

บทที่ 2

ตัวแปร ตัวดำเนินการ และคำสั่งพื้นฐานในการแสดงผล

2.1 ตัวแปร

ในมุมมองของนักเขียนโปรแกรม **ตัวแปร** (variable) เปรียบเสมือนกล่องมีชื่อที่บรรจุข้อมูลบางอย่างไว้ ซึ่งอาจจะเป็นตัวเลขหรือข้อความก็ได้ ส่วนในมุมมองของโปรแกรมเอง ตัวแปรเปรียบเสมือนตัวชี้บอก (indicator) ว่าข้อมูลหนึ่ง ๆ เก็บไว้ ณ ตำแหน่งใดของหน่วยความจำ ดังนั้นการกำหนดค่า การเก็บ การดึง และการลบค่าตัวแปร จึงเป็นสิ่งสำคัญอย่างมากในการเขียนโปรแกรม

2.1.1 หลักการตั้งชื่อของตัวแปรใน Scilab

1. ชื่อของตัวแปรไม่ควรมีความยาวเกิน 24 ตัวอักษร (ในกรณีที่เกิน Scilab จะพิจารณาเฉพาะตัวอักษร 24 ตัวแรกของชื่อเท่านั้น)
2. ชื่อของตัวแปรสามารถประกอบไปด้วยตัวอักษรพิมพ์ใหญ่ (A-Z) ตัวอักษรพิมพ์เล็ก (a-z) ตัวเลข (0-9) และสัญลักษณ์พิเศษ ("%", "_", "#", "!", "\$", "?") ปะปนกันได้ เช่น

x1	Days	Number_of_Student	go2school?
----	------	-------------------	------------

3. ชื่อของตัวแปรใน Scilab เป็นแบบไวต่ออักษรใหญ่เล็ก (case-sensitive) นั่นคือ ตัวอักษรพิมพ์ใหญ่และตัวอักษรพิมพ์เล็กถือว่าเป็นคนละตัวกัน เช่น ตัวแปรชื่อ A เป็นคนละตัวกับตัวแปรชื่อ a เป็นต้น
4. ชื่อของตัวแปรที่ตั้งขึ้นใหม่ต้องไม่ซ้ำกับชื่อของคำสั่งใด ๆ ที่มีอยู่แล้วใน Scilab
5. ตัวแปรที่มีชื่อขึ้นต้นด้วย “%” เป็นตัวแปรที่ Scilab ได้กำหนดค่าเอาไว้ให้แล้ว มักใช้สำหรับเก็บค่าคงตัวทางคณิตศาสตร์และตรรกศาสตร์ เช่น

ตัวแปร	ความหมาย
%pi	ค่า π (อัตราส่วนระหว่างเส้นรอบวงและเส้นผ่านศูนย์กลางของวงกลม) มีค่าประมาณ 3.1415927
%e	ค่า e (ฐานของลอการิทึมธรรมชาติ) มีค่าประมาณ 2.7182818 ...
%T	ค่าตรรกะจริง (true)
%F	ค่าตรรกะเท็จ (false)

2.1.2 การกำหนดค่าของตัวแปร

เราสามารถกำหนดค่าของตัวแปรได้โดยใช้คำสั่งในรูปของ

ชื่อของตัวแปร = ค่าของตัวแปร

โดยค่าของตัวแปรนั้น อาจเป็นข้อมูลจำพวกตัวเลขหรือข้อความก็ได้ ซึ่งผู้ใช้อาจกำหนดค่าด้วยตนเองโดยตรง หรือกำหนดโดยให้เป็นผลลัพธ์จากการคำนวณอื่น ๆ เช่น

-->x = 2	// กำหนดให้ตัวแปรชื่อ x มีค่าเท่ากับ 2
x = 2.	
-->str = "Good Morning!"	// กำหนดให้ตัวแปรชื่อ str มีค่าเป็นข้อความ
str = Good Morning!	

```
-->// กำหนดให้ตัวแปรชื่อ result มีค่าเท่ากับค่าของ 2 × (ค่าของตัวแปร x) + 1
-->result = 2*x + 1
    result =
        5.
```

ในการกำหนดค่าของตัวแปรนั้น ถ้าเดิมไม่เคยมีตัวแปรชื่อนั้นอยู่ในหน่วยความจำก่อน Scilab จะสร้างตัวแปรนั้นขึ้นโดยอัตโนมัติ แต่ถ้ามีตัวแปรชื่อนั้นอยู่ก่อนแล้ว Scilab จะกำหนดค่าของตัวแปรตัวนั้นใหม่ เช่น

```
-->clear                // ลบตัวแปรทุกตัวออกจากหน่วยความจำ
-->x = "Who am I?"      // กำหนดค่าของตัวแปรชื่อ x เป็นข้อความ (สร้างตัวแปรขึ้นใหม่)
    x =
    Who am I?
-->x = 45                // กำหนดค่าของตัวแปรชื่อ x เป็นตัวเลข (แก้ไขเฉพาะค่าของตัวแปร)
    x =
    45.
-->y = x                 // ให้ตัวแปรชื่อ y มีค่าเท่ากับค่าของตัวแปรชื่อ x
    y =
    45.
-->y = 2*y              // ให้ y มีค่าเพิ่มเป็น 2 เท่าของค่าเดิม
    y =
    90.
```

ข้อสังเกต ในการกำหนดให้ตัวแปรมีค่าเป็นข้อความนั้น จะต้องให้ข้อความนั้นอยู่ระหว่างเครื่องหมายอัญประกาศเดี่ยว (single quotation mark) ‘ ’ หรือเครื่องหมายอัญประกาศคู่ (double quotation mark) “ ” เสมอ

2.1.3 การลบตัวแปร

โดยทั่วไปแล้ว เมื่อเรากำหนดค่าของตัวแปรใดขึ้นมา ตัวแปรนั้นจะคงอยู่ในหน่วยความจำตลอดไปจนกว่าเราจะออกจากโปรแกรม Scilab แต่ถ้าในระหว่างที่ยังใช้โปรแกรม Scilab นั้นเราต้องการจะลบบางตัวแปรออกเพื่อคืนหน่วยความจำ เราสามารถทำได้โดยใช้คำสั่ง

```
clear <ชื่อตัวแปรที่ต้องการลบตัวแรก> <ชื่อตัวแปรที่ต้องการลบตัวต่อมา> ... <ชื่อตัวแปรที่ต้องการลบตัวสุดท้าย>
```

ในกรณีที่ต้องการลบตัวแปรทุกตัวออกจากหน่วยความจำ เราสามารถทำได้โดยใช้คำสั่ง clear โดยไม่ต้องระบุชื่อตัวแปร

ตัวอย่าง

```
-->x = 5;
-->y = 7;
-->z = 9;
-->clear x z
-->y
    y =
    7.
-->x
    !--error 4
Undefined variable: x
-->z
    !--error 4
Undefined variable: z
```

2.2 ตัวดำเนินการและความสัมพันธ์

บทนิยาม 2.2.1

ตัวดำเนินการ (operator) หมายถึง การกระทำอย่างหนึ่งระหว่างค่าของตัวแปรตัวหนึ่งหรือหลายตัวที่เป็นข้อมูลชนิดเดียวกัน เพื่อให้ได้ผลลัพธ์เป็นค่าใหม่ออกมา โดยที่ค่าใหม่นั้นเป็นข้อมูลชนิดเดิม

ตัวอย่าง 2.2.2

- เครื่องหมายบวก (+) เป็นตัวดำเนินการระหว่างจำนวนจริง 2 จำนวน เช่น $1 + 2 = 3$ เป็นต้น
- เครื่องหมายรากที่สอง ($\sqrt{\quad}$) เป็นตัวดำเนินการของจำนวนจริง 1 ตัว เช่น $\sqrt{2.25} = 1.5$ เป็นต้น

บทนิยาม 2.2.3

ความสัมพันธ์ (relation) หมายถึง การตรวจสอบเปรียบเทียบค่าของตัวแปร 2 ตัวที่เป็นข้อมูลชนิดเดียวกันว่าสอดคล้องกับเงื่อนไขบางอย่างหรือไม่ ซึ่งผลลัพธ์ที่ได้จากความสัมพันธ์จะมีเป็นค่าตรรกะจริง (true) หรือเท็จ (false) เท่านั้น

ตัวอย่าง 2.2.4

- เครื่องหมายมากกว่า (>) เป็นความสัมพันธ์ระหว่างจำนวนจริง 2 จำนวน เช่น $2 > 1$ มีค่าตรรกะเป็นจริง แต่ $1 > 5$ มีค่าตรรกะเป็นเท็จ เป็นต้น

โปรแกรม Scilab ได้กำหนดตัวดำเนินการและความสัมพันธ์ไว้หลายตัว ในที่นี้จะกล่าวถึงเพียงบางส่วนเท่านั้น

ตัวดำเนินการ

ตัวดำเนินการ	ความหมาย	ประเภทของข้อมูลที่สามารถดำเนินการได้
+	การบวก (addition)	จำนวนจริง 2 ตัว
-	การลบ (subtraction)	จำนวนจริง 2 ตัว
*	การคูณ (multiplication)	จำนวนจริง 2 ตัว
/	การหารทางขวา (right-hand division)	จำนวนจริง 2 ตัว (ตัวหารอยู่ทางขวา)
\	การหารทางซ้าย (left-hand division)	จำนวนจริง 2 ตัว (ตัวหารอยู่ทางซ้าย)
^	ยกกำลัง (exponentiation)	จำนวนจริง 2 จำนวน (ตัวแรกเป็นฐาน ตัวที่สองเป็นเลขชี้กำลัง)
~	นิเสธ (negation)	ค่าตรรกะหนึ่งตัว (เปลี่ยนจริงเป็นเท็จ เปลี่ยนเท็จเป็นจริง)
&	และ (and)	ค่าตรรกะ 2 ตัว (ผลที่ได้จะจริงก็ต่อเมื่อทั้งสองตัวเป็นจริง)
	หรือ (or)	ค่าตรรกะ 2 ตัว (ผลที่ได้จะจริงก็ต่อเมื่อมีบางตัวเป็นจริง)

ตัวอย่าง 2.2.5

```

-->x = 3;
-->y = 5;
-->z = 7;
-->x + y
ans =
    8.
-->y - z
ans =
    - 2.
-->x * y
ans =
    15.
-->y/z           // หา  $\frac{y}{z} = \frac{5}{7}$ 
ans =
    0.7142857
-->z\x           // หา  $\frac{x}{z} = \frac{3}{7}$ 
ans =
    0.4285714
-->y^x           // หา  $y^x = 5^3$ 
ans =
    125.
-->t = %T        // ให้ตัวแปร t มีค่าตรรกะเป็นจริง
t =
    T
-->f = %F        // ให้ตัวแปร f มีค่าตรรกะเป็นเท็จ
f =
    F
-->~t
ans =
    F
-->t & f          // จริงและเท็จเป็นเท็จ
ans =
    F
-->t | f          // จริงหรือเท็จเป็นจริง
ans =
    T

```

ความสัมพันธ์

ความสัมพันธ์	ความหมาย	ประเภทของข้อมูลที่สามารถดำเนินการได้
<	น้อยกว่าหรือไม่	จำนวนจริง 2 จำนวน
<=	น้อยกว่าหรือเท่ากับหรือไม่	จำนวนจริง 2 จำนวน
>	มากกว่าหรือไม่	จำนวนจริง 2 จำนวน
>=	มากกว่าหรือเท่ากับหรือไม่	จำนวนจริง 2 จำนวน
==	เท่ากันหรือไม่	จำนวนจริง 2 จำนวน

ตัวอย่าง 2.2.6

```
-->x = 3;
-->y = 5;
-->z = 7;
-->x < y
ans =
    T
-->z <= y
ans =
    F
-->z > x
ans =
    T
-->y >= x
ans =
    T
-->x == y
ans =
    F
```

2.3 คำสั่งพื้นฐานที่เกี่ยวข้องกับการแสดงผล

ในโปรแกรม Scilab มีคำสั่งที่เกี่ยวข้องกับการแสดงผลอยู่หลายคำสั่ง ในที่นี้เราจะกล่าวถึงเฉพาะคำสั่งต่อไปนี้ ซึ่งเป็นคำสั่งที่เกี่ยวข้องกับการแสดงผลทางจอภาพ (ผ่านทาง Scilab Console)

คำสั่ง disp

คำสั่ง disp (ย่อมาจากคำว่า display) เป็นคำสั่งที่ใช้ในการแสดงค่าของตัวแปร หรือข้อความ (string) ที่มีรูปแบบไม่ซับซ้อนมากนัก ตัวอย่างเช่น

```
-->disp(1+2)
    3.
-->x = 5;
-->disp(x)
    5.
-->disp("Hello World!")
Hello World!
```

คำสั่ง printf

คำสั่ง printf เป็นคำสั่งที่เหมาะสมสำหรับการแสดงผลที่มีความซับซ้อนมากขึ้น เช่น ต้องการแสดงข้อความพร้อมทั้งค่าของตัวแปรหลายตัว คำสั่ง printf มีข้อดีตรงที่เราสามารถจัดรูปแบบของค่าต่าง ๆ ก่อนที่จะแสดงผลได้ พิจารณาชุดคำสั่งต่อไปนี้

```
-->printf("Hello World!")
Hello World!

-->name = "James Bond";
-->printf("Hello World, %s\n", name)
Hello World, James Bond

-->age = 22;
-->printf("You are now %d years old\n", age)
You are now 22 years old

-->x = 12.34567901;
-->printf("The rounding of x to two decimal places is %.2f\n", x)
The rounding of x to two decimal places is 12.35
```

คำอธิบายตัวจัดการรูปแบบ (formatting) ของคำสั่ง `printf`

ตัวจัดการรูปแบบ	ความหมาย
<code>%s</code>	ข้อความ (string)
<code>%d</code>	จำนวนเต็ม (decimal)
<code>%.<x>f</code>	จำนวนจริงทศนิยม <x> ตำแหน่ง (เช่น <code>%.2f</code> หมายถึงจำนวนจริงที่มีทศนิยม 2 ตำแหน่ง)
<code>\t</code>	ช่องว่างระหว่างตัวอักษร ซึ่งมีระยะห่างเท่ากับการกดปุ่ม Tab 1 ครั้ง
<code>\n</code>	ขึ้นบรรทัดใหม่