

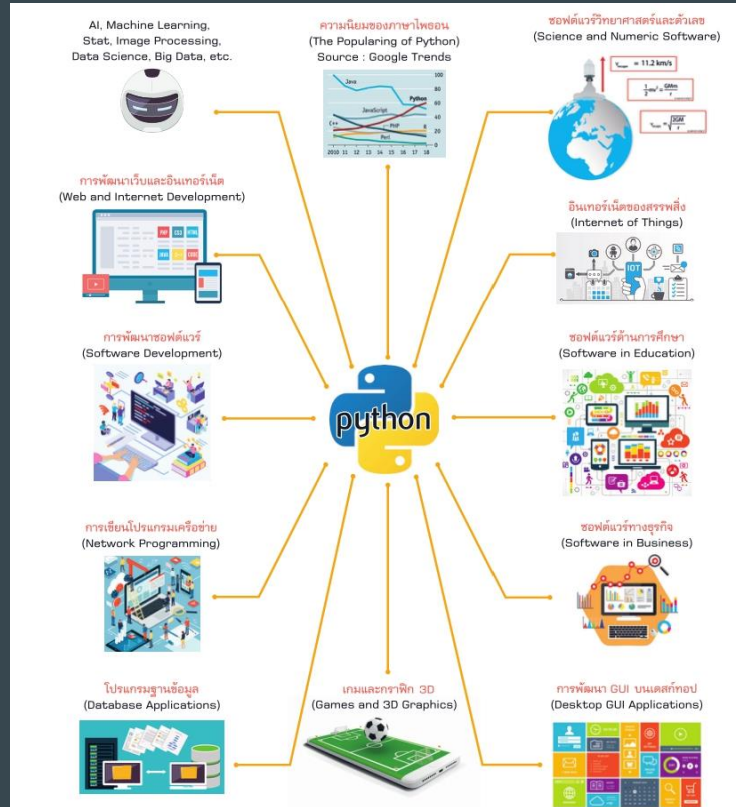
Unit 1

Python Language

ข้อดีของภาษา Python

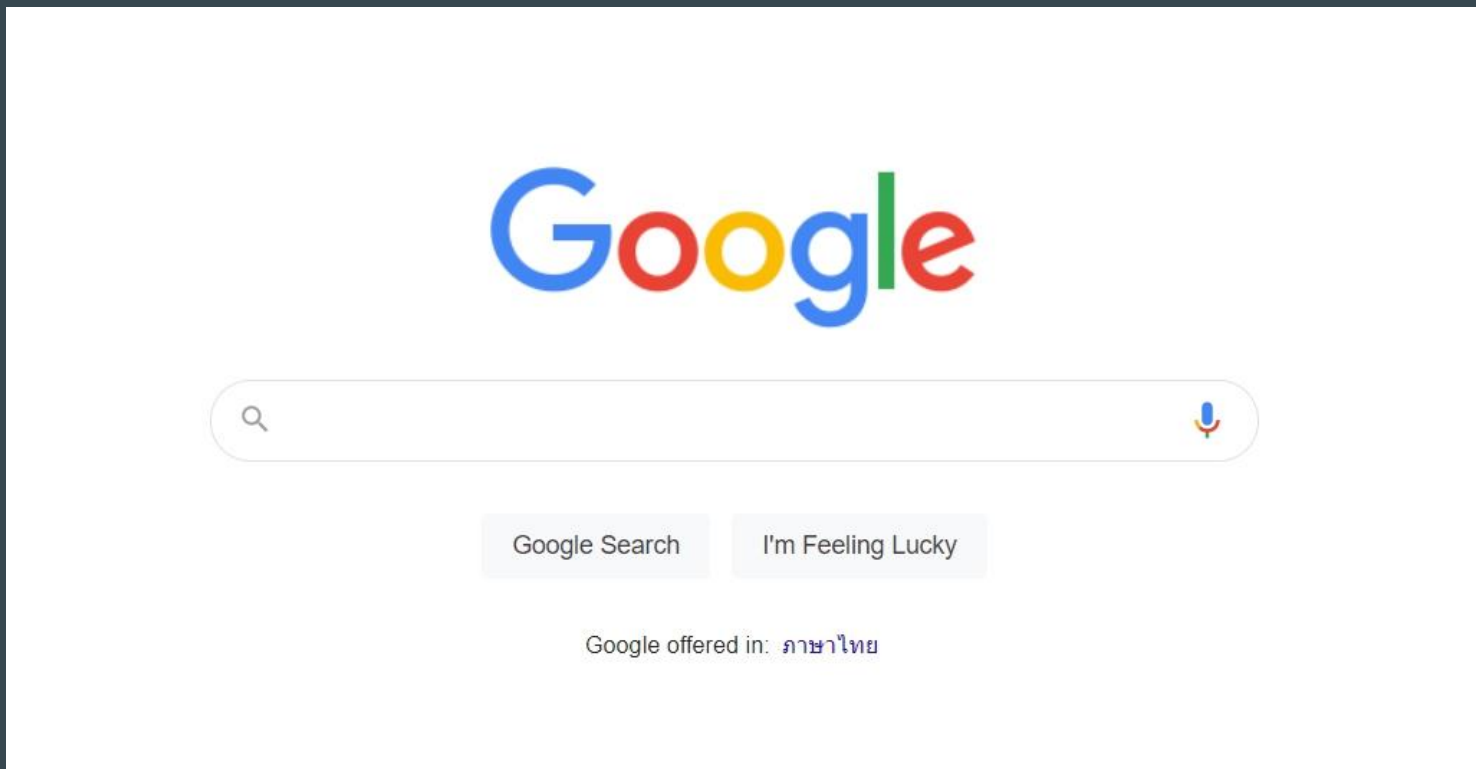
- มีโครงสร้างการเขียนคำสั่งโปรแกรมที่ง่าย ไม่ซับซ้อน เหมาะสำหรับผู้เริ่มต้น
- ไม่จำเป็นต้องกำหนดชนิดข้อมูลล่วงหน้าให้กับตัวแปร
- มี library จำนวนมาก ทั้งด้านคณิตศาสตร์และวิทยาศาสตร์ รวมถึงวิศวกรรมศาสตร์
- library การประมวลผลภาพ (Image Processing), library ทางด้านปัญญาประดิษฐ์ (AI), library การเรียนรู้ด้วยเครื่อง (Machine Learning)

ความสามารถของภาษา Python



▲ ไพธอนเป็นภาษาที่ครอบคลุมการทำงานเกือบทุกด้าน

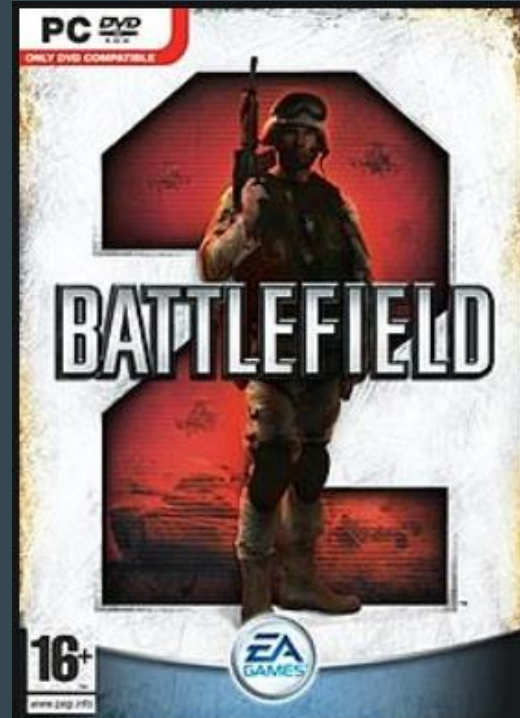
ความสามารถของภาษา Python



ความสามารถของภาษา Python



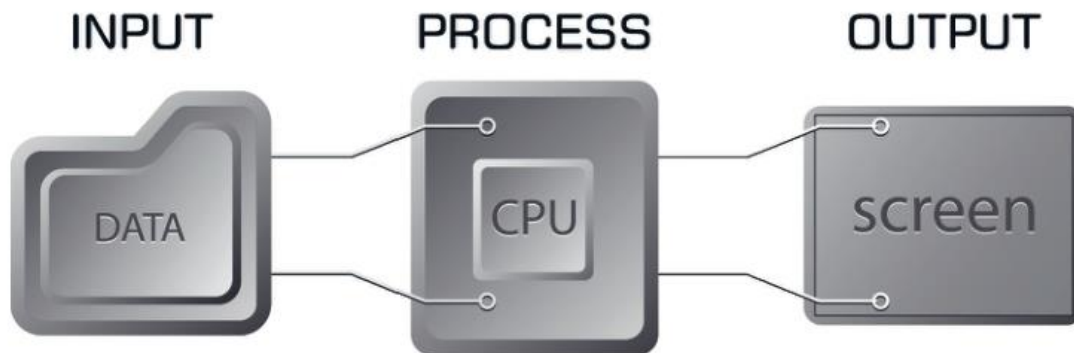
ความสามารถของภาษา Python



Unit 2

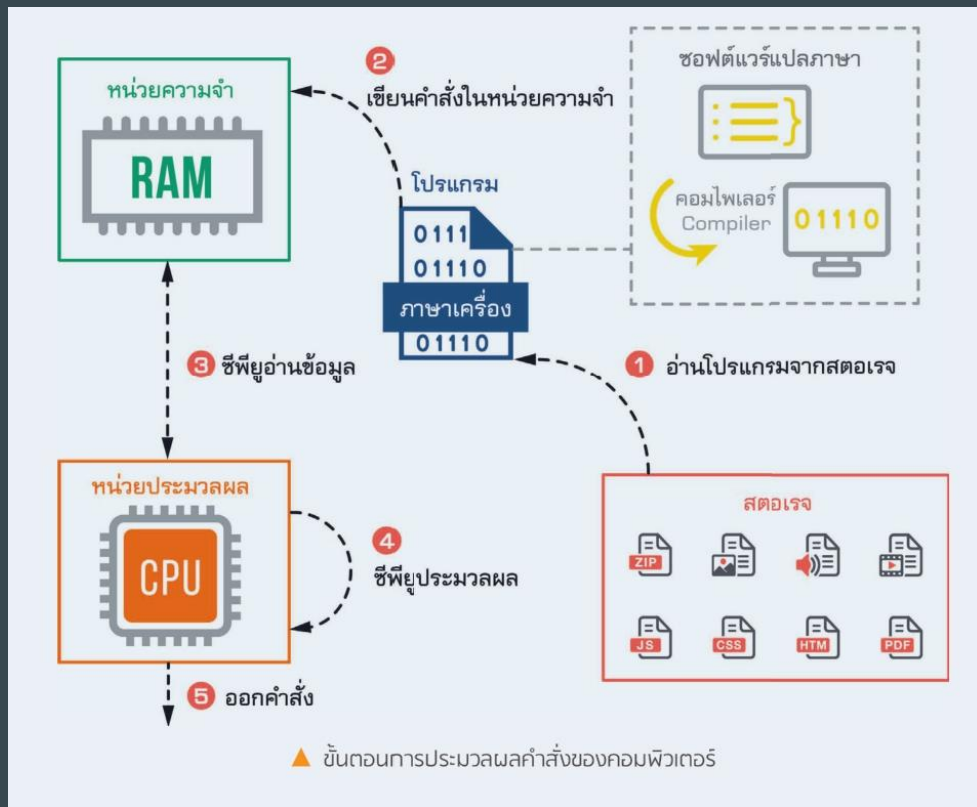
สิ่งที่ต้องรู้ก่อนเขียนโปรแกรม

หลักการทำงานของระบบคอมพิวเตอร์



▲ ระบบการทำงานพื้นฐานของคอมพิวเตอร์

ขั้นตอนการประมวลผลคำสั่ง



Compiler

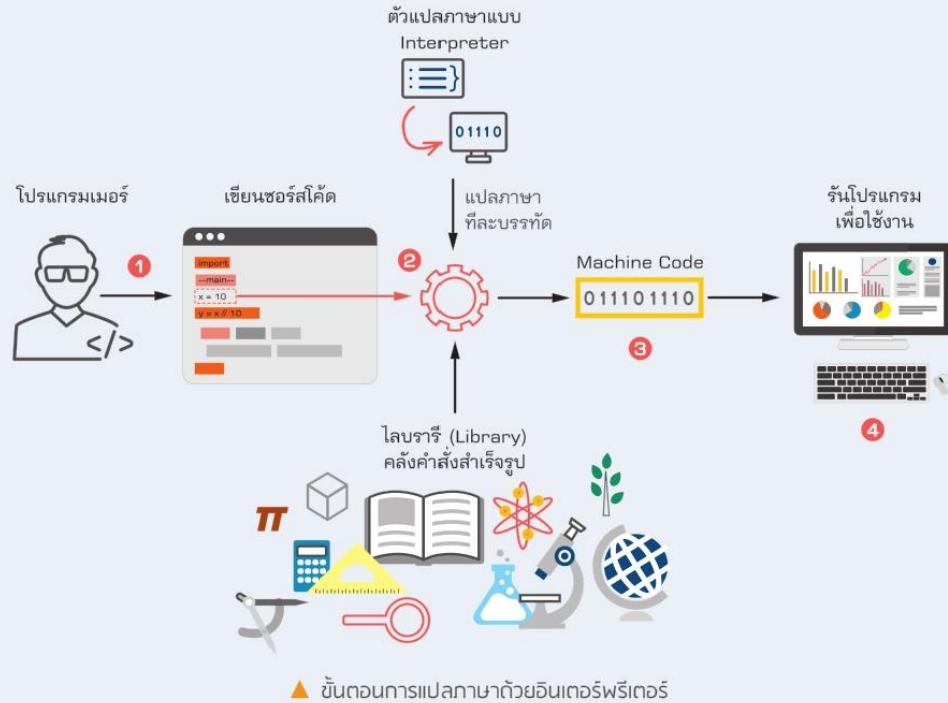


▲ ขั้นตอนการแปลภาษาด้วยตัวคอมไพเลอร์

Compiler

- C, C++, C#, Visual Basic, Swift
- ทำงานได้เร็วกว่า Interpreter
- ภาษาที่ใช้ compiler จะมีข้อกำหนดมาก ทำให้เรียนรู้ได้ช้ากว่าภาษาในกลุ่ม interpreter

Interpreter







Interpreter

- บางครั้งเรียกว่าภาษา script เช่น Python, JavaScript, PHP
- รูปแบบภาษาเรียนรู้และเข้าใจได้ง่าย
- ตรวจสอบความผิดพลาดได้ไม่ดีเท่า compiler

Unit 3

ฝึกลำดับความคิดด้วย

Flowchart

ชื่อสัญลักษณ์	สัญลักษณ์	ความหมาย
วงกลมหรือวงรี	 Start / Stop	ใช้สำหรับระบุเป็นจุดเริ่มต้น (Start) และจุดสิ้นสุด (Stop) ของโปรแกรม
สี่เหลี่ยมด้านขนาน	 Input / Output	ใช้สำหรับระบุเป็นอินพุตหรือเอาต์พุต (I/O) ระหว่างผู้ใช้งานกับคอมพิวเตอร์หรือโปรแกรม
สี่เหลี่ยมผืนผ้า	 Process	ใช้สำหรับการคำนวณหรือประมวลผล (Process) ของคอมพิวเตอร์
สี่เหลี่ยมขนมเปียกปูน	 Decision	ใช้สำหรับระบุทางเลือกหรือเงื่อนไขในกรณีที่เกิดการตัดสินใจมากกว่า 1 เส้นทาง
ลูกศร	 Direction	ใช้สำหรับระบุถึงทิศทางการทำงานของโปรแกรม (ขึ้น ลง ซ้าย หรือขวา)

Example

รับค่าตัวเลขจากผู้^๒ใช้ 2 จำนวน จากนั้นทำการหาผลรวมของ
ตัวเลขที่รับเข้ามา

Example

รับค่าตัวเลขจากผู้ใช้ 3 จำนวน จากนั้นทำการหาค่าเฉลี่ย

Example

รับค่าตัวเลขจากผู้ใช้ 2 จำนวน แล้วหาค่ามากที่สุด

Example

รับค่าขนาดหน้าอก เพื่อทำการตรวจสอบขนาดของเสื้อโดยมีเงื่อนไขดังนี้

- ถ้าขนาดหน้าอกไม่เกิน 34 จะได้เสื้อขนาด XS
- ถ้าขนาดหน้าอกไม่เกิน 36 จะได้เสื้อขนาด S
- ถ้าขนาดหน้าอกไม่เกิน 38 จะได้เสื้อขนาด M
- ถ้าขนาดหน้าอกไม่เกิน 40 จะได้เสื้อขนาด L
- ถ้าขนาดหน้าอกมากกว่า 40 จะได้เสื้อขนาด XL

Example

พิมพ์จำนวนเต็มตั้งแต่ 0 ถึง 10

Example

พิมพ์จำนวนเต็มตั้งแต่ 10 ถึง 0

Example

หาผลรวมตั้งแต่ 1 ถึง 10

Unit 4

Input & Output

Input & Output

- ใช้ `print()` ในการแสดงผลข้อมูล
- ใช้ `input()` ในการนำเข้าข้อมูล โดยข้อมูลที่เข้ามาจะอยู่ในรูปของ String
- หากต้องการข้อมูลเข้าที่เป็นตัวเลข ต้องทำการแปลงชนิดข้อมูลก่อน

Example

จงเขียนโปรแกรมรับตัวเลขเข้ามา 3 จำนวน โดยป้อนทีละจำนวน แล้ว
แสดงผลตัวเลขทั้งสามจำนวน โดยมีช่องว่างคั่น 1 ช่อง

Unit 5

Variables & Casting

Variable

ตัวแปร (Variable) -> ชื่อที่กำหนดขึ้นเพื่ออ้างอิงถึงตำแหน่งหน่วยความจำที่ใช้เก็บข้อมูล

- เก็บข้อมูลที่มีชนิดแตกต่างกันได้
- เปลี่ยนแปลงค่าข้อมูลได้

Casting

Casting -> การแปลงชนิดข้อมูล โดยใช้ Built-in Function ของ Python

- `int()` -> แปลงข้อมูลให้เป็นเลขจำนวนเต็ม
- `float()` -> แปลงข้อมูลให้เป็นเลขจำนวนจริง
- `str()` -> แปลงข้อมูลให้เป็น string

การตั้งชื่อตัวแปร

1. ต้องขึ้นต้นด้วยตัวอักษร ไม่ขึ้นต้นด้วยตัวเลขหรืออักขระพิเศษ ยกเว้น _ (underscore)
2. ตัวแปรสามารถประกอบด้วย A - Z, a - z, 0-9, _ (underscore) ได้ และไม่มีอักขระพิเศษประกอบ
3. ชื่อตัวแปรต้องไม่ซ้ำกับคำสั่ง เช่น if, while, import
4. ห้ามเป็นช่องว่างหรือเครื่องหมายอื่นๆ
5. การใช้ตัวอักษรพิมพ์เล็กและพิมพ์ใหญ่จะแตกต่างกัน เรียกว่า **case sensitive**

Unit 6

Data Types

ชนิดของข้อมูล (Data Types)

- ข้อมูลชนิดตัวเลข (Number) -> จำนวนเต็ม ทศนิยม จำนวนเชิงซ้อน
- ข้อมูลชนิดค่าความจริง (Boolean)
- ข้อมูลแบบเรียงลำดับ (Sequence) -> ข้อความ ลิสต์ ทูเปิล
- ข้อมูลชนิดเซต (Set)
- ข้อมูลชนิดดิกชันนารี (Dictionary)

ข้อมูลชนิดตัวเลข (Number)

- ข้อมูลเลขจำนวนเต็ม (Integer Number) -> 15
- ข้อมูลเลขจำนวนทศนิยม (Floating Point Number) -> 15.6
- ข้อมูลเลขจำนวนเชิงซ้อน (Complex Number) -> $3 + 5j$

Python จะรับรู้ชนิดข้อมูลจากการที่ผู้ใช้กำหนดค่า

ข้อมูลชนิดค่าความจริง (Boolean)

- เป็นข้อมูลที่มีค่าความจริงเป็นจริงหรือเป็นเท็จเท่านั้น
- แทนค่าความจริงด้วย True และแทนค่าความเท็จด้วย False
- $a = 10 > 3$ จะได้ True
- $a = 10 < 3$ จะได้ False

ข้อมูลแบบเรียงลำดับ(Sequence)

- ข้อมูลชนิดข้อความ (String) -> ข้อความ ตัวอักษร ตัวเลข ที่ไม่สามารถคำนวณได้ โดยอยู่ภายใน “ ”
- ข้อมูลชนิดลิสต์ (List) -> ข้อมูลที่เรียงต่อกันภายใน []
- ข้อมูลชนิดทูเปิล (Tuple) -> ข้อมูลที่เรียงต่อกันภายใน ()

Unit 7

Operators

Arithmetic Operator

Operator	Meaning
+	บวก
-	ลบ
*	คูณ
/	หาร
//	หารเอาผลจำนวนเต็ม
%	หารเอาเศษ
**	ยกกำลัง

Assignment Operator

Operator	Meaning
=	$a = 15$
+=	$a = a + 1$
-=	$a = a - 1$
*=	$a = a * 1$
/=	$a = a / 1$
//=	$a = a // 1$
%=	$a = a \% 1$
**=	$a = a ** 1$

Comparison Operator

Operator	Meaning
==	เท่ากับ
!=	ไม่เท่ากับ
>	มากกว่า
>=	มากกว่าหรือเท่ากับ
<	น้อยกว่า
<=	น้อยกว่าหรือเท่ากับ

Logical Operator

Operator	Meaning
and	และ
or	หรือ
not	นิเสธ

Logical Operator

p	q	p and q	p or q	not(p)	not(q)
True	True	True	True	False	False
True	False	False	True	False	True
False	True	False	True	True	False
False	False	False	False	True	True

ลำดับความสำคัญของเครื่องหมาย

- () สำคัญที่สุด
- ** สำคัญรองลงมา
- *, /, %, // มีความสำคัญเท่ากัน เจออันไหนมาก่อนก็ทำอันนั้นก่อน
- +, - สำคัญน้อยที่สุด
- ฉะนั้น ลำดับความสำคัญจากมากไปน้อย คือ (), **, *, /, %, //, +, -

Unit 8

MATH

Math

Function	Meaning
min()	หาค่าที่น้อยที่สุด
max()	หาค่าที่มากที่สุด
abs()	ถอดค่าสัมบูรณ์
pow()	ยกกำลัง
math.sqrt()	หารากที่สอง
math.ceil()	ปัดทศนิยมขึ้น (แสดงผลเป็นจำนวนเต็ม)
math.floor()	ปัดทศนิยมลง (แสดงผลเป็นจำนวนเต็ม)
math.pi	3.141592653589793

Unit 9

Booleans

Booleans

- ถ้าเชื่อมด้วย and ต้องเป็นจริงทั้งคู่ ถึงจะได้ผลลัพธ์เป็นจริง
- ถ้าเชื่อมด้วย or ต้องเป็นเท็จทั้งคู่ ถึงจะได้ผลลัพธ์เป็นเท็จ นอกนั้นเป็นจริง
- `bool(0)`, `bool("")` จะได้ผลลัพธ์เป็นเท็จ นอกนั้นเป็นจริง

Unit 10

If...Else

if (ถ้า)

เงื่อนไขต้องเป็นจริง จึงจะทำงาน

```
if condition:
```

```
    statement
```

if (ถ้า)

รับค่า n เป็นจำนวนเต็ม แล้วตรวจสอบว่าเป็นเลขคู่หรือไม่

```
n = int(input())  
if n % 2 == 0:  
    print("%d is even" % n)
```


if...else (ถ้า...มิฉะนั้น)

ถ้าเงื่อนไขใน if ไม่เป็นจริง จะทำ else

```
if condition:
```

```
    statement
```

```
else:
```

```
    statement
```

if...else (ถ้า...มิฉะนั้น)

รับค่า n เป็นจำนวนเต็ม แล้วตรวจสอบว่าเป็นเลขคู่หรือไม่

```
n = int(input())  
if n % 2 == 0:  
    print("%d is even" % n)  
else:  
    print("%d is odd" % n)
```

if...elif (ถ้า...มิฉะนั้น ถ้า...)

ถ้าเงื่อนไขใน if ไม่เป็นจริง จะทำ elif

```
if condition:
```

```
    statement
```

```
elif condition :
```

```
    statement
```

if...elif (ถ้า...มิฉะนั้น ถ้า...)

รับค่า n เป็นจำนวนเต็ม แล้วตรวจสอบว่าเป็นเลขคู่หรือไม่

```
n = int(input())  
if n % 2 == 0:  
    print("%d is even" % n)  
elif n % 2 != 0:  
    print("%d is odd" % n)
```

Nested if

```
if (condition) :  
    if (condition) :  
        Statement(s)  
    else:  
        Statement(s)  
elif (condition) :  
    if (condition) :  
        Statement(s)  
    elif (condition) :  
        Statement(s)  
elif (condition) :  
    Statement(s)  
else:  
    Statement(s)
```

Nested if

เขียนโปรแกรมโดยสอบถามรายได้ของผู้ใช้งานก่อนจะตัดสินใจว่า จะเลือกทำบัตรเครดิตแบบใด โดยต้องมีรายได้ตั้งแต่ 15000 บาท จึงจะทำบัตรเครดิตได้ และมีเงื่อนไขการทำบัตรดังนี้

- หากรายได้ต่ำกว่า 70000 บาท ให้ทำบัตรเงิน
- หากรายได้ต่ำกว่า 100000 บาท ให้ทำบัตรทอง
- หากรายได้มากกว่า 100000 บาท ให้ทำบัตร platinum

Unit 11

while loop

while

เงื่อนไขต้องเป็นจริง จึงจะทำงาน

```
while condition:
```

```
    statement
```


while

พิมพ์จำนวนเต็มตั้งแต่ 0 ถึง 10

```
i = 0;           //start
while i <= 10:   //stop
    print(i)
    i += 1;      //step
```

while

พิมพ์จำนวนเต็มตั้งแต่ 0 ถึง 10 โดยไม่เอาเลข 5

```
i = 0;
while i <= 10:
    if i == 5:
        i += 1;
        continue;
    print(i);
    i += 1;
```

While True (Infinite Loop)

รับค่าตัวเลขจากผู้ใช้เข้ามาเรื่อยๆ จนกว่าจะได้ -1 จึง
หยุด

```
while True:  
    n = int(input())  
    if n == -1:  
        break
```

Example

รับค่าจำนวนเต็มแล้วหาค่า **factorial** ของจำนวนนั้น

```
n = int(input())
```

```
i = n
```

```
sum = 1
```

```
while i >= 1:
```

```
    sum *= i
```

```
    i -= 1
```

```
print(sum)
```

Example

รับจำนวนเต็มบวก 5 จำนวน แล้วหาผลรวมทั้งหมด

```
i = 1
sum = 0
while i <= 5:
    n = int(input())
    sum += n
    i += 1
print(sum)
```

Analyse

รับค่าจำนวนเต็มแล้วหาค่า **factorial** ของจำนวนนั้น

$$5! = 5 * 4 * 3 * 2 * 1 = 120$$

step = -1

start = 5

stop = 1

Nested while

เงื่อนไขต้องเป็นจริง จึงจะทำงาน

```
while condition1:
```

```
    while condition2:
```

```
        statement2
```

```
    statement1
```

Unit 12

for loop

for

for data in range(start, stop, step)

for data in sequence

for

พิมพ์จำนวนเต็มตั้งแต่ 0 ถึง 10

```
for i in range(0, 11, 1):  
    print(i)
```

Unit 13

Strings

String

- สามารถเขียนอยู่ใน single quote (‘’) หรือ double quote (“”) ก็ได้

```
print("Hello")  
print('Hello')
```

String แบบหลายบรรทัด

- ใช้ double quote 3 ครั้ง หรือ single quote 3 ครั้ง

```
a = """Lorem ipsum dolor sit amet,  
consectetur adipiscing elit,  
sed do eiusmod tempor incididunt  
ut labore et dolore magna aliqua."""  
print(a)
```

```
Lorem ipsum dolor sit amet,  
consectetur adipiscing elit,  
sed do eiusmod tempor incididunt  
ut labore et dolore magna aliqua.
```

ผลลัพธ์

การเข้าถึง String

- ใช้ index ในการเข้าถึง

0	1	2	3	4	5	6	7	8	9	10
H	e	l	l	o		W	o	r	l	d
-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

การใช้ loop กับ String

- นิยมใช้ for เป็นหลัก

```
for x in "Hello":  
    print(x)
```

H
e
l
l
o

ผลลัพธ์

เช็คความยาว String

- ใช้ len()

```
a = "hello"  
print(len(a))
```

5

ผลลัพธ์

ตรวจสอบ String

- ใช้ in ในการตรวจสอบว่ามี string นั้นอยู่ใน string อีกตัวหรือไม่

```
a = "12345"  
print("2" in a)
```

True

ผลลัพธ์

การตัด String

a = "Hello, World!"

print(a[2:5]) → llo

print(a[:5]) → Hello

print(a[2:]) → llo, World!

print(a[-5:-2]) → orl

Unit 14

String Methods

Method	Meaning
capitalize()	แปลงตัวอักษรแรกให้เป็นตัวพิมพ์ใหญ่
title()	แปลงอักษรตัวแรกของแต่ละคำเป็นตัวพิมพ์ใหญ่
startswith()	สตริงขึ้นต้นด้วยค่าที่กำหนดหรือไม่ คืนค่า True ถ้าใช่
endswith()	สตริงลงท้ายด้วยค่าที่กำหนดหรือไม่ คืนค่า True ถ้าใช่
find()	ค้นหาตัวอักษร ถ้าพบจะคืนค่าเป็น index ถ้าไม่พบจะคืนค่า -1
isalnum()	ตรวจสอบว่าประกอบด้วยพยัญชนะ(a-z) หรือตัวเลข(0-9)หรือไม่ คืนค่า True ถ้าใช่
isalpha()	ตรวจสอบว่าประกอบด้วยพยัญชนะ(a-z) หรือไม่ คืนค่า True ถ้าใช่

Method	Meaning
isdigit()	ตรวจสอบว่าประกอบด้วยตัวเลข(0-9) หรือไม่ คืนค่า True ถ้าใช่
islower()	ตรวจสอบว่าเป็นพยัญชนะตัวพิมพ์เล็กหรือไม่ คืนค่า True ถ้าใช่
isupper()	ตรวจสอบว่าเป็นพยัญชนะตัวพิมพ์เล็กหรือไม่ คืนค่า True ถ้าใช่
lower()	แปลงข้อความให้เป็นตัวพิมพ์เล็ก
upper()	แปลงข้อความให้เป็นตัวพิมพ์ใหญ่
replace()	แทนที่สตริง
strip()	ลบช่องว่างที่ตำแหน่งแรกและตำแหน่งสุดท้าย
split()	แบ่งข้อความด้วยค่าที่กำหนด คืนค่าเป็น list
swapcase()	เปลี่ยนจากตัวพิมพ์ใหญ่เป็นพิมพ์เล็กและเปลี่ยน จากพิมพ์เล็กเป็นพิมพ์ใหญ่

Unit 15

ASCII CODE

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	\$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(72	48	H	104	68	h
9	09	Horizontal tab	41	29)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	61	3D	=	93	5D]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

Example

รับอักขระตัวพิมพ์ใหญ่เข้ามา 1 ตัว แล้วทำการแปลงให้เป็นตัวพิมพ์เล็ก

```
str1 = input()
lower = ord(upper) + 32
print(chr(lower))
```


Unit 16

Lists

Method	Meaning
append()	เพิ่ม element ไปที่ลำดับสุดท้ายของ list
clear()	ลบ element ทั้งหมดใน list
copy()	คัดลอก list
count()	นับจำนวน element ใน list ด้วยค่าที่ระบุ
extend()	เพิ่ม element(หลายตัว) ไปที่ลำดับสุดท้ายของ list
Index()	คืนค่าตำแหน่งของ element ใน list
insert()	เพิ่ม element โดยอ้างอิงตำแหน่ง
pop()	ลบ element โดยอ้างอิงตำแหน่ง
remove()	ลบ element ด้วยค่าที่เจาะจง
reverse()	กลับลำดับ element ใน list
sort()	เรียงลำดับ element ใน list

Unit 17

Functions

Functions

- ไม่มีพารามิเตอร์ (parameter) และไม่มีการคืนค่า (return)
- ไม่มีพารามิเตอร์ (parameter) แต่มีการคืนค่า (return)
- มีพารามิเตอร์ (parameter) แต่ไม่มีการคืนค่า (return)
- มีพารามิเตอร์ (parameter) และมีการคืนค่า (return)

Scope of Variables

- Global Variable → ใช้งานได้ทุกที่ในโปรแกรม
- Local Variable → ใช้งานได้เฉพาะภายในฟังก์ชันที่มีการประกาศไว้เท่านั้น เมื่อใช้งานเสร็จ ตัวแปรจะถูกลบทิ้ง