



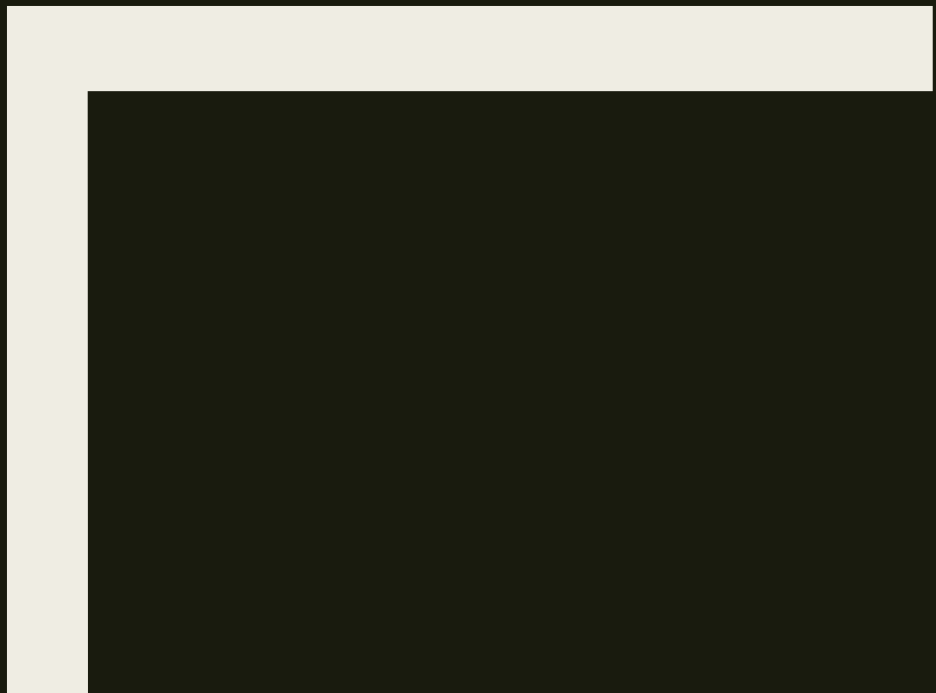
BASIC PYTHON



Presented by
Sittthipat Impoka
Team Sdi

Content

- Install Python & PyCharm
- Start Your First Program
- Comment
- Variable and Print
- Operation
- Function
- Standard Coding











INSTALL PYTHON & PYCHARM

Downloads python



Downloads python

Release version	Release date	Click for more	
Python 3.7.1	2018-10-20	 Download	Release Notes
Python 3.6.7	2018-10-20	 Download	Release Notes
Python 3.5.6	2018-08-02	 Download	Release Notes
Python 3.4.9	2018-08-02	 Download	Release Notes
Python 3.7.0	2018-06-27	 Download	Release Notes
Python 3.6.6	2018-06-27	 Download	Release Notes
Python 2.7.15	2018-05-01	 Download	Release Notes
Python 3.6.5	2018-03-28	 Download	Release Notes

Downloads python

Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		045fb3440219a1f6923fefdbabde63342	17496336	SIG
XZ compressed source tarball	Source release		a80ae3cc478460b922242f43a1b4094d	12642436	SIG
macOS 64-bit/32-bit installer	Mac OS X	for Mac OS X 10.6 and later	9ac8c85150147f679f213addd1e7d96e	25193631	SIG
macOS 64-bit installer	Mac OS X	for OS X 10.9 and later	223b71346316c3ec7a8dc8bff5476d84	23768240	SIG
Windows debug information files	Windows		4c61ef61d4c51d615cbe751480be01f8	25079974	SIG
Windows debug information files for 64-bit binaries	Windows		680bf74bad3700e6b756a84a56720949	25858214	SIG
Windows help file	Windows		297315472777f28368b052be734ba2ee	6252777	SIG
Windows x86-64 MSI installer	Windows	for AMD64/EM64T/x64	0ffa44a86522f9a37b916b361eebc552	20246528	SIG
Windows x86 MSI installer	Windows		023e49c9fba54914ebc05c4662a93ffe	19304448	SIG

Downloads Pycharm


- Pycharm คืออะไร



Pycharm คือ tools สำหรับการเขียน Python


Downloads Pycharm

JetBrains s.r.o. [CZ] | <https://www.jetbrains.com/pycharm/download/#section=windows>

 **PyCharm**

What's New Features Docs & Demos Buy [Download](#)

[Windows](#) [macOS](#) [Linux](#)



Professional

Full-featured IDE
for Python & Web
development

[Version: 2018.3](#)
[Build: 183.4284.139](#)
[Released: November 21, 2018](#)

[System requirements](#)
[Installation Instructions](#)
[Previous versions](#)

[DOWNLOAD](#)

Free trial

Community

Lightweight IDE
for Python & Scientific
development

[DOWNLOAD](#)

Free, open-source

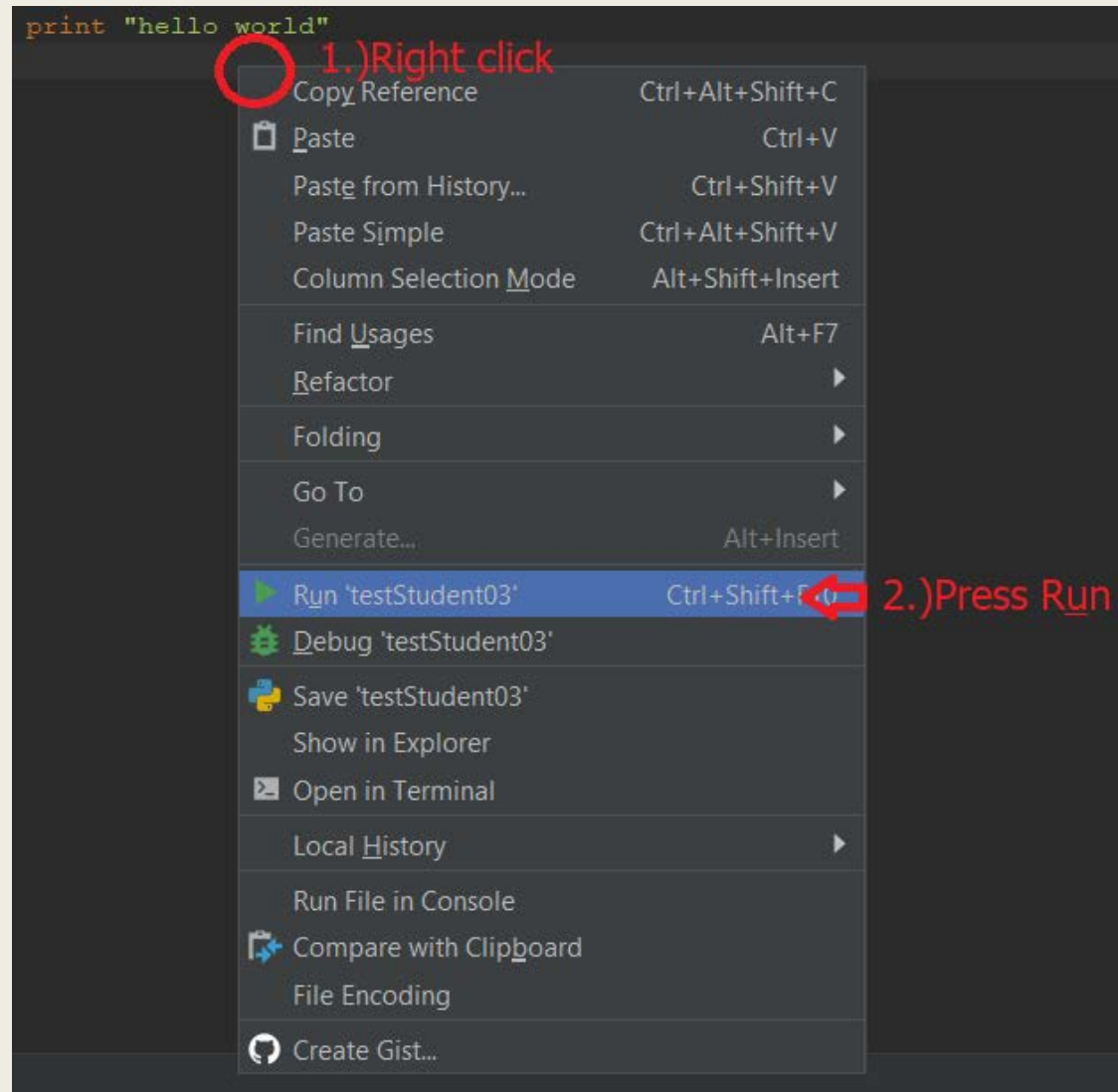


START YOUR
FIRST PROGRAM

Hello.py

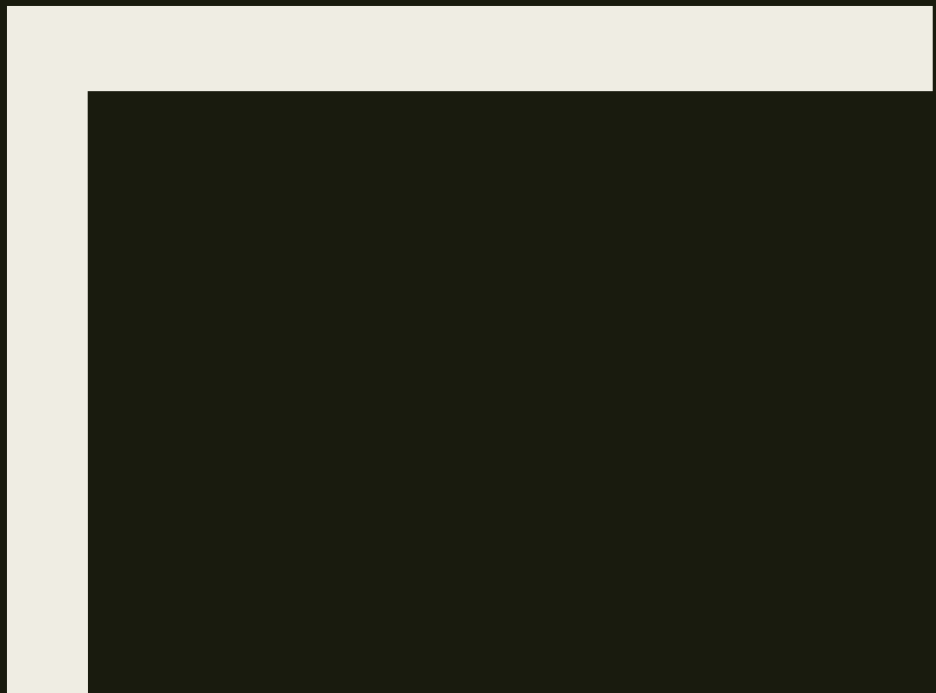
```
print "hello world"
```

How to run your first program



Output your first program

```
hello world
```



COMMENT

What is comment?

คือการใส่คำอธิบายลงในโปรแกรม โดยจะไม่ส่งผลใด ๆ กับโปรแกรม

Example

```
print "hello world" # this is comment
```

ตัวอักษรหลัง # ในบรรทัด
จะเป็น comment ทั้งหมด

Output

```
hello world
```

How to comment Thai message

Example

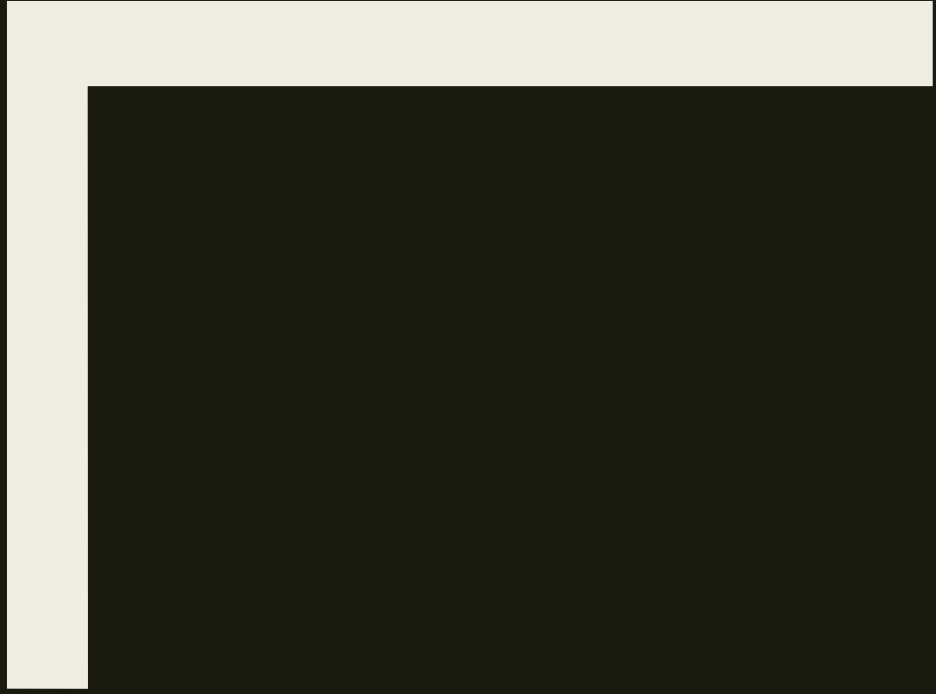
```
# -*- coding: utf-8 -*-
```

```
print "hello world" # วิธีคอมเมนต์ภาษาไทย
```

พิมพ์ code
-*- coding: utf-8 -*-
ที่บรรทัดบนสุด

Output

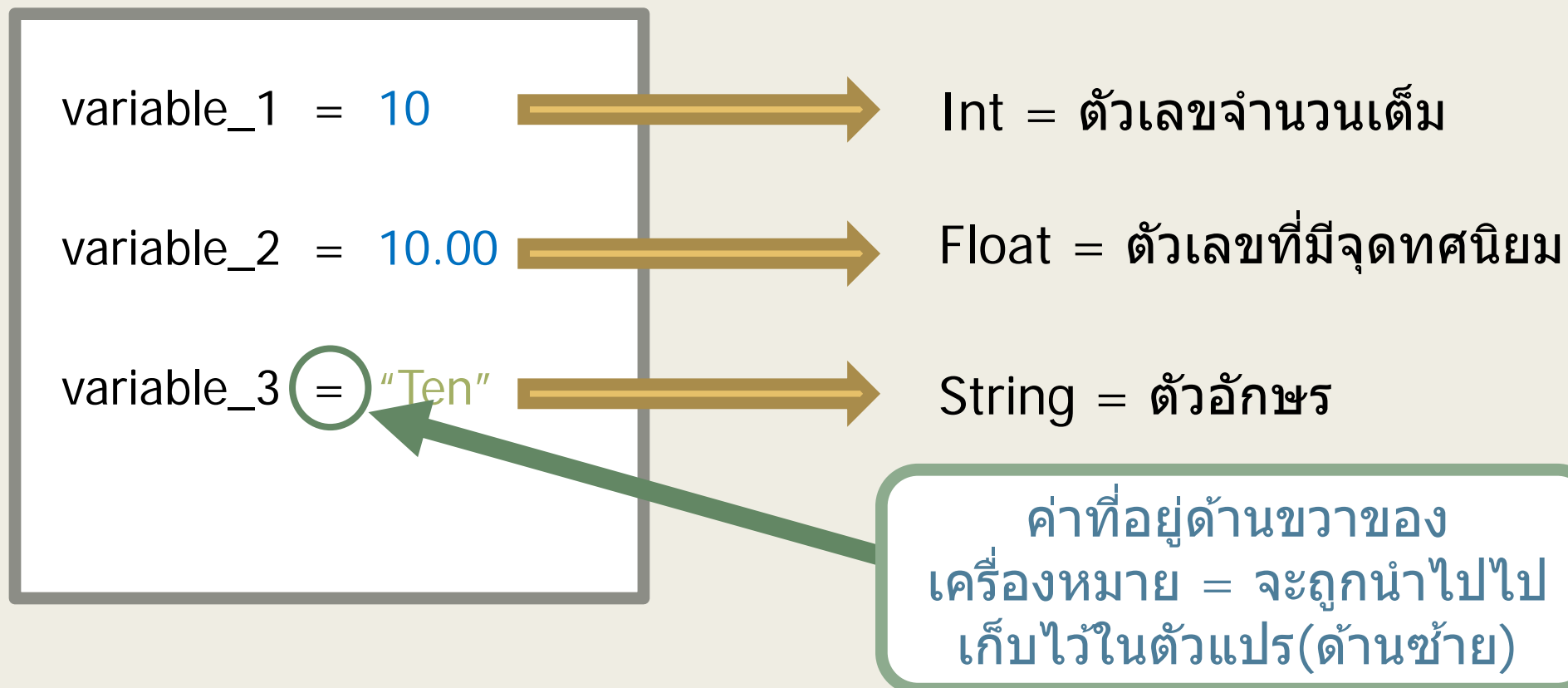
```
hello world
```

VARIABLES AND PRINT

What is variable?

- คือชื่อหรือเครื่องหมายที่กำหนดขึ้นสำหรับใช้เก็บค่าในหน่วยความจำ



What is null or none?

คือ ตัวแปรที่ไม่ได้มีค่าใด ๆ

Example

```
A = None
```

```
B = ""
```

ในกรณีของ B ค่าจะไม่ใช้
None แต่จะมีค่าเป็นช่องว่าง

Create multi values of variable

การสร้างตัวแปรหลายตัวพร้อม ๆ กัน

Example



```
Integer_1, Integer_2, Integer_3 = 10, 20, 30
```

The diagram illustrates the assignment of multiple values to multiple variables. Three colored arcs connect the variables to their respective values: a red arc from Integer_1 to 10, a green arc from Integer_2 to 20, and a yellow arc from Integer_3 to 30.

จากการประกาศตัวแปรข้างต้นตัวแปรจะมีค่าดังนี้

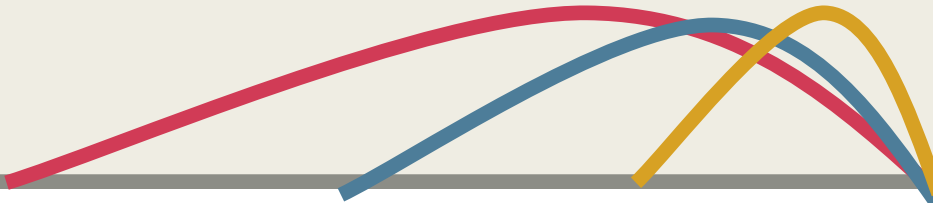
Integer_1 = 10

Integer_2 = 20

Integer_3 = 30

Create multi variable

Example



Integer_1 = Integer_2 = Integer_3 = 10

The diagram illustrates the assignment of the value 10 to three variables. Three colored arcs (red, blue, and yellow) originate from the number 10 and point to Integer_1, Integer_2, and Integer_3 respectively, showing that all three variables share the same memory address.

จากการประกาศตัวแปรข้างต้นตัวแปรจะมีค่าดังนี้

Integer_1 = 10

Integer_2 = 10

Integer_3 = 10

Create name of variable

การตั้งชื่อตัวแปรของ python ต้องเป็นไปตามกฎดังนี้

1. เริ่มต้นด้วยตัวอักษรตั้งแต่ 1 ตัวขึ้นไป
2. ตัวแปรห้ามมีช่องว่าง
3. ห้ามมีสัญลักษณ์พิเศษเช่น #, ?, \$, ...
4. ตัวแปรต้องไม่ซ้ำกับคำสงวน (ดูได้ในหน้าถัดไป)
5. ตัวแปรใช้ตัวพิมพ์เล็กพิมพ์ใหญ่ มีความหมายต่างกัน

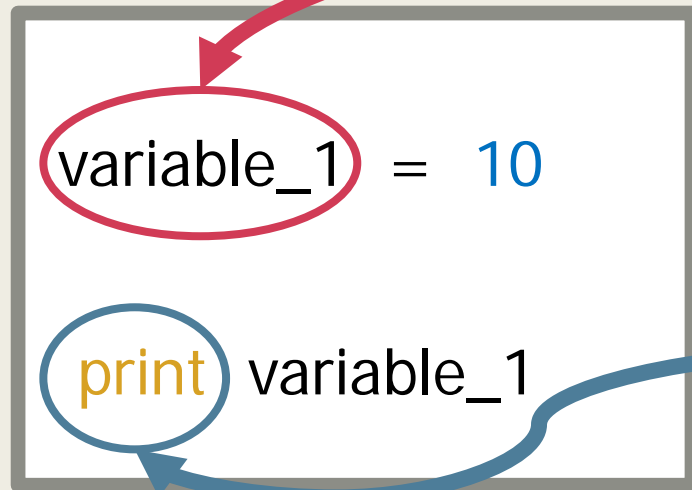
Reserved word

and, as, assert, break, class, continue, def, del, elif, else, except, exec, finally, for, from, global, if, import, in, is, lambda, not, or, pass, print, raise, return, try, while, with, yield

What is print?

- Print คือ การแสดงค่าออกทางหน้าจอ

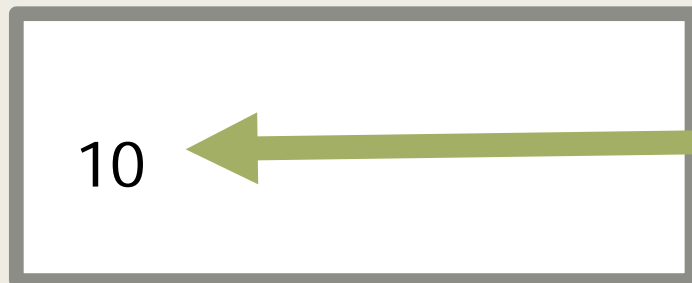
Example



ประกาศตัวแปรเพื่อมารับค่า

พิมพ์ `print` แล้วเว้นช่องว่าง 1 ช่อง
ตามด้วย ตัวแปร หรือ string

Output



ถ้าหาก `print` ตัวแปรจะได้ค่าที่
อยู่ในตัวแปร

What is raw_input

คือการรับค่าทาง keyboard(String) มาเก็บไว้ในตัวแปร

Example

```
your_answer = raw_input("How old are you : ");  
print "Now your age is : " , your_answer
```

Output

How old are you : |

Enter

How old are you : 20
Now your age is : 20

ประกาศ raw_input
เพื่อรับค่าจาก keyboard
("ใส่ค่าที่ต้องการแสดง")
เช่น คำถาม

Click 1 ครั้งพิมพ์
คำตอบเสร็จแล้ว
กดปุ่ม enter
สมมติว่าพิมพ์ 20

What is input

คือการรับค่าทาง keyboard (Int,Float) มาเก็บไว้ในตัวแปร

Example

```
your_answer = input("How old are you : ");  
print "Now your age is : " , your_answer
```

Output

How old are you : |

Enter

How old are you : 20.0
Now your age is : 20.0

ประกาศ input
เพื่อรับค่าจาก keyboard
("ใส่ค่าที่ต้องการแสดง")
เช่น คำถาม

Click 1 ครั้งพิมพ์
คำตอบเสร็จแล้ว
กดปุ่ม enter
สมมติว่าพิมพ์ 20.0

Print string directly

Example

```
variable_1 = "dog cat bird"  
print " you can print string directly"  
print variable_1
```

สามารถ **print** string
โดยตรงได้

Output

```
you can print string directly  
dog cat bird
```

Start new line string in output


Example

```
print " your string is too long? \n press  back slash n"
```

Output

```
your string is too long?  
press back slash n
```

กรณี String มีความยาว
สามารถพิมพ์ \n เพื่อเริ่ม
บรรทัดใหม่ใน output ได้



Start new line string in code

Example

```
print " your string is too long?" \  
      "press enter or use \"
```

กรณี String มีความยาว
สามารถพิมพ์ \ หรือ enter
เพื่อเริ่มบรรทัดใหม่ใน
code ได้


Output

```
your string is too long? press enter or use \
```

Double quote and single quote

Example

การเรียกใช้ หรือ ประกาศ string
จะใช้ single quote('...')
หรือ double quote("...") ก็ได้



```
print "welcome to work shop class"
```

```
print 'you should choose double quote or single quote'
```

Output

```
welcome to work shop class
```

```
you should choose double quote or single quote
```

Print string connect by +

Example

```
variable_1 = "you can connect string "  
variable_2 = "by use +"  
print variable_1 + variable_2
```

Output

```
you can connect string by use +
```

เวลา **print** สามารถ
ใช้เครื่องหมาย +
เพื่อต่อ string ได้
**แต่ไม่สามารถใช้
เครื่องหมาย + ต่อ
string กับ int หรือ
float

Print multi variable

Example

```
A = "ten ="
```

```
B = 10
```

```
print A,B
```

การ **print** ตัวแปรหลายๆตัว ทำได้โดย
การ ใส่เครื่องหมาย , คั่นระหว่างตัวแปร

Output

```
ten = 10
```


Print values of String by %s

Example

```
A = "fish"  
print "cat eat %s" %A
```

ประกาศ %s ใน string

ประกาศ
%(ตัวแปรที่ต้องการจะใส่)
ต่อท้าย string

Output

```
cat eat fish
```

ตัวแปรจะแสดงค่า ใน
ตำแหน่งที่ประกาศ %s ไว้

Print values of `Int` by `%d`

Example

```
A = 10
```

```
print "ten = %d" %A
```

ประกาศ `%d` ใน string

ประกาศ
`%`(ตัวแปรที่ต้องการจะใส่)
ต่อท้าย string

Output

```
ten = 10
```

ตัวแปรจะแสดงค่า ใน
ตำแหน่งที่ประกาศ `%s` ไว้

Print values of Float by %f

Example

```
A = 10
```

```
B = 10.00
```

```
print "ten = %f"
```

```
print "ten = %f"
```

```
%A
```

```
%B
```

ประกาศ %f ใน string

ประกาศ
%(ตัวแปรที่ต้องการจะใส่)
ต่อท้าย string

Output

```
ten = 10.000000
```

```
ten = 10.000000
```

สามารถใช้ %f print Int
ได้โดย

Print multi values by %

Example

```
A = "fish"  
B = "rat"  
print "cat eat %s and %s" %(A,B)
```

Output

```
cat eat fish and rat
```

กรณีหลายตัวแปรให้
ประกาศ % ตามด้วย() ซึ่ง
เราจะใส่ตัวแปรใน () โดย
ใช้เครื่องหมาย , คั่น
ระหว่างตัวแปร

Tip how to check type variable by **print type**

Example

```
B = "this is string"
```

```
B = 55
```

```
print B
```

```
print type(B)
```

ประกาศ
type(ตัวแปรที่ต้องดูประเภท)

Output

```
55
```

```
<type 'int'>
```

จะเห็นได้ว่า ตัวแปรจะรับเอา
ค่าล่าสุดมาเป็นค่าปัจจุบัน

Tip how to print specific character

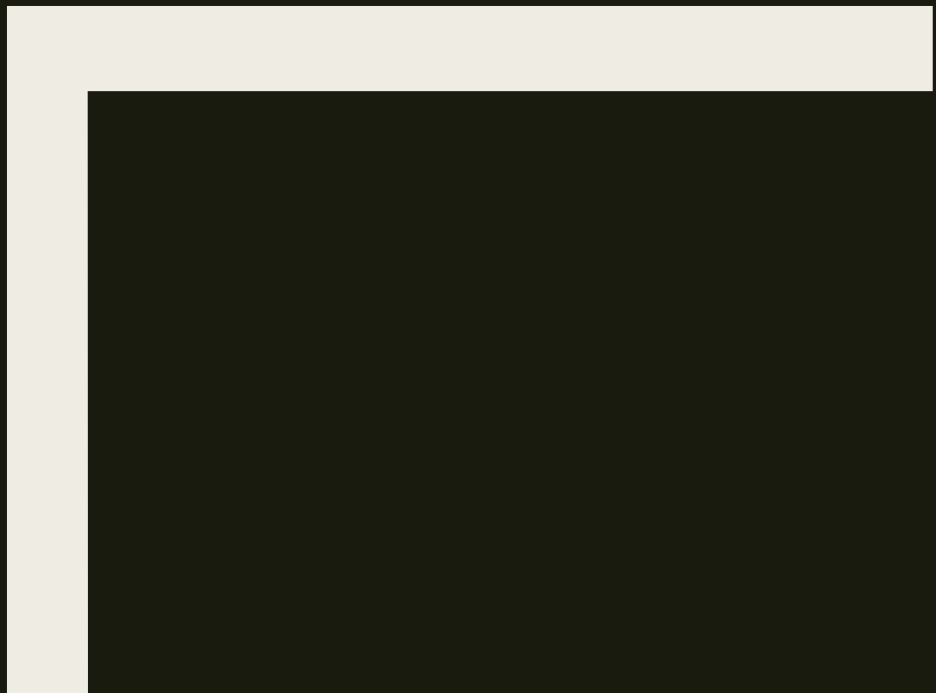
Example

```
print "\\"
```

ประกาศ \ แล้วตามด้วย
อักขระพิเศษ

Output

```
"
```



OPERATION

What is Arithmetic operators?

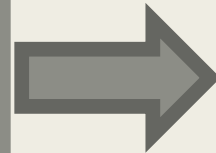
ตัวดำเนินการทางคณิตศาสตร์

Operator	Name	Example
+	Addition	$a + b$
-	Subtraction	$a - b$
*	Multiplication	$a * b$
/	Division	a / b
//	Division and floor	$a // b$
%	Modulo	$a \% b$
**	Power	$a ** b$

Example output from operator

Example

```
a = 5
b = 3
print "a + b = ",a+b
print "a - b = ",a-b
print "a * b = ",a*b
print "a / b = ",a/b
print "a // b = ",a//b
print "a % b = ",a%b
print "a ** b = ",a**b
```



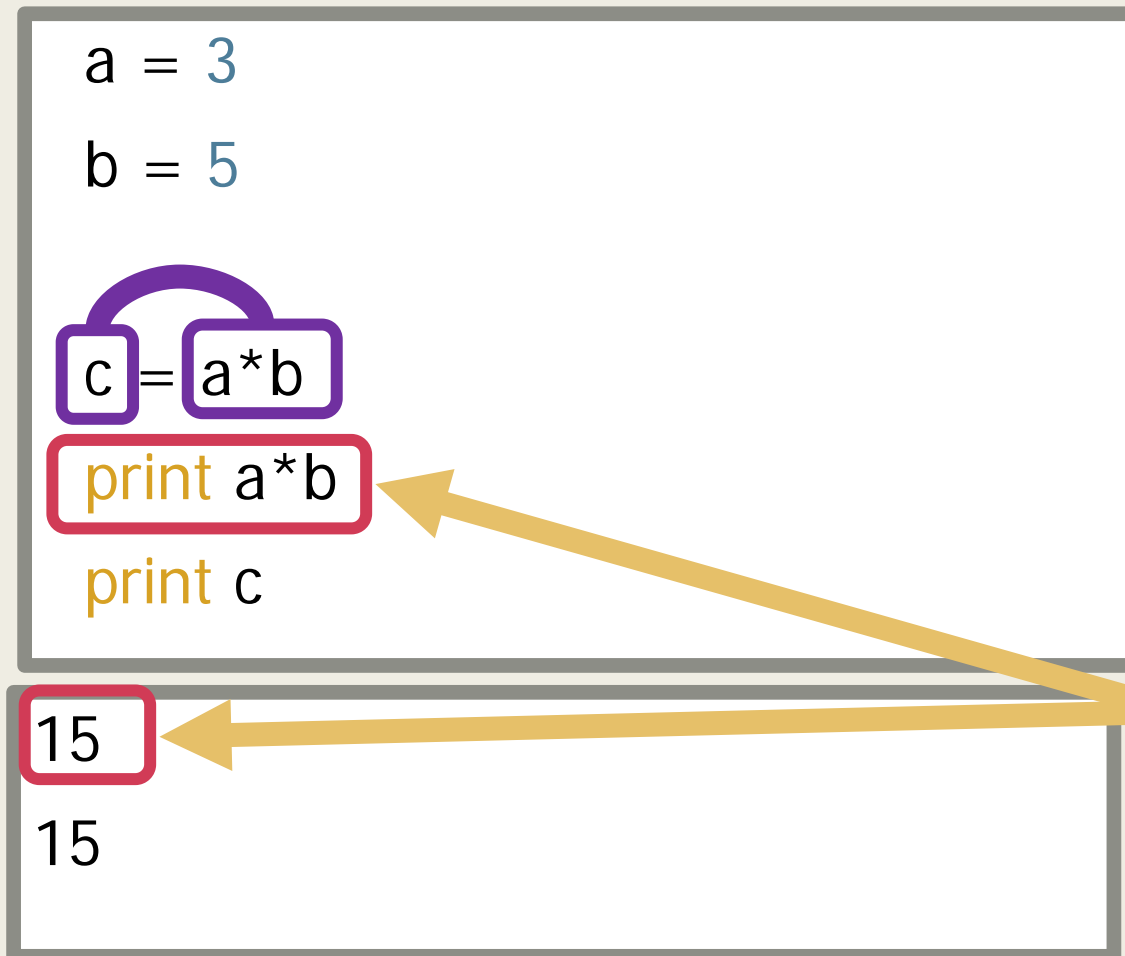
Output

```
a + b = 8
a - b = 2
a * b = 16
a / b = 1
a // b = 1
a % b = 2
a ** b = 125
```

Print output from operator directly

Example

Output

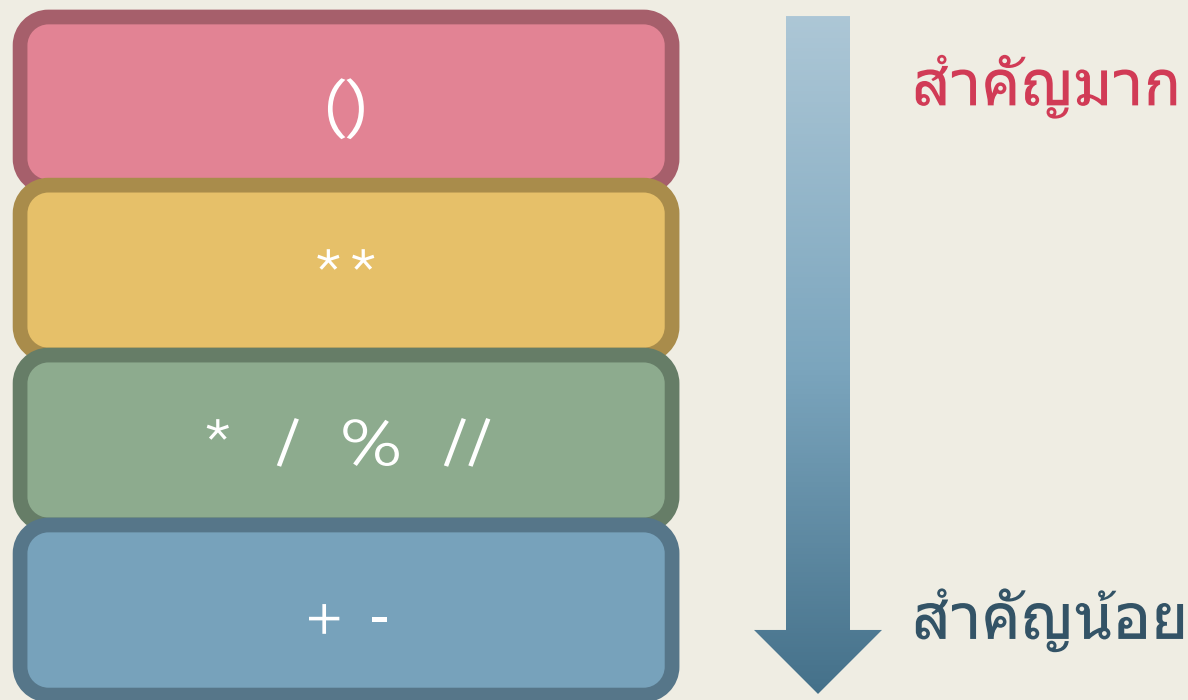


จะเห็นได้ว่า สามารถ ใช้ operator ตอน print ค่าได้ เลย(หรือจะประกาศตัวแปร รับค่าที่คำนวณจาก operator ก็ได้)

What is python operator Precedence

ลำดับความสำคัญของตัวดำเนินการใน Python Operators
(สำคัญมากทำก่อน สำคัญน้อยทำทีหลัง)

ตารางแสดงความสำคัญของตัวดำเนินการจาก มาก ไป น้อย



Example of operator Precedence

Example

```
a = 5
b = 3
c = 7
print "First line answer = " ,a*b-c
print "Second line answer = " ,a*(b-c)
```

Output

```
First line answer = 8
Second line answer = -20
```

จะเห็นได้ว่า การ
ประกาศใช้ operator
ในเครื่องหมาย ()
python จะคำนวณค่า
ใน () ก่อน

Exercise

$$18/6 * (4-1) = ?$$

Answer



or



Exercise

มีกล่องสี่เหลี่ยม 2 กล่อง จงหาว่า ปริมาตรของกล่องแรกเป็นกี่เท่าของกล่องที่ 2
โดยให้ทำการ print คำตอบด้วย

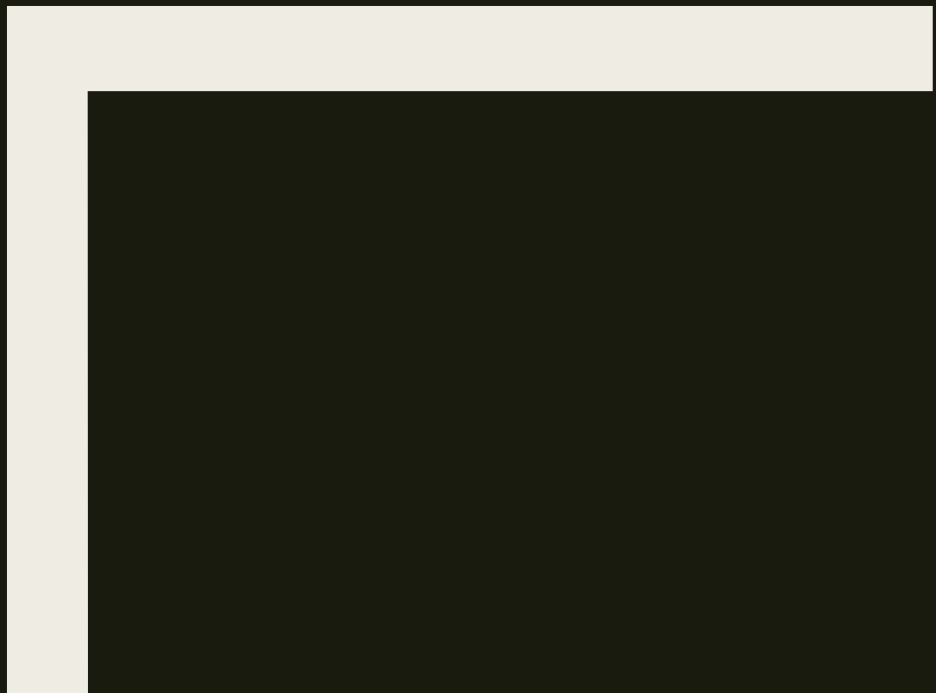
กำหนดให้

กล่องแรก กว้าง = 10 / ยาว = 15 / สูง = 10

กล่องสอง กว้าง = 2 / ยาว = 10 / สูง = 25

สูตรหาปริมาตร

ปริมาตรของกล่องสี่เหลี่ยม = กว้าง x ยาว x สูง

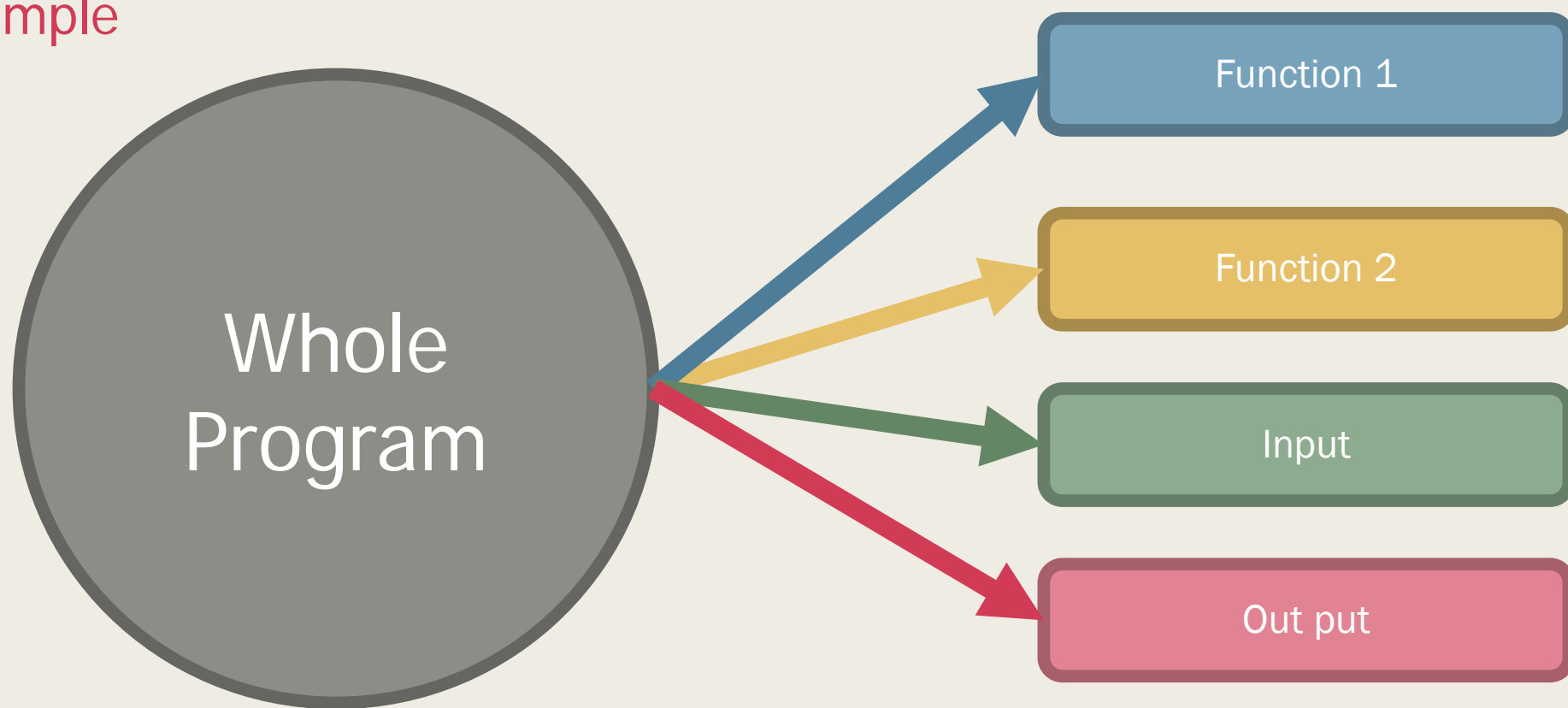


FUNCTION

What is Function?

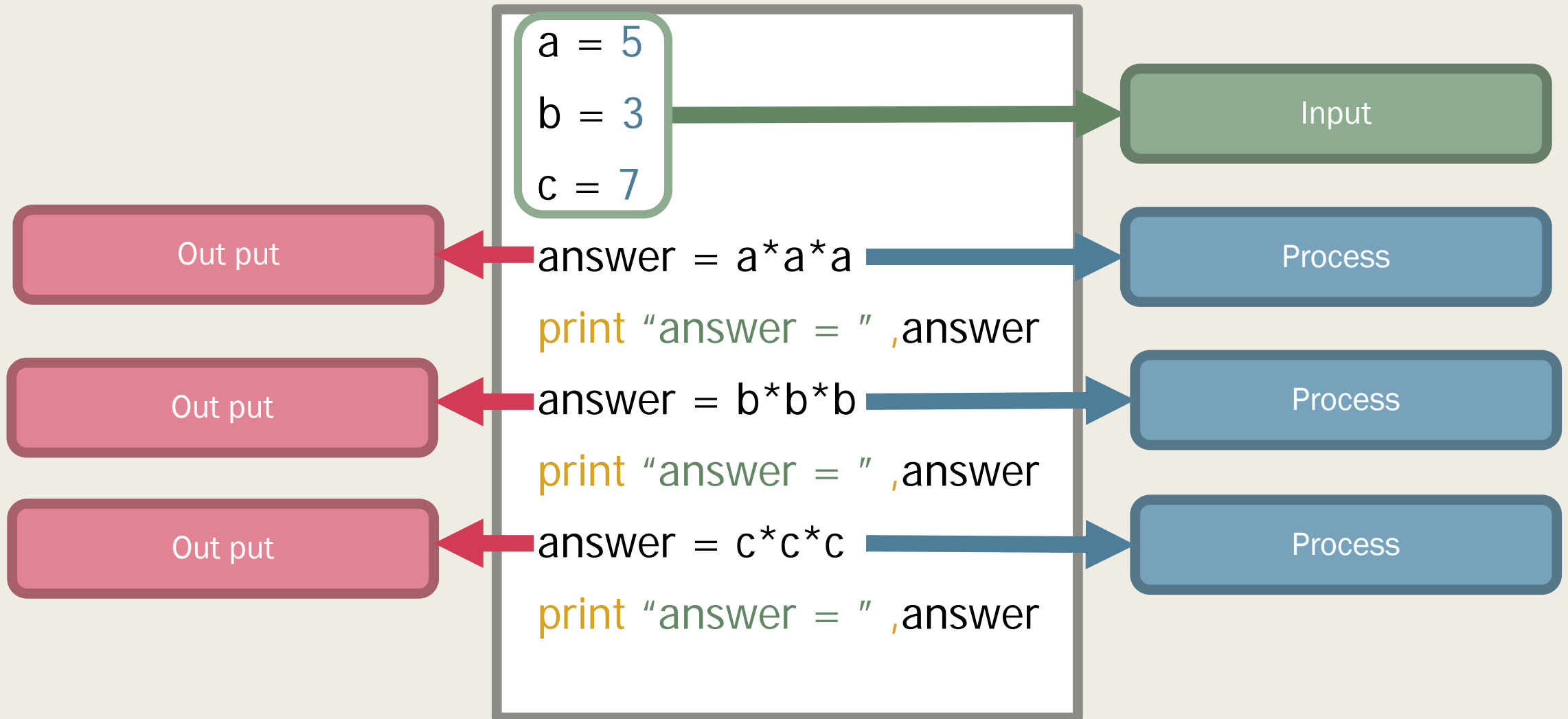
คือ การแตกปัญหาออกเป็นปัญหาย่อย ๆ ซึ่ง Function ที่มีหน้าที่ทำสิ่งต่าง ๆ ที่กำหนดไว้ตามแต่ Function

Example



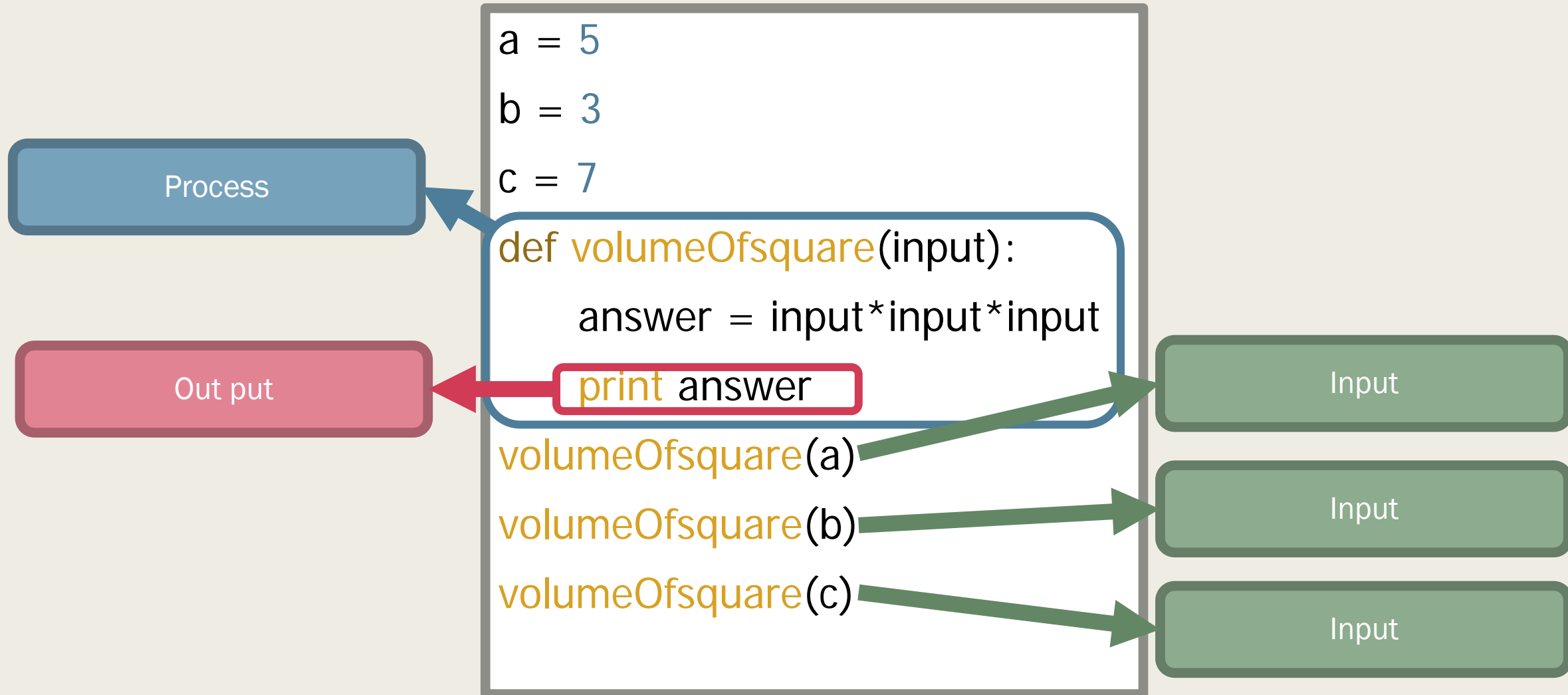
No function

ตัวอย่างโปรแกรมหาพื้นที่ของลูกบาศก์สี่เหลี่ยม **ที่ไม่ทำเป็น** function



Use function

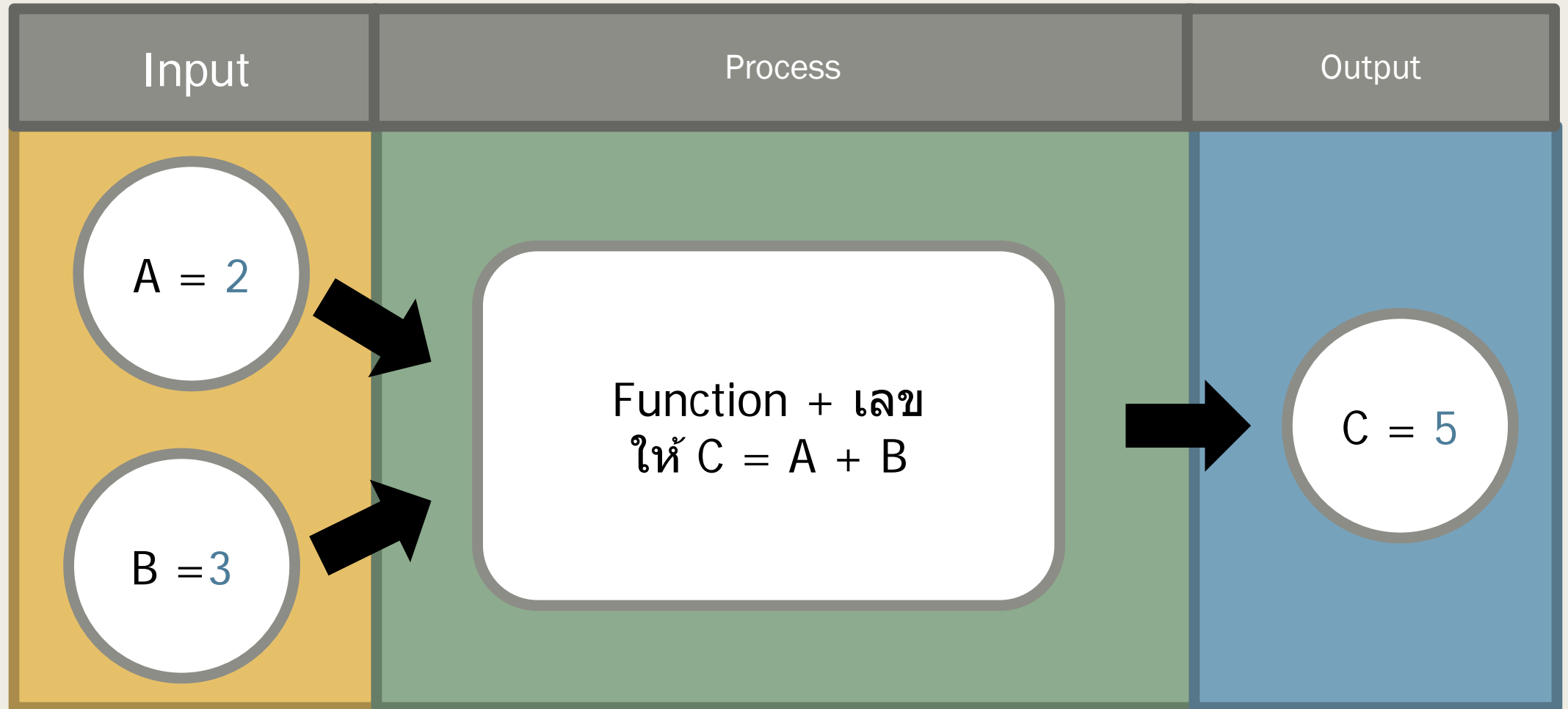
ตัวอย่างโปรแกรมหาพื้นที่ของลูกบาศก์สี่เหลี่ยม **ที่ทำเป็น** function



Process of function

Process ของ function แบ่งเป็น 3 ส่วน 1.Input 2.Process 3.Output

Example ตัวอย่างแสดง process ของ function + เลข



How good function

- 1) เขียนโปรแกรมได้ง่ายขึ้น
- 2) โอกาสที่จะผิดพลาดน้อยลง
- 3) นำกลับไปใช้กับโปรแกรมอื่น ๆ ได้
- 4) ลดขนาดของโปรแกรมลง ลดเวลา, ค่าใช้จ่าย, ซ่อมบำรุงโปรแกรม

How to Create Function?

1) พิมพ์ `def`

2) สร้างชื่อให้ function

3) พิมพ์ `()`: เพื่อเสร็จสิ้น
การประกาศ function

Tab สำคัญมาก
ส่วนใหญ่ code
process ของ
function จะอยู่
หลังจากกด tab 1
ครั้งจากด้านซ้าย
สุดของบรรทัด

```
def myFunction ():
```

Tab

```
    print "Hello function"
```

```
    print "not function"
```

4) หลังจากพิมพ์ `()`: เสร็จ
กด enter เพื่อสร้าง
process ว่าอยากให้
function ทำอะไร

ไม่อยู่ใน function

How to use your function?

Example

```
def myFunction ():  
    print "Hello function"
```

Output



ไม่มี output
เนื่องจาก ยังไม่มี
การเรียกใช้
function

How to use your function?

Example

```
def myFunction ():  
    print "Hello function"
```

myFunction ()

วิธีเรียกใช้ function
ประกาศชื่อ function
ตามด้วย ()

Output

Hello function

Return

Example

```
A = 2
def plus(integer_1):
    c = integer_1 + integer_1
    return c
answer = plus(A)
print answer
```

Output

4

Input ประกาศ
ให้ A = 2

ตัวแปรที่อยู่ใน () บ่ง
บอกว่า function นี้
ต้องการตัวแปรกี่ตัว
function นี้ คือ 1 ตัว

ประกาศ function plus
โดยให้ return c

ส่งค่า A ไปใน function
plus

การเรียกใช้ function ที่มีการ
return ต้องประกาศตัวแปรมารับค่า
ประกาศ answer มารับค่า

Multiple input

Example

ไม่จำเป็นต้อง
return ตัวแปร

```
A = 2
B = 3
def plus (integer_1, integer_2):
    return integer_1 + integer_2
answer = plus(A, B)
print answer
```

หาก function ต้องการ
ตัวแปรมากกว่า 1 ตัว
ตอน เรียกใช้ก็ต้องส่ง
ตัวแปร ไปในจำนวนที่
function ต้องการ
Function นี้ต้องการตัว
แปร 2 ตัว

Output

5

Multiple output

Example

```
A = 2
B = 3
def plus (integer_1,integer_2):
    a_2 = integer_1+ integer_1
    b_2 = integer_2+ integer_2
    return a_2,b_2
answer_1,answer_2 = plus(A,B)
print answer_1
print answer_2
```

สามารถ return ค่าได้
หลายค่า เวลาประกาศ
ตัวแปรรับก็ประกาศตาม
จำนวนที่ return มา

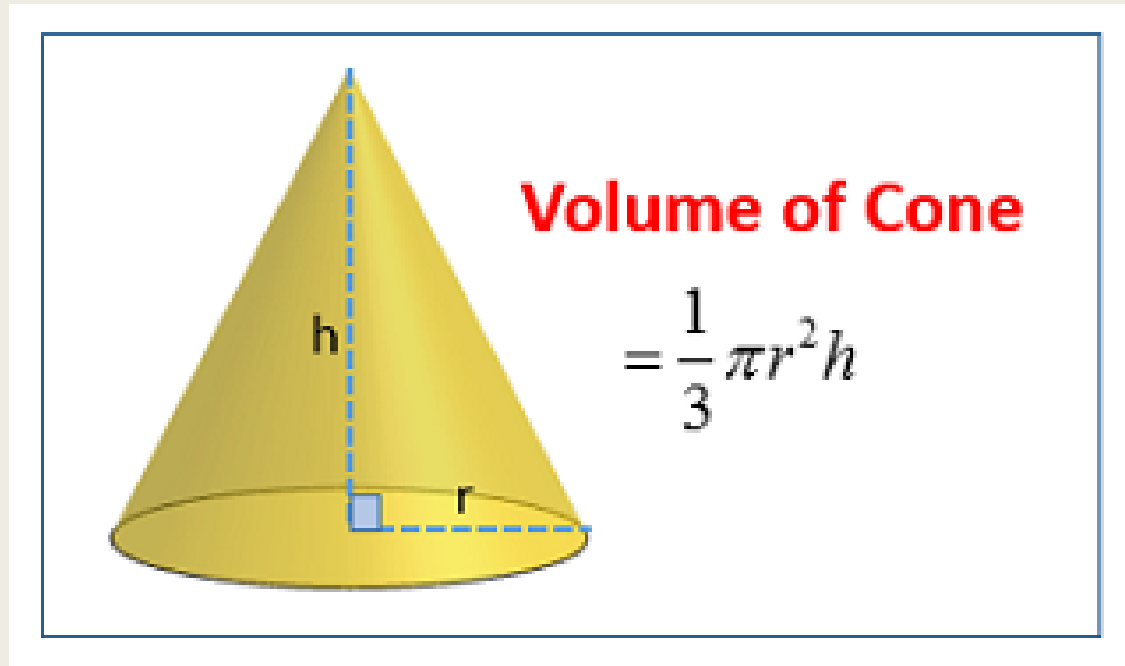
Output

4

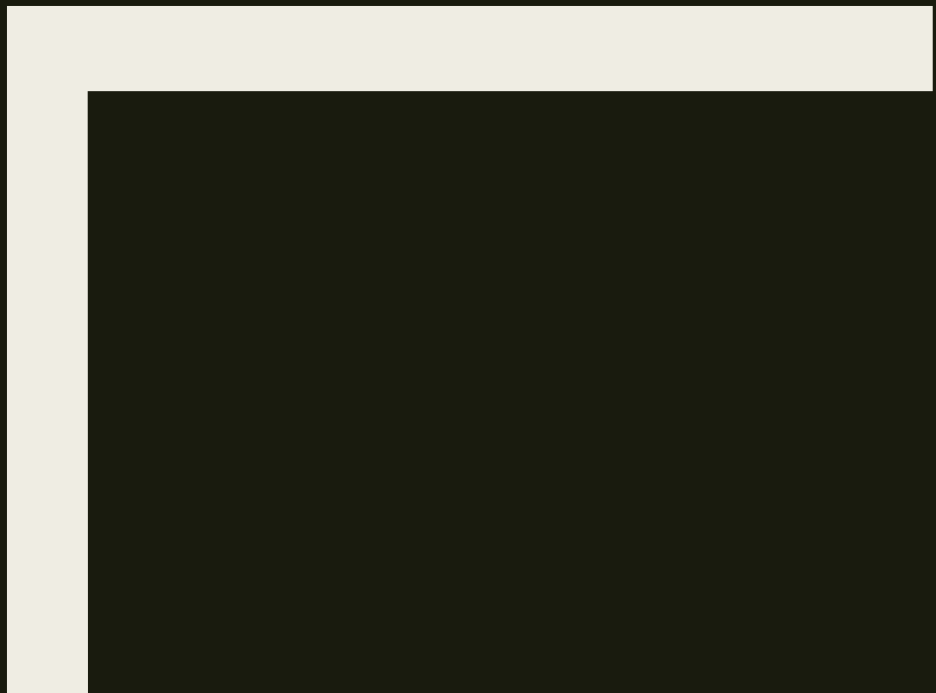
6

Exercise

จงสร้าง function ที่คำนวณปริมาตร ของพีระมิดทรงกรวย โดยให้รับค่ารัศมี และ ความสูงจาก keyboard **โดยให้ทำการ print คำตอบด้วย**



กำหนดให้ h = ความสูง, r = รัศมี(**จะต้องใช้ ****), $\pi = 3.141$



STANDARD CODING

What is standard coding?

คือ การตั้งมาตรฐาน รูปแบบ/วิธีการเขียนโค้ด ให้อยู่ในมาตรฐานเดียวกัน

What is utility of standard coding?

เราจะได้โค้ด ที่อยู่บน standard เดียวกัน หน้าตาเหมือนกัน คนอื่นอ่านง่ายเรา
อ่านง่าย

คำถาม ?

standard coding สำหรับโปรแกรมเมอร์คนเดียว จำเป็นหรือเปล่า?

Name

1.การตั้งชื่อตัวแปรต้องสื่อความหมาย

Example

```
normal_temperature = 25
```

2.การตั้งชื่อตัวแปรต้องเป็น snake_case

Example

```
volume_of_square = 50
```

3.การตั้งชื่อฟังก์ชันให้เป็น lower camel case

Example

```
■ def camelBack ():
```



THANK YOU