บทที่ 1
ความรู้เบื้องต้นของภาษาจาวา

บรรยายโดย ผศ.ดร.ธราวิเชษฐ์ ธิติจรูญโรจน์

หัวข้อ

01 การรับค่าผ่านทางคีย์บอร์ดและแสดงผล

02 ตัวแปร (Variable)

03 ตัวดำเนินการ (Operation)

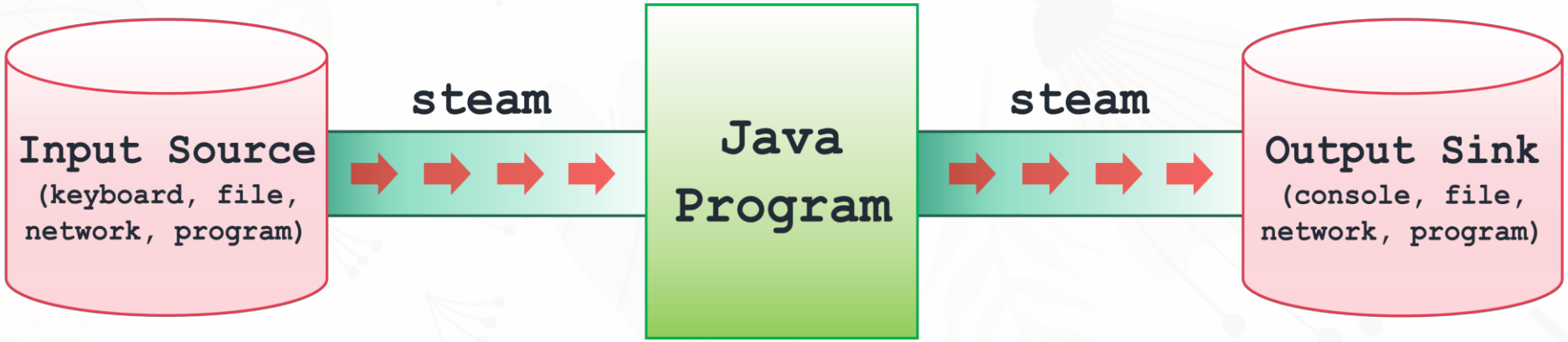
หัวข้อ

01 การรับค่าผ่านทางคีย์บอร์ดและแสดงผล

02 ตัวแปร (Variable)

03 ตัวดำเนินการ (Operation)

ช่องทางการสื่อสาร



ประกอบด้วย 3 ส่วนหลัก ได้แก่ ต้นทาง (input source) ช่องทางการแลกเปลี่ยนข้อมูล (steam) และปลายทาง (output sink)

การแสดงผลทางจอภาพ

การแสดงผลผ่านทางจอภาพ

```
System.out.print("...");
```

ให้พิมพ์ข้อความภายในเครื่องหมายคำพูด "..." เพื่อแสดงผลออกทางจอภาพ

line (new line)

```
System.out.println("...");
```

ให้พิมพ์ข้อความภายในเครื่องหมายคำพูด "..." เพื่อแสดงผลออกทางจอภาพ ก่อนขึ้นบรรทัดใหม่

การแสดงผลทางจอภาพ (ตัวอย่างโปรแกรม)

```
public class Main {
    public static void main(String[] args) {
        System.out.print("line 1");
        System.out.print("line 2");
        System.out.println("line 3");
        System.out.println("line 4");
    }
}
```

→ line 1 line 2 line 3
→ line 4
→

การแสดงผลทางจอภาพ (ตัวอย่างโปรแกรม)

```
public class Main {
    public static void main(String[] args) {
        System.out.println("line 1");
        System.out.print("line 2");
        System.out.println("line 3");
        System.out.print("line 4");
    }
}
```

line1
line2line3
line4

Scanner = libralies รับค่าของ Java

การรับค่าผ่านทางคีย์บอร์ด

Folder

ขั้นที่ 1 นำเข้า Package (ชุดคำสั่งที่จำเป็น)

```
import java.util.*;      หรือ
import java.util.Scanner;
```

→ ALL

↳ เอาแค่ libraries Scanner

ขั้นที่ 2 สร้าง Object เพื่อเชื่อมต่อระหว่างคีย์บอร์ดกับโปรแกรม

ชื่อตัวแปร

ท่อเชื่อมระหว่างโปรแกรมกับ keyboard

```
Scanner sc = new Scanner(System.in);
```

โดยที่ sc คือ object ที่ทำหน้าที่เป็นตัวกลางในการรับค่าจากคีย์บอร์ดมาสู่โปรแกรม

การรับค่าผ่านทางคีย์บอร์ด

ขั้นที่ 3 การรับค่าผ่านทางคีย์บอร์ด

กรณีต้องการรับค่าเป็น String

ชื่อตัวแปร ชื่อตัวแปร

```
String str = sc.nextLine();
```

กรณีต้องการรับค่าเป็น Integer

ชื่อตัวแปร

```
int num = sc.nextInt();
```

กรณีต้องการรับค่าเป็น float

ชื่อตัวแปร

```
float num = sc.nextFloat();
```

กรณีต้องการรับค่าเป็น double

ชื่อตัวแปร

```
double num = sc.nextDouble();
```

การรับค่าผ่านทางคีย์บอร์ด (ตัวอย่างโปรแกรม)

```
import java.util.*; 1

public class Main {
    public static void main(String[] args) {

สร้างท่อ 2  Scanner sc = new Scanner(System.in);
รับค่า 3  String str = sc.nextLine();

    }
}
```

การรับค่าผ่านทางคีย์บอร์ด (ตัวอย่างโปรแกรม)

```
import java.util.*;

public class Main {
    public static void main(String[] args) {

สร้างท่อ  Scanner sc = new Scanner(System.in);
รับค่า    int num1 = sc.nextInt();
          double num2 = sc.nextDouble();
    }
}
```

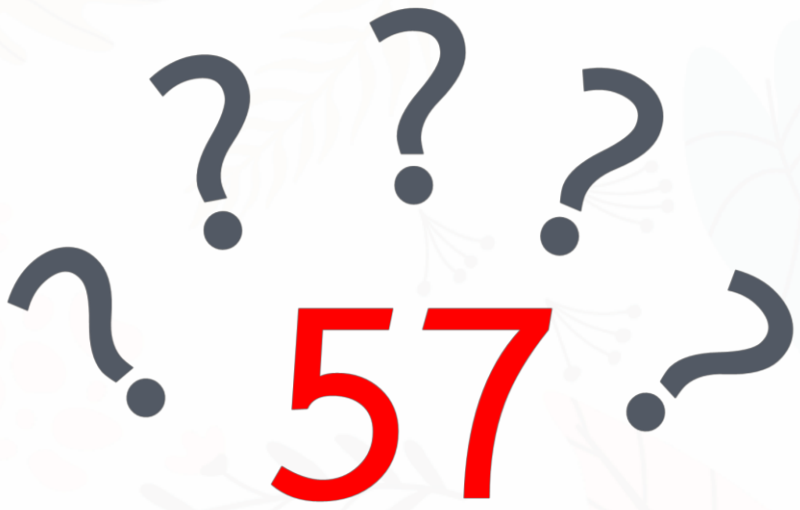
หัวข้อ

01 การรับค่าผ่านทางคีย์บอร์ดและแสดงผล

02 ตัวแปร (Variable)

03 ตัวดำเนินการ (Operation)

ตัวแปร (Variable)



ตัวแปร คือ สิ่งที่ใช้สำหรับการจัดเก็บข้อมูลหรือสารสนเทศ เพื่อนำมา

(1) จัดการหรืออ้างอิงการใช้งานในภายหลัง

(2) ติดป้ายบ่งบอกว่าสิ่งที่เก็บคืออะไร

(3) เป็นการให้ความหมายของสิ่งที่จัดเก็บ

ตัวแปร (Variable)

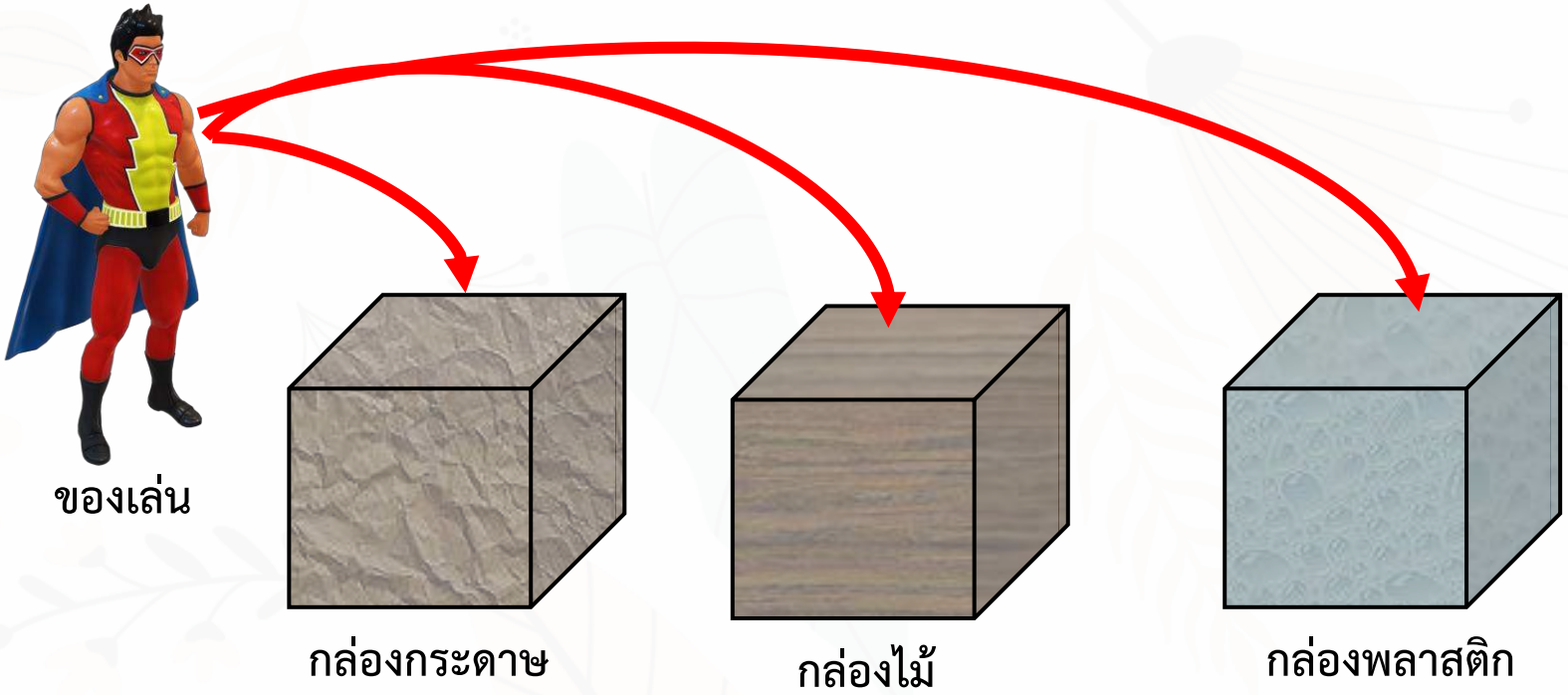
ตัวแปร คือ สิ่งที่ใช้สำหรับการจัดเก็บข้อมูลหรือสารสนเทศ เพื่อนำมา

(1) จัดการหรืออ้างอิงการใช้งานในภายหลัง

(2) ติดป้ายบ่งบอกว่าสิ่งที่เก็บคืออะไร

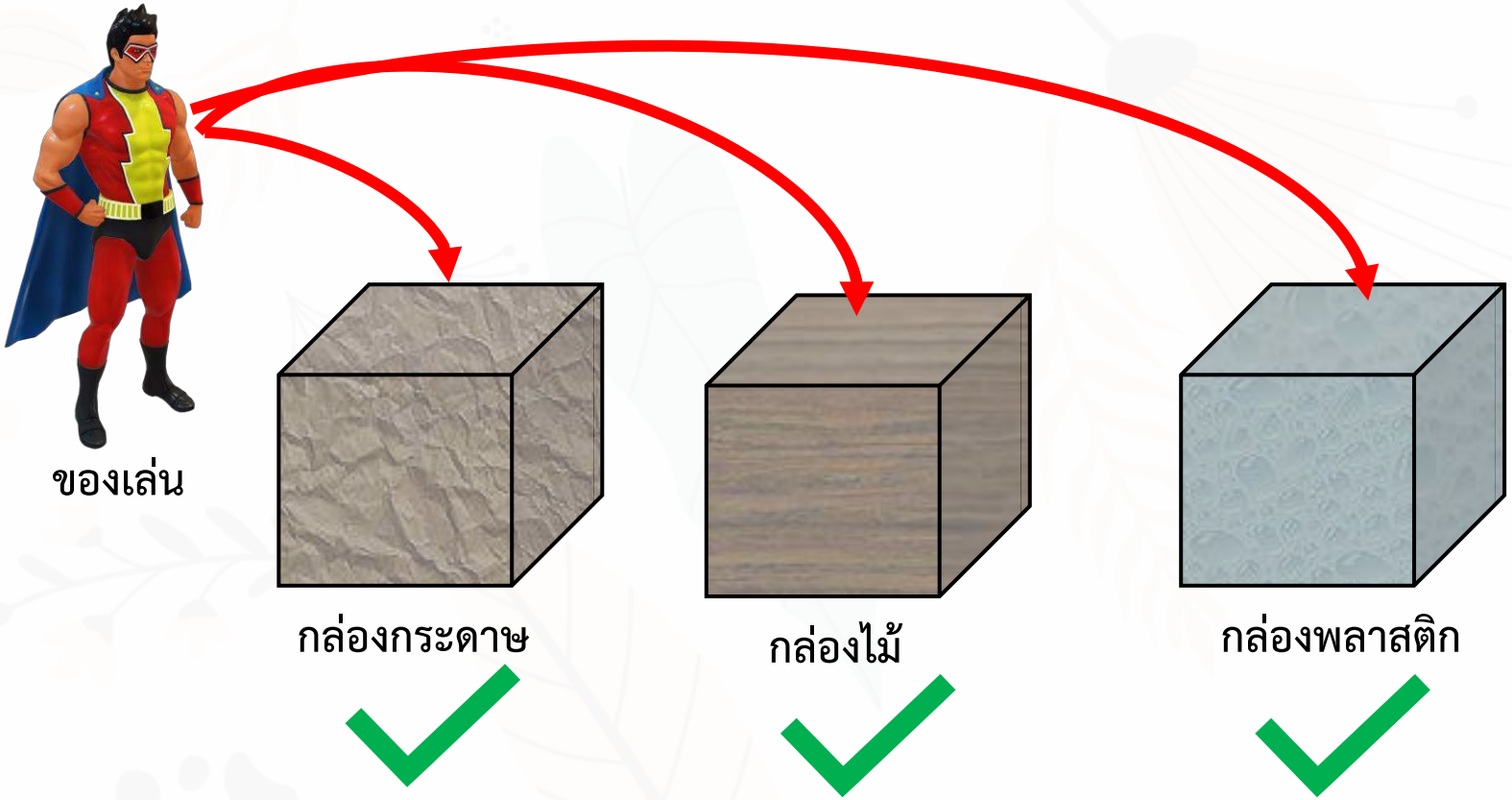
(3) เป็นการให้ความหมายของสิ่งที่จัดเก็บ

ตัวแปร (Variable)

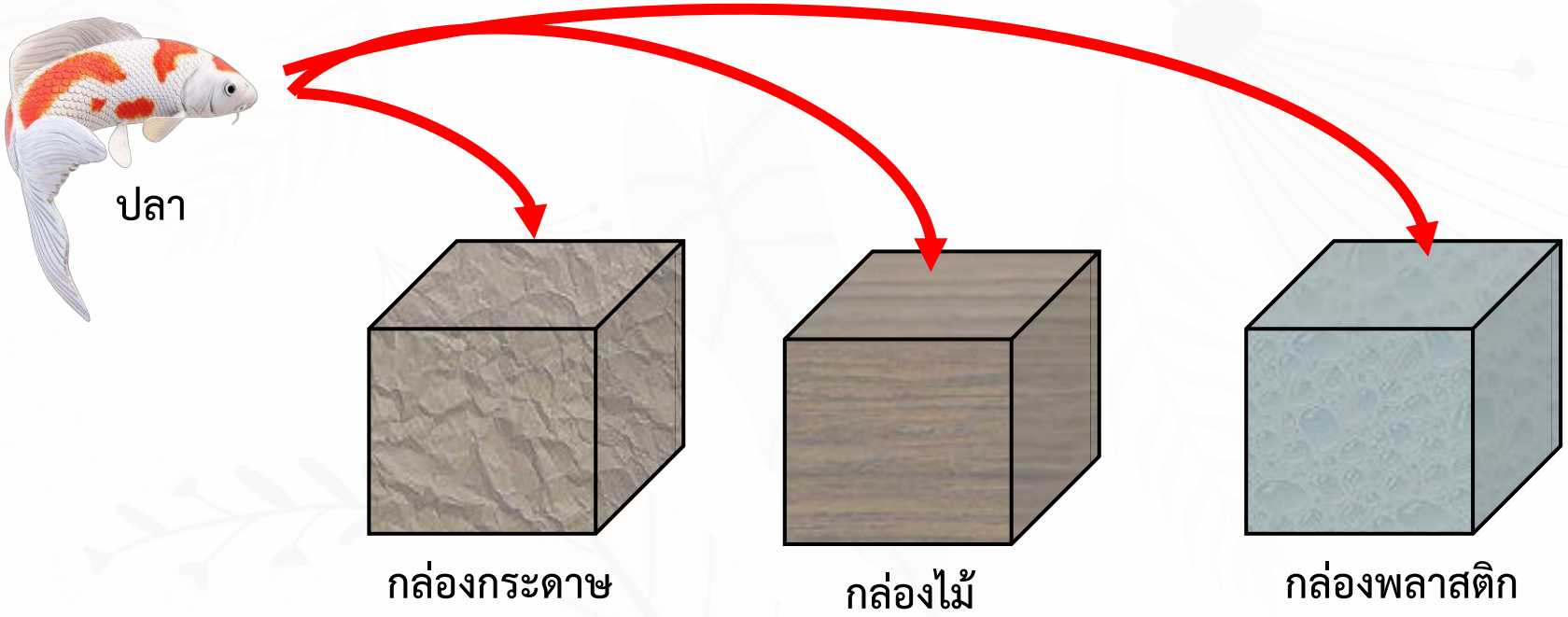


นักศึกษาสามารถมองว่าตัวแปรเป็นกล่องที่ใช้สำหรับเก็บของได้ ซึ่งการเลือกกล่องสำหรับเก็บของ นักศึกษาต้องพิจารณาจาก 2 ปัจจัย ได้แก่ (1) ชนิดของกล่อง และ (2) ขนาดของกล่อง

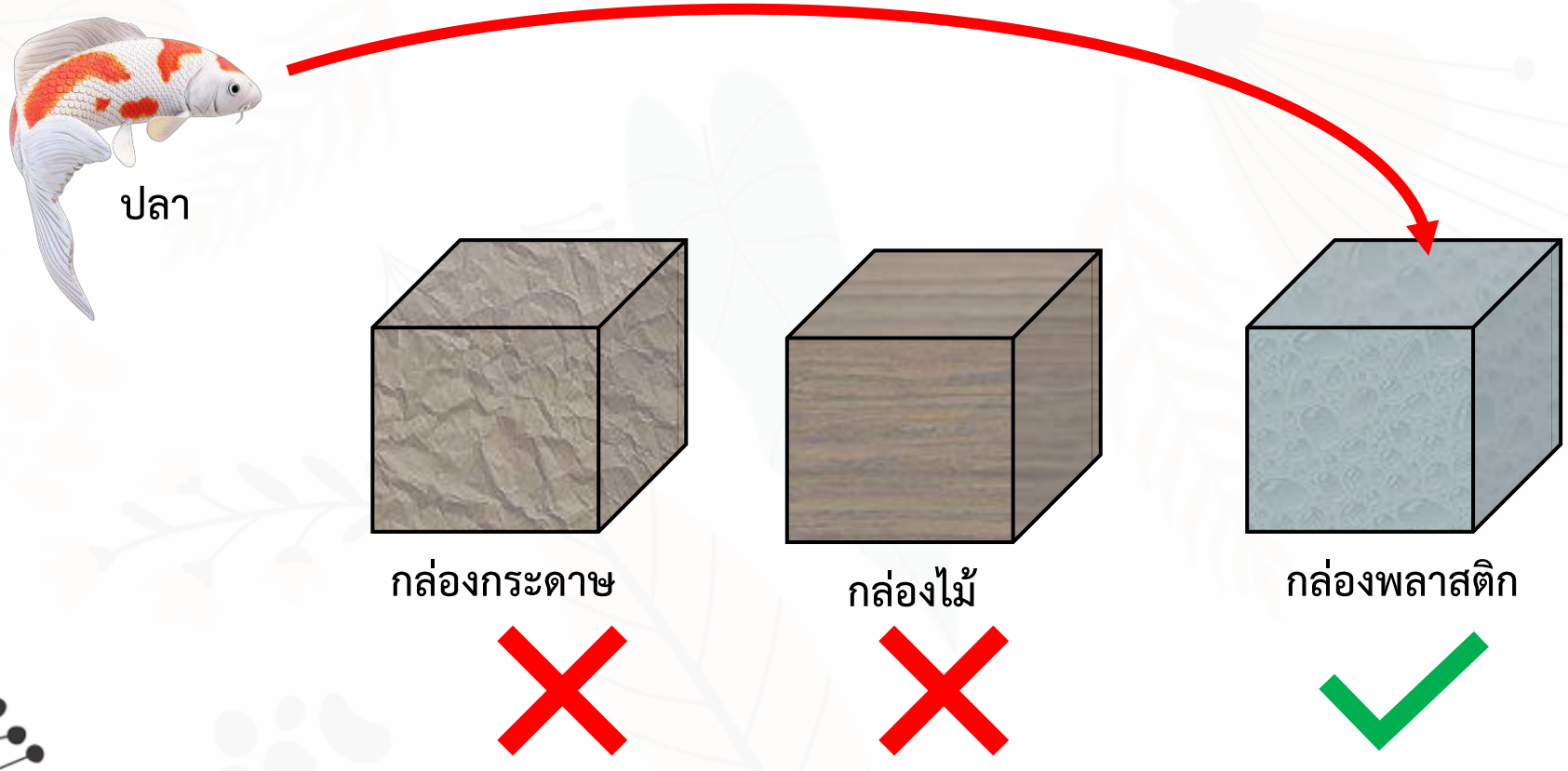
ตัวแปร (Variable)



ตัวแปร (Variable)

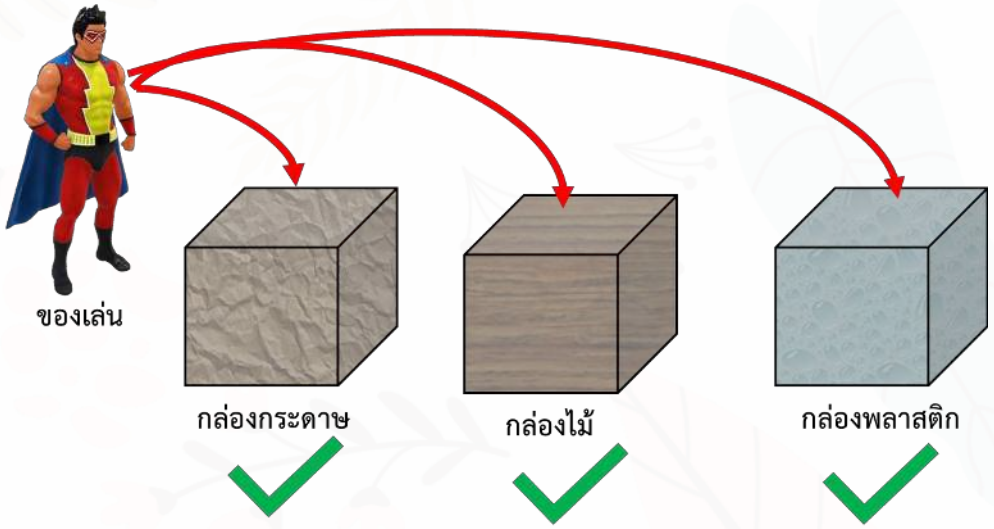


ตัวแปร (Variable)

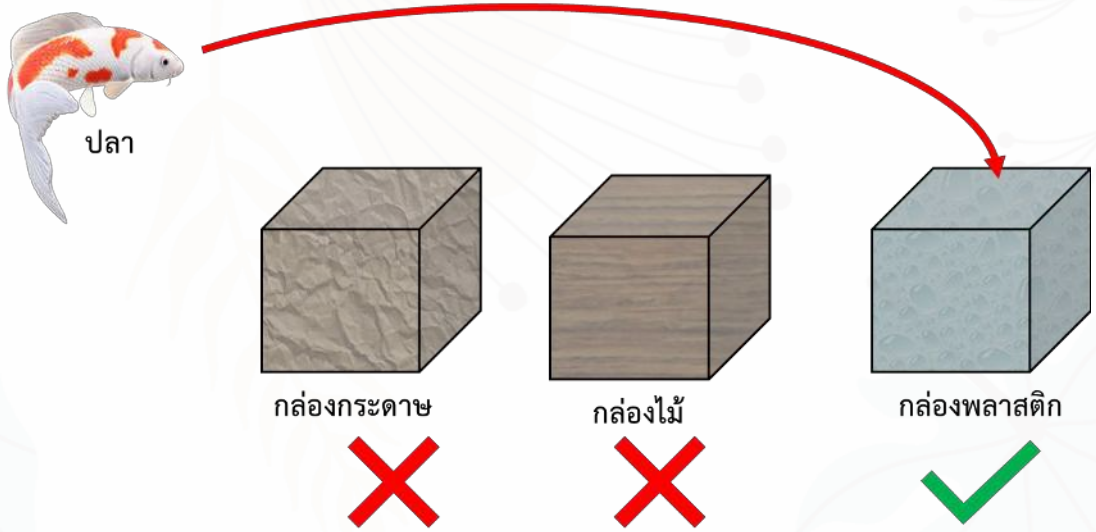


ตัวแปร (Variable)

จากตัวอย่างจะพบว่า “นักศึกษาต้องเลือกชนิดของกล่องให้เหมาะสมกับประเภทของที่ต้องการจัดเก็บ”

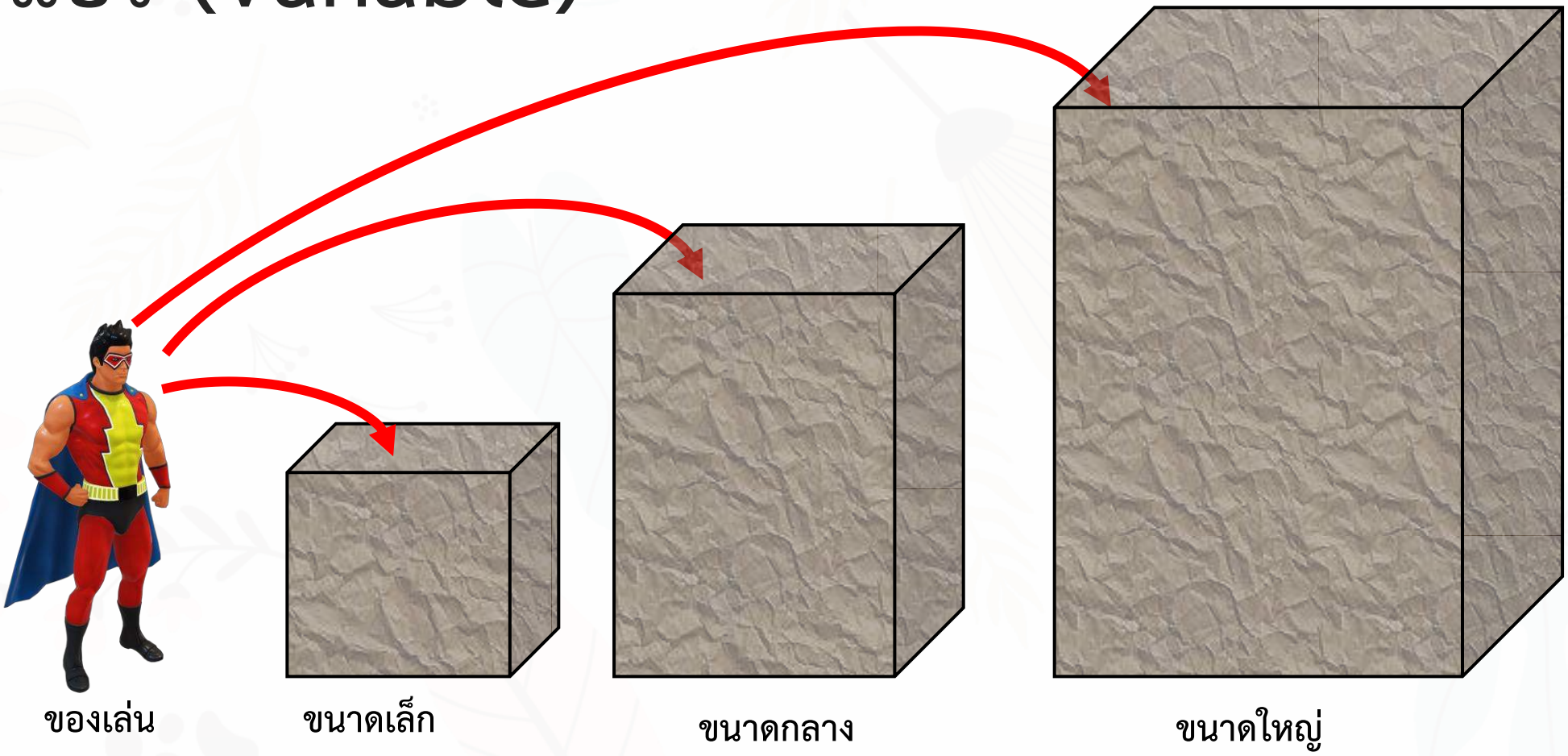


01



02

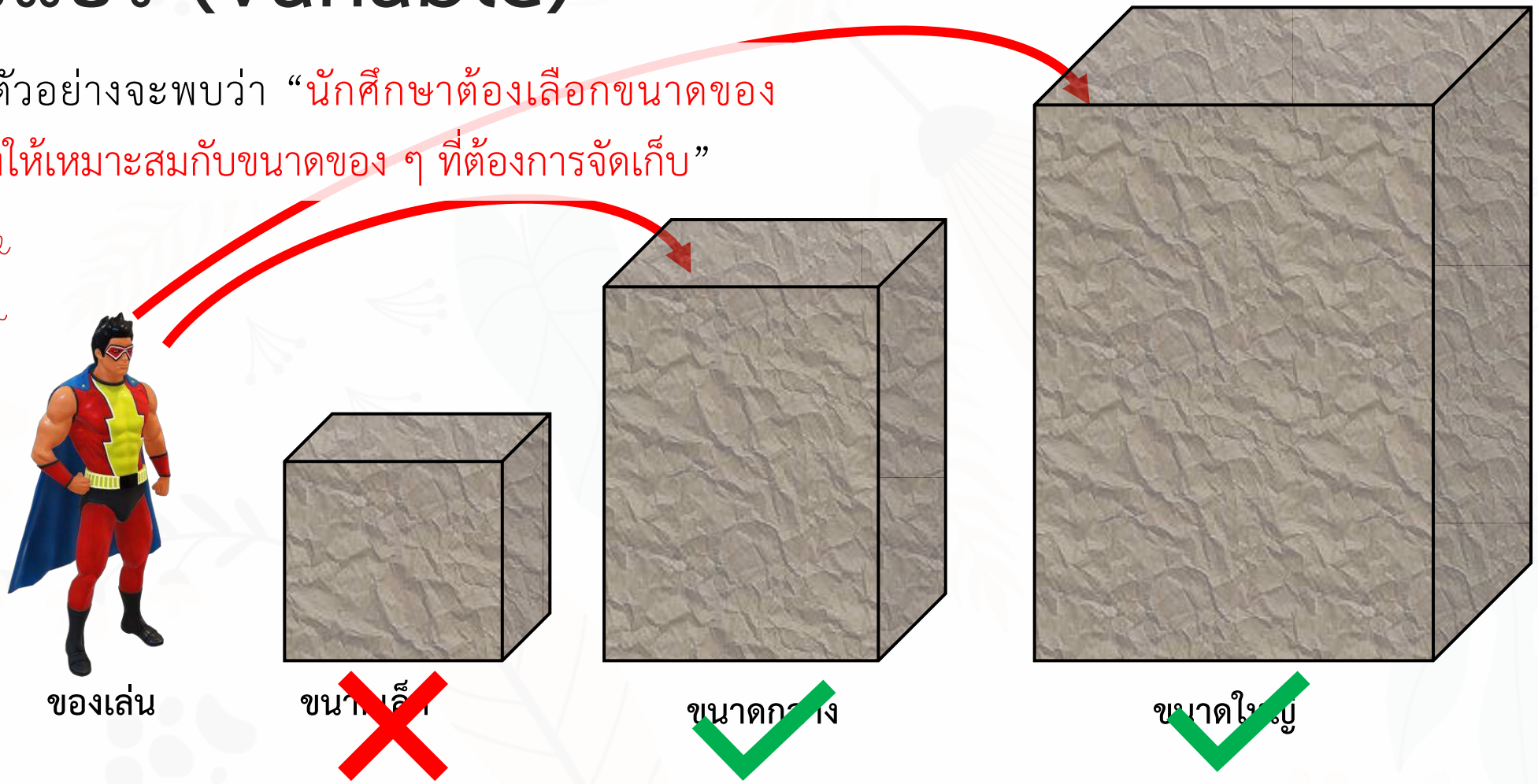
ตัวแปร (Variable)



ตัวแปร (Variable)

จากตัวอย่างจะพบว่า “นักศึกษาต้องเลือกขนาดของกล่องให้เหมาะสมกับขนาดของ ๆ ที่ต้องการจัดเก็บ”

1) เลือก type
2) " size



ตัวแปร (Variable)

ตัวแปร สามารถแบ่งออกเป็นกลุ่ม ได้ 3 แบบ ดังนี้

แบ่งตาม
สิ่งที่เก็บ

- ตัวแปรชนิด ข้อมูลพื้นฐาน
- ตัวแปรชนิด ข้อมูลแบบอ้างอิง

01

แบ่งตาม
การเปลี่ยนแปลง

- ตัวแปรคงที่ (ค่าคงที่)
- ตัวแปร

02

แบ่งตาม
จำนวนค่าที่เก็บ

- ตัวแปรเดียว
- ตัวแปรชุด

03

ตัวแปร (Variable)

ตัวแปร สามารถแบ่งออกเป็นกลุ่ม ได้ 3 แบบ ดังนี้

- แบ่งตามสิ่งที่เก็บ
  - ตัวแปรชนิดข้อมูลพื้นฐาน → Value
  - ตัวแปรชนิดข้อมูลแบบอ้างอิง → address, ref., object

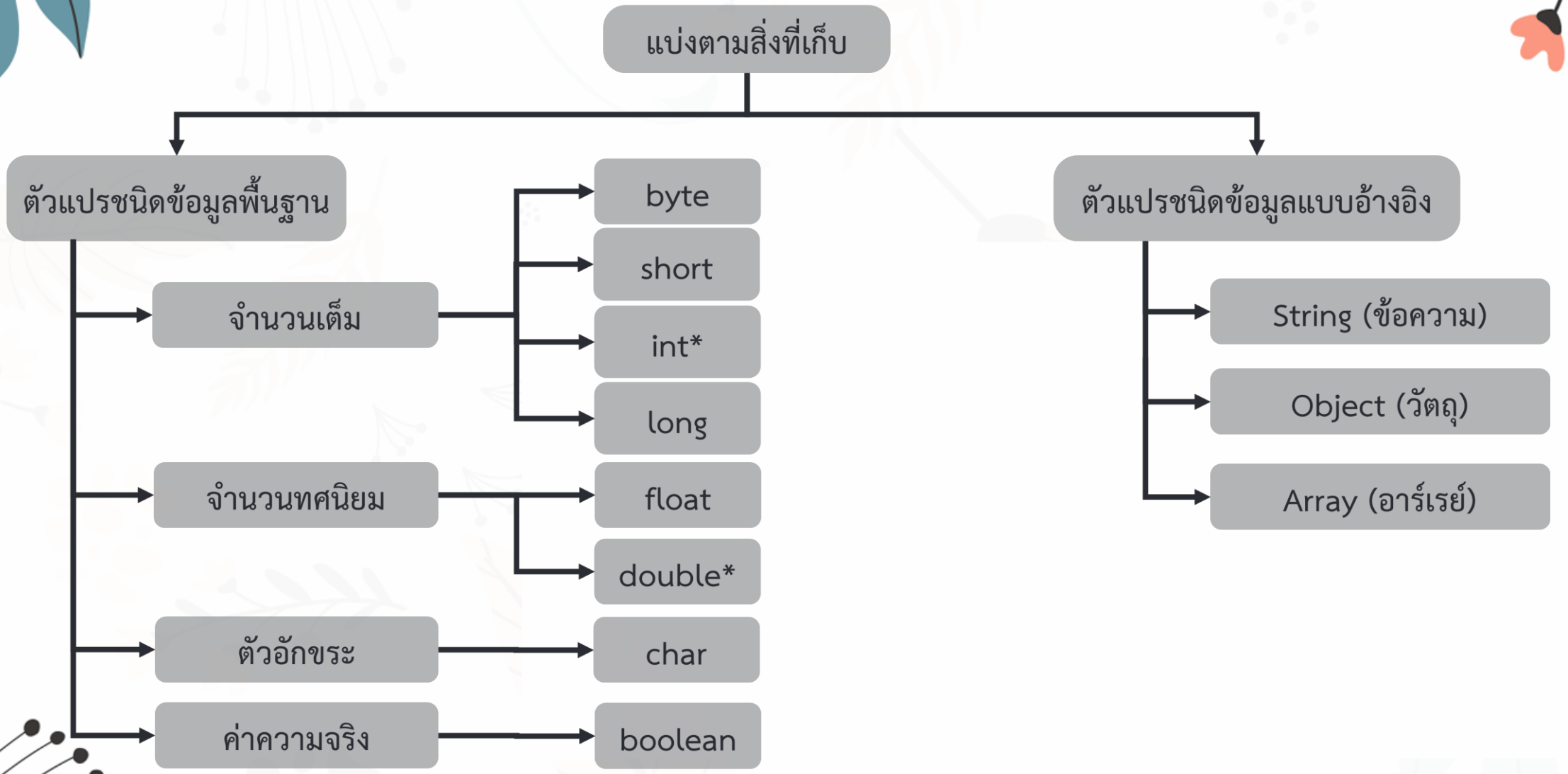
01

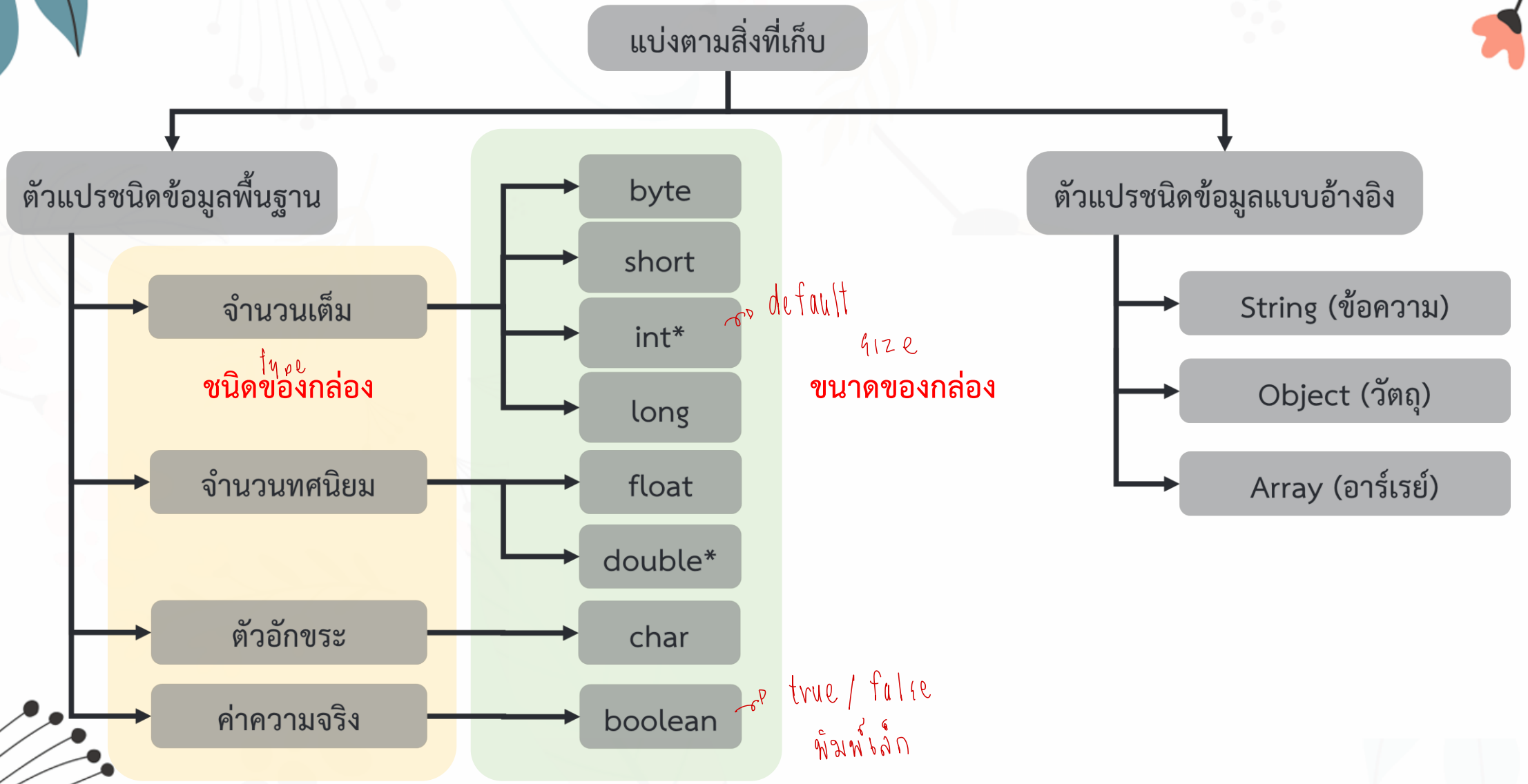
- แบ่งตามการเปลี่ยนแปลง
  - ตัวแปรคงที่ (ค่าคงที่)
  - ตัวแปร

02

- แบ่งตามจำนวนค่าที่เก็บ
  - ตัวแปรเดียว
  - ตัวแปรชุด

03


แบ่งตามสิ่งที่เก็บ
ตัวแปรชนิดข้อมูลพื้นฐาน
จำนวนเต็ม
byte
short
int*
long
จำนวนทศนิยม
float
double*
ตัวอักขระ
char
ค่าความจริง
boolean
ตัวแปรชนิดข้อมูลแบบอ้างอิง
String (ข้อความ)
Object (วัตถุ)
Array (อาร์เรย์)


แบ่งตามสิ่งที่เก็บ
ตัวแปรชนิดข้อมูลพื้นฐาน
ตัวแปรชนิดข้อมูลแบบอ้างอิง
จำนวนเต็ม
จำนวนทศนิยม
ตัวอักขระ
ค่าความจริง
byte
short
int*
long
float
double*
char
boolean
String (ข้อความ)
Object (วัตถุ)
Array (อาร์เรย์)
type
ชนิดของกล่อง
size
ขนาดของกล่อง
default
true / false
พิมพ์เล็ก

ขนาดและช่วงค่าของชนิดข้อมูลแบบพื้นฐาน

ชนิดข้อมูล	ขนาด (Bit/Byte)	ช่วงค่า	ค่าเริ่มต้น
boolean	1 (Not Precisely define)	true หรือ false	false
char	16 / 2	'\u0000' ถึง '\uFFFF' (**16** bit Unicode character)	\u0000
byte	8 / 1	-128 ถึง +127	0
short	16 / 2	-32,768 ถึง +32,767	0
int	32 / 4	-2^{31} ถึง $+2^{31}-1$	0
long	64 / 8	-2^{63} ถึง $+2^{63}-1$	0L
float	32 / 4	-3.40×10^{-38} -3.40E−38 ถึง +3.40E+38 (IEEE 754 floating point standard)	0.0f
double	64 / 8	-1.80E−308 ถึง +1.80E+308	0.0d หรือ 0.0

ตัวแปร (Variable) - ตัวเลขจำนวนเต็ม

ในภาษาจาวานิยมกำหนดค่าเริ่มต้นของข้อมูลชนิดตัวเลขจำนวนเต็มนิยมเป็น int แต่ถ้าต้องเก็บขนาดใหญ่ (มีจำนวนหลักที่มากขึ้น) ก็จะใช้ชนิดข้อมูลเป็น long ซึ่งอาจจะมีตัวอักษร l หรือ L ต่อท้ายหรือไม่ก็ได้ อาทิเช่น

ค่า	ความหมาย
2l	เลขฐานสิบที่มีค่าเป็น 2 ซึ่งเป็นข้อมูลชนิด long
077L	เลขฐานแปดที่เป็นข้อมูลชนิด long
0xBAACL	เลขฐานสิบหกที่เป็นข้อมูลชนิด long

หมายเหตุ ตัวเลขจำนวนเต็มในภาษาจาวาจะถูกมองเป็น int เสมอ

ตัวแปร (Variable) - ตัวเลขทศนิยม

ในภาษาจาวานิยมกำหนดค่าเริ่มต้นของข้อมูลชนิดตัวเลขทศนิยมนิยมเป็น double ซึ่งอาจจะมีตัวอักษร D หรือ d ต่อท้ายหรือไม่ก็ได้ อาทิเช่น

```
2.718F หรือ 3.14f
```

→ ให้มองว่าเป็น float

ขณะที่ ข้อมูลชนิดตัวเลขทศนิยมที่มีชนิดข้อมูลเป็น float จะมีตัวอักษร F หรือ f ต่อท้าย อาทิเช่น

```
2.718D หรือ 2.718
```

หมายเหตุ ตัวเลขทศนิยมในภาษาจาวาจะถูกมองเป็น double เสมอ

ตัวแปร (Variable) - ตรรกะ

ในภาษาจาวาข้อมูลชนิดตรรกะมีอยู่ 2 ค่า คือ true และ false

ตัวแปร (Variable) - ตัวอักขระ

single quote = char
double quote = str

ข้อมูลชนิดตัวอักขระใช้เพื่อแสดงตัวอักขระหนึ่งตัว ซึ่งถูกเก็บอยู่ในรูปของมาตรฐาน Unicode ขนาด 16 บิต โดยจะมีค่าตั้งแต่ ‘\u0000’ ถึง ‘\uFFFF’ ตัวอย่างเช่น

```
char letter = '\u0041';
```

เป็นการประกาศตัวแปรที่ชื่อว่า letter ให้เป็นข้อมูลชนิด char โดยมีค่าเป็น \u0041 ซึ่งมีค่าเท่ากับตัวอักษร A หรืออาจจะเขียนเป็นตัวอักขระโดยตรงเลย เช่น

```
char letter = 'A';
```

มีอักษรตัวเดียว ONLY

เป็นการประกาศตัวแปรที่ชื่อว่า letter ให้เป็นข้อมูลชนิด char โดยมีค่าเป็นตัวอักษร A เช่นเดียวกับคำสั่งก่อนหน้านี้

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	□

https://naveenr.net/content/images/2017/03/ascii-codes.gif

ตัวแปร (Variable) - ตัวอักขระ

ข้อมูลชนิดตัวอักขระพิเศษเพื่อช่วยในการแสดงผล

อักขระ	'\b'	'\t'	'\n'	'\r'	'\\'	'\''	'\"'
Unicode	'\u000B'	'\u0009'	'\u000A'	'\u000D'	'\u005C'	'\u0027'	'\u0022'
ความหมาย	Backspace	Tab	New line	Return	Backslash	Single quote	Double quote

ตัวแปร (Variable)

ตัวแปร สามารถแบ่งออกเป็นกลุ่ม ได้ 3 แบบ ดังนี้

แบ่งตาม
สิ่งที่เก็บ

- ตัวแปรชนิด ข้อมูลพื้นฐาน
- ตัวแปรชนิด ข้อมูลแบบอ้างอิง

01

แบ่งตาม
จำนวนค่าที่เก็บ

- ตัวแปรเดียว
- ตัวแปรชุด

02

แบ่งตาม
การเปลี่ยนแปลง

- ตัวแปรคงที่ (ค่าคงที่)
- ตัวแปร

03

ชนิดข้อมูลแบบอ้างอิง

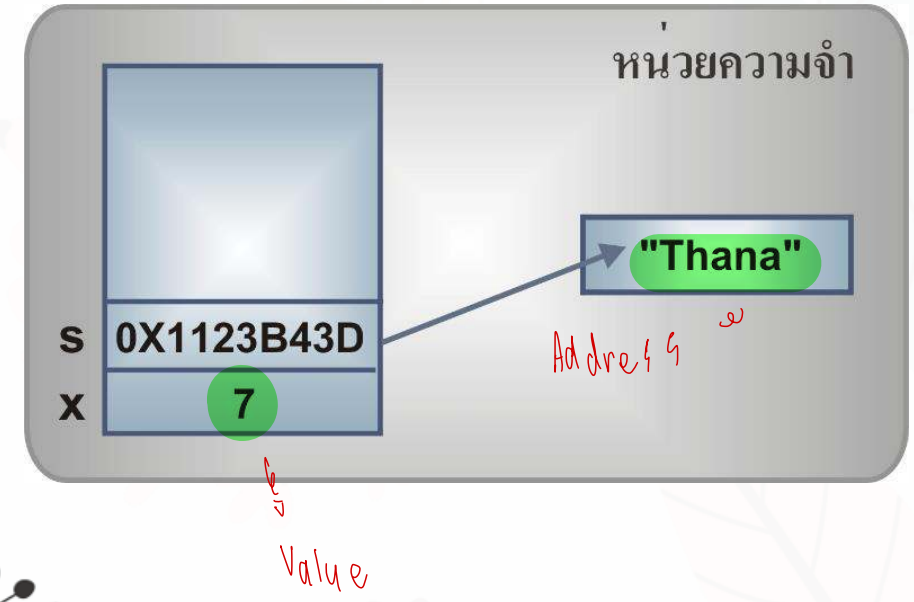
ชนิดข้อมูลแบบอ้างอิง คือ ออปเจ็ค (Object) ในภาษาจาวา โดยแบ่งออกเป็น 2 แบบ ได้แก่

(1) ชนิดข้อมูลที่เป็นคลาส

(2) ชนิดข้อมูลที่เป็นอะเรย์

ชนิดข้อมูลแบบอ้างอิง - ออปเจ็ค (Object)

ซึ่งชนิดข้อมูลแบบอ้างอิงจะมีวิธีการเก็บข้อมูลในหน่วยความจำที่แตกต่างจากการเก็บข้อมูลของชนิดข้อมูลแบบพื้นฐาน ตัวอย่างเช่น

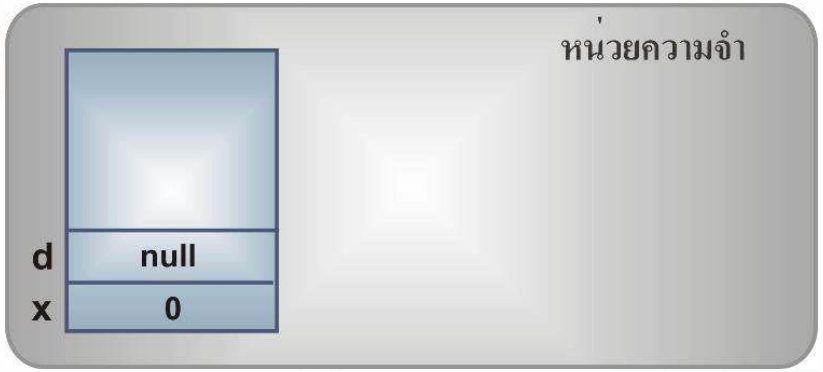



```
int x = 7;
```

```
String s = new String("Thana");
```

ชนิดข้อมูลแบบอ้างอิง - ออปเจ็ค (Object)

01

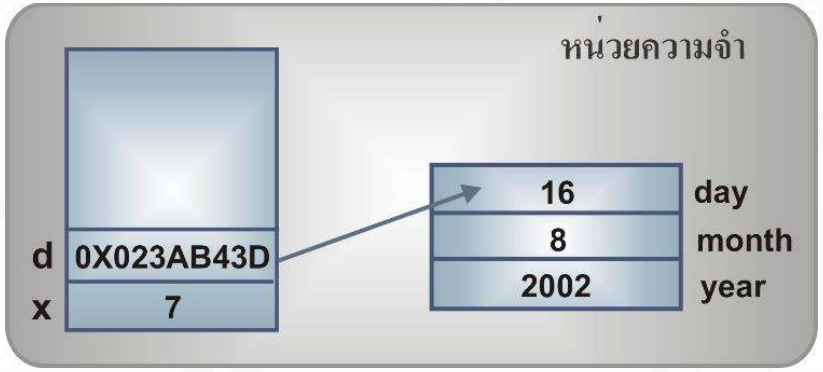


การประกาศตัวแปร

```
int x;
```

```
Date d;
```

02



การกำหนดค่าตัวแปร

```
x = 7;
```

```
D = new Date(16, 8, 2002);
```

ชนิดข้อมูลแบบอ้างอิง - คลาส String

String เป็นชนิดข้อมูลที่เก็บข้อมูลเป็นตัวอักษรหรือข้อความ ซึ่งในภาษาจาวาถือว่าเป็นคลาสหรือออปเจ็คชนิดหนึ่ง ที่มีข้อแตกต่างจากออปเจ็คทั่วไปดังนี้

String เป็นออปเจ็คที่มีค่าคงที่ข้อมูลซึ่งก็คือข้อความใดๆ ที่อยู่ภายในเครื่องหมาย double quote (“ ”)

```
"This is a java course"
```

String Literal คือการสร้างและกำหนดค่า String โดยอาศัย “ ” ตัวอย่างเช่น

1

```
String s = "Taravichet Titijaroonroj";
```

อาศัยคำสั่ง new สำหรับการสร้างและกำหนดค่า String

2

```
String s = new String("Taravichet Titijaroonroj");
```

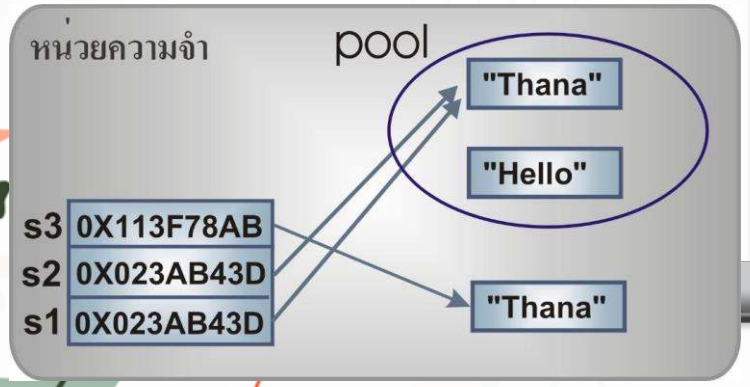
คำอธิบายโปรแกรม

กรณีที่ไม่ใช้คำสั่ง new

ภาษาจาวาจะดูจาก String Pool ว่ามีข้อความเดิมอยู่หรือไม่

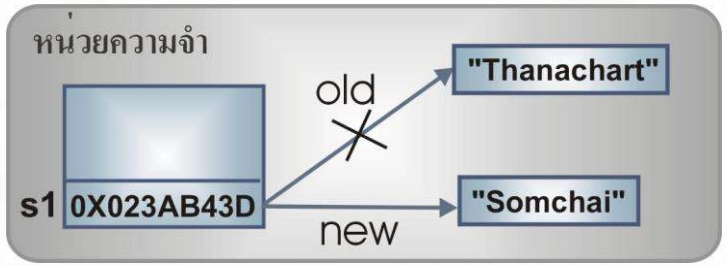
กรณีที่ใช้คำสั่ง new

ภาษาจาวาจะสร้างข้อความใหม่และจองเนื้อที่ในหน่วยความจำเสมอ




```java
public class Main {
    public static void main(String args[]){
        String s1 = "Thana";
        String s2 = "Thana";
        String s3 = new String("Thana");
    }
}
```

คำอธิบายโปรแกรม




```
public class Main {
    public static void main(String args[]){
            String s1;
            s1 = "Thanachart";
            s1 = "Somchai";
    }
}
```

วิธีการประกาศตัวแปร

การประกาศตัวแปรในภาษาจาวาจำเป็นต้องกำหนดชนิดข้อมูล (Data Type) ก่อน จากนั้น เว้นวรรค ก่อนกำหนดชื่อตัวแปร และจบท้ายด้วยสัญลักษณ์ semicolon

ชนิดข้อมูล ชื่อตัวแปร ;

หลักการตั้งชื่อที่ปฏิบัติกันโดยทั่วไปจะต้องเป็นไปตามกฎการตั้งชื่อดังนี้

- ประกอบด้วยด้วยอักขระ 0-9, A-Z, a-z, _ หรือ $ เท่านั้น
- อักขระตัวแรกต้องไม่ใช้ 0-9
- ต้องไม่ตรงกับคีย์เวิร์ด
- ในภาษาจาวาเป็น case sensitive ทำให้ myVariable แตกต่างจาก MyVariable

คีย์เวิร์ดที่ใช้ในภาษาจาวา

abstract	continue	for	new	switch
assert	default	goto	package	synchronized
boolean	do	if	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp	volatile
const	float	native	super	while

ตัวอย่างของการตั้งชื่อในภาษาจาวา

- MyVariable
- _MyVariable
- $x
- This_is_also_a_variable
- isMoney
- stu2

- My Variable
- 9pns
- a+c
- Hello'World
- public

ตัวอย่างของการตั้งชื่อในภาษาจาวา

ประเภท	หลักการ	ตัวอย่าง
Class	• จะขึ้นต้นด้วยตัวอักษรพิมพ์ใหญ่แล้วตามด้วยตัวอักษรพิมพ์เล็กหรือตัวเลข โดยจะใช้ตัวอักษรพิมพ์ใหญ่เฉพาะอักษรนำของแต่ละคำที่ตามมาในชื่อ • ควรเป็นคำนาม	Sample, HelloWorld, MyBigBoss, Student, GraduateStudent เป็นต้น
Variable or Attribute	• จะขึ้นต้นด้วยตัวอักษรพิมพ์เล็ก โดยจะใช้ตัวอักษรพิมพ์ใหญ่เฉพาะอักษรนำของแต่ละคำที่ตามมาในชื่อ • ควรเป็นคำนามหรือเป็นชื่อสั้น ๆ	x, i, name, id, gpa, firstName, myFirstName เป็นต้น
Method	• จะใช้หลักการเดียวกับการตั้งชื่อ Variable or Attribute • ควรเป็นคำกริยา	getName, setName, showDetails, getMyFirstName เป็นต้น
Constant	• จะใช้ตัวอักษรพิมพ์ใหญ่ทั้งหมด และจะแยกคำโดยใช้เครื่องหมาย _ (underscore) • ควรเป็นคำนาม	ID, MIN_GPA เป็นต้น

วิธีการประกาศตัวแปร

ชนิดข้อมูล ชื่อตัวแปร ;

- int
- double
- char
- boolean

- (A-Z), (a-z), (0-9), (_, $)
- ห้ามขึ้นต้นด้วย (0-9)
- num ≠ Num ≠ NUM
- ไม่ซ้ำกับคีย์เวิร์ด

วิธีการประกาศตัวแปร

```
public class Main {
    public static void main(String args[]){
                int age ;
                double weight ;
                char grade ;
    }
}
```

วิธีการกำหนดค่าตัวแปร

การกำหนดค่าตัวแปรในภาษาจาวาจำเป็นต้องอาศัยเครื่องหมายเท่ากับ (=) ดังตัวอย่าง

ชื่อตัวแปร = ค่า ;

โดยที่ฝั่งซ้ายของเครื่องหมายเท่ากับ คือ ตัวแปร และฝั่งขวาคือค่าที่ต้องการจัดเก็บ

```
public class Main {
    public static void main(String args[]){
        int age ;
        double weight ;
        char grade ;

        age = 10 ;
        weight = 120.5 ;
        grade =  'B';

    }
}
```

ประกาศ

กำหนดค่า

วิธีการประกาศและกำหนดค่าตัวแปร

ในภาษาจาวานักศึกษาสามารถประกาศและกำหนดค่าตัวแปรในคำสั่งเดียวกันได้

ชนิดข้อมูล ชื่อตัวแปร = ค่า ;

โดยที่ ตัวแปรจะถูกสร้างก่อน จากนั้น จึงนำค่าที่ต้องการจัดเก็บไปใส่ในตัวแปรที่พึ่งสร้างขึ้น

```
public class Main {
    public static void main(String args[]){
        int age = 10 ;
        double weight = 120.5 ;
        char grade =  'B';
    }
}
```

ประกาศ
+ กำหนดค่า

วิธีการประกาศและกำหนดค่าตัวแปร

```
public class Main {
    public static void main(String args[]){
            int amount;
            double x = 1, y;
            float price, wholeSalePrice;
            double radius = 3.14;
            char c = 'a';
            double y = x+4*3;
            amount = 121+14;
    }
}
```

วิธีการประกาศและกำหนดค่าตัวแปร

```
public class Main {
    public static void main(String args[]){
            int x = 5, y;
            boolean b1;
            float z = 3.414f;
            double w;
            y = 4;
            b1 = (x > y);  -> b1 = true;
            w = x * 3.2;   -> w = 16.0;

            System.out.println("x=" +x+ "y=" +y);
            System.out.println("b1=" + b1);
            System.out.println("z=" +z+ "w=" +w);
    }
}
```

ตัวแปร (Variable)

ตัวแปร สามารถแบ่งออกเป็นกลุ่ม ได้ 3 แบบ ดังนี้

แบ่งตาม
สิ่งที่เก็บ

- ตัวแปรชนิด ข้อมูลพื้นฐาน
- ตัวแปรชนิด ข้อมูลแบบอ้างอิง

01

แบ่งตาม
การเปลี่ยนแปลง

- ตัวแปรคงที่ (ค่าคงที่)
- ตัวแปร

02

แบ่งตาม
จำนวนค่าที่เก็บ

- ตัวแปรเดียว
- ตัวแปรชุด

03

ตัวแปรค่าคงที่

ค่าคงที่ คือ ตัวแปรที่ใช้จัดเก็บและอ้างอิงถึงข้อมูล (Value) ที่ไม่สามารถเปลี่ยนแปลงค่าที่จัดเก็บได้หลังกำหนดค่าเรียบร้อยแล้ว การใช้ค่าคงที่ช่วยให้การเขียนโปรแกรมเป็นเรื่องง่ายและชัดเจน ตัวอย่างเช่น ค่าคงที่แรงโน้มถ่วง (Gravity) เป็น 9.8 การประกาศและกำหนดค่าคงที่ในภาษาจาวาจะมีรูปแบบคล้ายกับการประกาศตัวแปรปกติ แต่จะเพิ่มคำว่า "final" ข้างหน้าเพื่อระบุว่าเป็นค่าคงที่

ค่าจะไม่เปลี่ยนแปลงแล้ว

```
final ชนิดข้อมูล ชื่อตัวแปร = ค่า ;
```

ตัวอย่างเช่น

```
final int num = 10;
```

```
final int num;
num = 10;
```

ตัวอย่างโปรแกรม
สำหรับตัวแปรค่าคงที่

```
public class Main {
    public static void main(String args[]){

        final int MAXIMUM = 10;
        final double MIN_GPA;

        System.out.println("Max is " + MAXIMUM);

        MIN_GPA = 2.00;

        System.out.println("Min GPA is " + MIN_GPA);

        // MIN_GPA = 3.00;   //illegal
    }
}
```

ตัวแปร (Variable)

ตัวแปร สามารถแบ่งออกเป็นกลุ่ม ได้ 3 แบบ ดังนี้

แบ่งตาม
สิ่งที่เก็บ

- ตัวแปรชนิดข้อมูลพื้นฐาน
- ตัวแปรชนิดข้อมูลแบบอ้างอิง

01

แบ่งตาม
การเปลี่ยนแปลง

- ตัวแปรคงที่ (ค่าคงที่)
- ตัวแปร

02

แบ่งตาม
จำนวนค่าที่เก็บ

- ตัวแปรเดียว
- ตัวแปรชุด

03

ตัวแปร (Variable)

ตัวแปร สามารถแบ่งออกเป็นกลุ่ม ได้ 3 แบบ ดังนี้

แบ่งตาม
สิ่งที่เก็บ

- ตัวแปรชนิด ข้อมูลพื้นฐาน
- ตัวแปรชนิด ข้อมูลแบบอ้างอิง

01

แบ่งตาม
การเปลี่ยนแปลง

- ตัวแปรคงที่ (ค่าคงที่)
- ตัวแปร

02

แบ่งตาม
จำนวนค่าที่เก็บ

- ตัวแปรเดียว
- ตัวแปรชุด

03

หัวข้อ

01 การรับค่าผ่านทางคีย์บอร์ดและแสดงผล

02 ตัวแปร (Variable)

03 ตัวดำเนินการ (Operation)

ตัวดำเนินการในภาษาจาวา

- กำหนดค่า (Assignment Operator)

  เครื่องหมายเท่ากับ (=) 1) กำหนดค่า 2) เปรียบเทียบ (compare)

- คอมเมนต์ (Comment Operator)

  //, /* ... */, //* ... */

- คณิตศาสตร์ (Arithmetic Operator)

  +, -, *, /, %

- การคำนวณและการกำหนดค่า (Compound Operator)

  +=, -=, *=, /=, %=

- การเพิ่มลดค่า (Increment & Decrement Operator)

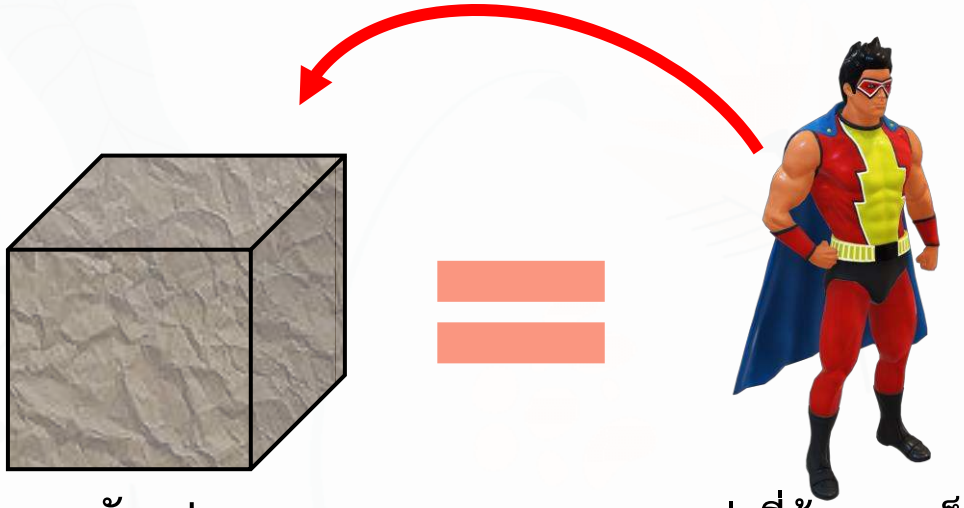
  ++, --

- เชื่อมข้อความ (Concatenate Operator)

  เครื่องหมายบวก (+)

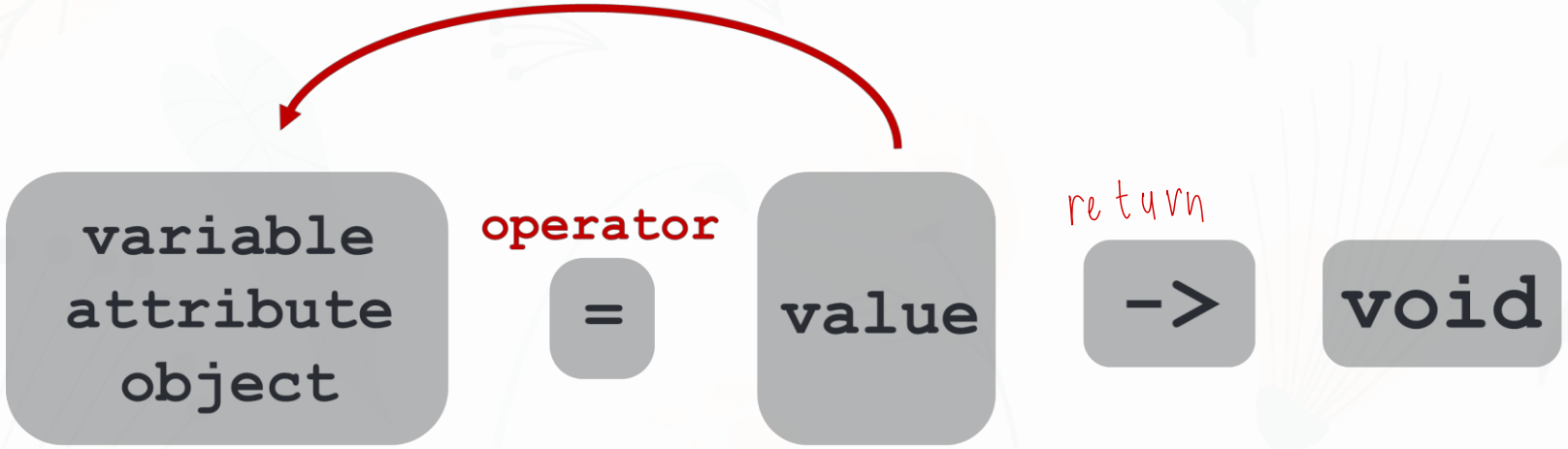
กำหนดค่า (Assignment Operator)

ในภาษาจาวาเครื่องหมายเท่ากับ (=) ใช้สำหรับการกำหนดค่า โดยจะนำค่าจากทางฝั่งขวาของเครื่องหมายเท่ากับไปเก็บในตัวแปรฝั่งซ้าย



ซึ่งนิพจน์ของเครื่องหมายเท่ากับ (=) หลังทำงานเสร็จจะเป็น void โดยที่ นิพจน์ (Statement) คือ หน่วยของคำสั่งในไวยากรณ์ของภาษาโปรแกรม และ void เป็นตัวบ่งบอกว่า Method หรือ นิพจน์นั้น ไม่มีค่าที่จะส่งกลับ (Return)

กำหนดค่า (Assignment Operator)



คอมเมนต์ (Comment Operator)

คอมเมนต์เขียนไว้เพื่อ (1) อธิบายโปรแกรม (2) ให้ผู้อ่านเข้าใจโปรแกรมง่ายยิ่งขึ้น (3) ช่วยทำให้การแก้ไขและปรับปรุงโปรแกรมเป็นไปได้ง่ายขึ้น ในภาษาจาวากำหนดรูปแบบของการเขียนคอมเม็นต์ไว้สามรูปแบบดังนี้

- คอมเม็นต์สำหรับข้อความบรรทัดเดียว

```
// This is a comment
```

- คอมเม็นต์สำหรับข้อความตั้งแต่หนึ่งบรรทัดขึ้นไป

```
/* This is also a comment */
```

- คอมเม็นต์สำหรับข้อความที่ต้องการสร้างเป็นไฟล์เอกสารที่เป็นไฟล์ประเภท HTML

```
/** This is a comment for documentation */
```

ตัวอย่างโปรแกรมสำหรับคอมเมนต์

```
public class Main {
    public static void main(String args[]){

        int a = 10; // b = 20, c = 5;
        /*
        System.out.println(10 + 20 + 30);
        System.out.println(a + 20 + c);
        System.out.println(a + b + "30")
        System.out.println("a" + b + c);
        */

        System.out.println("I'm Bank");
    }
}
```

ตัวอย่างโปรแกรมสำหรับคอมเมนต์

```
/*
This program is to show
how to write comments
*/
public class Main {

        // Main method
        public static void main(String args[]) {

        /** This is a comment for documentation */
                System.out.println("Document");

        }
}
```

คณิตศาสตร์ (Arithmetic Operator)

ตัวดำเนินการทางคณิตศาสตร์ในจาวาใช้สำหรับการดำเนินการทางคณิตศาสตร์ทั่วไป ซึ่งประกอบด้วย +, -, *, /, % ตัวอย่างเช่น

เครื่องหมายทางคณิตศาสตร์	เครื่องหมายทางคอมพิวเตอร์	ความหมาย	ตัวอย่างการใช้งาน
+	+	บวก	a+b
-	−	ลบ	a−b
×	*	คูณ	a*b
÷	/	หาร	a/b
	%	เศษจากการหาร	a%b

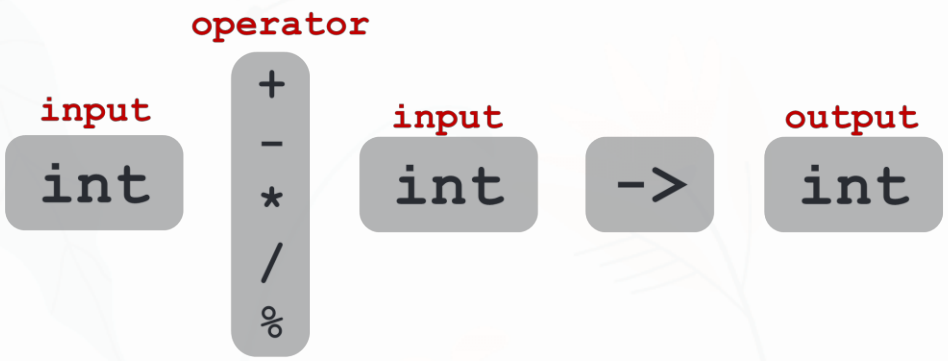
คณิตศาสตร์ (Arithmetic Operator)

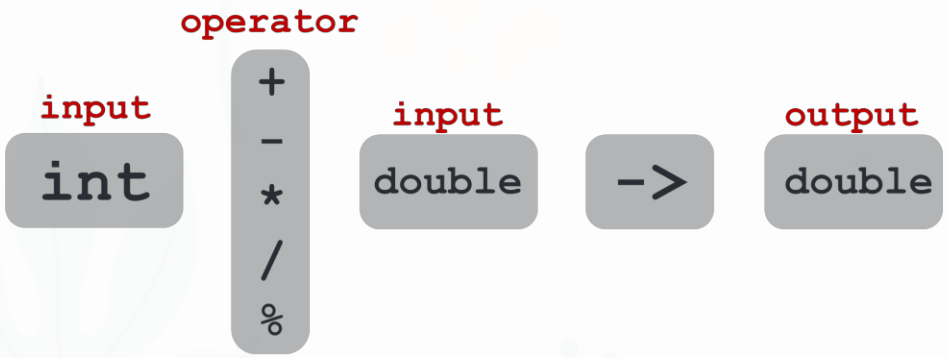
ตัวดำเนินการทางคณิตศาสตร์ในจาวาใช้สำหรับการดำเนินการทางคณิตศาสตร์ทั่วไป ซึ่งประกอบด้วย

Operand	Operator	Operand	Return	Output
Input	Action	Input		Output
จำนวนเต็ม	+	จำนวนเต็ม		จำนวนเต็ม
ทศนิยม	-	ทศนิยม		ทศนิยม
ทศนิยม	*	จำนวนเต็ม		ทศนิยม
	/			
	%			

คณิตศาสตร์ (Arithmetic Operator)

ตัวดำเนินการทางคณิตศาสตร์ในจาวาใช้สำหรับการดำเนินการทางคณิตศาสตร์ทั่วไป ซึ่งประกอบด้วย





```java
public class Main {
   public static void main(String args[]) {
        long longData = 100;
        int intData  = 5;
        double doubleData = 2;
        float floatData = 4f;

        System.out.println("long + long ="      + ((Object)(longData+longData)).getClass());
        System.out.println("long + int ="       + ((Object)(longData+intData)).getClass());
        System.out.println("long + double ="    + ((Object)(longData+doubleData)).getClass());
        System.out.println("long + float ="   + ((Object)(longData+floatData)).getClass());

        System.out.println("int + long ="       + ((Object)(intData+longData)).getClass());
        System.out.println("int + int ="        + ((Object)(intData+intData)).getClass());
        System.out.println("int + double ="     + ((Object)(intData+doubleData)).getClass());
        System.out.println("int + float ="    + ((Object)(intData+floatData)).getClass());

        System.out.println("double + long ="    + ((Object)(doubleData+longData)).getClass());
        System.out.println("double + int ="     + ((Object)(doubleData+intData)).getClass());
        System.out.println("double + double ="  + ((Object)(doubleData+doubleData)).getClass());
        System.out.println("double + float =" + ((Object)(doubleData+floatData)).getClass());

        System.out.println("float + long = "    + ((Object)(floatData+longData)).getClass());
        System.out.println("float + int = "     + ((Object)(floatData+intData)).getClass());
        System.out.println("float + double = "  + ((Object)(floatData+doubleData)).getClass());
        System.out.println("float + float = " + ((Object)(floatData+floatData)).getClass());
    }
}
```

จากตัวอย่างสามารถสรุปได้ 2 ประเด่น ดังนี้

ประเด่นที่ 1 ชนิดข้อมูล

จำนวนเต็ม	+	จำนวนเต็ม	=	จำนวนเต็ม
ทศนิยม	+	ทศนิยม	=	ทศนิยม
ทศนิยม	+	จำนวนเต็ม	=	ทศนิยม

ประเด่นที่ 2 ขนาด

small	+	small	=	small
large	+	large	=	large
large	+	small	=	large

```
long + long     = class java.lang.Long
long + int      = class java.lang.Long
long + double   = class java.lang.Double
long + float    = class java.lang.Float

int + long      = class java.lang.Long
int + int       = class java.lang.Integer
int + double    = class java.lang.Double
int + float     = class java.lang.Float

double + long   = class java.lang.Double
double + int    = class java.lang.Double
double + double= class java.lang.Double
double + float = class java.lang.Double

float + long    = class java.lang.Float
float + int     = class java.lang.Float
float + double = class java.lang.Double
float + float   = class java.lang.Float
```

ตัวอย่างโปรแกรมสำหรับตัวดำเนินการทางคณิตศาสตร์

```
public class Main {
   public static void main(String args[]) {
        int a = 10 + 7;
        System.out.println("line 1: "+a);

        a = 10 - 7;
        System.out.println("line 2: "+a);

        a = 10 * 7;
        System.out.println("line 3: "+a);

        a = 10 / 2;
        System.out.println("line 4: "+a);
    }
}
```

การคำนวณและกำหนดค่า (Compound Operator)

ตัวดำเนินการที่ผสมการทำงานของตัวดำเนินการทางคณิตศาสตร์และการกำหนดค่าเข้าด้วยกัน และเขียนคำสั่งในรูปแบบย่อ

Operand	Arithmetic Operator	Assignment Operator	Operand	Return →	Void
Variable	Action1	Action2	Input		Output

โดยมีเงื่อนไข ดังต่อไปนี้

- ฝั่งซ้ายของตัวดำเนินการต้องเป็นตัวแปรที่ต้องการแก้ไขค่าเสมอ
- ฝั่งขวาของตัวดำเนินการเป็นค่าที่ต้องการนำไปคำนวณ ซึ่งอาจจะอยู่ในรูปแบบ ค่า นิพจน์ ตัวแปร เมธอด ก็ได้
- ต้องเขียนตัวดำเนินการคณิตศาสตร์และตามด้วยเครื่องหมายเท่ากับทันที
- ลำดับการทำงานจะเริ่มจากตัวดำเนินการคณิตศาสตร์และตามด้วยเครื่องหมายตามลำดับ
- นิพจน์ของตัวดำเนินการชนิดนี้จะคืนค่าเป็น void

การคำนวณและกำหนดค่า (Compound Operator)

```
a = a + 3;
a = a - 3;
a = a * 3;
a = a / 3;
a = a % 3;
```

```
a += 3;
a -= 3;
a *= 3;
a /= 3;
a %= 3;
```

ตัวอย่างโปรแกรมสำหรับการคำนวณและกำหนดค่า

```
public class Main {
   public static void main(String args[]) {
        int a = 10, b = 5;
        System.out.println("line 1: "+a);

        a += 4;
        System.out.println("line 2: "+a);

        a += (4+3);
        System.out.println("line 3: "+a);

        a += (b-3*2);  ต่อไปไม่มีวงเล็บก็นับว่ามี
        System.out.println("line 4: "+a);
    }
}
```

การเพิ่มลดค่า (Increment & Decrement Operator)

ตัวดำเนินการเพิ่มลดค่าทีละ 1 ให้กับตัวแปรที่กำหนด หรือกล่าวได้คือ การทำงานของตัวดำเนินการทางคณิตศาสตร์ด้วยเครื่องหมายบวกหรือเครื่องหมายลบทีปรับค่าละ 1 ก่อนนำผลลัพธ์ที่ได้ไปจัดเก็บในตัวแปร และเขียนคำสั่งในรูปแบบย่อ

```
a = a + 1;
a = a - 1;
```

→

```
a += 1;
a -= 1;
```

→

ใช้ก่อนแล้ว +1 +1 แล้วค่อยใช้

```
a++ or ++a;
a-- or --a;
```

การใช้งานเครื่องหมาย ++ หรือ - - มีด้วยกัน 2 รูปแบบ ดังนี้

- เขียนขึ้นต้นด้วยตัวแปรแล้วจึงตามด้วยเครื่องหมาย ++ หรือ - - หมายความว่า ให้นำค่าที่เก็บไว้ในตัวแปรไปใช้งานก่อน แล้วจึงดำเนินการเพิ่มหรือลดค่าในตัวแปรหนึ่ง
- เขียนขึ้นต้นด้วยเครื่องหมาย ++ หรือ - - แล้วจึงตามด้วยตัวแปร หมายความว่า ให้นำค่าที่เก็บไว้ในตัวแปรไปดำเนินการเพิ่มหรือลดค่าในตัวแปรหนึ่ง แล้วจึงนำไปใช้งาน

นิพจน์ของตัวดำเนินการชนิดนี้จะคืนค่าเป็น void

ตัวอย่างโปรแกรม

สำหรับการเพิ่มลดค่าที่ 1

```
public class Main {
   public static void main(String args[]) {
        int a = 10;
        System.out.println("line 1: "+a);
        System.out.println("line 2: "+ (a++));
        System.out.println("line 3: "+a);

        int b = 10;
        System.out.println("line 1: "+b);
        System.out.println("line 2: "+ (++b));
        System.out.println("line 3: "+b);

    }
}
```

ตัวอย่างโปรแกรม

สำหรับการเพิ่มลดค่าที่ 2

```
public class Main {
   public static void main(String args[]) {

        int a = 10;
        System.out.println("line 1: "+a);
        System.out.println("line 2: "+ (--a));
        System.out.println("line 3: "+b);

        int b = 10;
        System.out.println("line 1: "+b);
        System.out.println("line 2: "+ (b--));
        System.out.println("line 3: "+b);

    }
}
```

ตัวอย่างโปรแกรม

สำหรับการเพิ่มลดค่าที่ 3

```
public class Main {
   public static void main(String args[]) {
        int a = 10;
        System.out.println("line 1: "+a);

        a += 4*3;
        System.out.println("line 2: "+a);

        a += ++a;   -> a = a + (++a)
        System.out.println("line 3: "+a);

        a += ++a*2;   -> a = a + (++a * 2)
        System.out.println("line 4: "+a);
    }
}
```

ตัวอย่างโปรแกรม

สำหรับการเพิ่มลดค่าที่ 4

```
public class Main {
   public static void main(String args[]) {
        int x, y;
        x = 5;
        y = x++;

        System.out.println("x = "+x+" y = "+y);
        y = ++x;
        System.out.println("x = "+x+" y = "+y);
    }
}
```

ลำดับความสำคัญของตัวดำเนินการ

ตัวดำเนินการเพิ่มลดค่าทีละ 1 ให้กับตัวแปรที่กำหนด หรือกล่าวได้คือ การทำงานของตัวดำเนินการทางคณิตศาสตร์ด้วยเครื่องหมายบวกหรือเครื่องหมายลบทีปรับค่าละ 1 ก่อนนำผลลัพธ์ที่ได้ไปจัดเก็บในตัวแปร และเขียนคำสั่งในรูปแบบย่อ

ลำดับ	เรียงจาก	ตัวดำเนินการ
1	ขวาไปซ้าย (R to L)	3 ++, --, 2 +, -, ~, !, ( ) 1
2	ซ้ายไปขวา (L to R)	4 *, /, %
3	ซ้ายไปขวา (L to R)	5 +, -
4	ขวาไปซ้าย (R to L)	6 =, +=, -=, *=, /=, %=, <<=, >>=, >>>=, &=, ^=, |=

ลำดับความสำคัญของตัวดำเนินการ

ให้นักศึกษาคำนวณหาผลลัพธ์ตามลำดับดังนี้

```
x = 2+3*4-(7+2);
```

$= 2+3\times4-9 = 2+12-9 = 5$ #

ขั้นที่ 1 คำนวณหาผลลัพธ์ค่า 7+2 ทำให้ได้

```
x = 2+3*4-9
```

ขั้นที่ 2 คำนวณหาผลลัพธ์ค่า 3*4 ทำให้ได้

```
x = 2+12-9
```

ขั้นที่ 3 คำนวณหาผลลัพธ์ค่า 2+12 ทำให้ได้

```
x = 14-9
```

ขั้นที่ 4 คำนวณหาผลลัพธ์ค่า 14-9 ทำให้ได้

```
x = 5
```

ตัวดำเนินการที่ใช้กับข้อมูลชนิด String

String เป็นออปเจ็คที่มีตัวดำเนินการที่ใช้ในการเชื่อมข้อความสองข้อความเข้าด้วยกัน โดยใช้เครื่องหมาย + อาทิเช่น

```
String s1 = “Hello” + “ World”;
```

ตัวดำเนินการในการเชื่อมข้อความสามารถใช้เชื่อมข้อมูลชนิด String กับตัวถูกดำเนินการที่เป็นชนิดข้อมูลอื่น ๆ ภาษาจาวาจะแปลงชนิดข้อมูลดังกล่าวให้เป็นชนิด String โดยอัตโนมัติอาทิเช่น คำสั่ง

```
String s1 = “This";
String s2 = s1 + “ is a test ”;
String s3 = s1+4;
```

นิพจน์ของตัวดำเนินการชนิดนี้จะคืนค่าเป็น String และรับค่าเป็น String

มี 2 ความหมาย
1) การ คนต
2) การเอา str มาต่อกัน

ตัวดำเนินการที่ใช้กับข้อมูลชนิด String

2 ฝั่งเป็นตัวเลข = sum

1 ใน = ต่อ 1 ใน

input		input		output
int	+	String	->	String
	operator			

input		input		output
int	+	String	->	String
	operator			

input		input		output
double	+	String	->	String
	operator			

ตัวอย่างโปรแกรมสำหรับตัวดำเนินการที่ใช้กับข้อมูลชนิด String

```
public class Main {
   public static void main(String args[]) {

        int a = 10, b = 20, c = 5;

        System.out.println(10 + 20 + 30);
        System.out.println(a + 20 + c);

        System.out.println(a + b + “30”)
        System.out.println(“a” + b + c);

        System.out.println(a + “20” + c);
        System.out.println(“a” + b + c);

        System.out.println(“I’m ” + (b + c));
        System.out.println(“I’m ” + b + c);
    }
}
```

- **ตัวดำเนินการที่มีการคืนค่า**

ตัวดำเนินการ	ประกอบด้วย
คณิตศาสตร์	+, -, *, /, %
การเปรียบเทียบ	< , > , <= , >= , == , !=
ตรรกศาสตร์	!, &&, & , ||, |
เชื่อมข้อความ	+

- **ตัวดำเนินการที่ไม่มีการคืนค่า**

ตัวดำเนินการ	ประกอบด้วย
กำหนดค่า	=
คอมเมนต์	//, /* ... */, /**... */
การคำนวณและการกำหนดค่า	+=, -=, *=, /=, %=
การเพิ่มลดค่า	++, --

ตัวดำเนินการในภาษาจาวา

- กำหนดค่า (Assignment Operator)

  เครื่องหมายเท่ากับ (=)

- คอมเมนต์ (Comment Operator)

  //, /* ... */, //* ... */

- คณิตศาสตร์ (Arithmetic Operator)

  +, -, *, /, %

- การคำนวณและการกำหนดค่า (Compound Operator)

  +=, -=, *=, /=, %=

- การเพิ่มลดค่า (Increment & Decrement Operator)

  ++, --

- เชื่อมข้อความ (Concatenate Operator)

  เครื่องหมายบวก (+)

ความแตกต่างระหว่าง void และ null

คำสั่ง void และ null เป็นคำที่เกี่ยวข้องกับการ ไม่มีค่า (No Value) แต่มีวัตถุประสงค์ที่แตกต่างกันและใช้ในบริบทที่แตกต่างกัน

คำสั่ง	ความหมาย	การใช้งาน	การเปรียบเทียบ
void	คือ "void" ถูกใช้เฉพาะในบริบทของประเภทการส่งคืนเมธอดหรือนิพจน์ จะมีความหมายว่า "จะไม่ส่งคืนค่าใด ๆ” กลับคืน	ใช้งานได้กับ • เมธอด • นิพจน์	ไม่สามารถเปรียบเทียบกับสิ่งใดได้ เนื่องจากจะไม่ส่งคืนค่า
null	เป็นค่า ไม่ใช่ Data Type มันแสดงถึงการไม่มีค่าสำหรับ Data Type ประเภทอ้างอิง จะมีความหมายว่า "เป็นค่าเริ่มต้นสำหรับการอ้างอิงวัตถุ บ่งชี้ว่าข้อมูลอ้างอิงไม่ได้ชี้ไปที่วัตถุใดๆ ในหน่วยความจำ”	ใช้งานกับ • การอ้างอิงวัตถุ	สามารถเปรียบเทียบการอ้างอิงวัตถุกับ "null" เพื่อดูว่าชี้ไปที่วัตถุใดๆ หรือไม่

หัวข้อ

01 การรับค่าผ่านทางคีย์บอร์ดและแสดงผล

02 ตัวแปร (Variable)

03 ตัวดำเนินการ (Operation)

Thanks !