

Computers & Programming

Chapter 12: Function



Computer Engineering, Kasetsart University Kamphaeng Sean Campus

Contents



Function Overview

Using User-defined Function

Variable Lifetime

Function Overview

- ◆ ฟังก์ชัน คือ ส่วนย่อยของโปรแกรมที่เขียนขึ้นมาใช้งานอย่างใดอย่างหนึ่ง โดยการอ้างถึงด้วยชื่อที่กำหนดขึ้น เช่น
 - `sqrt()` -- ใช้คำนวณค่ารากที่สองของจำนวนใดๆ
 - `sin()` -- ใช้คำนวณค่า `sin` ของจำนวนใดๆ
- ◆ ฟังก์ชันสามารถถูกเรียกใช้จากส่วนหลักของโปรแกรมหรือ ภายในฟังก์ชันอื่นก็ได้
 - ฟังก์ชันอื่นจะทำงานได้ก็ต่อเมื่อถูกเรียกใช้เท่านั้น

3

2 Types of Function

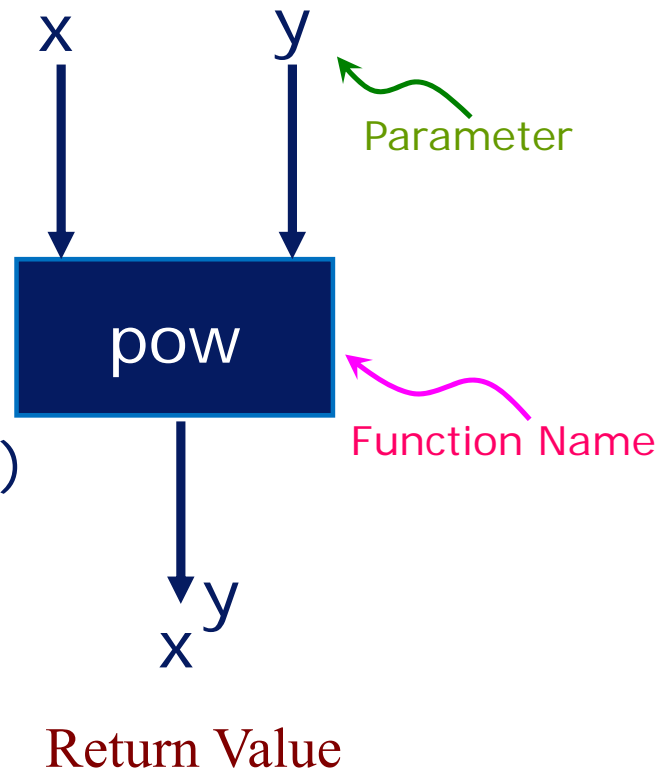
1. ฟังก์ชันมาตรฐาน (Pre-defined function in Python)
 - `math.pow(x,y)`
 - `math.sqrt(25)`
 - `len()`
2. ฟังก์ชันที่ผู้ใช้กำหนดเอง (User-defined function)
 - `sayhi(5)`
 - `add(x,y)`

4

Function Component

- ◆ องค์ประกอบของฟังก์ชัน
 - ชื่อฟังก์ชัน
 - พารามิเตอร์ (parameters)
 - การส่งค่ากลับ (return values)

- ◆ ตัวอย่าง
 - `Z=math.pow(2, 3)`



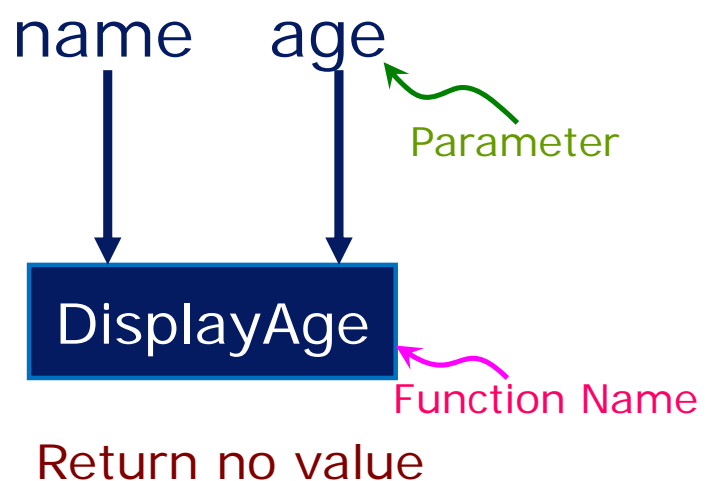
5

Function Component

- ◆ องค์ประกอบของฟังก์ชัน
 - ชื่อฟังก์ชัน
 - พารามิเตอร์ (parameters)
 - การส่งค่ากลับ (return values)

- ◆ ตัวอย่าง

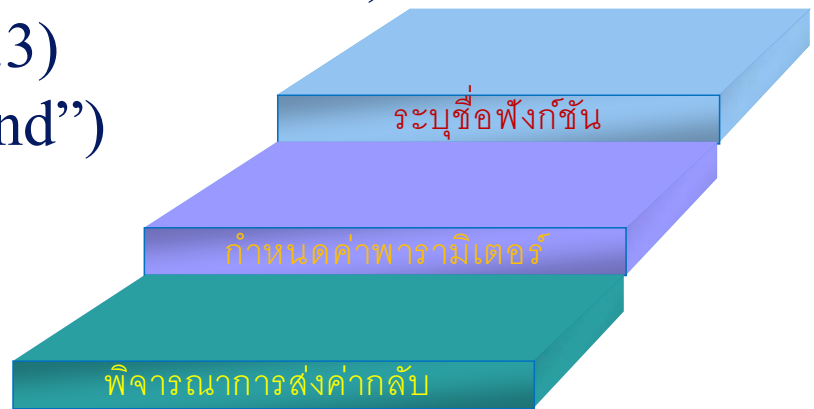
- `DisplayAge("Harry", 17)`



6

Using Pre-defined function

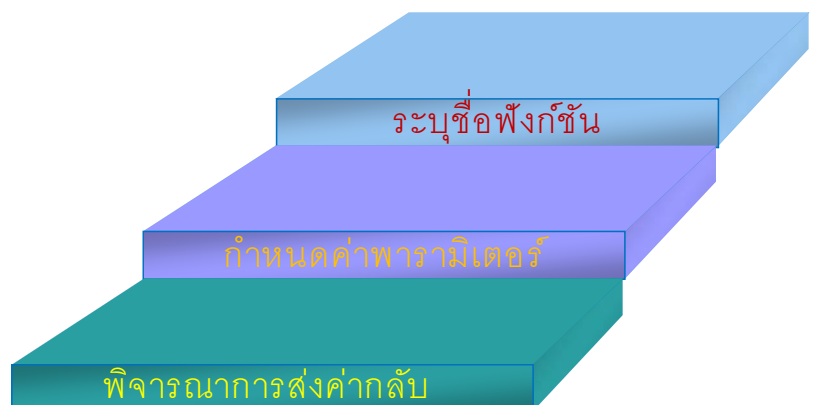
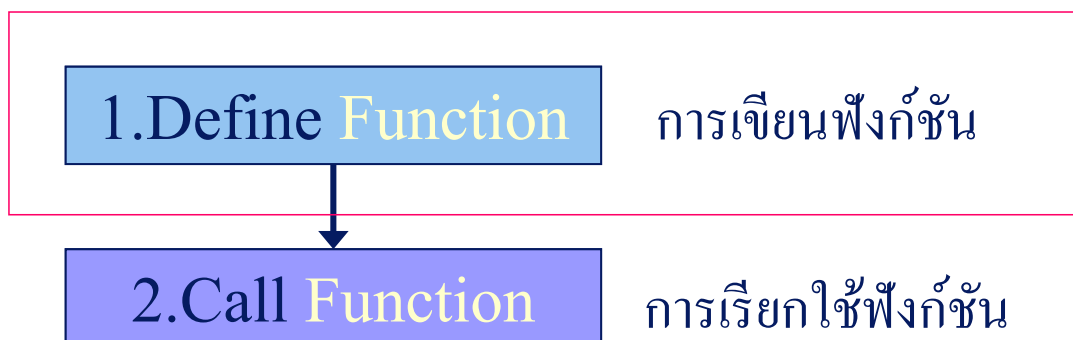
- `y = math.sqrt(x)`
- `y = math.sqrt(b * b - 4 * a * c)`
- `z = math.ceil(3.3)`
- `w = str.find ("and")`



การเรียกใช้ฟังก์ชัน (Call Function)

7

Using User-defined Function



8

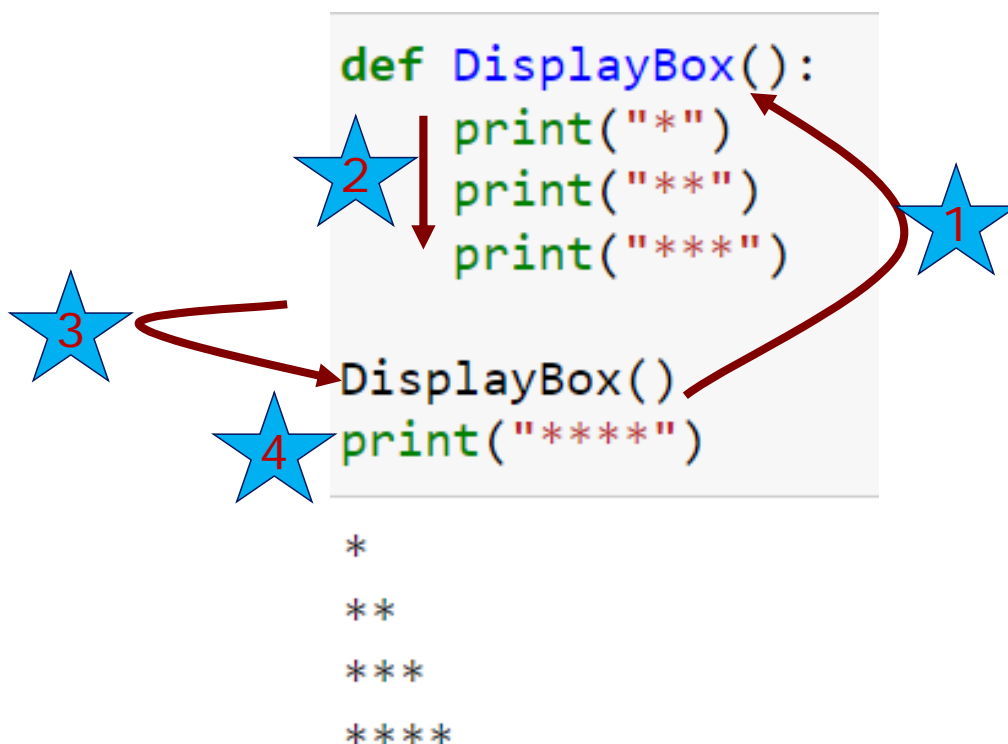
Using User-defined Function

```
def function-name(<parameter list>):  
    const/variable declaration  
    statements  
    return
```

- ชื่อฟังก์ชัน (function_name): ตั้งชื่อตามกฎหมายการตั้งชื่อเหมือนตัวแปร
- พารามิเตอร์ (จะมีหรือไม่มีก็ได้) : เขียนในรูปของการประกาศตัวแปรที่ละตัว คั่นด้วย , เช่น (a, b)
- การส่งค่ากลับ (return value)
 - ➡ สามารถส่งค่ากลับด้วยตัวแปรหรือค่าคงที่ได้ 0 , 1
 - ➡ ภายในฟังก์ชันไม่จำเป็นต้องส่งค่ากลับเสมอไป

9

Example I



Declaration Examples

```
def displaybox():
```

```
def calbox(x,y):
```

11

Function with No Return Value

◆ ฟังก์ชันที่ไม่มีการส่งค่ากลับ (No Return Value)

```
def function-name(<parameter list>):  
    const/variable declaration  
    statements
```

12

Example II : No Return Value without Parameter Function

◆ โปรแกรมร้องเพลงวันเกิดให้เอ้

```
print("Happy Birthday to you")
print("Happy Birthday to you")
print(" Happy Birthday , my dear Ae")
print("Happy Birthday to you")
```

13

Example II : No Return Value without Parameter Function

```
def HappyBirthAe():
    print("Happy Birthday to you")
    print("Happy Birthday to you")
    print("Happy Birthday, my dear Ae")
    print("Happy Birthday to you")
```

#ส่วนหลักของโปรแกรม
HappyBirthAe()

```
Happy Birthday to you
Happy Birthday to you
Happy Birthday, my dear Ae
Happy Birthday to you
```

14

Example II: No Return Value without Parameter Function

```
1 def HappyBirthAe():
2     print("Happy Birthday to you")
3     print("Happy Birthday to you")
4     print("Happy Birthday, my dear Ae")
5     print("Happy Birthday to you")
6
7 HappyBirthAe()
```

1. เริ่มจากบรรทัดที่ 1 การกำหนดนิยามฟังก์ชันชื่อ HappyBirthAe()
2. อ่านบรรทัดถัดไปเรื่อยๆ จนกว่าจะถึงบรรทัดที่ Block ของฟังก์ชันจะสิ้นสุด ในบรรทัดที่ 6
3. ฟังก์ชัน HappyBirthAe() ถูกนิยามไว้ทำนั้นยังไม่ถูกสั่งให้ทำงาน (เหมือนกับการประกาศตัวแปร)
4. เมื่อโปรแกรมรันมาถึงบรรทัดที่ 7 มีการเรียกใช้ฟังก์ชัน HappyBirthAe()
5. โปรแกรมจะทำการทบทวนว่า ฟังก์ชัน HappyBirthAe() ได้ถูกนิยามไว้หรือไม่ ถ้าเกิดมีการนิยามไว้ก่อนหน้านี้ให้รันคำสั่งที่อยู่ในฟังก์ชัน นั่นคือคำสั่ง บรรทัดที่ 2-5
6. จบการทำงาน

Example II: No Return Value without Parameter Function

```
1 def HappyBirthAe():
2     print("Happy Birthday to you")
3     print("Happy Birthday to you")
4     print("Happy Birthday, my dear Ae")
5     print("Happy Birthday to you")
6
7 HappyBirthAe()
```

```
1 def HappyBirthAe():
2     print("Happy Birthday to you")
3     print("Happy Birthday to you")
4     print("Happy Birthday, my dear Ae")
5     print("Happy Birthday to you")
6
7 HappyBirthAe()
```

ย่อหน้าผิด ทำให้คำสั่งนี้ไม่อยู่ในขอบเขตของฟังก์ชัน

```
Happy Birthday to you
Happy Birthday to you
Happy Birthday to you
Happy Birthday, my dear Ae
```


Example II : No Return Value without Parameter Function

```
def HappyBirthAe():  
    print("Happy Birthday to you")  
    print("Happy Birthday to you")  
    print("Happy Birthday, my dear Ae")  
    print("Happy Birthday to you")
```

#ส่วนหลักของโปรแกรม

HappyBirthAe()

ถ้าอยากร้องเพลง 2 รอบ ?

ถ้าอยากร้องเพลง 4 รอบ ?

17

Example II : No Return Value without Parameter Function

```
def HappyBirthAe():  
    print("Happy Birthday to you")  
    print("Happy Birthday to you")  
    print("Happy Birthday, my dear Ae")  
    print("Happy Birthday to you")
```

#ส่วนหลักของโปรแกรม

ถ้าอยากร้องเพลง 10 รอบ ?

18

Example II : No Return Value without Parameter Function

ถ้าอยากลองเพลงให้อั้มบ้าง

```
def HappyBirthAe():  
    print("Happy Birthday to you")  
    print("Happy Birthday to you")  
    print("Happy Birthday, my dear Ae")  
    print("Happy Birthday to you")
```

```
def HappyBirthAum():  
    print("Happy Birthday to you")  
    print("Happy Birthday to you")  
    print("Happy Birthday, my dear Aum")  
    print("Happy Birthday to you")
```

#ส่วนหลักของโปรแกรม

```
HappyBirthAe()  
HappyBirthAum()
```

Happy Birthday to you
Happy Birthday to you
Happy Birthday, my dear Ae
Happy Birthday to you

Happy Birthday to you
Happy Birthday to you
Happy Birthday, my dear Aum
Happy Birthday to you

Example II : No Return Value without Parameter Function

```
def HappyBirthAe():  
    print("Happy Birthday to you")  
    print("Happy Birthday to you")  
    print("Happy Birthday, my dear Ae")  
    print("Happy Birthday to you")
```

```
def HappyBirthAum():  
    print("Happy Birthday to you")  
    print("Happy Birthday to you")  
    print("Happy Birthday, my dear Aum")  
    print("Happy Birthday to you")
```

```
def HappyBirthBoth():  
    HappyBirthAe()  
    HappyBirthAum()
```

#ส่วนหลักของโปรแกรม

```
HappyBirthBoth()
```

Example III : No Return Value with Parameter Function

◆ ถ้าเราอยากร้องเพลงวันเกิดอีกหลายๆ คน ?

```
def HappyBirthday(person):  
    print("Happy Birthday to you")  
    print("Happy Birthday to you")  
    print("Happy Birthday,my dear", person)  
    print("Happy Birthday to you")
```

#ส่วนหลักของโปรแกรม

```
HappyBirthday("Ae")
```

```
HappyBirthday("Aum")
```

```
Happy Birthday to you  
Happy Birthday to you  
Happy Birthday, my dear Ae  
Happy Birthday to you
```

```
Happy Birthday to you  
Happy Birthday to you  
Happy Birthday, my dear Aum  
Happy Birthday to you
```

21

Example III : No Return Value with Parameter Function

◆ ถ้าเราอยากรับชื่อคนอื่นๆ

```
def HappyBirthday(person):  
    print("Happy Birthday to you")  
    print("Happy Birthday to you")  
    print("Happy Birthday,my dear ",person)  
    print("Happy Birthday to you")
```

#ส่วนหลักของโปรแกรม

```
name = input("Enter Name : ")
```

```
HappyBirthday(name)
```

Enter name: Harry

```
Happy Birthday to you  
Happy Birthday to you  
Happy Birthday, my dear Harry  
Happy Birthday to you
```

22

Quick Check I

◆ เติมส่วนของฟังก์ชันให้ร้องเพลงในโอกาสอื่นๆ ?

```
def HappyOccasion(person, ):
    print("Happy ", , "to you")
    print("Happy ", , "to you")
    print("Happy ", , "my dear ", person)
    print("Happy ", , "to you")
```

ส่วนหลักของโปรแกรม

```
HappyOccasion("Harry", "Monday")
```

Happy Monday to you

Happy Monday to you

Happy Monday, my dear Harry

Happy Monday to you

23

Quick Check II

◆ ให้นิสิตเขียนโปรแกรมเพื่อคำนวณหาพื้นที่วงกลมโดยการสร้างฟังก์ชันแบบไม่ส่งค่ากลับ โดยการป้อนค่าพารามิเตอร์คือ รัศมีของวงกลม ไปให้ฟังก์ชันชื่อ circle_area()

◆ ตัวอย่างการรันของโปรแกรม

Radius of circle: 3

The circle area is 28.2744

24

Quick Check II

```
pi = 3.1416

radius = input("Radius of circle : ")
radius = float(radius)


area = pi*radius*radius

print("The circle area is ",area)
```

25

Quick Check II

```
def circle_area( ):
    

    radius = input("Please Enter Radius = ")
    radius = float(radius)
    
```

26

Function with return value

- ◆ ฟังก์ชันที่มีการส่งค่ากลับ เป็นฟังก์ชันที่สามารถส่งค่ากลับได้ โดยใช้คำสั่ง `return` ตามด้วยตัวแปรหรือค่าคงที่

```
def function-name(<parameter list>):  
    const/variable declaration  
    statements  
    return value
```

ค่าตัวแปรหรือค่าคงที่
ที่ส่งค่ากลับ

27

Example IV: Return Value with Parameter Function

```
def HappyBirthday(person):  
    if(person == 'Ae'):  
        return "Diamond"  
    elif(person == 'Aum'):  
        return "Macbook"  
    else:  
        return "Pocky"  
  
name = input("Enter Name : ")  
gift = HappyBirthday(name)  
print(name, "get ",gift,"as a gift")
```

“Diamond”

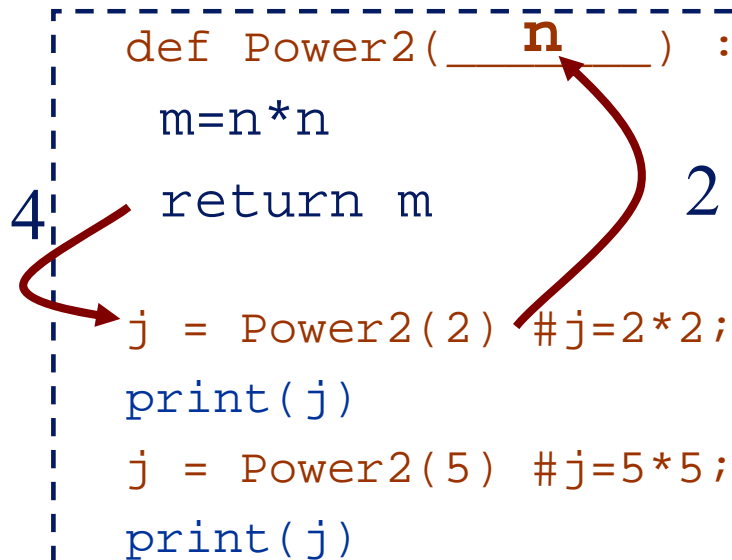
“Ae”

Enter name: Ae
Ae get Diamond as a gift

28

Example V: Return Value with Parameter Function

```
def Power2(n) :  
    m=n*n  
    return m  
4  
j = Power2(2) #j=2*2;  
print(j)  
j = Power2(5) #j=5*5;  
print(j)
```



4
25

29

Quick Check III

- ◆ ให้นิสิตเขียนโปรแกรมเพื่อคำนวณหาพื้นที่ผิวข้างของทรงกระบอกโดยการสร้างฟังก์ชันแบบส่งค่ากลับ โดยการป้อนค่าพารามิเตอร์ คือ ความสูงและรัศมีของทรงกระบอกไปให้ฟังก์ชันชื่อ `cylinder_surface`
- ◆ สูตรคือ $2\pi r(r + h)$
- ◆ ตัวอย่างการรันของโปรแกรม

Radius of cylinder: 3

Height of cylinder: 5

The cylinder surface area is 150.7968

Quick Check III

```
def cylinder_surface ( ) :  
      
  
r = input("Radius of cylinder: ")  
r = float(r)  
h = input("Height of cylinder:")  
h = float(h)  
      
print("The cylinder surface area is ",area)
```

31

Variable Lifetime

- ◆ ตัวแปร มีขอบเขตการใช้งานของมัน
- ◆ วงจรชีวิตของตัวแปร
 - กำหนดค่าตัวแปร → เริ่มต้นชีวิต
 - หากตัวแปรยังไม่ถูกกำหนดค่า = ยังไม่มีชีวิต → ใช้งานไม่ได้
 - ใช้งาน
 - ตัวแปรใช้งานได้ภายในฟังก์ชันที่ตัวแปรถูกสร้างไว้เท่านั้น
 - ทำลายอัตโนมัติเมื่อจบฟังก์ชัน

32

Example VI: Variable Lifetime

```
def cylinder_surface (r, h) :  
    area = 2*pi*r*(r+h)  
    return area  
  
r = input("Radius of cylinder: ")  
r = float(r)  
h = input("Height of cylinder:")  
h = float(h)  
area= cylinder_surface(r,h)  
  
pi= 3.414  
print("The cylinder surface area is ",area)
```

pi ไม่อยู่ในขอบเขตของ block นี้

33

Example VII: Variable Lifetime

```
1. def lifetime():  
2.     for i in range(5):  
3.         print(i)  
4.  
5. lifetime()  
6. print(i)
```

0
1
2
3
4

i ไม่อยู่ในขอบเขตของ block นี้

```
-----  
NameError                                Traceback (most recent call last)  
Input In [2], in <cell line: 6>()  
      3         print(i)  
      5 lifetime()  
----> 6 print(i)  
  
NameError: name 'i' is not defined
```

34

Quick check IV(1)

- ◆ จงเขียนฟังก์ชัน Power4() ที่มี n เป็นพารามิเตอร์ และส่งค่า n^4 กลับเป็นผลลัพธ์ // Power4(2) → 16 โดยไม่ใช้ math.pow

```
def Power4(_____):
```

- ◆ เขียนคำสั่งเรียกใช้ฟังก์ชันที่สร้างขึ้นในการหาค่า $a * b^4$

35

Quick check IV(2)

- ◆ จงเขียนฟังก์ชัน Findmin(x,y) ที่ส่งค่าที่น้อยกว่าระหว่าง x กับ y ออกจากฟังก์ชัน

```
def Findmin(_____):
```

- ◆ เขียนคำสั่งเรียกใช้ฟังก์ชันที่สร้างขึ้นในการหาค่าน้อยที่สุด ระหว่าง 101, 37

Quick check V

จงแสดงผลการทำงานของโปรแกรมต่อไปนี้

```
def myfunc1(d, e):  
    print("myfunc1: d = ",d )  
    print("myfunc1: e = ",e )
```

```
def myfunc2(d, e):  
    d=30  
    print("myfunc2: d = ",d )  
    print("myfunc2: e = ",e )  
    return d+e
```

```
#main program  
a=10  
b=20  
c=0  
print("main: a = ",a )  
print("main: b = ",b )
```

```
myfunc1(a,b);  
print("main: c = ",c )  
c=myfunc2(a,b)  
print("main: c = ",c )
```

37

References

◆ Hands-on Python tutorial

anh.cs.luc.edu/python/handson/3.1/handsonHtml/functions.html

◆ Python ๑๐๑ หนังสือสอนเขียนโปรแกรมภาษา Python

<https://www.eng.chula.ac.th/th/20535>