

7/3/2568

การทดลองที่ CE-OOP 06

เรื่อง Graphic User Interface: GUI

จัดทำโดย

นาย คณพัฒน์ รุ่งรพีพรพงษ์ 67200030

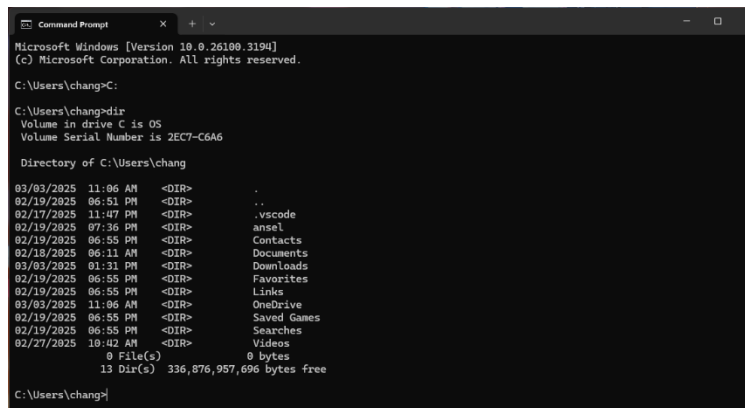
เสนอ

อาจารย์ นภัสรพี สิทธิวัจน์

ตอนที่ 1 ทฤษฎีที่เกี่ยวข้อง

การสร้างส่วนประสานกับผู้ใช้งานแบบกราฟฟิก (Graphic User Interface: GUI) คือการทำให้ผู้ใช้ทำงานได้ง่าย

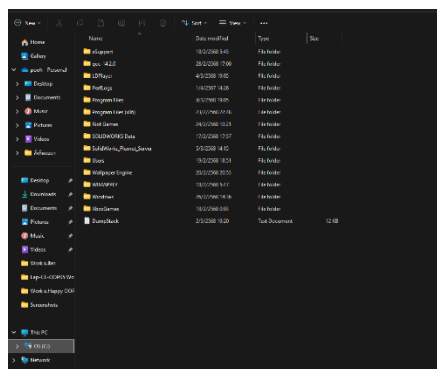
สะดวกสบายด้วยกราฟฟิก สาเหตุที่ต้องสร้าง GUI เป็นเพราะการทำงานกับโค้ดโดยตรงนั้น ผู้ใช้งานต้องมีความเข้าใจและความชำนาญการใช้งานโปรแกรมในขั้นที่ดี แต่ผู้ใช้งานทั่วไปไม่อาจทำงานกับโปรแกรมนี้ได้ การสร้างเป็น GUI จะส่งผลให้ผู้ใช้งานทุกคนทำงานสามารถกับโปรแกรมนั้นได้อย่างง่ายดาย ในรูปที่ 1-1 คือการเปิดดูข้อมูลในไดรฟ์ D ด้วยการพิมพ์คำสั่ง (Command line) คำสั่งนั้นคือ D: dir



รูปที่ 1-1 การเปิดดูข้อมูลในไดรฟ์ D ด้วย Command line

จากรูปที่ 1-1 ข้อมูลที่ปรากฏให้เห็นได้บอกวันที่ใช้งานล่าสุด เวลาสุดท้ายที่เปิดใช้งาน เพิ่มข้อมูลหรือไม่ ขนาดไฟล์

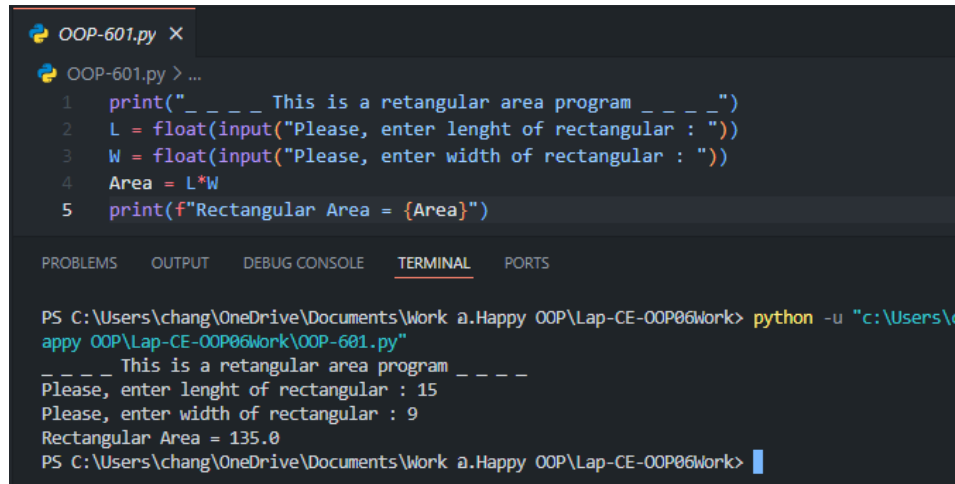
และชื่อไฟล์ (DIR > Directory คือแฟ้มข้อมูล) ผู้อ่านจะเห็นว่าต้องใช้ความรู้พอประมาณสำหรับทำความเข้าใจ หากเปลี่ยนเป็น
การใช้ GUI ผู้อ่านจะเข้าใจข้อมูลได้ง่ายกว่าเดิม ดังรูปที่ 1-2



รูปที่ 1-2 การเปิดดูข้อมูลในไดรฟ์ D ด้วยการใช้ GUI

ต่อจากนี้ผู้เขียนจะกล่าวถึงการใช้งานโปรแกรมภาษา Python โดยตรงเปรียบเทียบกับการใช้งาน GUI ด้วยภาษา Python

เพื่อให้ผู้อ่านได้เห็นการใช้งานที่ลำบากกับสะดวกสบาย ดังรูปที่ 1-3 คือโปรแกรมสำหรับคำนวณพื้นที่สี่เหลี่ยม



```
OOP-601.py X
OOP-601.py > ...
1 print("_ _ _ _ This is a rectangular area program _ _ _ _")
2 L = float(input("Please, enter length of rectangular : "))
3 W = float(input("Please, enter width of rectangular : "))
4 Area = L*W
5 print(f"Rectangular Area = {Area}")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\chang\OneDrive\Documents\Work a.Happy OOP\Lap-CE-OOP06Work> python -u "c:\Users\
appy OOP\Lap-CE-OOP06Work\OOP-601.py"
_ _ _ _ This is a rectangular area program _ _ _ _
Please, enter length of rectangular : 15
Please, enter width of rectangular : 9
Rectangular Area = 135.0
PS C:\Users\chang\OneDrive\Documents\Work a.Happy OOP\Lap-CE-OOP06Work> |
```

รูปที่ 1-3 การคำนวณพื้นที่สี่เหลี่ยมด้วยโค้ดภาษา Python

จากรูปที่ 1-3 ผู้อ่านจะเห็นได้ว่ามีทั้งซอร์สโค้ด (ในส่วนสำหรับเขียนโค้ด) และการใช้งานโค้ด (ในส่วนของ Terminal) ซึ่ง

ต้องป้อนค่าตัวเลข 10 กับ 6 และได้คำตอบเป็น 60 ผู้ใช้งานทั่วไปไม่สามารถเข้าใจและใช้งานได้ แต่ผู้อ่านได้เรียนรู้การเขียนโค้ด

ภาษา Python บนโปรแกรม VS Code แล้ว ผู้อ่านเข้าใจและสั่งให้โค้ดทำงานได้อย่างสบาย

จากนี้ ผู้เขียนจะปรับปรุงโค้ดของรูปที่ 1-3 ให้กลายเป็น GUI ผู้เขียนอยากให้ผู้อ่านได้เข้าใจและรับรู้ก่อนว่า การเปลี่ยน

โค้ดที่เขียนอย่างคุ้นเคยให้กลายเป็นการเขียนโค้ดเพื่อสร้าง GUI จะมีความซับซ้อนเพิ่มขึ้นมีหลายคำสั่งที่เคยเขียนอย่างคุ้นเคย

จะต้องเปลี่ยนให้เหมาะกับสร้าง GUI ผู้อ่านลองนึกถึงการแปลภาษาไทยที่คุ้นเคยให้กลายเป็นภาษาอังกฤษ มันต้องมีการแก้ไขคำ

และไวยากรณ์บ้าง คิดอย่างนี้จะได้เข้าใจง่ายมากขึ้น

มาถึงตรงนี้ผู้อ่านจะมีความเข้าใจมากขึ้นและเห็นภาพมากขึ้นแล้ว ต่อจากนี้ผู้เขียนจะนำเสนอคำสั่งที่เกี่ยวข้องกับการ

สร้าง GUI ซึ่งผู้อ่านจะได้พอเจอบ่อยมาก บ่อย ค่อนข้างบ่อย เจอเป็นระยะ และพบเจอไม่บ่อย ทั้งหมดที่ว่ามัล้วนคู่ควรให้ผู้อ่าน

ตอนที่ 2 พื้นฐานการสร้าง GUI

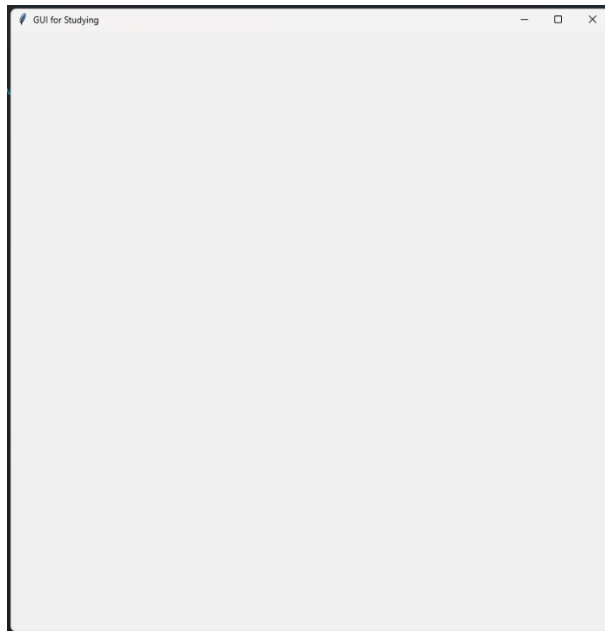
1. ให้ผู้อ่านเขียนโค้ดตามตัวอย่างในรูปที่ 2-1 ซึ่งเป็นโค้ดสำหรับสร้างหน้าต่าง 1 หน้าเพื่อใช้เป็นหน้าต่างสำหรับทำงาน (GUI)

```
OOP-602.py > ...
1  from tkinter import *
2
3  myWindow = Tk() #สร้างหน้าต่างสำหรับทำงาน (สร้างหน้า GUI)
4  myWindow.title('GUI for Studying') #ข้อความที่มมซ้ายบน เพื่อบอกว่าหน้าต่างนี้แสดงอะไร
5  myWindow.geometry('800x800') #กำหนดขนาดของหน้าต่างสำหรับทำงาน หน่วยเป็นพิกเซล
6
7  myWindow.mainloop() #สั่งให้หน้าต่างสำหรับทำงาน ที่ชื่อ myWindow ทำงานจนกว่าผู้ใช้งานจะกดปิดเอง
```

รูปที่ 2-1 การเริ่มต้นสร้างหน้าต่างสำหรับทำงาน (First GUI)

2. ผลการรันโปรแกรมเป็นอย่างไร

Ans: เป็นดังนี้



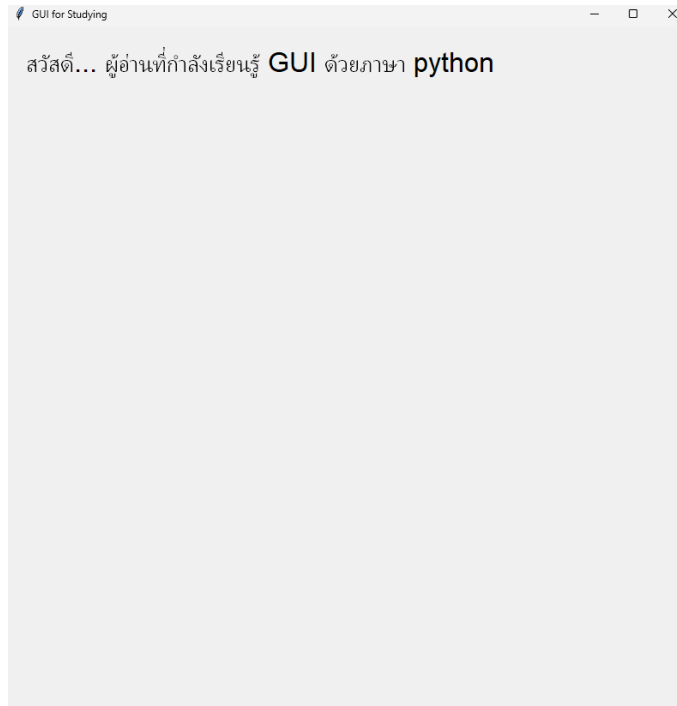
3. ให้ผู้อ่านเขียนโค้ดตามตัวอย่างในรูปที่ 2-2 ซึ่งเป็นการนำโค้ดรูปที่ 2-1 มาพัฒนาต่อ โดยเติมการแสดงข้อความเข้าไป

```
OOP-601.py  OOP-602.py  OOP-6023.py
OOP-602.py > ...
1  from tkinter import *
2
3  myWindow = Tk() #สร้างหน้าต่างสำหรับทำงาน (สร้างหน้า GUI)
4  myWindow.title('GUI for Studying') #ข้อความที่มมซ้ายบน เพื่อบอกว่าหน้าต่างนี้แสดงอะไร
5  myWindow.geometry('800x800') #กำหนดขนาดของหน้าต่างสำหรับทำงาน หน่วยเป็นพิกเซล
6
7  myMessagell = Label(myWindow, text='สวัสดี... ผู้อ่านที่กำลังเรียนรู้ GUI ด้วยภาษา python', font='AngsanaNew 24')
8  myMessagell.place(x=20, y=20)
9
10 myWindow.mainloop() #สั่งให้หน้าต่างสำหรับทำงาน ที่ชื่อ myWindow ทำงานจนกว่าผู้ใช้งานจะกดปิดเอง
```

รูปที่ 2-2 การสร้างข้อความบนหน้าต่างสำหรับทำงาน

4. ผลการรันโปรแกรมเป็นอย่างไร

Ans:



5. โปรดอธิบายโค้ดบรรทัดที่ 7-8 มาให้เข้าใจ

Ans: บรรทัดที่ 7 สร้างกล่อง ข้อความขึ้นมา 1 ข้อความ ใน ตัวแปร myMessage1 แสดงบนหน้าต่างชื่อ myWindow

บรรทัดที่ 8 วางข้อความ ที่พิกัด x=20, y=20

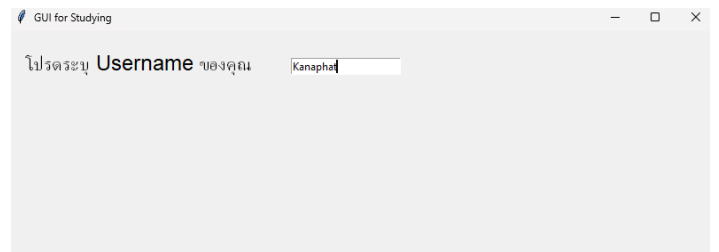
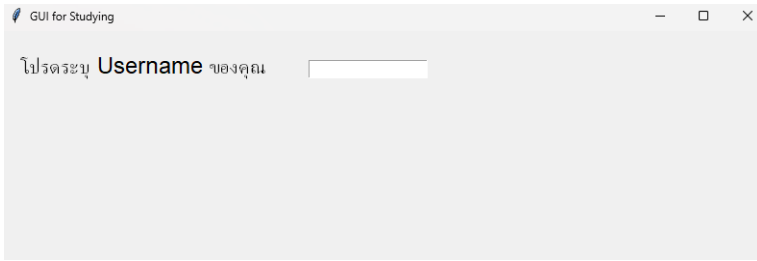
6. ให้ผู้อ่านเขียนโค้ดตามตัวอย่างในรูปที่ 2-3 ซึ่งเป็นการนำโค้ดรูปที่ 2-2 มาพัฒนาต่อ โดยเพิ่มช่องรับข้อมูลเข้าไป

```
OOP-605.py • OOP-602.py X
OOP-602.py > ...
1  from tkinter import *
2
3  myWindow = Tk() #สร้างหน้าต่างสำหรับทำงาน (สร้างหน้า GUI)
4  myWindow.title('GUI for Studying') #ข้อความที่มมาย้ายบน เพื่อบอกว่าหน้าต่างนี้แสดงอะไร
5  myWindow.geometry('800x800') #กำหนดขนาดของหน้าต่างสำหรับทำงาน หน่วยเป็นพิกเซล
6
7  myMessage1 = Label(myWindow, text='โปรดระบุ Username ของคุณ', font='AngsanaNew 18')
8  myMessage1.place(x = 20, y = 20)
9  myBox1 = Entry(myWindow)
10 myBox1.place(x = 320, y = 30)
11
12 myWindow.mainloop() #สั่งให้หน้าต่างสำหรับทำงาน ที่ชื่อ myWindow ทำงานจนกว่าผู้ใช้งานจะกดปิดเอง
```

รูปที่ 2-3 การสร้างข้อความและช่องรับข้อมูลบนหน้าต่างสำหรับทำงาน

7. ผลการรันโปรแกรมเป็นอย่างไร

Ans:



8. คำสั่ง Entry มีไว้เพื่ออะไร

Ans: Entry มีไว้รับข้อความ จาก User ไปทำงาน

9. บรรทัดที่ 9 Entry(myWindow) คำสั่งนี้หมายความว่าอย่างไร โปรดอธิบายให้เข้าใจ

Ans: สร้างกล่องข้อความว่างเปล่าขึ้นมา 1 กล่อง ในหน้าต่าง myWindow ในตัวแปล myBox1

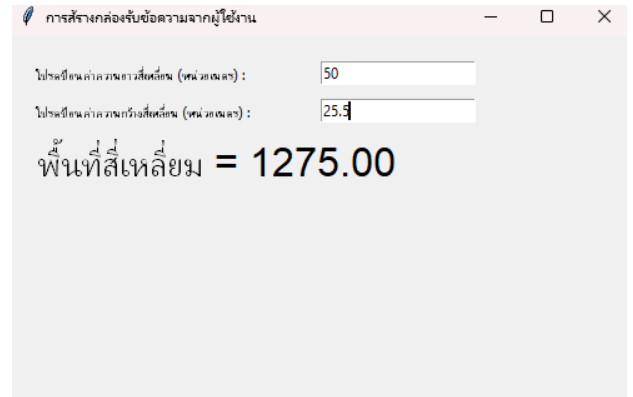
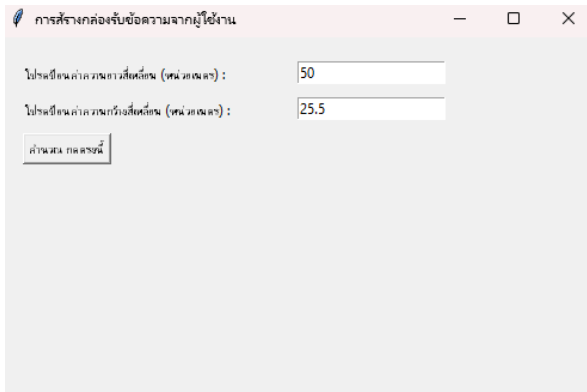
10. ให้ผู้อ่านเขียนโค้ดตามรูปที่ 2-4 ซึ่งเป็นโค้ดที่มีการรับข้อมูล สร้างปุ่ม สร้างฟังก์ชัน (เมื่อกดปุ่มแล้วให้ไปทำงานที่ฟังก์ชัน)

```
OOP-601.py OOP-602.py OOP-603.py
OOP-6023.py > ...
1 import tkinter as tk
2
3 win = tk.Tk()
4 win.title('การสร้างกล่องรับข้อความจากผู้ใช้งาน')
5 win.geometry('500x300')
6 #สร้างข้อความ เพื่อบอกข้อมูล
7 L1 = tk.Label(win, text = 'โปรดป้อนค่าความยาวสี่เหลี่ยม (หน่วยเมตร) : ')
8 L1.place(x=20, y=20)
9 W = tk.Label(win, text = 'โปรดป้อนค่าความกว้างสี่เหลี่ยม (หน่วยเมตร) : ')
10 W.place(x=20, y=50)
11 #สร้างกล่องรับข้อความ
12 Box_L = tk.Entry(win)
13 Box_L.place(x=250, y=20)
14 Box_W = tk.Entry(win)
15 Box_W.place(x=250, y=50)
16 #สร้างการทำงานให้กับปุ่ม cmd
17 def Rectangular_Area():
18     value1 = float(Box_L.get())
19     value2 = float(Box_W.get())
20     Ans = value1*value2
21     txt3 = tk.Label(win, text = 'พื้นที่สี่เหลี่ยม = %.2f' %Ans, font = 'AngsanaNew 24')
22     txt3.place(x = 20, y = 80)
23
24 cmd = tk.Button(win, text = 'คำนวณ กดตรงนี้', command = Rectangular_Area)
25 cmd.place(x = 20, y = 80)
26
27 win.mainloop()
```

รูปที่ 2-4 การสร้าง GUI สำหรับคำนวณพื้นที่สี่เหลี่ยม

11. เมื่อรันโปรแกรมแล้ว ผู้อ่านต้องป้อนข้อมูลเป็นตัวเลขเข้าไปในช่องรับข้อมูล ผลการรันโปรแกรมเป็นอย่างไร

Ans:



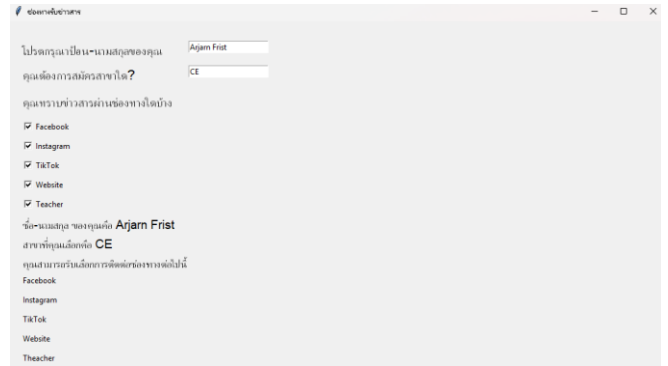
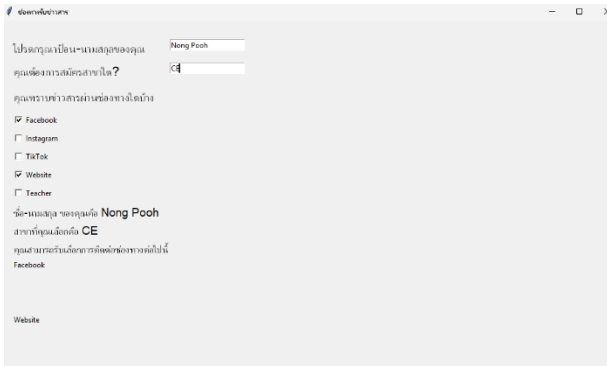
12. ให้ผู้อ่านเขียนโค้ดตามรูปที่ 2-5 ซึ่งเป็นโค้ดที่ใช้ Check Button เพื่อเลือกการกระทำได้หลายอย่าง

```
OOP-604.py X c.c
C:\Users\chang > OneDrive > Documents > Work a.Happy OOP > Lap-CE-OOP06Work > OOP-604.py > ...
1 from tkinter import *
2 from tkinter import IntVar
3
4 myWindow = Tk()
5 myWindow.title("ห้องทางวิชาการ")
6 myWindow.geometry("1024x720")
7 myMessage1 = Label(myWindow, text="โปรดกรอกนาม-นามสกุลของคุณ", font="AngsanaNew 16")
8 myMessage1.place(x=20, y=30)
9 myDataBox1 = Entry(myWindow)
10 myDataBox1.place(x=280, y=30)
11 myMessage2 = Label(myWindow, text="คุณต้องการสมัครสาขาใด?", font="AngsanaNew 16")
12 myMessage2.place(x=20, y=68)
13 myDataBox2 = Entry(myWindow)
14 myDataBox2.place(x=280, y=68)
15
16 def DisplayInfo():
17     userName = myDataBox1.get()
18     text1 = Label(myWindow, text="ชื่อ-นามสกุล ของคุณคือ " + userName, font="Angsana 14")
19     text1.place(x=20, y=300)
20     userMajor = myDataBox2.get()
21     text2 = Label(myWindow, text="สาขาที่คุณเลือกคือ " + userMajor, font="Angsana 14")
22     text2.place(x=20, y=330)
23     text9 = Label(myWindow, text="คุณสามารถรับการติดต่อของทางต่อไปนี้", font="Angsana 14")
24     text9.place(x=20, y=360)
25     if (myCheck1.get() == 1):
26         text4 = Label(myWindow, text="Facebook")
27         text4.place(x=20, y=390)
28     if (myCheck2.get() == 1):
29         text5 = Label(myWindow, text="Instagram")
30         text5.place(x=20, y=420)
31     if (myCheck3.get() == 1):
32         text6 = Label(myWindow, text="TikTok")
33         text6.place(x=20, y=450)
34     if (myCheck4.get() == 1):
35         text7 = Label(myWindow, text="Website")
36         text7.place(x=20, y=480)
37     if (myCheck5.get() == 1):
38         text8 = Label(myWindow, text="Teacher")
39         text8.place(x=20, y=510)
40
41 myCheck1 = IntVar();myCheck2 = IntVar();myCheck3 = IntVar();myCheck4 = IntVar();myCheck5 = IntVar()
42 myMessage3 = Label(myWindow, text="คุณทราบข่าวสารผ่านช่องทางใดบ้าง", font="AngsanaNew 16")
43 myMessage3.place(x=20, y=110)
44
45 myCheckButton1 = Checkbutton(myWindow, text="Facebook", variable=myCheck1, onvalue=1, offvalue=0, command=DisplayInfo)
46 myCheckButton1.place(x=20, y=150); myCheckButton1.var = myCheck1
47 myCheckButton2 = Checkbutton(myWindow, text="Instagram", variable=myCheck2, onvalue=1, offvalue=0, command=DisplayInfo)
48 myCheckButton2.place(x=20, y=180); myCheckButton2.var = myCheck2
49 myCheckButton3 = Checkbutton(myWindow, text="TikTok", variable=myCheck3, onvalue=1, offvalue=0, command=DisplayInfo)
50 myCheckButton3.place(x=20, y=210); myCheckButton3.var = myCheck3
51 myCheckButton4 = Checkbutton(myWindow, text="Website", variable=myCheck4, onvalue=1, offvalue=0, command=DisplayInfo)
52 myCheckButton4.place(x=20, y=240); myCheckButton4.var = myCheck4
53 myCheckButton5 = Checkbutton(myWindow, text="Teacher", variable=myCheck5, onvalue=1, offvalue=0, command=DisplayInfo)
54 myCheckButton5.place(x=20, y=270); myCheckButton5.var = myCheck5
55
56 myWindow.mainloop()
```

รูปที่ 2-4 การสร้าง GUI สำหรับคำนวณพื้นที่สี่เหลี่ยม

13. ผลลัพธ์การรันโปรแกรมเป็นอย่างไร (ขอผลลัพธ์ 2 ครั้งที่ใส่ข้อมูลไม่เหมือนกัน)

Ans:



14. จากโค้ดรูปที่ 2-4 มีโค้ดอยู่บรรทัดหนึ่งที่แสดงข้อความไม่ถูกต้อง นั่นคือโค้ดบรรทัดที่เท่าไร และต้องแก้ข้อความจากอะไรให้เป็นอะไรถึงจะถูกต้อง

Ans:

15. โปรดอธิบายโค้ดบรรทัดที่ 41 บรรทัดที่ 45 กับ 46 มาให้เข้าใจ

Ans: โค้ดบรรทัดที่ 38 เปลี่ยนจากคำว่า Instagram เปลี่ยนเป็น Teacher

ตอนที่ 3 การประยุกต์ใช้ GUI กับสถิติ

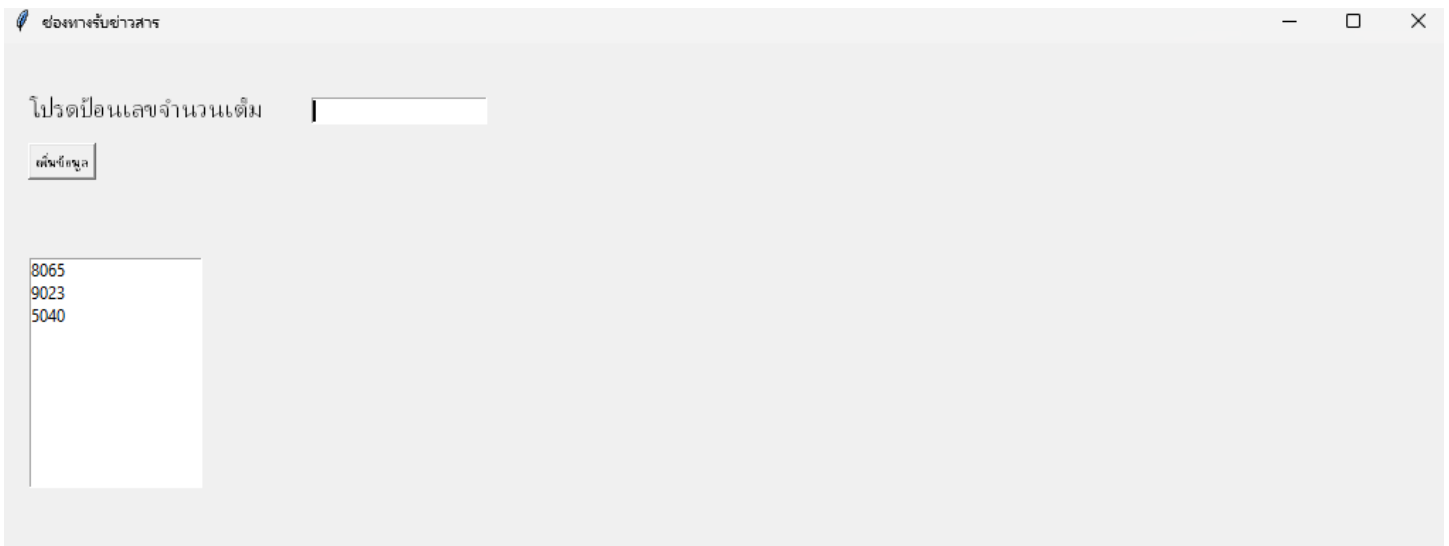
16. ให้ผู้อ่านเขียนโค้ดตามรูปที่ 3-1 ซึ่งเป็นการป้อนข้อมูลเก็บไว้ในตัวแปรแบบ List และนำข้อมูลออกมาแสดง

```
OOP-605.py X
OOP-605.py > ...
1  from tkinter import *
2  from tkinter import IntVar
3
4  myWindow = Tk()
5  myWindow.title('ช่องทางรับข่าวสาร')
6  myWindow.geometry('1024x720')
7  myMessage = Label(myWindow, text = 'โปรดป้อนเลขจำนวนเต็ม', font = 'AngsanaNew 16' )
8  myMessage.place(x = 20, y = 30)
9  myDataBox1 = Entry(myWindow)
10 myDataBox1.place(x = 220, y = 38)
11
12 def CollectedData():
13     dataList = myDataBox1.get()
14     if dataList:
15         myListBox.insert(END, dataList)
16         myDataBox1.delete(0, END)
17
18 cmd1 = Button(myWindow, text = 'เพิ่มข้อมูล', command = CollectedData)
19 cmd1.place(x = 20, y = 70)
20
21 myListBox = Listbox(myWindow)
22 myListBox.place(x = 20, y = 150)
23
24 myWindow.mainloop()
```

รูปที่ 3-1 การสร้าง GUI สำหรับเก็บข้อมูลไว้ในตัวแปรแบบ List

17. ให้ผู้อ่านรันโปรแกรมและลองป้อนตัวเลขเข้าไปสัก 3 ค่า ผลลัพธ์เป็นอย่างไร

Ans:



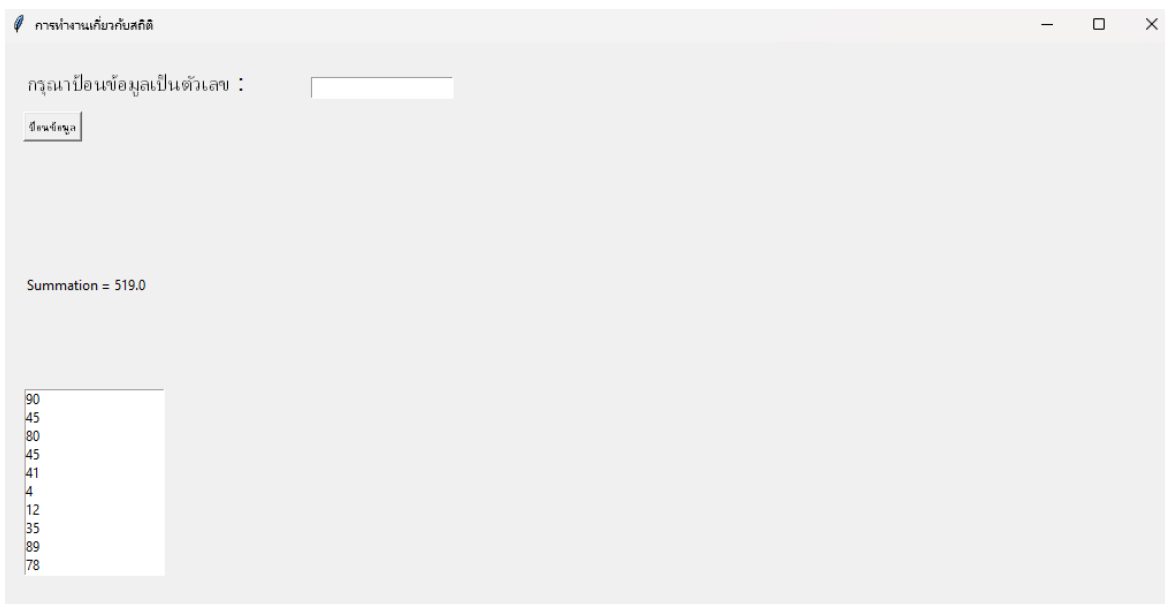
18. ผู้เขียนจะนำโค้ดรูปที่ 3-1 มาพัฒนาเพิ่ม โดยมีเจตนาทำให้หาค่าผลรวมได้ ขอให้ผู้อ่านเขียนโค้ดตามรูปที่ 3-2

```
OOP-605.py OOP-606.py X
OOP-606.py > ...
1 from tkinter import *
2 from tkinter import IntVar
3
4 # สร้างฟังก์ชันสำหรับเพิ่มข้อมูล (Function to add data)
5 def add_data():
6     data = myBox1.get()
7     if data:
8         value = float(data)
9         items.append(value)
10        displayBox.insert(END, data)
11        myBox1.delete(0, END)
12        mySummation()
13
14 # สร้างฟังก์ชันสำหรับคำนวณผลรวม (Function to summation display)
15 def mySummation():
16     total_sum = sum(items)
17     sum_label.config(text=f'Summation = {total_sum}')
18
19 # สร้างหน้าต่างหลักสำหรับทำงาน (Create the main window)
20 win = Tk()
21 win.title('การทำงานเกี่ยวกับสถิติ')
22 win.geometry('1024x720')
23
24 # สร้างข้อความเพื่อแสดงที่ต้องการ (Create the message for information)
25 myText1 = Label(win, text='กรุณาป้อนข้อมูลเป็นตัวเลข : ', font='AngsanaNew 16')
26 myText1.place(x=20, y=20)
27
28 # สร้างกล่องป้อนข้อมูล (Create an entry widget for input)
29 myBox1 = Entry(win)
30 myBox1.place(x=270, y=30)
31
32 # สร้างปุ่มสำหรับป้อนข้อมูล (Create a button to add data)
33 myButton1 = Button(win, text='ป้อนข้อมูล', command=add_data)
34 myButton1.place(x=20, y=60)
35
36 # สร้างกล่องแสดงข้อมูลของ list (Create a listbox to display the data)
37 displayBox = Listbox(win)
38 displayBox.place(x=20, y=300)
39
40 # สร้างข้อความสำหรับแสดงผลรวม (Create a label to display summation)
41 sum_label = Label(win, text='Summation = 0.00')
42 sum_label.place(x=20, y=200)
43
44 # สร้างตัวแปร list เพื่อเก็บข้อมูล (Initialize an empty list to store data)
45 items = []
46
47 # เริ่มการทำงาน (Run the application)
48 win.mainloop()
```

รูปที่ 3-2 การสร้าง GUI สำหรับเก็บข้อมูลไว้ในตัวแปรแบบ List และหาค่าผลรวม

19. ผลการรันโปรแกรมเป็นอย่างไร (ให้ผู้อ่านป้อนตัวเลขเข้าไป 10 ค่า)

Ans:



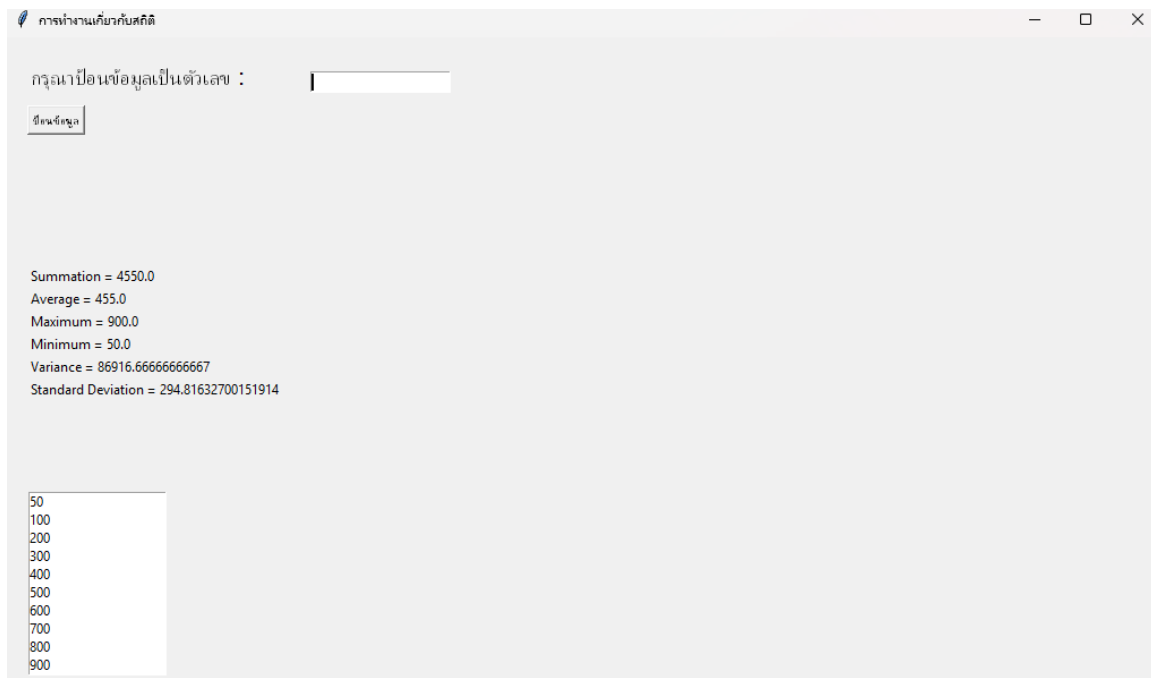
Checkpoint1 ให้ผู้อ่านเพิ่มการหาค่าเฉลี่ย หาค่าสูงสุด หาค่าต่ำสุด ค่าความแปรปรวน และค่าเบี่ยงเบนมาตรฐาน เมื่อเขียน

โค้ดเสร็จแล้ว โปรตแสดงผลลัพธ์และโค้ดให้เห็นอย่างชัดเจน (ป้อนตัวเลขสัก 10 ค่า)

```
OOP-605.py OOP-606.py x
OOP-606.py > ...
1 from tkinter import *
2 from tkinter import IntVar
3 import statistics
4
5 # สร้างฟังก์ชันสำหรับเพิ่มข้อมูล (Function to add data)
6 def add_data():
7     data = myBox1.get()
8     if data:
9         value = float(data)
10        items.append(value)
11        displayBox.insert(END, data)
12        myBox1.delete(0, END)
13        mySummation()
14        myAverage()
15        myMaximum()
16        myMinimum()
17        myVariance()
18        myStdev()
19
20 # สร้างฟังก์ชันสำหรับคำนวณผลรวม (Function to summation display)
21 def mySummation():
22     total_sum = sum(items)
23     sum_label.config(text=f'Summation = {total_sum}')
24
25 def myAverage():
26     total_Avg = sum(items)/len(items)
27     Avg_label.config(text=f'Average = {total_Avg}')
28
29 def myMaximum():
30     total_Max = max(items)
31     Max_label.config(text=f'Maximum = {total_Max}')
32
33 def myMinimum():
34     total_Min = min(items)
35     Min_label.config(text=f'Minimum = {total_Min}')
36
37 def myVariance():
38     if len(items) > 1: # ต้องมีข้อมูลอย่างน้อย 2 ค่า
39         total_Var = statistics.variance(items)
40         Var_label.config(text=f'Variance = {total_Var}')
41     else:
42         Var_label.config(text='Variance = Not enough data') # แสดงข้อความแจ้งเตือน ว่าข้อมูลยังไม่พอ
43
44 def myStdev():
45     if len(items) > 1: # ต้องมีข้อมูลอย่างน้อย 2 ค่า
46         total_Stdev = statistics.stdev(items)
47         Stdev_label.config(text=f'Standard Deviation = {total_Stdev}')
48     else:
49         Stdev_label.config(text='Standard Deviation = Not enough data') # แสดงข้อความแจ้งเตือน ว่าข้อมูลยังไม่พอ
```

```
21
22 # สร้างหน้าต่างหลักสำหรับทำงาน (Create the main window)
23 win = Tk()
24 win.title('การทำงานเกี่ยวกับสถิติ')
25 win.geometry('1024x720')
26
27 # สร้างข้อความเพื่อแสดงที่ต้องการ (Create the message for information)
28 myText1 = Label(win, text='กรุณาป้อนข้อมูลเป็นตัวเลข : ', font='AngsanaNew 16')
29 myText1.place(x=20, y=20)
30
31 # สร้างกล่องป้อนข้อมูล (Create an entry widget for input)
32 myBox1 = Entry(win)
33 myBox1.place(x=270, y=30)
34
35 # สร้างปุ่มสำหรับป้อนข้อมูล (Create a button to add data)
36 myButton1 = Button(win, text='ป้อนข้อมูล', command=add_data)
37 myButton1.place(x=20, y=60)
38
39 # สร้างกล่องแสดงข้อมูลของ list (Create a listbox to display the data)
40 displayBox = Listbox(win)
41 displayBox.place(x=20, y=400)
42
43 # สร้างข้อความสำหรับแสดงผลรวม (Create a label to display summation)
44 sum_label = Label(win, text='Summation = 0.00')
45 sum_label.place(x=20, y=200)
46 Avg_label = Label(win, text='Average = 0.00')
47 Avg_label.place(x=20, y=220)
48 Max_label = Label(win, text='Maximum = 0.00')
49 Max_label.place(x=20, y=240)
50 Min_label = Label(win, text='Minimum = 0.00')
51 Min_label.place(x=20, y=260)
52 Var_label = Label(win, text='Variance = 0.00')
53 Var_label.place(x=20, y=280)
54 Stdev_label = Label(win, text='Standard Deviation = 0.00')
55 Stdev_label.place(x=20, y=300)
56
57 # สร้างตัวแปร list เพื่อเก็บค่าข้อมูล (Initialize an empty list to store data)
58 items = []
59
60 # เริ่มการทำงาน (Run the application)
61 win.mainloop()
```

Result:



Checkpoint2 จากโค้ดรูปที่ 3-2 การหาค่าผลรวมได้เกิดขึ้นทันทีหลังจากการป้อนตัวเลขเข้าไป ให้ผู้อ่านแก้ไขโค้ดเป็น กดปุ่มเพื่อหาค่าผลรวม เมื่อกดปุ่มค่าผลรวมจึงแสดงผล ต่อจากนั้นเพิ่มปุ่มหาค่าเฉลี่ย ค่าสูงสุด ค่าต่ำสุด ค่าความแปรปรวน และค่าเบี่ยงเบนมาตรฐาน เมื่อกดปุ่มเหล่านี้ค่าที่ต้องการจะแสดงออกบน GUI ที่สร้างไว้ โปรดแสดงโค้ดและผลลัพธ์ให้ชัดเจน

```
OOP-605.py OOP-607.py X
OOP-607.py > ...
1 from tkinter import *
2 from tkinter import IntVar
3 import statistics
4
5 # สร้างฟังก์ชันสำหรับเพิ่มข้อมูล (Function to add data)
6 def add_data():
7     data = myBox1.get()
8     if data:
9         value = float(data)
10        items.append(value)
11        displayBox.insert(END, data)
12        myBox1.delete(0, END)
13
14 def Find_sum():
15     mySummation()
16 def Find_Avg():
17     myAverage()
18 def Find_Max():
19     myMaximum()
20 def Find_Min():
21     myMinimum()
22 def Find_Var():
23     myVariance()
24 def Find_Stdev():
25     myStdev()
26
27
28 # สร้างฟังก์ชันสำหรับคำนวณผลรวม (Function to summation display)
29 def mySummation():
30     if len(items) >= 1:
31         total_sum = sum(items)
32         sum_label.config(text=f'Summation = {total_sum}')
33     else:
34         sum_label.config(text = 'Not enough data')
35
36 def myAverage():
37     if len(items) >= 1:
38         total_Avg = sum(items)/len(items)
39         Avg_label.config(text=f'Average = {total_Avg}')
40     else:
41         Avg_label.config(text = 'Not enough data')
42
```

```
OOP-605.py OOP-607.py X
OOP-607.py > ...
43 |
44 def myMaximum():
45     if len(items) >= 1:
46         total_Max = max(items)
47         Max_label.config(text=f'Maximum = {total_Max}')
48     else:
49         Max_label.config(text = 'Not enough data')
50
51
52 def myMinimum():
53     if len(items) >= 1:
54         total_Min = min(items)
55         Min_label.config(text=f'Minimum = {total_Min}')
56     else:
57         Min_label.config(text = 'Not enough data')
58
59
60 def myVariance():
61     if len(items) > 1: # ต้องมีข้อมูลอย่างน้อย 2 ค่า
62         total_Var = statistics.variance(items)
63         Var_label.config(text=f'Variance = {total_Var}')
64     else:
65         Var_label.config(text='Variance = Not enough data') # แสดงข้อความแจ้งเตือน ว่าข้อมูลยังไม่พอ
66
67
68 def myStdev():
69     if len(items) > 1: # ต้องมีข้อมูลอย่างน้อย 2 ค่า
70         total_Stdev = statistics.stdev(items)
71         Stdev_label.config(text=f'Standard Deviation = {total_Stdev}')
72     else:
73         Stdev_label.config(text='Standard Deviation = Not enough data') # แสดงข้อความแจ้งเตือน ว่าข้อมูลยังไม่พอ
74
75
76 # สร้างหน้าต่างหลักสำหรับทำงาน (Create the main window)
77 win = Tk()
78 win.title('การหาค่าทางสถิติ')
79 win.geometry('1024x720')
80
81 # สร้างข้อความเพื่อแสดงที่ต้องการ (Create the message for information)
82 myText1 = Label(win, text='กรุณาป้อนข้อมูลเป็นตัวเลข : ', font='AngsanaNew 16')
83 myText1.place(x=20, y=20)
84
85 # สร้างกล่องป้อนข้อมูล (Create an entry widget for input)
86 myBox1 = Entry(win)
87 myBox1.place(x=270, y=30)
88
89 # สร้างปุ่มสำหรับป้อนข้อมูล (Create a button to add data)
90 myButton1 = Button(win, text='ป้อนข้อมูล', command=add_data)
91 myButton1.place(x=20, y=60)
92
```

```
OOP-605.py OOP-607.py X
OOP-607.py > ...
88 myButton1 = Button(win, text='ป้อนข้อมูล', command=add_data)
89 myButton1.place(x=20, y=60)
90
91 myButton2 = Button(win, text='หาผลรวม "กดตรงนี้"', command=Find_sum, font = 'Thaisaraban 8')
92 myButton2.place(x=20, y=90)
93
94 myButton3 = Button(win, text='หาค่าเฉลี่ย "กดตรงนี้"', command=Find_Avg, font = 'Thaisaraban 8')
95 myButton3.place(x=20, y=120)
96
97 myButton4 = Button(win, text='หาค่ามากที่สุด "กดตรงนี้"', command=Find_Max, font = 'Thaisaraban 8')
98 myButton4.place(x=20, y=150)
99
100 myButton5 = Button(win, text='หาค่าน้อยสุด "กดตรงนี้"', command=Find_Min, font = 'Thaisaraban 8')
101 myButton5.place(x=20, y=180)
102
103 myButton6 = Button(win, text='หาค่าแปรปรวน "กดตรงนี้"', command=Find_Var, font = 'Thaisaraban 8')
104 myButton6.place(x=20, y=210)
105
106 myButton7 = Button(win, text='หาค่าเบี่ยงเบน "กดตรงนี้"', command=Find_Stdev, font = 'Thaisaraban 8')
107 myButton7.place(x=20, y=240)
108
109
110
111 # สร้างกล่องแสดงข้อมูลของ list (Create a listbox to display the data)
112 displayBox = Listbox(win)
113 displayBox.place(x=20, y=300)
114
115 # สร้างข้อความสำหรับแสดงผลรวม (Create a label to display summation)
116 sum_label = Label(win, text='Summation = 0.00', font = 'AngsanaNew 8')
117 sum_label.place(x=150, y=90)
118 Avg_label = Label(win, text='Average = 0.00', font = 'AngsanaNew 8')
119 Avg_label.place(x=150, y=120)
120 Max_label = Label(win, text='Maximum = 0.00', font = 'AngsanaNew 8')
121 Max_label.place(x=150, y=150)
122 Min_label = Label(win, text='Minimum = 0.00', font = 'AngsanaNew 8')
123 Min_label.place(x=150, y=180)
124 Var_label = Label(win, text='Variance = 0.00', font = 'AngsanaNew 8')
125 Var_label.place(x=150, y=210)
126 Stdev_label = Label(win, text='Standard Deviation = 0.00', font = 'AngsanaNew 8')
127 Stdev_label.place(x=150, y=240)
128
129 # สร้างค่าแปร list เพื่อเก็บค่าข้อมูล (Initialize an empty list to store data)
130 items = []
131
132 # เริ่มการทำงาน (Run the application)
133 win.mainloop()
```

Result:

การทำงานเกี่ยวกับสถิติ

กรุณาป้อนข้อมูลเป็นตัวเลข :

Summation = 0.00

Average = 0.00

Maximum = 0.00

Minimum = 0.00

Variance = 0.00

Standard Deviation = 0.00

การทำงานเกี่ยวกับสถิติ

กรุณาป้อนข้อมูลเป็นตัวเลข :

Summation = 73041.0

Average = 0.00

Maximum = 0.00

Minimum = 0.00

Variance = 0.00

Standard Deviation = 0.00

4500
5000
6000
8500
9000
6500
10000
7000
8000
8541

การทำงานเกี่ยวกับสถิติ

กรุณาป้อนข้อมูลเป็นตัวเลข :

Summation = 73041.0

Average = 7304.1

Maximum = 0.00

Minimum = 0.00

Variance = 0.00

Standard Deviation = 0.00

4500
5000
6000
8500
9000
6500
10000
7000
8000
8541

การทำงานเกี่ยวกับสถิติ

กรุณาป้อนข้อมูลเป็นตัวเลข :

Summation = 73041.0

Average = 7304.1

Maximum = 10000.0

Minimum = 0.00

Variance = 0.00

Standard Deviation = 0.00

4500
5000
6000
8500
9000
6500
10000
7000
8000
8541

การทำงานเกี่ยวกับสถิติ

กรุณาป้อนข้อมูลเป็นตัวเลข :

Summation = 73041.0

Average = 7304.1

Maximum = 10000.0

Minimum = 4500.0

Variance = 0.00

Standard Deviation = 0.00

4500
5000
6000
8500
9000
6500
10000
7000
8000
8541

การทำงานเกี่ยวกับสถิติ

กรุณาป้อนข้อมูลเป็นตัวเลข :

Summation = 73041.0

Average = 7304.1

Maximum = 10000.0

Minimum = 4500.0

Variance = 3244434.766666666

Standard Deviation = 0.00

4500
5000
6000
8500
9000
6500
10000
7000
8000
8541

การทำงานเกี่ยวกับสถิติ

กรุณาป้อนข้อมูลเป็นตัวเลข :

Summation = 73041.0

Average = 7304.1

Maximum = 10000.0

Minimum = 4500.0

Variance = 3244434.766666666

Standard Deviation = 1801.2314583824775

4500
5000
6000
8500
9000
6500
10000
7000
8000
8541