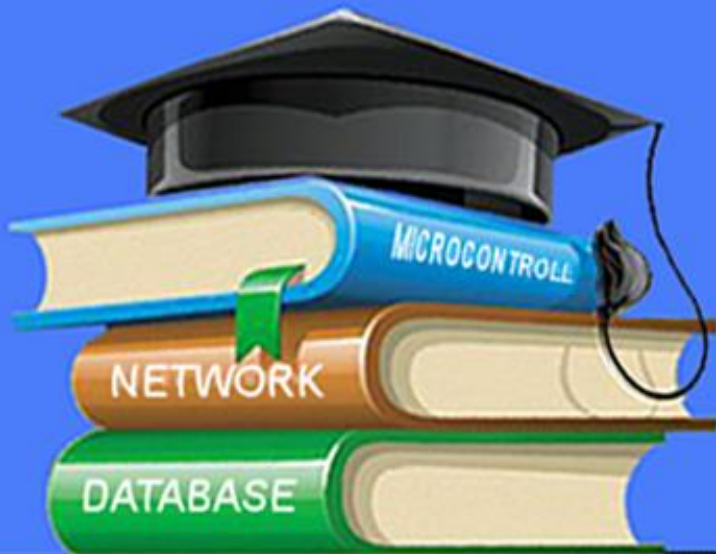


*Computers & Programming* 

# *Chapter 2: Python Basic Concept*



# Contents



**Python Overview**

**Variable and Constant**

**Statement**

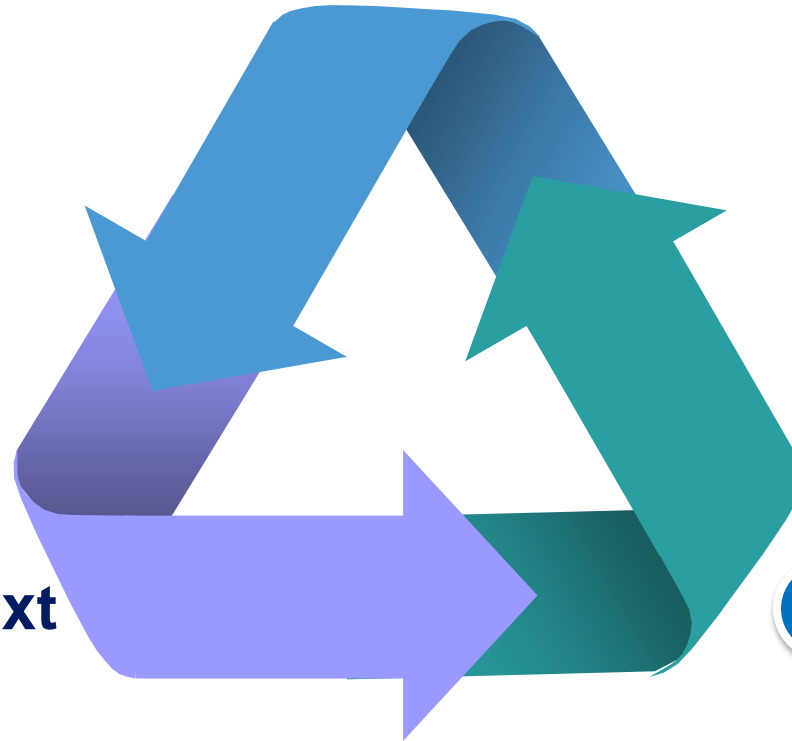
**Expression**

**math Class**

**Python**  
Basic Concept

# Python Program

1 case-sensitive



2 comment text

3 indentation rules

# Python Program

1 case-sensitive



2 comment text

3 indentation rules

## 1. Case-sensitive

ตัวพิมพ์ใหญ่  
ตัวพิมพ์เล็ก  
ถือว่าเป็นคนละตัว

```
pi = 1  
Pi = 2  
PI = 3  
pI = 4  
  
print(pi)
```

1



```
pi = 1  
pi = 2  
pi = 3  
  
print(pi)
```

3



# Python Program

1 case-sensitive

2 comment text



3 indentation rules

## 2. Comment text

คอมเมนต์ใช้เครื่องหมาย

“(3-single quote) หรือ #

- คอมไพเลอร์จะไม่ประมวลผล
- ใช้อธิบายโปรแกรมให้คนอื่น
- หรือเพื่อความเข้าใจของตัวเอง

```
'''
โปรแกรมนี้พัฒนาขึ้นในวันที่ 28 เมษายน 2565
'''

pi = 3.1416

radius = input("Please Enter Radius = ") #ขั้นตอนรับค่าจากคีย์บอร์ด
radius = int(radius)

area = pi*radius*radius

print(area)
```

# Python Program

1

case-sensitive

2

comment text



3

indentation rules

## 3. Indentation Rules

- รูปแบบการเขียนโปรแกรมภาษา Python โดยใช้หลักการเยื้องโค้ด หรือแท็บ เพื่อบ่งบอกว่า โค้ดบรรทัดเหล่านี้ทำงานอยู่ภายบล็อกใด เนื่องจากภาษา Python ไม่มีเครื่องหมายปีกกา { } จึงต้องใช้หลัก Indentation แทน

\*หมายเหตุ: ในบางครั้งไม่สามารถใช้ผสมกันได้ระหว่างเว้นวรรคกับแท็บ

# Python Program

1 case-sensitive

2 comment text



3 indentation rules

## 3. Indentation Rules

```
STATEMENT
if CONDITION:
    if CONDITION:
        STATEMENT
    else
        STATEMENT
STATEMENT
```



```
CODE BLOCK 1 BEGIN
CODE BLOCK 1 CONTINUES
CODE BLOCK 2 BEGIN
    CODE BLOCK 3 BEGIN
CODE BLOCK 2 CONTINUES
    CODE BLOCK 3 CONTINUES
CODE BLOCK 1 CONTINUES
```

# Python Program

1 case-sensitive

2 comment text



3 indentation rules

## 3. Indentation Rules

ผิดหลัก

indentation rules เนื่องจาก  
ได้ครอบอยู่ในบล็อกเดียวกัน

```
pi = 1
    pi = 2
pi = 3
```

```
print(pi)
```

```
File "<ipython-input-21-28a9ec281af3>", line 2
```

```
    pi = 2
    ^
```

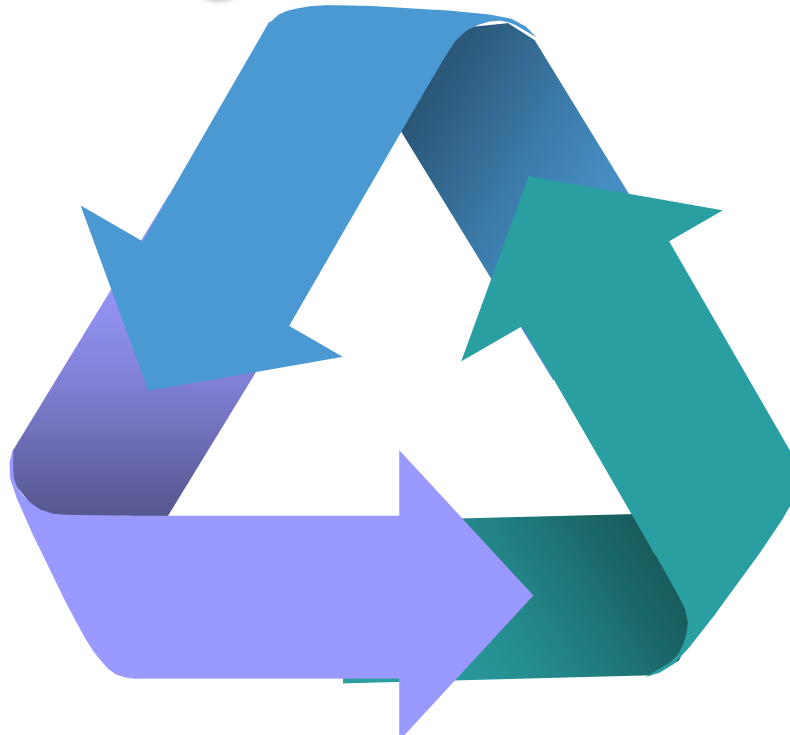
```
IndentationError: unexpected indent
```



# Python Program

ตัวพิมพ์ใหญ่, พิมพ์เล็ก ถือว่าเป็นคนละตัว

## 1 case-sensitive



ใช้หลักการแท็บเพื่อบ่งบอกว่า โค้ดบรรทัดเหล่านี้ทำงานอยู่ภายในบล็อกใด

## 2 comment text

คอมเมนต์ใช้เครื่องหมาย `'''` (3-single quote) หรือ `#` โดยคอมไพเลอร์จะไม่ประมวลผล.

## 3 indentation rules



# Reserved words

คำสงวน (Reserved words or keywords) คือ

คำที่ถูกกำหนดให้มีความหมายเฉพาะ และห้ามใช้เพื่อการอื่น  
ที่ไม่ได้กำหนดเอาไว้ของคอมไพเลอร์

## ตัวอย่างคำสงวน

def

import

global

lambda

return

## ความหมาย

พื้นที่ที่กำหนดนิยาม

การอ้างอิง Library

เรียกใช้ตัวแปรระหว่างนิยาม

การสร้างนิยามโดยไม่ตั้งชื่อ

ส่งค่ากลับจากฟังก์ชันไปยังที่เรียกใช้ฟังก์ชัน



# Reserved words

and	default	inline	pret_cast	typename
and_eq	delete	int	return	union
asm	do	long	short	unsigned
auto	double	mutable	signed	using
bitand	dynamic_cast	lambda	sizeof	virtual
bitor	else	new	static	void
bool	enum	not	static_cast	volatile
break	explicit	not_eq	struct	wchar_t
case	export	operator	switch	while
catch	extern	or	template	xor
char	false	or_eq	this	xor_eq
class	float	private	throw	def
compl	for	protected	true	
const	goto	public	try	
const_cast	if	register	typedef	



# Contents



Python Overview

**Variable and Constant**

Statement

Expression

math Class

**Python**  
Basic Concept

# Variable

## ตัวแปร (variable) คือ

สัญลักษณ์ที่สร้างขึ้นในโปรแกรมคอมพิวเตอร์เพื่อใช้เรียกแทนตำแหน่งของหน่วยความจำ และตัวแปรทุกตัวที่จะใช้งานในโปรแกรมได้จะต้องทำการประกาศก่อนเสมอ

m

44

int

n

77

int

## กฎการตั้งชื่อตัวแปร

- 1 จะต้องไม่เป็นตัวอักษรพิเศษ(เช่น \*%^+\$/) และต้องต่อเนื่องกันไม่มี  
การเว้นช่องว่าง สามารถใช้ขีดล่าง (\_) ได้
- 2 อักขระตัวแรกต้องไม่ใช่ตัวเลข
- 3 จะต้องไม่ใช่คำสงวน (*reserved words*).
- 4 ตัวพิมพ์ใหญ่, พิมพ์เล็ก ถือว่าเป็นคนละตัว (*case-sensitive*)

# Variable

## กฎการตั้งชื่อตัวแปร

- 1 จะต้องไม่เป็นตัวอักษรพิเศษ (เช่น \*%^+\$) และต้องต่อเนื่องกันไม่มีการเว้นช่องว่าง สามารถใช้ขีดล่าง (\_) ได้
- 2 อักขระตัวแรกต้องไม่ใช่ตัวเลข
- 3 จะต้องไม่ใช่คำสงวน (*reserved words*)
- 4 ตัวพิมพ์ใหญ่, พิมพ์เล็ก ถือว่าเป็นคนละตัว (*case-sensitive*)

## ตัวอย่างการตั้งชื่อตัวแปร

S&P



H2O



2You



The student



The-student



The\_student



int



A



5



# Variable

## กฎการตั้งชื่อตัวแปร

- 1 จะต้องไม่เป็นตัวอักษรพิเศษ (เช่น \*%^+\$) และต้องต่อเนื่องกันไม่มีการเว้นช่องว่าง สามารถใช้ขีดล่าง (\_) ได้
- 2 อักขระตัวแรกต้องไม่ใช่ตัวเลข
- 3 จะต้องไม่ใช่คำสงวน (*reserved words*)
- 4 ตัวพิมพ์ใหญ่, พิมพ์เล็ก ถือว่าเป็นคนละตัว (*case-sensitive*)

## ตัวอย่างการตั้งชื่อตัวแปร

\_test



2morrow



What?



theDog



return



Elizabeth\_the\_2nd



123x





# Python Variable Declaration

- ❖ การประกาศตัวแปรเป็นการแนะนำสัญลักษณ์ที่จะใช้ในโปรแกรม นอกเหนือจากค่าคงวน
  - ❖ ตัวแปรสามารถเก็บข้อมูลได้เท่านั้น
- ❖ การประกาศตัวแปร**ไม่จำเป็นต้อง**ระบุประเภทของข้อมูลที่ตัวแปรนั้นจะเก็บ

## ไวยากรณ์การประกาศตัวแปร (Syntax)

```
variable_name = initial_value
```

```
variable1, variable2,..., variableN = initial_value1, initial_value2,...,initial_valueN
```



# Python Variable Declaration

## ตัวอย่างการประกาศตัวแปร

score = 99

score

99

(int)

c = "A"

c

"A"

(string)

grade = 'A'

grade

'A'

(string)

'A' มาจากข้อมูลที่มีอยู่เดิม  
ในหน่วยความจำ



## ชนิดข้อมูล (Data type)

Type	Description	Range*
bool	Boolean value	true / false
int	Integer	123
float	Floating point number	123.456
complex	Complex number	$x + yj$
string	alphanumeric or character variables	"ABCD"

1 byte = 8 bits



# Example1: circle.py

```
pi = 3.1416
radius = 10

area = pi*radius*radius
print(area)
```

314.16

```
pi = 3.1416
radius = 10

area = pi*radius*radius
print(int(area))
```

314



```
x = 3
y = float(3)

print ("X is ", type(x) , "and Y is ", type(y))

X is  <class 'int'> and Y is  <class 'float'>
```



# Contents



Python Overview

Variable and Constant

**Statement**

Expression

math Class

**Python**  
Basic Concept



# Statement

Assignment Statement

Input & Output Statement





# Assignment Statement

## ข้อกำหนดเกี่ยวกับการกำหนดค่า

- การกำหนดค่า (assignment) ให้กับตัวแปร
- สามารถทำได้ด้วยการใช้เครื่องหมาย **assignment (=)**

## ไวยากรณ์ (Syntax):

`variable = expression`

## ข้อสังเกต

- คือการนำเอาค่าจากทางด้านขวาของเครื่องหมายเท่ากับไปเก็บไว้ทางด้านซ้ายของเครื่องหมายเท่ากับ
- ทางด้านซ้ายของเครื่องหมายเท่ากับสามารถเป็นสมการทางคณิตศาสตร์ หรือการคำนวณแบบต่าง ๆ ได้
- ทางด้านซ้ายของเครื่องหมายเท่ากับจะต้องเป็นตัวแปรเท่านั้น

# Assignment Statement

## ตัวอย่างการกำหนดค่า

score = 9

# กำหนดให้ตัวแปรชื่อ score มีค่าเท่ากับ 9

score

9

(int)

x = 3\*y

# กำหนดให้ตัวแปรชื่อ x มีค่าเท่ากับ 3 เท่าของค่าที่อยู่ในตัวแปร y

x

45

(int)

y

15

(int)

a, b, c = 5.0 # ประกาศ a, b, c พร้อมใส่ค่าเริ่มต้นให้ c

a = b = c\*0.5 # กำหนดค่าให้ a มีค่าเท่ากับ b และ b มีค่า 0.5 เท่าของ c

a

2.5

(float)

b

2.5

(float)

c

5.0

(float)





# Input & Output Statement

## ฟังก์ชันมาตรฐานสำหรับรับค่าและแสดงผล

**input()** คำสั่งที่เชื่อมโยงกับอุปกรณ์นำข้อมูลเข้า (input) มาตรฐาน คือ แป้นพิมพ์ โดยจะรับค่าในรูปแบบข้อความ ( String ) เท่านั้น

**print()** คำสั่งที่เชื่อมโยงกับอุปกรณ์แสดงผล (output) มาตรฐาน คือ จอภาพ



# Input Statement

## ไวยากรณ์ (Syntax):

```
variable = input()
```

## ตัวอย่างการใช้คำสั่ง input()

- `x = input()`
- `x, y = input("Enter two values: ").split()`

## วิธีการป้อนอินพุตสำหรับคำสั่ง input()

คำสั่ง `input()` สามารถรับได้หลายค่าโดย

- ป้อนแยกกันด้วยช่องว่าง (กด spacebar)
- ส่งค่าด้วยการกดเว้นบรรทัด (กด Enter)

เช่น: 3 6 (Enter)



# Output Statement

## ไวยากรณ์ (Syntax):

```
print (expression1,expression2 , ... ,expressionN)  
      {Variable/Integer/float/'String'/"String literal"}
```

## ตัวอย่างการใช้คำสั่ง print

- `print ("How many pennies do you have?")`

```
How many pennies do you have?
```

- `print("Total value = " , total )`

```
Total value = 53.95
```

- `print(12345)`

```
12345
```

- `print('A')`

```
A
```

# Output Statement

โดยปกติแล้วคำสั่ง `print()` จะจบข้อความด้วยการเป็นบรรทัด (“\n”) หากต้องจบข้อความด้วยรูปแบบอื่นจะต้องใช้คำสั่ง

`print (expression1,...,expressionN , end = “ตัวอักษรสุดท้ายของข้อความ”)`

## ตัวอย่างการใช้คำสั่ง `print`

- `print ("Hello engineering student" , end = " ")`
- `print("Welcome to 204111 class")`

Hello engineering student Welcome to 204111 class

- `x = 2`
- `print(x , end = "n")`

2n



## Example3: coins3.py

```
pennies = input("How many pennies do you have? ")  
nickels = input("How many nickels do you have? ")  
  
total = float(pennies)*0.01 + float(nickels)*0.05  
print("Total value = " , total )
```

How many pennies do you have? 3

How many nickels do you have? 4.654

## Quick check 2

1. จงหาข้อผิดพลาด และแก้ไขคำสั่งต่อไปนี้

```
pi = 3.1416
```

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

```
area = pi*radius*radius
```

```
print(area)
```

2. จงเขียนคำสั่งรับเลขจำนวนเต็มสามตัว ชื่อตัวแปร a,b,c

```
a,b,c = input().split()
```

```
a = input()
```

```
b = input()
```

```
c = input()
```



## Quick check 2

### 3. จงบอกค่าของตัวแปรซึ่งรับค่าตัวแปรคำสั่ง

```
x, y = input("Enter two values: ").split()
```

ถ้ากำหนดค่าจากคีย์บอร์ดตามตัวอย่างนี้ 25 52

ตัวแปร `minimum` มีค่าเท่ากับ 25

ตัวแปร `maximum` มีค่าเท่ากับ 52

# Contents



Python Overview

Variable and Constant

Statement

**Expression**

math Class

**Python**  
Basic Concept





# Statement

Arithmetic Expression

Relational Expression

Logical Expression



# Arithmetic Expression

## ตัวดำเนินการ

1.  $+$  คือ ตัวดำเนินการบวก
2.  $-$  คือ ตัวดำเนินการลบ
3.  $*$  คือ ตัวดำเนินการคูณ
4.  $/$  คือ ตัวดำเนินการหาร แบ่งเป็นการหารแบบ integer และ float
5.  $\%$  คือ ตัวดำเนินการหารเอาเศษ

## ข้อแตกต่างตัวดำเนินการ / และ %

$$17/5 = 3.4$$

$$\begin{array}{r} 3.4 \\ 5 \overline{) 17} \\ \underline{15} \\ 2 \end{array}$$

$$17\%5 = 2$$

# Arithmetic Expression

## ตัวอย่างการใช้ตัวดำเนินการ

1.  $4.5+3$  → 7.5
2.  $0.5*3$  → 1.5
3.  $5.0/2$  → 2.5
4.  $5/2$  → 2.5
5.  $5\%2$  → 1

## ตัวอย่างการแบ่งสมการทางคณิตศาสตร์เป็นภาษา Python

1.  $\frac{a+5}{a}$  →  $(a+5)/a$
2.  $\frac{5a+3}{a} + a$  →  $((5*a)+3)/a + a$   
หรือ  $(5*a+3)/a+a$



# Quick Check 3

- $3/5$
- $3/5.0$
- $25 \% 3$
- $17 + 8 / 5 - 7$
- $17.0 + 8.0 / 5.0 - 7.0$
- $17 + 8 / 5.0 - 7.0$
- $10 \% 3 - 5 / 2$
- $10 \% (3 - 4) / 2$

0.6

0.6

1

11.6

11.6

11.6

-1.5

0.0

# Arithmetic Expression

## ตัวดำเนินการ

1. **+=** คือ ตัวดำเนินการเพิ่มค่า
2. **-=** คือ ตัวดำเนินการลดค่า

## ตัวอย่างการใช้ตัวดำเนินการเพิ่มค่า และตัวดำเนินการลดค่า

$x += \rightarrow x = x + 1$

$x -= \rightarrow x = x - 1$

```
x=1  
x-=1  
print(x)
```

0

```
x=1  
x+=2  
print(x)
```

3

# Arithmetic Expression

## ตัวอย่างการใช้ตัวดำเนินการ

### ตัวดำเนินการ

1.  $+=$
2.  $-=$
3.  $*=$
4.  $/=$
5.  $\%=$

$$x+=1 \quad \rightarrow \quad x = x+1$$

$$x+=5 \quad \rightarrow \quad x = x+5$$

$$x*=7 \quad \rightarrow \quad x = x*7$$

$$x-=9 \quad \rightarrow \quad x = x-9$$

$$x-=-9 \quad \rightarrow \quad x = x-(-9)$$

$$x/=5 \quad \rightarrow \quad x = x/5$$

$$x\%=5 \quad \rightarrow \quad x = x\%5$$

$$x\%=x \quad \rightarrow \quad x = x\%x$$



## Example4: coins5.py

```
pennies = input("How many pennies do you have? ")
nickels = input("How many nickels do you have? ")

value = float(pennies) + 5 * float(nickels) + 10
dollar = value/100.0
cent = value%100
print("Total value = " , float(dollar) ,"dollar and ",int(cent)," cent")
```

# Contents



Python Overview

Variable and Constant

Statement

Expression

**math Class**

**Python**  
Basic Concept



## ฟังก์ชันในไลบรารี math

ฟังก์ชัน	หน้าที่
<code>math.sqrt()</code>	ค่ารากที่สองของ $x$
<code>math.pow(x, y)</code>	$x^y$
<code>math.sin(x)</code>	ค่าไซน์ (sine) ของ $x$ (หน่วยเรเดียน)
<code>math.cos(x)</code>	ค่าโคไซน์ (cosine) ของ $x$ (หน่วยเรเดียน)
<code>math.tan(x)</code>	ค่าแทนเจนต์ (tangent) ของ $x$ (หน่วยเรเดียน)
<code>math.exp(x)</code>	$e^x$
<code>math.log(x)</code>	(natural log) $\ln(x)$ , $x > 0$
<code>math.log10(x)</code>	(log ฐานสิบ) $\lg(x)$ , $x > 0$
<code>math.ceil(x)</code>	ปัดเลขทศนิยมให้เป็นมีค่าเป็นจำนวนเต็มน้อยสุดที่ $\geq x$
<code>math.floor(x)</code>	ปัดเลขทศนิยมให้เป็นมีค่าเป็นจำนวนเต็มมากที่สุดที่ $\leq x$
<code>math.fabs(x)</code>	ค่า absolute $ x $

# math Class

ไวยากรณ์ (Syntax):

```
function_name(expression1, expression2, ..., expressionn);
```

ตัวอย่าง

```
math.sqrt(x*5)
```

$$\sqrt{5x}$$

```
math.floor(x/y)
```

$$\left\lfloor \frac{x}{y} \right\rfloor$$

```
math.pow(x+y, z)
```

$$(x + y)^z$$

```
math.ceil(x-y)
```

$$\lceil x - y \rceil$$



## Example5: ปริมาตรทรงกลม

```
import math

radius = input()
radius = float(radius)

vol = 4 * 3.1416 * math.pow(radius,3)/3;

print(vol)
```

# Thank You !

