

CHAPTER 02

การดำเนินการและประมวลผลข้อมูล (Data Operations and Processing)

1. ประเภทข้อมูลและตัวแปร (Data Types and Variables)

1. ประเภทข้อมูลหรือชนิดตัวแปร (Data Types)

1.1 ประเภทข้อมูลแบบดั้งเดิม (Primitive Data Types) แบ่งออกเป็น 4 กลุ่ม ได้แก่

1) **ตัวเลขจำนวนเต็ม (Integer)** → คือ จำนวนเต็มบวก จำนวนเต็มลบ จำนวนเต็มศูนย์

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
byte	8 bits	-128 ถึง 127	0
short	16 bits	-32,768 ถึง 32,767	0
int	32 bits	-2,147,483,648 ถึง 2,147,483,647	0
long	64 bits	-9,223,372,036,854,775,808 ถึง 9,223,372,036,854,775,807	0L

2) **ตัวเลขจำนวนจริง (Floating Point / Real Number)** คือ จำนวนที่มีจุดทศนิยม

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
float	32 bits	-3.40292347E+38 ถึง 3.40292347E+38	0.0F
double	64 bits	-1.79769313486231570E+308 ถึง 1.79769313486231570E+308	0.0D

3) **ตัวอักขระ (Character)** → คือ ตัวอักษร (Letter) หรือตัวเลข (Digit) หรือสัญลักษณ์ (Symbol) เพียง 1 ตัว

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
char	16 bits	'\u0000' ถึง '\uFFFF' (0 - 65535) เช่น 'A' หรือ 'a' (คล่อมด้วยเครื่องหมาย '...')	'\u0000'

4) **ตัวตรรกะ (Boolean)** → คือ ค่าความจริง ซึ่งมีเพียงสองค่าคือ จริง (True) หรือ เท็จ (False)

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
boolean	ไม่ระบุ	true หรือ false	false

1.2 ประเภทข้อมูลแบบอ้างอิง (Reference Data Types) แบ่งออกเป็น 1 กลุ่ม (ยังมีมากกว่านี้) ได้แก่

1) **สตริง (String) หรือข้อความ** → คือ กลุ่มของอักขระหรือสายอักขระตั้งแต่ 0 ตัวขึ้นไปมาเรียงต่อกัน

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
String	ไม่ระบุ	เช่น "Java Chula" หรือ "1" หรือ "F50" หรือ "" (Empty String) (คล่อมด้วยเครื่องหมาย "...")	null

1.3 การเรียงลำดับขนาดและศักดิ์ศรีของประเภทข้อมูล (เฉพาะตัวเลข) มีดังนี้

byte	short	int	long	float	double
เล็ก			ใหญ่		

2. การตั้งชื่อตัวแปร

1) กฎการตั้งชื่อตัวแปร (บังคับและต้องทำตาม)

- (1) ชื่อตัวแปรสามารถประกอบไปด้วย ตัวอักษร ตัวเลข สัญลักษณ์ '\$' (Dollar Sign) และสัญลักษณ์ '_' (Underscore) เช่น num, \$_\$, _hey, \$1, xxx3, test1_\$ เป็นต้น
- (2) ชื่อตัวแปรห้ามขึ้นต้นชื่อด้วยตัวเลข เช่น 3girl1man, 2you, 5hundredMile เป็นต้น
- (3) ชื่อตัวแปรห้ามมีช่องว่างและสัญลักษณ์พิเศษอื่นๆ นอกเหนือจากที่กล่าวไว้ในข้อที่ 1
- (4) ชื่อตัวแปรห้ามตั้งชื่อซ้ำกับคำสงวน (Reserved Words / Keywords) ซึ่งมี 53 คำดังต่อไปนี้

abstract	assert	boolean	break	byte	case
catch	char	class	const *	continue	default
do	double	else	enum	extends	false
final	finally	float	for	goto *	if
implements	import	instanceof	int	interface	long
native	new	null	package	private	protected
public	return	short	static	strictfp	super
switch	synchronized	this	throw	throws	transient
true	try	void	volatile	while	
- (5) ชื่อตัวแปรห้ามตั้งชื่อซ้ำกันภายในเมทอดเดียวกัน (แต่ต่างเมทอดกันอาจตั้งชื่อซ้ำกันได้)
- (6) ชื่อตัวแปรมีความยาวได้ไม่จำกัด (จะยาวกี่กิโลเมตรก็ได้ แต่ควรให้มีความยาวเหมาะสม)
- (7) ตัวอักษรพิมพ์ใหญ่และพิมพ์เล็กมีความหมายต่างกันเสมอ (Case-Sensitive) เช่น var ต่างกับ Var

2) ธรรมเนียมการตั้งชื่อตัวแปร (ไม่บังคับแต่ควรทำตาม)

- (1) ชื่อตัวแปร (และชื่อเมทอด) นิยมขึ้นต้นด้วยอักษรพิมพ์เล็ก / ชื่อคลาสนิยมขึ้นต้นด้วยอักษรพิมพ์ใหญ่
- (2) ชื่อตัวแปรนิยมใช้เป็นภาษาอังกฤษ (แต่อาจใช้เป็นภาษาใดก็ได้ ขึ้นกับการรองรับของ Editor ที่เขียน)
- (3) ชื่อตัวแปรควรตั้งชื่อให้สื่อความหมายกับค่าของตัวแปรนั้น เช่น stdID, firstName, birthDate เป็นต้น
- (4) ชื่อตัวแปรถ้าประกอบด้วยคำหลายคำให้เขียนคำแรกขึ้นต้นด้วยอักษรพิมพ์เล็ก ส่วนคำถัดๆ ไปขึ้นต้นด้วยอักษรพิมพ์ใหญ่ เช่น hostName, numberOfCourse, currentTime, firstDayOfMonth เป็นต้น
- (5) ชื่อตัวแปรถ้าเป็นค่าคงที่นิยมตั้งชื่อด้วยอักษรพิมพ์ใหญ่ทั้งหมด เช่น PI, MONTH_OF_YEAR เป็นต้น
- (6) พยายามหลีกเลี่ยงการใช้สัญลักษณ์ '\$' และ สัญลักษณ์ '_'

3. การประกาศและกำหนดค่าตัวแปร

1) การประกาศตัวแปร

```
<ประเภทข้อมูล> <ชื่อตัวแปร>;
```

ตัวอย่าง int x; double num; String lastName; boolean check;

2) การกำหนดค่าตัวแปร

อาจเป็นค่าคงที่ (Constant) หรือค่าจากตัวแปรอื่นก็ได้

<ชื่อตัวแปรที่ประกาศไว้แล้ว> = <ค่าของตัวแปร>;

ซ้ายต้องใหญ่หรือพอกับขวา

ตัวอย่าง `x = 5; num = 4.01; lastName = "Rukchula"; check = true; a = b;`

3) การประกาศพร้อมกับการกำหนดค่าตัวแปร (รวมวิธีที่ 1 และ 2 เข้าด้วยกัน)

ซ้ายตั้งขวายัด

<ประเภทข้อมูล> <ชื่อตัวแปร> = <ค่าของตัวแปร>;

ตัวอย่าง `int x = n; double num = 4.01; char ch = 'A'; float y = 2.9f;`

4) การประกาศและกำหนดค่าตัวแปรหลายตัวที่มีประเภทข้อมูลเดียวกัน

คั่นด้วย Comma

<ประเภทข้อมูล> <ชื่อตัวแปร>, <ชื่อตัวแปร> = <ค่าของตัวแปร>, ...;

ตัวอย่าง `int x = n, year, age = 18, z, a;`

4. ค่าของตัวแปรตามประเภทข้อมูล

1) ค่าของจำนวนเต็ม

ประเภทข้อมูล	รูปแบบการเขียน / ตัวอย่างค่าที่จัดเก็บ
byte และ short	เขียนเลขจำนวนเต็มทั่วไป เช่น <code>byte x = 12; short y = 199;</code>
int (ฐานสิบ)	เขียนเลขจำนวนเต็มทั่วไป เช่น <code>int dec = 26;</code>
int (ฐานแปด)	เขียน 0 (ศูนย์) นำหน้าเลขจำนวนเต็ม เช่น <code>int oct = 032;</code>
int (ฐานสิบหก)	เขียน 0x หรือ 0X นำหน้าเลขจำนวนเต็ม เช่น <code>int hex = 0x1A;</code>
long	เขียนเลขจำนวนเต็มแล้วมีอักษร l หรือ L ต่อท้าย เช่น <code>long n = 26L;</code>

2) ค่าของจำนวนจริง

ประเภทข้อมูล	รูปแบบการเขียน / ตัวอย่างค่าที่จัดเก็บ
float	เขียนเลขจำนวนจริงแล้วมีอักษร f หรือ F ต่อท้าย เช่น <code>float f = 1.3F;</code>
double (รูปทั่วไป)	เขียนเลขจำนวนจริงทั่วไป เช่น <code>double d = 1.3;</code>
double (รูปเต็ม)	เขียนเลขจำนวนจริงแล้วมีอักษร d หรือ D ต่อท้าย เช่น <code>double d = 1.3D;</code>
double (รูป $a \times 10^b$)	เขียนค่า a ในรูปของเลขจำนวนเต็มหรือเลขจำนวนจริงทั่วไป ตามด้วยอักษร e หรือ E และตามด้วยค่า b ในรูปของเลขจำนวนเต็ม (เท่านั้น) เช่น <ul style="list-style-type: none"> <code>double s = 93.478e8;</code> มีค่าเท่ากับ 93.478×10^8 <code>double t = 631E-5;</code> มีค่าเท่ากับ 631×10^{-5} <code>double u = 2.0e+200;</code> มีค่าเท่ากับ 2.0×10^{200}

3) ค่าของตัวอักขระและสตริง

ประเภทข้อมูล	รูปแบบการเขียน / ตัวอย่างค่าที่จัดเก็บ
char (รูปทั่วไป)	เขียนอักขระทั่วไปคล่อมด้วย Single Quote เช่น <code>char c = 'a';</code>

ประเภทข้อมูล	รูปแบบการเขียน / ตัวอย่างค่าที่จัดเก็บ
char (รูป Unicode)	เขียนรหัส Unicode คล่อมด้วย Single Quote เช่น char c = '\u0043'; <ul style="list-style-type: none"> '\u0030' หรือ รหัส 48 (ฐานสิบ) → มีค่าเท่ากับอักขระ '0' '\u0041' หรือ รหัส 65 (ฐานสิบ) → มีค่าเท่ากับอักขระ 'A' '\u0061' หรือ รหัส 97 (ฐานสิบ) → มีค่าเท่ากับอักขระ 'a' '\u00E0' หรือ รหัส 3585 (ฐานสิบ) → มีค่าเท่ากับอักขระ 'ณ'
char (รูปพิเศษ)	เขียนรหัสพิเศษคล่อมด้วย Single Quote เช่น char tab = '\t';
String (รูปทั่วไป)	เขียนสายอักขระคล่อมด้วย Double Quote เช่น String s = "Bank";
String (รูปอ็อบเจ็ค)	สร้างอ็อบเจ็คจากคลาส String เช่น String t = new String("CU");

โจทย์ข้อที่ 1 [ระดับง่าย] จงพิจารณาคำสั่งต่อไปนี้ผิด (☒) หรือถูก (☑) พร้อมบอกเหตุผล (15 คะแนน)

- ☐ integer x;
- ☐ int x = 1.5;
- ☐ byte x = 1000;
- ☐ int null = 5;
- ☐ Int y = 1;
- ☐ float cal = 15.763;
- ☐ String id = "1";
- ☐ char c = 'Tuksin';
- ☐ String s1 = 'Apisit';
- ☐ String 3Type = "One Two Three";
- ☐ short _\$abcdefghijklmnopqrstuvwxyz0123456789\$ _;
- ☐ boolean flag = 1;
- ☐ float ตัวเลข = 139L;
- ☐ String _ = "ㄆㄇㄉㄅㄆㄇㄉㄅㄆㄇㄉㄅ";
- ☐ int ch = '\u0044';

โจทย์ข้อที่ 2 [ระดับปานกลาง] จงเติมเต็มคำสั่งการประกาศและกำหนดค่าของตัวแปรต่อไปนี้ให้สมบูรณ์ พร้อมทั้งระบุค่าที่เก็บอยู่ในตัวแปรแต่ละข้อ (15 คะแนน)

ข้อ	ประเภทตัวแปร	การประกาศและกำหนดค่าตัวแปร	ค่าที่เก็บในตัวแปร
1.		var1 = 014;	
2.		var2 = 014f;	
3.	int	var3 = 'C';	

ข้อ	ประเภทตัวแปร	การประกาศและกำหนดค่าตัวแปร	ค่าที่เก็บในตัวแปร
4.		var4 = 0E-0;	
5.	double	var5 = 0x1D;	
6.		var6 = 1.0e2.0;	
7.		var7 = .1000;	
8.		var8 = 08;	
9.		var9 = null;	
10.		var10 = 100E012;	
11.		var11 = 1f + "";	
12.		var12 = 1e1 + "";	
13.		var13 = 0d;	
14.		var14 = 3.f;	
15.		var15 = 0xbeef;	

2. การรับเข้าและส่งออกข้อมูล (Input and Output the Data)

1. การรับเข้าข้อมูล (Input the Data)

การรับเข้าข้อมูลในภาษาจาวามีอยู่หลายวิธีด้วยกัน แต่ในเอกสารชุดนี้จะนำเสนอการรับเข้าข้อมูลที่นิยมใช้กันมากในปัจจุบัน นั่นคือการรับเข้าข้อมูลด้วยคลาส Scanner ซึ่งมีขั้นตอนการใช้งานดังต่อไปนี้

- 1) การนำเข้าคลาส Scanner ก่อนเรียกใช้งานคลาส Scanner จะต้องนำเข้าคลาส Scanner ก่อนเสมอ โดยใช้คำสั่ง `import java.util.Scanner;` ซึ่งจะระบุไว้ก่อนเขียนหัวคลาส ดังตัวอย่าง

```

1 import java.util.Scanner;
2 public class Test {
3     ...
4 }
```

- 2) การสร้างตัวอ่านจากแป้นพิมพ์ เป็นการกำหนดชื่อตัวอ่านเพื่อใช้สำหรับรับค่าข้อมูลต่างๆ จากแป้นพิมพ์ โดยจะสร้างเพียงครั้งเดียวเท่านั้น และจะใช้งานตัวอ่านนั้นตลอดทั้งโปรแกรม ซึ่งมีคำสั่งดังนี้

```
Scanner <ชื่อตัวอ่าน> = new Scanner(System.in);
```

```

1 import java.util.Scanner;
2 public class Test {
3     public static void main(String[] args) {
4         Scanner kb = new Scanner(System.in);
5         ...
6     }
7 }
```

System.in หมายถึง
แป้นพิมพ์ (keyboard)

ตัวอ่านชื่อว่า kb

- 3) การรับค่าข้อมูลด้วยตัวอ่านจากแป้นพิมพ์ ตัวอ่านจากแป้นพิมพ์ที่สร้างขึ้นสามารถรับค่าข้อมูลประเภทจำนวนเต็ม จำนวนจริง และสตริง โดยใช้คำสั่งและเมทอดต่อไปนี้

- 1) คำสั่งรับค่าจำนวนเต็มประเภท int โดยใช้เมทอด nextInt()

```
int <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.nextInt();
```

หลังชื่อเมทอด nextInt() ต้องตามด้วยวงเล็บเสมอ จึงได้เป็น nextInt() (ห้ามลืมเด็ดขาด)

เช่น int num = kb.nextInt(); (รับค่า int จากตัวอ่าน kb มาเก็บไว้ที่ตัวแปร num)

- 2) คำสั่งรับค่าจำนวนเต็มประเภท long โดยใช้เมทอด nextLong()

```
long <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.nextLong();
```

เช่น long n = kb.nextLong(); (รับค่า long จากตัวอ่าน kb มาเก็บไว้ที่ตัวแปร n)

- 3) คำสั่งรับค่าจำนวนจริงประเภท float โดยใช้เมทอด nextFloat()

```
float <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.nextFloat();
```

เช่น float f = kb.nextFloat(); (รับค่า float จากตัวอ่าน kb มาเก็บไว้ที่ตัวแปร f)

- 4) คำสั่งรับค่าจำนวนจริงประเภท double โดยใช้เมทอด nextDouble()

```
double <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.nextDouble();
```

เช่น double d = kb.nextDouble(); (รับค่า double จากตัวอ่าน kb มาเก็บไว้ที่ตัวแปร d)

- 5) คำสั่งรับค่าข้อมูลประเภท String ที่ละบรรทัด โดยใช้เมทอด nextLine()

```
String <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.nextLine();
```

เช่น String s1 = kb.nextLine(); (รับค่า String ทั้งบรรทัด จากตัวอ่าน kb มาเก็บไว้ที่ตัวแปร s1)

- 6) คำสั่งรับค่าข้อมูลประเภท String ที่ละคำหรือที่ละช่อง (แต่ละช่องคั่นด้วยช่องว่างหรือ \t หรือ \n) โดยใช้เมทอด next()

```
String <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.next();
```

เช่น String s2 = kb.next(); (รับค่า String คำแรก จากตัวอ่าน kb มาเก็บไว้ที่ตัวแปร s2)

โจทย์ข้อที่ 3 [ระดับง่าย] จงเติมเต็มคำสั่งการรับค่าของตัวแปรในแต่ละข้อต่อไปนี้ให้สมบูรณ์ พร้อมทั้งระบุค่าที่เก็บอยู่ในตัวแปรแต่ละข้อ (15 คะแนน)

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

ข้อ	ประเภทตัวแปร	การรับค่าจากแป้นพิมพ์เพื่อเก็บยังตัวแปรที่กำหนด	ค่าที่ป้อนเข้ามาทางแป้นพิมพ์	ค่าที่เก็บในตัวแปร
1.		var1 = kb.nextInt();	087	
2.		var2 = kb.nextLong();	-0	

ข้อ	ประเภทตัวแปร	การรับค่าจากแป้นพิมพ์เพื่อเก็บยังตัวแปรที่กำหนด	ค่าที่ป้อนเข้ามาทางแป้นพิมพ์	ค่าที่เก็บในตัวแปร
3.	int		7.0	
4.		var4 = kb.nextFloat();	22.50000	
5.	long		46L	
6.	float		7.2F	
7.	double		-0.1	
8.	double		-13	
9.	double	var10 = kb.nextInt();	1	
10.	double	var11 = kb.nextInt();	1.0	
11.	int	var12 = kb.nextDouble();	1.0	
12.		var13 = kb.nextLine();	22.50000	
13.		var14 = kb.nextLine();	Hi java CU	
14.		var15 = kb.next();	Hi java CU	
15.	String	var16 = kb.nextInt();	191	

2. การส่งออกข้อมูล (Output the Data)

การส่งออกข้อมูลหรือการแสดงผลข้อมูลสามารถทำได้หลายช่องทาง แต่ช่องทางที่พื้นฐานที่สุดนั้นก็คือจอภาพ (Monitor/Screen) โดยรายละเอียดของคำสั่งเบื้องต้นได้กล่าวไปแล้วในบทที่ 1 ซึ่งมีคำสั่งดังนี้

- 1) การแสดงผลออกทางจอภาพแบบไม่ขึ้นบรรทัดใหม่ โดยใช้เมทอด print()

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

System.out หมายถึงจอภาพ

- 2) การแสดงผลออกทางจอภาพแบบขึ้นบรรทัดใหม่ โดยใช้เมทอด println()

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

สิ่งที่สามารถระบุและแสดงผลภายในวงเล็บของเมทอดทั้งสองได้นั้นมีดังนี้

- 1) ตัวแปร เช่น System.out.print(num);
- 2) ตัวเลข เช่น System.out.print(13);
- 3) นิพจน์ เช่น System.out.print(x * y + (10 - z));
- 4) ข้อความหรือสตริง เช่น System.out.print("Hello Java");
- 5) ข้อ 1-4 ปะปนกัน เช่น System.out.print("Money = " + (x * y + 92.4) + " Baht");

ใช้เครื่องหมาย + เป็นตัวเชื่อม
แต่ละช่วงเข้าด้วยกัน

3. รูปแบบการเขียนโปรแกรมรับเข้าและส่งออกข้อมูลที่ต้องปฏิบัติตาม มีรายละเอียดดังตัวอย่างต่อไปนี้

```

1 import java.util.Scanner;           // ต้องนำเข้าคลาส Scanner
2 public class Test {
3     public static void main(String[] args) {
4         Scanner kb = new Scanner(System.in); // ต้องสร้างตัวอ่าน
5         System.out.print("Please enter number: ");
6         int num = kb.nextInt();
7
8         ...                          // หลังชื่อเมทอดอย่าลืมใส่วงเล็บ
9
10        System.out.println("Result is " + num);
11    }
12 }

```

ต้องแสดงข้อความว่าต้องการรับเข้าข้อมูลอะไร โดยใช้เมทอด print()

ต้องแสดงผลลัพธ์ โดยใช้เมทอด println()

โปรแกรมจะแสดงผลลัพธ์ออกทางจอภาพดังนี้

Please enter number:

โปรแกรมจะแสดงข้อความเพื่อแจ้งให้ผู้ใช้ทราบว่า
ระบบกำลังรอรับค่าตัวเลขหนึ่งค่าจากแป้นพิมพ์

```
Please enter number: 15
```

ผู้ใช้พิมพ์ตัวเลข 15 ผ่านทางแป้นพิมพ์พร้อมทั้งกดปุ่ม Enter เพื่อขึ้นบรรทัดใหม่ ซึ่งทำให้ค่าของตัวเลข 15 เข้าไปเก็บอยู่ในตัวแปร num

```
Please enter number: 15
Result is 15
```

โปรแกรมแสดงค่าที่เก็บอยู่ในตัวแปร num
ดังนี้ เปรียบภาพ

โจทย์ข้อที่ 4 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับเลขประจำตัวนิสิต ชื่อ-นามสกุล นิสิต อายุ นิสิต และผลการเรียนเฉลี่ยสะสมของนิสิต เข้ามาทางแป้นพิมพ์เก็บไว้ในตัวแปร id, name, age และ gpa ตามลำดับ และแสดงค่าของทั้งสี่ตัวแปรนั้นออกทางจอภาพให้สวยงาม (10 คะแนน)

[illegible]

โจทย์ข้อที่ 5 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็ม สองจำนวนเข้ามาทางแป้นพิมพ์เก็บไว้ในตัวแปร a และ b ตามลำดับ แล้วทำการสลับค่าของสองตัวแปรนั้น โดยที่ค่าของตัวแปร a หลังการสลับจะเท่ากับค่าเก่าของตัวแปร b และค่าของตัวแปร b หลังการสลับจะเก็บค่าเก่าของตัวแปร a พร้อมทั้งแสดงผลลัพธ์ออกทางจอภาพตามตัวอย่างต่อไปนี้ (10 คะแนน)

[illegible]

3. ตัวดำเนินการและคลาสมาตรฐาน (Operators and Standard Class)

1. ตัวดำเนินการ (Operators) แบ่งออกเป็น 5 กลุ่มหลักดังต่อไปนี้
 - 1) ตัวดำเนินการทางคณิตศาสตร์ (Arithmetic Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
+	การบวก	$x + y$ หรือ $a + b + c$
-	การลบ	$x - y$ หรือ $a - b - c$
*	การคูณ	$x * y$ หรือ $a * b * c$
/	การหาร	x / y หรือ $a / b / c$
%	การหารเอาแต่เศษ (Modulo)	$x \% y$ หรือ $a \% b \% c$

- 2) ตัวดำเนินการเพิ่มค่าและลดค่า (Increment and Decrement Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
x++	ใช้งานตัวแปร x ก่อนเพิ่มค่าขึ้นอีก 1	ให้ $x = 2, y = 1$ จะได้ว่า $x++ + y = 3$
++x	เพิ่มค่าขึ้นอีก 1 ก่อนใช้งานตัวแปร x	ให้ $x = 2, y = 1$ จะได้ว่า $++x + y = 4$
x--	ใช้งานตัวแปร x ก่อนลดค่าลงอีก 1	ให้ $x = 2, y = 1$ จะได้ว่า $x-- + y = 3$
--x	ลดค่าลงอีก 1 ก่อนใช้งานตัวแปร x	ให้ $x = 2, y = 1$ จะได้ว่า $--x + y = 2$

3) ตัวดำเนินการเปรียบเทียบค่า (Comparison/Relational Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
==	เท่ากัน	$x == y$ (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)
!=	ไม่เท่ากัน	$x != y$ (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)
<	น้อยกว่า	$x < y$ (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)
>	มากกว่า	$x > y$ (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)
<=	น้อยกว่าหรือเท่ากัน	$x <= y$ (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)
>=	มากกว่าหรือเท่ากัน	$x >= y$ (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)

4) ตัวดำเนินการทางตรรกศาสตร์ (Logical Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
&&	และ (AND)	$(x >= y) \ \&\& \ (x != 0)$
	หรือ (OR)	$(x == y) \ \ (x == z)$
!	นิเสธ (NOT)	$!(x <= y)$

5) ตัวดำเนินการกำหนดค่า (Assignment Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
=	เท่ากับ	$x = y$ นำค่า y เก็บไว้ใน x
+=	$x += y$ มีค่าเท่ากับ $x = x + y$	$x += 14$ จะได้ว่า $x = x + 14$
-=	$x -= y$ มีค่าเท่ากับ $x = x - y$	$x -= 1$ จะได้ว่า $x = x - 1$
*=	$x *= y$ มีค่าเท่ากับ $x = x * y$	$x *= y--$ จะได้ว่า $x = x * y--$
/=	$x /= y$ มีค่าเท่ากับ $x = x / y$	$x /= 5.8$ จะได้ว่า $x = x / 5.8$

โจทย์ข้อที่ 6 [ระดับง่าย] จงแสดงผลลัพธ์จากส่วนของโปรแกรมต่อไปนี้ (5 คะแนน)

ข้อ	ส่วนของโปรแกรม	ผลลัพธ์ที่แสดง
1.	<code>int a = 21, b = 0; System.out.println("a" + a - b);</code>	
2.	<code>double x = 1.0, y = 1.000; System.out.println(x > y);</code>	
3.	<code>double a = 1.6; a += 5.1; System.out.println(a % 4);</code>	
4.	<code>int x = 6; System.out.println(x-- + ++x - --x - x-- + x++ + x);</code>	
5.	<code>boolean a = true, b = true, c = false; System.out.println(a b && c);</code>	

2. กฎการพิจารณาลำดับตัวดำเนินการ (เฉพาะที่ใช้กันบ่อยๆ)

ลำดับ	ชื่อตัวดำเนินการ / ตัวดำเนินการ	ตัวอย่างการทำงาน
1.	วงเล็บ (Parentheses)	(...)
2.	เรียกเมทอด (Method Call)	()
		$x - (2 + y)$ จะทำ $(2 + y)$ ก่อนนำ x มาลบออก
		$\text{Math.sqrt}(a + 2)$ จะทำ $\text{Math.sqrt}(\dots)$ ก่อน

ลำดับ	ชื่อตัวดำเนินการ / ตัวดำเนินการ		ตัวอย่างการทำงาน
3.	ค่าบวก (Unary Plus)	+	2 * +a จะทำ +a ก่อนนำ 2 ไปคูณกับคำตอบ
	ค่าลบ (Unary Minus)	-	-a / 3 จะทำ -a ก่อนนำคำตอบไปหาร 3
4.	คูณ (Multiplication)	*	x * 3 + 2 จะทำ x * 3 ก่อนนำคำตอบไปบวก 2
	หาร (Division)	/	3 - 5 / x จะทำ 5 / x ก่อนนำ 3 มาลบออก
	หารเอาแต่เศษ (Modulo)	%	y % 2 + 1 จะทำ y % 2 ก่อนนำคำตอบไปบวก 1
5.	บวก (Addition)	+	x + 17 > 19 จะทำ x + 17 ก่อนทำ >
	ลบ (Subtraction)	-	x - 9.5 = 42.5 จะทำ x - 9.5 ก่อนทำ =
6.	การเปรียบเทียบค่า : น้อยกว่า (<), น้อยกว่าเท่ากับ (<=), มากกว่า (>), มากกว่าเท่ากับ (>=)		
7.	การเปรียบเทียบค่า : เท่ากัน (==), ไม่เท่ากัน (!=)		
8.	และ (Logical AND)	&&	a b && c จะทำ && ก่อน
9.	หรือ (Logical OR)		x = d > 3 d <= 15 จะทำ ก่อนทำ =
10.	เท่ากับ (Assignment)	=	จะถูกพิจารณาหลังสุด

หมายเหตุ ในกรณีที่ตัวดำเนินการมีลำดับความสำคัญเท่ากัน ให้พิจารณาการทำงานจากซ้ายไปขวา

โจทย์ข้อที่ 7 [ระดับง่าย] จงแสดงลำดับการทำงานที่ถูกต้องของตัวดำเนินการของนิพจน์ต่อไปนี้ โดยให้เขียนหมายเลขลำดับกำกับไว้บนตัวดำเนินการนั้น (5 คะแนน)

- 1) $a * (b + -(c / d) / e) + (f - g \% h)$
- 2) $(1.0 / 2.0) * \text{Math.sin}(x - \text{Math.PI} / \text{Math.sqrt}(y))$
- 3) $a * b + (-c) / d / e * f - g \% h$
- 4) $\text{cal} = (1 - x \% 2.5 + y * z) + 3 * k - -7 / w$
- 5) $\text{cal} = x == 5 \ || \ x <= 3 \ \&\& \ x >= -3 \ || \ (x != 0)$

3. ประเภทข้อมูลที่เป็นผลลัพธ์สุดท้ายจากการดำเนินการ

สามารถสรุปความสัมพันธ์ของประเภทข้อมูล ตัวเลข ตัวอักขระ ตัวตรรกะ และ สตริง ได้ 20 รูปแบบ ดังนี้

รูปแบบ	ข้อมูลที่ 1	กลุ่มของตัวดำเนินการ	ข้อมูลที่ 2	ประเภทข้อมูลสุดท้าย
1.	ตัวเลข	+ - * / %	ตัวเลข	ตัวเลข (ที่มีศักดิ์ศรีสูงสุด)
2.	ตัวเลข	= += -= *= /=	ตัวเลข ^[1]	ตัวเลข (ของข้อมูลที่ 1)
3.	ตัวเลข	== != < > <= >=	ตัวเลข	boolean
4.	ตัวเลข	+ - * / % = += -= *= /=	char	ตัวเลข

รูปแบบ	ข้อมูลที่ 1	กลุ่มของตัวดำเนินการ	ข้อมูลที่ 2	ประเภทข้อมูลสุดท้าย
5.	ตัวเลข	++ --	-	ตัวเลข
6.	-	++ --	ตัวเลข	ตัวเลข
7.	char	+ - * / %	char	ตัวเลข (โดยทั่วไปคือ int)
8.	char	= += -= *= /=	char	char
9.	char	== != < > <= >=	char	boolean
10.	char	+ - * / %	ตัวเลข	ตัวเลข
11.	char	+= -= *= /=	ตัวเลข	char
12.	char	=	ตัวเลข ^[2]	char
13.	char	++ --	-	char
14.	-	++ --	char	char
15.	String	+	ทุกประเภท	String
16.	ทุกประเภท	+	String	String
17.	String	=	String	String
18.	String	== !=	String	boolean
19.	boolean	= && 	boolean	boolean
20.	-	!	boolean	boolean (ค่าตรงข้าม)

^[1] ประเภทข้อมูลของข้อมูลที่ 1 ต้องมีศักดิ์ศรีสูงกว่าหรือเท่ากับประเภทข้อมูลของข้อมูลที่ 2

^[2] ประเภทข้อมูลของข้อมูลที่ 2 ต้องเป็นตัวเลขจำนวนเต็มที่มีศักดิ์ศรีต่ำกว่าหรือเท่ากับ int เท่านั้น

โจทย์ข้อที่ 8 [ระดับง่าย] จงระบุประเภทข้อมูลที่เป็นผลลัพธ์สุดท้ายและค่าคำตอบจากการดำเนินการของนิพจน์ต่อไปนี้ (10 คะแนน)

ข้อ	นิพจน์	ประเภทข้อมูลสุดท้าย	คำตอบ
1.	(5 + 4) + ((3 + 2L) - 1)		
2.	(5 - 2 / 1) + 0.000 + '\u0000'		
3.	5.0F * 6 - ('D' - 'A')		
*4.	2 % (0.4 + 0.3F)		
*5.	!("Java" == "Jaba");		
6.	(13 / 5) + (2 / 3)		
7.	'0' - '1' + "5" + false		
8.	4E1 / 2		
9.	!false true && false !true		
10.	0.0 / (10 + 'x' - 'Y') != 0L		

4. การเปลี่ยนประเภทข้อมูล (Type Conversions)

- 1) **เปลี่ยนแบบอัตโนมัติ** โดยทั่วไปคอมพิวเตอร์ของภาษาจาวาจะปรับเปลี่ยนประเภทข้อมูลให้อัตโนมัติตามความเหมาะสม ซึ่งเป็นไปตามความสัมพันธ์ทั้ง 19 รูปแบบที่ได้กล่าวไปในหัวข้อที่แล้ว ดังตัวอย่างต่อไปนี้

```
double a = 34 + 1.0F; //a = 35.0
float b = 4L / 3; //b = 1.0F
```

```
int c = 'A'; //c = 65
char d = 65; //d = 'A'
```

- 2) **เปลี่ยนโดยการบังคับตามที่ระบุ (Casting)** จะเป็นการบังคับหรือกำหนดประเภทข้อมูลที่ต้องการจะให้ เป็น โดยการใส่วงเล็บชื่อประเภทข้อมูลหน้าข้อมูลที่จะเปลี่ยน ดังรูปแบบคำสั่งและตัวอย่างต่อไปนี้

(<ประเภทข้อมูล>) <ข้อมูล/ตัวแปร>

ใช้กับข้อมูลประเภทตัวเลข (เต็ม/จริง) และอักขระ เท่านั้น

```
byte b = 50;
b = (byte)(b * 2); //b = 100
```

```
char c = '1';
int num = (int) c; //num = 49
```

```
double x = 5.99;
long y = (long) x; //y = 5
```

```
int ten = (int) "10"; //Error
boolean f = (boolean) 1; //Error
```

โจทย์ข้อที่ 9 [ระดับง่าย] จงระบุประเภทข้อมูลที่เป็นผลลัพธ์สุดท้ายและค่าคำตอบจากการดำเนินการของนิพจน์หรือคำสั่งต่อไปนี้ (10 คะแนน)

ข้อ	นิพจน์	ประเภทข้อมูลสุดท้าย	คำตอบ
1.	(byte)(3 + 5.4) / 7		
2.	(3 + 5.4) / (float) 7		
3.	(float)((3 + 2) / 7)		
4.	char c = (double) 'A';		
5.	char c = (int) 'b'; c += 1.7;		
6.	char c = 67 + (int) 3.91		
7.	(int) 19.8E-1		
8.	(short)((double) 1 / 2) == 00L + 0		
9.	(byte) 011 + (short) 1.9991		
10.	(byte) '0' - '6' + (char) 1.5		

5. คลาสมาตรฐานทางคณิตศาสตร์ (Mathematic Class) มีเมทอดที่สำคัญดังนี้

ลำดับ	รูปแบบคำสั่ง	ประเภทข้อมูลรับเข้า	หน้าที่
1.	Math.toRadians(degree)	double	เปลี่ยนองศาเป็นเรเดียน
2.	Math.toDegrees(radian)	double	เปลี่ยนเรเดียนเป็นองศา
3.	Math.sin(radian)	double	คำนวณค่าไซน์ (Sine)
4.	Math.cos(radian)	double	คำนวณค่าโคไซน์ (Cosine)

ลำดับ	รูปแบบคำสั่ง	ประเภทข้อมูลรับเข้า	หน้าที่
5.	Math.sqrt(x)	double	คำนวณค่ารากที่สอง
6.	Math.pow(x, y)	double, double	คำนวณค่าเลขยกกำลัง x^y
7.	Math.abs(x)	double	คำนวณค่าสัมบูรณ์ (Absolute)
8.	Math.max(x, y)	double, double	คำนวณค่ามากกว่าระหว่าง x กับ y
9.	Math.min(x, y)	double, double	คำนวณค่าน้อยกว่าระหว่าง x กับ y
10.	Math.log10(x)	double	คำนวณค่าลอการิทึมฐานสิบ
11.	Math.PI	ค่าคงที่ PI ซึ่งเท่ากับ 3.141592653589793...	
12.	Math.random()	ไม่มีข้อมูลรับเข้า	สร้างตัวเลขสุ่มในช่วง [0.0, 1.0)
	(int)(Math.random() * (b - a + 1) + a) สร้างตัวเลขสุ่มจำนวนเต็มในช่วง a ถึง b		

หมายเหตุ ผลลัพธ์ที่ได้ทุกข้อจะเป็นประเภท double (ยกเว้นข้อที่ 12 ส่วนล่าง ผลลัพธ์จะเป็นประเภท int)

โจทย์ข้อที่ 10 [ระดับง่าย] จงเขียนคำสั่งคำนวณ (Assignment Statement) จากนิพจน์ต่อไปนี้ให้ถูกต้องตามหลักไวยากรณ์ของภาษาจาวา พร้อมทั้งประกาศประเภทตัวแปรที่ใช้ให้ครบสมบูรณ์ (6 คะแนน)

ข้อ	นิพจน์	คำสั่ง
1.	$\frac{2\pi r^2}{\sqrt{b^2 - 4ac}}$	
2.	$\frac{ m - n }{2mn} - \frac{\sqrt{p^2 - q^2}}{p + q}$	
3.	$\frac{3x^3 + 4y^4}{(x + y)(x - y)} + \pi xy$	

โจทย์ข้อที่ 11 [ระดับง่าย] จงเขียนคำสั่งภาษาจาวาที่ถูกต้องสมบูรณ์ เพื่อกำหนดค่าให้กับตัวแปร n โดยใช้การสุ่มค่าตัวเลขจำนวนเต็มตามเงื่อนไขต่อไปนี้ (6 คะแนน)

ข้อ	เงื่อนไข	คำสั่ง
1.	สุ่มตัวเลข 0 – 9	
2.	สุ่มตัวเลข 1 – 10	
3.	สุ่มตัวเลข 0 – 10	
4.	สุ่มตัวเลข 0 – 987	
5.	สุ่มตัวเลข 1 – 500	
6.	สุ่มตัวเลข 2 – 600	

โจทย์ข้อที่ 12 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็มที่มีความยาว 4 หลักหนึ่งค่าเข้ามาทางแป้นพิมพ์แล้วทำการสลับหลักของตัวเลขนั้นดังตัวอย่างการแสดงผลต่อไปนี้ (10 คะแนน)

1	2	3	4

Enter number: 1234
 Reversed number: 4321

Enter number: 9753
 Reversed number: 3579

โจทย์ข้อที่ 13 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็มจากแป้นพิมพ์ยาว 6 หลัก แล้วทำการตัดตัวเลข 2 หลักแรก และ 2 หลักสุดท้าย เพื่อนำมาคำนวณหาค่าผลบวกและผลต่างของสองตัวเลขนั้น พร้อมทั้งแสดงผลลัพธ์ออกทางจอภาพให้สวยงาม (10 คะแนน)

1	2	3	4

โจทย์ข้อที่ 14 [ระดับยาก] ธนาคารลานเกียร์แห่งประเทศไทยได้ว่าจ้างให้นิสิตชั้นปีที่ 1 ของคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ซึ่งกำลังเรียนวิชาการเขียนโปรแกรมด้วยภาษาจาวาเบื้องต้น เพื่อสร้างเครื่องคำนวณจำนวนเงิน (Money Machine) โดยเมื่อผู้ใช้ระบุจำนวนเงินเข้ามา โปรแกรมจะทำการคำนวณว่าจะต้องใช้ธนบัตรใบละ 1000 บาท 500 บาท 100 บาท 50 บาท และ 20 บาท อย่างละกี่ใบ และเหรียญ 10 บาท 5 บาท 2 บาท และ 1 บาท อย่างละกี่เหรียญ โดยมีตัวอย่างของผลการทำงานดังต่อไปนี้ (15 คะแนน)

[illegible]

Do you know?

```
1 / 0           =
1.0 / 0.0       =
-1.0 / 0.0      =
1.0 / (1.0 / 0) =
0.0 / 0.0       =
```