

07 ฟังก์ชัน (Function)

Function

python มี Function 2 ประเภท คือ

- Library function (ไลบรารี ฟังก์ชัน)
- User defined function (ฟังก์ชันที่เขียนขึ้นเอง)

Libray function

เป็น function ที่ python จัดเตรียมไว้ให้ ผู้ใช้สามารถเรียกใช้งานได้เลย โดยการเรียกใช้งานจะมี 2 แบบ คือ

1. เรียกผ่านชื่อฟังก์ชันโดยตรง เรียกว่า Build in function
2. เรียกผ่านไฟล์ที่มีการจัดเก็บฟังก์ชันไว้ เรียกว่า Module function

1. Build in function

สามารถเรียกใช้งานผ่านชื่อฟังก์ชันตามด้วยข้อมูลที่ต้องการส่งให้ function ในวงเล็บ เมื่อ function ทำงานเสร็จแล้วก็จะคืนค่าผลลัพธ์กลับคืนมา

<https://docs.python.org/3/library/functions.html> (<https://docs.python.org/3/library/functions.html>)

In [4]:

```
# input = รับข้อมูลจากแป้นพิมพ์
data = input()
```

20

In [5]:

```
# print = แสดงผลข้อมูลทางจอภาพ
print(data)
```

20

In [6]:

```
#แปลงข้อมูล
float(data)

int(data)

str(data)
```

Out[6]:

'20'

In [7]:

```
#ตรวจสอบชนิดข้อมูล  
type(data)
```

Out[7]:

str

In [10]:

```
data1 = 2  
data2 = 3  
data3 = 4
```

In [11]:

```
#หาค่า data1 ยกกำลัง data2  
pow(data1,data2)
```

Out[11]:

8

In [19]:

```
#หาค่าผลรวม a  
a = (1, 2, 3, 4, 5)  
x = sum(a)  
print(x)
```

In [21]:

```
#หาค่ามากที่สุด  
max(data1,data2,data3)
```

Out[21]:

4

In [22]:

```
#หาค่าน้อยที่สุด  
min(data1,data2,data3)
```

Out[22]:

2

2. Module function

การเรียกใช้ function ประเภทนี้ต้อง import ชื่อ module ก่อน ซึ่งความหมายของ module ก็คือ ไฟล์ที่มีการรวบรวมฟังก์ชันต่างๆ เราสามารถเขียนฟังก์ชันและเก็บเป็น module ไว้ใช้งานเองได้ เรียกว่า user defined function

การ import สามารถทำได้ 3 รูปแบบคือ

1. import modulename

import ชื่อ module โดยตรง

2. from modulename import functionname1 [, functionname2,.....,functionname_n]

import ชื่อ function จากชื่อ module

3. from modulename import *

import ฟังก์ชันทั้งหมดที่มีอยู่ใน module

In [23]:

```
import math
import datetime

print(math.factorial(5))
print(datetime.date.today())
```

120
2020-08-31

In [24]:

```
from math import factorial
from datetime import date,datetime
```

In [25]:

```
from math import *
from datetime import *
```

User defined function (ฟังก์ชันที่เขียนขึ้นเอง)

เป็นกลุ่มคำสั่งที่ถูกกำหนดขึ้นมาเพื่อทำงานอย่างใดอย่างหนึ่ง โดยสามารถเขียนฟังก์ชันหลายๆฟังก์ชันรวมกันใน ไฟล์ หนึ่งๆ เรียกว่า module รูปแบบการเขียน คือ

```
def functionName([parameter]):  
  
    statements  
    .  
    .  
    [return varValue]
```

ในการเรียกใช้งานฟังก์ชันสามารถเรียกใช้ฟังก์ชันโดยมีรูปแบบดังนี้

```
[result =] functionName([argument])
```

ตัวแปร global

ตัวแปร global จะถูกกำหนดไว้ในส่วนของโปรแกรมหลัก สามารถเรียกใช้หรือเข้าถึงได้ทุกตำแหน่งโปรแกรม

In [26]:

```
def showdata1():  
    print('show data1', data)  
  
def showdata2():  
    print('show data2',data)  
  
data = 5  
showdata1()  
showdata2()
```

```
show data1 5  
show data2 5
```

In [27]:

```
def showdata1():  
    global g1  
    g1 = 20  
    print('show data1', data)  
    print('show global', g1)  
  
def showdata2():  
    print('show data2', data)  
    print('show global2', g1)  
  
data = 5  
showdata1()  
showdata2()
```

```
show data1 5  
show global 20  
show data2 5  
show global2 20
```

ตัวแปร local

ตัวแปร local จะถูกกำหนดไว้ภายในฟังก์ชันหนึ่ง ขอบเขตการเรียกใช้งานก็จะถูกจำกัดภายในฟังก์ชันนั้น

In [28]:

```
def showdata1():  
    data1 = 10  
    print('show data1', data1)  
  
def showdata2():  
    data2 = 20  
    print('show data2', data2)  
    showdata1()  
  
showdata1()  
showdata2()
```

```
show data1 10  
show data2 20  
show data1 10
```

อาร์กิวเมนต์ (Argument) และ พารามิเตอร์ (Parameter)

- อาร์กิวเมนต์ (Argument) คือ ตัวแปรที่ส่งไปให้ฟังก์ชันพร้อมกับการเรียกใช้ฟังก์ชันในกรณีที่มีจำนวนมากกว่าหนึ่งค่าให้คั่นด้วยเครื่องหมาย ","
- พารามิเตอร์ (Parameter) คือ ตัวแปรที่ทำหน้าที่รับค่าอาร์กิวเมนต์ที่ส่งมาให้งานในฟังก์ชัน ในกรณีที่มีจำนวนมากกว่าหนึ่งค่าให้คั่นด้วยเครื่องหมาย ";"

In [29]:

```
def my_function(fname, lname): #create function 2 input parameter
    fullname = fname + " " + lname
    print(fullname)
```

In [30]:

```
my_function("somchai", "rukthai") #call function with 2 argument

my_function("somchai rukthai")
```

somchai rukthai

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-30-a27cc1c16670> in <module>
      1 my_function("somchai", "rukthai") #call function with 2 argument
      2
----> 3 my_function("somchai rukthai")
```

TypeError: my_function() missing 1 required positional argument: 'lname'

Default Parameter Value

In [31]:

```
def my_function(country = "Thailand"):
    print("I am from " + country)

my_function("Sweden")
my_function("India")
my_function()
my_function("Brazil")
```

I am from Sweden
I am from India
I am from Thailand
I am from Brazil

Passing a List as an Argument

In [32]:

```
def my_function(food):
    for x in food:
        print(x)

fruits = ["apple", "banana", "cherry"]

my_function(fruits)
```

apple
banana
cherry

Return Values

In [33]:

```
def my_function(x):  
    return 5 * x  
  
print(my_function(3))  
print(my_function(5))  
print(my_function(9))
```

15
25
45

In []: