไอคอน', variable=self.v_radio,
RB2 = ttk.Radiobutton(SGUI, text='ไม่ใช้ไอคอน', variable=self.v_radio)
RB1.pack(pady=20)
RB2.pack()

check = view_product_status(pid)
print('CHECK:', check)
if check[-1] == 'show':
    RB1.invoke() # ถ้าสถานะเป็น show จะเลือก ใช้ไอคอน
else:
    RB2.invoke()
```

## เพิ่มโค้ดในฟังก์ชัน insert\_table

```
def insert_table(self):
    self.table_product.delete(*self.table_product.get_children())
    data = View_product_table_icon()
    print(data)
    for d in data:
        row = List(d) #convert tuple to list

        check = view_product_status(row[0])

        # ใช้วิธีสถานะของการนำสินค้าไปทำบุญ
        if check[-1] == 'show':
            row.append('✓')

    self.table_product.insert('', 'end', value=row)
```

# ไปที่ไฟล์ menufunction.py เพิ่มโค้ดในฟังก์ชัน change\_status

```
pycache_ GUI-Member-Product.py loong.ico loong.png memberdb.py memberdb.sqlite3 menufunction.py productdb.py productdb.sqlite3 tab1.png tab2.png tab3.png tab4.png transaction.csv
  71     print('CHECK:', check)
  72     if check[-1] == 'show':
  73         RB1.invoke() # ถ้าสถานะเป็น show จะเลือก ใช้วีโอค่อน
  74     else:
  75         RB2.invoke()
  76
  77     # RB2.invoke() #ตั้งค่า default ของ radio
  78
  79     # Dropdown
  80
  81     # dropdown = ttk.Combobox(SGUI, values=['ใช้วีโอค่อน', 'ไม่ใช้วีโอค่อน'])
  82     # dropdown.pack()
  83     # dropdown.set('ใช้วีโอค่อน')
  84
  85     # dropdown.bind('<<ComboboxSelected>>', lambda x=None: print(dropdown.get()))
  86     def check_close():
  87         print('closed')
  88         SGUI.destroy() #ปิดหน้าต่าง
  89         self.insert_table()
  90
  91
  92     SGUI.protocol('WM_DELETE_WINDOW', check_close)
  93
  94     SGUI.mainloop()
  95
  96     def command(self):
  97         self.popup()
```



# นำรายการสินค้ามาโชว์เป็น icon ไปที่ไฟล์ GUI-Member-Product.py คอมเม้นท์ตัว button

```
4
5 ...
6
7 B = ttk.Button(CF1,text='ลาเต้',image=icon_tab3,compound='top',command=lambda m='latte': AddMenu(m))
8 B.grid(row=0,column=0,ipadx=20,ipady=10)
9 B = ttk.Button(CF1,text='คาปูชิโน',image=icon_tab3,compound='top',command=lambda m='cappuccino': AddMenu(m))
0 B.grid(row=0,column=1,ipadx=20,ipady=10)
1 B = ttk.Button(CF1,text='เอสเพรสโซ',image=icon_tab3,compound='top',command=lambda m='espresso': AddMenu(m))
2 B.grid(row=0,column=2,ipadx=20,ipady=10)
3
4 # ROW1
5 B = ttk.Button(CF1,text='ชาเขียว',image=icon_tab3,compound='top',command=lambda m='greentea': AddMenu(m))
6 B.grid(row=1,column=0,ipadx=20,ipady=10)
7 B = ttk.Button(CF1,text='ชาเย็น',image=icon_tab3,compound='top',command=lambda m='icetea': AddMenu(m))
8 B.grid(row=1,column=1,ipadx=20,ipady=10)
9 B = ttk.Button(CF1,text='ชาร้อน',image=icon_tab3,compound='top',command=lambda m='hottea': AddMenu(m))
0 B.grid(row=1,column=2,ipadx=20,ipady=10)
1 ...
```



เพิ่มโคลั่ด



ไปที่ไฟล์ menufunction.py เพิ่มโค้ดในฟังก์ชัน change\_status



ไปที่ไฟล์ menufunction.py เพิ่มโค้ดในฟังก์ชัน change\_status



ไปที่ไฟล์ menufunction.py เพิ่มโค้ดในฟังก์ชัน change\_status