

Description

การใช้งานฟังก์ชันของคุณที่ผ่านมานั้น เป็นลักษณะของการส่งข้อมูลเข้าฟังก์ชันเพียงอย่างเดียว ในโจทย์ข้อนี้ ฉันจะเสนอวิธีการส่งข้อมูลออกจากฟังก์ชัน

ก่อนอื่น ฉันขอยกตัวอย่างที่ง่ายที่สุดก่อน

```
"""Source code's docstring"""
def function():
    """Function's docstring"""
    varie_1 = input()
```

```
function()
```

...ทำไมดูง่าย

เนื้อหาใจความของคำสั่งนี้คือ

ฟังก์ชัน input จะทำงานให้โปรแกรมรอผู้ใช้งานป้อนข้อมูลเข้าโปรแกรม

และฟังก์ชันนั้นจะส่งค่าผลลัพธ์ออกมาจากฟังก์ชัน

ซึ่งฉันได้สร้างตัวแปร **varie_1** ขึ้นมาเพื่อเก็บผลลัพธ์ของฟังก์ชัน input ไว้

อธิบายละเอียดก็คือ

ฟังก์ชัน input มีการส่งค่าออก (**Return value**) เป็นข้อความที่ผู้ใช้ป้อนเข้าไป

ฟังก์ชันที่ **Return value** นั้น เรียกว่า **Fruitful function**

ที่ผ่านมานั้นคุณได้ใช้ built-in function input

ก็คือคุณได้ใช้ฟังก์ชันที่มีการส่งข้อมูลออกมาจากฟังก์ชันไปแล้ว โดยที่คุณอาจจะไม่รู้ตัว

แต่ฟังก์ชันที่คุณสร้างมาตลอดนั้น ไม่มีการส่งข้อมูลใดๆออกมา

ฟังก์ชันลักษณะดังกล่าวเรียกว่า **Void function**

******หากคำศัพท์ใดๆที่คุณไม่คุ้นเคย ฉันคาดหวังว่าคุณจะมี dictionary ติดตัวคุณอยู่ในขณะที่คุณเรียน, อ่าน

Think Python, หรือแม้แต่ทำโจทย์ของฉันอยู่

******ไม่ว่าจะเป็นในรูปแบบหนังสือ หรือเว็บไซต์ออนไลน์ก็ตาม

******ฉันแนะนำ dict.longdo.com

******และ add-on ของ Chrome ชื่อ **Google Dictionary**

******ฉันไม่ได้คำโฆษณา คุณก็รู้ว่าฉันเป็นโปรแกรม

จากนี้ ฉันจะนำเสนอการสร้าง **Fruitful function** ให้คุณ

```
"""Source code's docstring"""
def function():
    """Function's docstring"""
    varie_1 = input()
    varie_2 = int(varie_1)

    result_1 = function_1(varie_2)
    result_2 = function_2(varie_2)

    print(result_1, result_2)

def function_1(param):
    return param

def function_2(param):
    return param
```

```
return param+1
```

```
function()
```

function_1 ได้มีการใช้ Keyword return
เป็นคำสั่งให้ฟังก์ชันนั้นสิ้นสุดการทำงาน และส่งค่าออกมาจากฟังก์ชัน
ในที่นี้คือการส่งค่าของตัวแปร **param** ออกมา
ซึ่ง **param** ก็เป็น Parameter ของ function นี้
กล่าวได้ว่า ฟังก์ชันนี้ไม่ได้ทำงานอื่นเลย นอกจากรับค่า และส่งค่าเดิมออกมา

การใช้ return คือการสั่งให้ฟังก์ชันนั้นสิ้นสุดการทำงานโดยปริยาย
เพราะฉะนั้นคุณควรออกแบบวางแผนฟังก์ชันของคุณให้ทำงานได้ตรงความต้องการของคุณ
หากมีคำสั่ง return หลายตัว
ฟังก์ชันนั้นจะทำงาน return ตัวแรกที่สามารถทำงานได้

function_2 มีลักษณะเหมือนกับ function_1
แต่จะเพิ่มค่า +1 ให้กับ **param**
ค่าที่ส่งออกจาก function_2 นั้นจะมีค่าเพิ่มขึ้น 1
คุณควรทดลองด้วยตัวคุณเอง
และในเมื่อ **param** ใน function_2 นั้นได้มีการทำงานคณิตศาสตร์
argument ที่ส่งเข้า function_2 นั้นจะต้องคาดหวังว่าจะเป็นจำนวนจริง (Integer หรือ Floating point)

ลองดูตัวอย่างการใช้ **Fruitful function** แบบจริงจัง

```
def function1():
    varie_1 = input()
    varie_1 = int(varie_1)

    varie_2 = function2(varie_1)
    print(varie_2)

def function2(param):
    param2 = param
    param *= 64

    param += param2

    return param%256
```

```
function1()
```

คุณควรทดลองด้วยตัวคุณเอง
ต่อไปนี้เป็นงานของคุณ

งานของคุณคือให้สร้างฟังก์ชันตามที่ฉันกำหนดให้ และแสดงผลออกมา

ชื่อตัวแปรที่ฉันใช้ใน **Specification** เป็นการใช้เพื่อให้ดูสะอาดตา
การเขียนโปรแกรมของคุณจริงๆก็ควรจะเขียนให้ถูกต้องตามหลักการตั้งชื่อตัวแปร ฟังก์ชัน ฯลฯ

Specification

➡ Input Specification

3 บรรทัด; หมายความว่า มีการเรียกใช้ฟังก์ชัน input 3 ครั้ง; คุณจะกำหนดไว้ที่ไหนก็ได้ในโปรแกรมคุณ แต่ต้องใช้ให้ครบทั้งสามครั้ง

x

y

z

เป็นจำนวนเต็มทั้งหมด

➡ Output Specification

5 บรรทัด เป็นไปตามฟังก์ชันที่กำหนดให้ แสดงผลออกมาฟังก์ชันละ 1 บรรทัดตามลำดับ
 หากมีขั้นตอนใดที่ฉันไม่ได้อธิบายในโจทย์ข้อนี้หรือที่ผ่านๆมา คุณสามารถหาอ่านได้ใน Think Python หรือถามเจ้านายฉัน
 ฉันอาจจะเขียนวิธีการใดๆที่คุณอาจต้องการในข้อถัดๆไป หากข้อนี้คุณไม่สามารถทำได้ คุณสามารถเปิดโจทย์ข้ออื่นดูก่อนได้
 อาจจะได้ความคิดดีๆขึ้นมา

$$f(x) = x + 1 \quad (1)$$

$$f(y) = 7y^3 + 2y^2 - 31y + 1 \quad (2)$$

$$f(z) = -z \quad (3)$$

$$f(x, y) = (x + y)(x - y) \quad (4)$$

$$f(x, y, z) = \frac{y - \sqrt{y^2 - 4xz}}{2x} \quad (5)$$

≡ Sample Case

➡ Sample Input

1
3
2

32
444

➡ Sample Output

2
115
-2
-8
1.0

33
613079197

51

-51
-196112
0.11583185723609102

10
90
100

11
5116411
-100
-8000
1.2984378812835757

⌚ Time Remaining

0

Day

1

Hour

3

Minutes

10

Seconds

792

i Information

Time Limit

1 Second

Memory Limit

32 MB

Language

py

Deadline

21 September 2016, 21:00

Submission Limit

Unlimit

Restrict Word

No Restrict

Testcase

10 case

Rejudge Testcase

0 case

Full Score

100 Point

Bonus Score

100

Your Score

PPPPPPPPP (https://ejudge.it.kmitl.ac.th/problem/submission/310087)

📁 Submit File

Choose File No file chosen

Choose File No file chosen

Submit