

Description

ความสามารถอีกอย่างที่คอมพิวเตอร์ทำได้คือการจดจำ
โจทย์ข้อนี้ฉันจะอธิบายการใช้งานหน่วยความจำในรูปแบบตัวแปร (**Variable**)

ที่จริงภาษาโปรแกรมในปัจจุบันก็ใช้ตัวแปรกันเป็นหลักจนแทบจะไม่ได้ใช้หน่วยความจำในรูปแบบอื่นเลย

ลองดูชุดคำสั่งนี้ก่อน

```
"""Source code's docstring"""
def functionName():
    """Function's docstring"""

    variable_01 = 4
    variable_02 = 3.14
    variable_03 = "Output"

    print(variable_01)
    print(variable_01 * variable_02)
    print(variable_01, variable_02, variable_03)
```

3 คำสั่งแรกคือลักษณะการสร้างและกำหนดค่าให้กับตัวแปร

ฉันใช้ชื่อตัวแปร **variable_01**, **variable_02** และ **variable_03** กำหนดค่าเป็น 4, 3.14, และ "Output" ตามลำดับ

ลักษณะการสร้างและกำหนดค่าตัวแปรจะประกอบด้วยสามส่วนเสมอคือ

1. ชื่อตัวแปร: การตั้งชื่อตัวแปรที่ดี คุณควรได้เรียนแล้ว
2. เครื่องหมายการกำหนดค่า = ซึ่งก็คือเครื่องหมายเท่ากับ เป็นสัญลักษณ์ที่ใช้ในภาษา Python
3. ค่าของตัวแปร หรือความหมายของตัวแปรนั้นๆ ซึ่งในภาษา Python นั้น เป็นไปได้หลายอย่างมาก หลายอย่างเสียจนคงอธิบายไม่หมดในข้อนี้

ขาดส่วนใดส่วนหนึ่ง อาจนำไปสู่การเขียนผิดไวยากรณ์ (Invalid syntax) หรือความหมายผิดพลาดได้ (Semantic Error)

การพิมพ์ชื่อตัวแปรใดๆ มีความหมายให้เรียกค่าของตัวแปรนั้นๆออกมาใช้งาน

ใน print ที่หนึ่ง ได้นำเสนอการเรียกค่าของตัวแปร **variable_01**

การพิมพ์ชื่อตัวแปรผิด หรือพิมพ์ตัวแปรที่ไม่เคยถูกสร้าง จะนำไปสู่ Runtime error

หลักการใช้ตัวแปรคุณควรเรียนในห้องเรียน

ใน print ที่สอง ได้มีการเรียกใช้ค่าของตัวแปร **variable_01** และ **variable_02**

และนำมาคูณกัน

ลักษณะการทำงาน ก็เหมือนกับที่คุณนั่งทำโจทย์คณิตศาสตร์แล้วแทนค่าตัวแปรนั้นแหละ

กำหนดให้ **x = 1**, **y = 10** บลาๆๆ และให้ **x + y** จะได้ **11**

อะไรประมาณนี้

และใน print ที่สาม นำเสนอการแสดงผลลัพธ์หลายค่า โดยที่ไม่เกี่ยวข้องกัน

ในฟังก์ชัน print สามารถใช้ , (comma) คั่นข้อมูล

ข้อมูลดังกล่าวจะถูกคั่นด้วยเว้นวรรค 1 ช่อง

ต่อไปเป็นการทำงานตัวแปรพื้นฐาน ในลักษณะการคำนวณต่อเนื่องแบบคณิตศาสตร์มัธยม


```

"""Source code's docstring"""
def functionName():
    """Function's docstring"""

    x = 10
    y = 16
    z = 1

    a1 = 5*x**2 + 5*y + z
    a2 = x / y * z
    a3 = 10+(y-(y**2 - 4*x*z)**0.5)/(2*x)

```

คำสั่งข้างต้นมีค่าเทียบเท่ากับ

$$x = 10$$

$$y = 16$$

$$z = 1$$

$$a1 = 5x^2 + 5y + z$$

$$a2 = x \div y \times z$$

$$a3 = 10 + \frac{y - \sqrt{y^2 - 4xz}}{2x}$$

ซึ่งมีผลเหมือนกับการดำเนินการทางคณิตศาสตร์
คุณสามารถลอง print(a1, a2, a3) เพื่อแสดงผลลัพธ์ได้

ก่อนถึงงานของคุณ มีกรณีหนึ่งที่ควรรู้

```

"""Source code's docstring"""
def functionName():
    """Function's docstring"""

    variable_01 = 10

    variable_01 = variable_01 + 5

    variable_01 += 37

```

ในคำสั่งที่สอง เป็นการเรียกค่าของตัวแปร **variable_01** ออกมา; นำไปบวก 5; แล้วจึงกลับมากำหนดค่าให้กับตัวแปร **variable_01**

ซึ่งก็คือตัวแปรตัวมันเอง

การกระทำในลักษณะนี้ เป็นเรื่องปกติของโปรแกรมคอมพิวเตอร์

เพราะการใช้เครื่องหมายเท่ากับ (=) ในภาษาคอมพิวเตอร์เช่น Python มีความหมายคือการกำหนดค่า ไม่ใช่การเท่ากันในลักษณะของสมการคณิตศาสตร์

ดังนั้นจึงไม่ใช่เรื่องผิดปกติแต่อย่างใด แกรมยังมีชื่อเรียกด้วย
และยังมีหลายชื่อด้วยเช่น Updating variable, Incremental, Decremental อื่นๆ แล้วแต่ความหมายที่คุณจะ
ใช้งานในตอนนั้น
แกรมยังมีรูปย่ออีก
ในคำสั่งที่สาม มีคุณลักษณะแบบเดียวกันกับ
`variable_01 = variable_01 + 37`

นอกจาก += แล้ว
ยังมี -= *= /= %= **= อีก
ล้วนเป็นรูปย่อทั้งสิ้น

มาถึงงานของคุณ

งานของคุณคือ ให้แสดงผลลัพธ์ของตัวแปร **days, hours, minutes, seconds, milliseconds** จากสมการ
ทั้งหมดที่ให้มา

$$start_here = 492137954293754252786$$
$$milliseconds = start_here$$
$$seconds = milliseconds // 1000$$
$$milliseconds = milliseconds \% 1000$$
$$minutes = seconds // 60$$
$$seconds = seconds \% 60$$
$$hours = minutes // 60$$
$$minutes = minutes \% 60$$
$$days = hours // 24$$
$$hours = hours \% 24$$

by Chotipat Pornavalai (<https://ejudge.it.kmitl.ac.th/account/6>)

8 August 2017, 13:45

Specification

➡ Input Specification

➡ Output Specification

➡ Input Specification

➡ Output Specification

ไม่มีค่าส่งเข้า

หนึ่งบรรทัด ประกอบไปด้วยตัวเลขจำนวนเต็ม 5 ตัว
 คั่นด้วยเว้นวรรคหนึ่งช่อง
days hours minutes seconds milliseconds

☰ Sample Case

➡ Sample Input

➡ Sample Output

🕒 Time Remaining

4

Days

3

Hours

35

Minutes

35

Seconds

440

i Information

Time Limit

1 Second

Memory Limit

32 MB

Language

py

Deadline

5 December 2017, 23:59

Submission Limit

Unlimit

Restrict Word

No Restrict

Required Word

No Restrict

Testcase

1 case

Rejudge Testcase

0 case

Full Score

100 Point

Bonus Score

100

Your Score

 (https://ejudge.it.kmitl.ac.th/problem/submission/468319)