

## CHAPTER 04

## คำสั่งวนซ้ำ (Iteration Statements)

### 1. คำสั่งวนซ้ำแบบเบื้องต้นด้วยคำสั่ง while

#### 1. นิยามและข้อกำหนดของคำสั่งวนซ้ำหรือทำซ้ำ

- 1) คำสั่งวนซ้ำหรือทำซ้ำ (Iteration/Repetition) คือ คำสั่งที่ใช้สำหรับวนรอบเพื่อทำงานอย่างใดอย่างหนึ่งตามเงื่อนไข (Condition) ที่กำหนดขึ้น และจะวนทำงานแบบนั้นไปเรื่อยๆ ตราบใดที่เงื่อนไขยังเป็นจริง
- 2) คำสั่งวนซ้ำในภาษาจาวา ประกอบไปด้วย 3 ชุดคำสั่งได้แก่ [1] while, [2] do-while และ [3] for

#### 2. คำสั่งวนซ้ำด้วยคำสั่ง while มี 3 ส่วนหลักดังรูปแบบคำสั่งต่อไปนี้

```

1 Initial;
2 while(Condition) {
3     Statement(s);
4     Update;
5 }

```

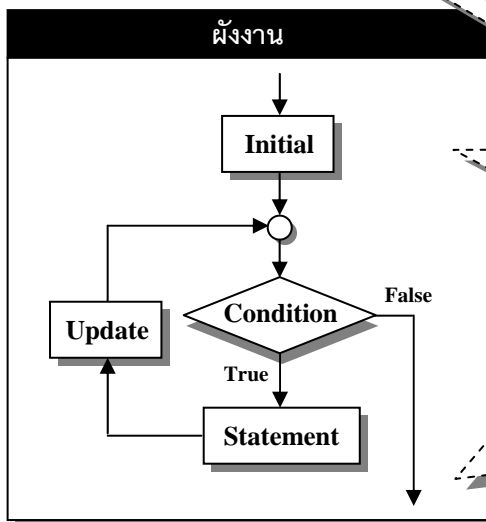
① คำตั้งต้น (Initial) เป็นการกำหนดค่าเริ่มต้นสำหรับการวนซ้ำ

② เงื่อนไข (Condition) จะเป็นนิพจน์ตรรกะ (Boolean Expression) ที่แสดงถึงค่าความจริงว่าเมื่อไรจะวนซ้ำ เมื่อไรจะสิ้นสุดการวนซ้ำ

③ การเปลี่ยนรอบ (Update) เป็นการเปลี่ยนรอบการวนซ้ำ (ถ้าไม่ระบุส่วนนี้จะเกิดการวนซ้ำแบบไม่รู้จบ หรือ Infinite Loop)

ภายในคำสั่ง while ถ้ามีเพียง 1 คำสั่งไม่ต้องมีเครื่องหมายปีกกาเปิดปิดก็ได้

พิจารณาเงื่อนไขของคำสั่ง while ว่าเป็นจริงหรือไม่ ถ้าเป็นจริงจะเข้าทำคำสั่งทุกๆ คำสั่ง ภายในคำสั่ง while และจะวนทำงานแบบนั้นไปเรื่อยๆ จนกว่าเงื่อนไขของคำสั่ง while จะเป็นเท็จ ก็จะสิ้นสุดการวนซ้ำ



#### ตัวอย่าง

```

1 int i = 1;
2 while(i <= 5) {
3     System.out.println(i);
4     i++;
5 }

```

กำหนดค่าเริ่มต้นให้ตัวแปร i เท่ากับ 1 แล้วทำการวนซ้ำเพื่อแสดงค่า i ตั้งแต่ 1 ถึง 5 โดยในแต่ละรอบการวนซ้ำจะเพิ่มจำนวนรอบ (เพิ่มค่า i) ขึ้นทีละ 1 (โดยใช้คำสั่ง i++) ดังนั้นผลลัพธ์ที่ได้คือ 12345

โจทย์ข้อที่ 1 [ระดับง่าย] จงแสดงผลลัพธ์จากส่วนของโปรแกรมตามค่า n ที่กำหนดให้ต่อไปนี้ (5 คะแนน)

```

1 int i = 0, j = 5;
2 while (i <= n) {
3     System.out.println(n++);
4     i = i + 2; }
5 while (j > n) {
6     System.out.println(--j);
7     j--;
8 }

```

| n = 1 | n = 0 | n = -1 | n = 3 | n = -2 |
|-------|-------|--------|-------|--------|
|       |       |        |       |        |
|       |       |        |       |        |
|       |       |        |       |        |
|       |       |        |       |        |

โจทย์ข้อที่ 2 [ระดับง่าย] จงเขียนเฉพาะคำสั่ง `while` ตามเงื่อนไขต่อไปนี้ (10 คะแนน)

1) แสดงคำว่า "Java" ขึ้นบนจอภาพ 100 ครั้ง (5 คะแนน)

|  |  |  |
|--|--|--|
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

2) แสดงเลขคู่ 10 ตัวแรกออกทางจอภาพ โดยเริ่มต้นที่ตัวเลข 2 (5 คะแนน)

|  |  |  |
|--|--|--|
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

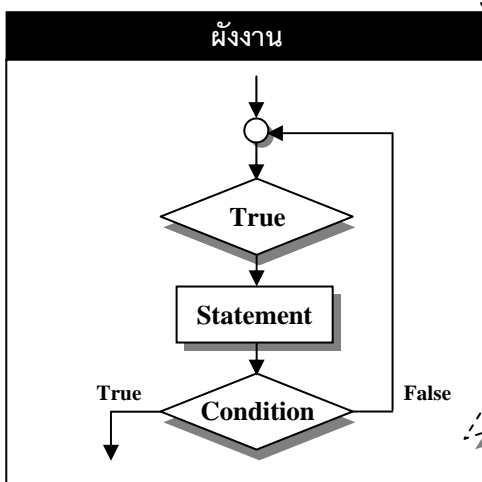
3. คำสั่งวนซ้ำด้วยคำสั่ง `while (true)` มีรูปแบบดังต่อไปนี้

```

1 while(true) {
2     Statement(s);
3     if(Condition) break;
4 }

```

คำสั่ง `while(true)` มีเงื่อนไข (Condition) เป็นจริงตลอดกาล ดังนั้นจะวนซ้ำแบบไม่รู้จบ



การออกจากคำสั่ง `while(true)` ทำได้เพียงวิธีเดียวเท่านั้น คือใช้คำสั่ง `if(...) break;` โดยเมื่อเงื่อนไขของคำสั่ง `if` เป็นจริงก็จะสิ้นสุดการวนซ้ำ

เข้าทำคำสั่งทุกๆ คำสั่ง ภายในคำสั่ง `while` ทันทีเพราะว่าเงื่อนไขเป็นจริงตลอดกาล และจะวนทำงานแบบนี้ไปเรื่อยๆ จนกว่าเงื่อนไขของคำสั่ง `if` ซึ่งอยู่ภายในคำสั่ง `while` เป็นจริง ก็จะ `break` การทำงาน และสิ้นสุดการวนซ้ำ

ตัวอย่าง

```

1 Scanner kb = new ...;
2 while(true) {
3     int x = kb.nextInt();
4     if(x == 0) break;
5 }

```

วนรับค่าตัวเลขจำนวนเต็ม 1 จำนวนทางแป้นพิมพ์ แล้วตรวจสอบว่าตัวเลขที่รับเข้ามานั้นเท่ากับ 0 หรือไม่ ถ้าใช่ก็จะสิ้นสุดการวนซ้ำ แต่ถ้าไม่เช่นนั้นก็จะวนรับค่าตัวเลขจำนวนเต็มจากแป้นพิมพ์ไปเรื่อยๆ

โจทย์ข้อที่ 3 [ระดับง่าย] จงเขียนเฉพาะคำสั่ง while(true) ตามเงื่อนไขต่อไปนี้ (10 คะแนน)

- 1) แสดงคำว่า “เกรียน” ขึ้นบนจอภาพ 1000 ครั้ง (5 คะแนน)

|  |  |  |  |
|--|--|--|--|
|  |  |  |  |
|--|--|--|--|

- 2) สุ่มตัวเลขตั้งแต่ 0-10 แล้วแสดงผลขึ้นบนจอภาพไปเรื่อยๆ จะจบเมื่อสุ่มได้เลข 5 (5 คะแนน)

|  |  |  |  |
|--|--|--|--|
|  |  |  |  |
|--|--|--|--|

โจทย์ข้อที่ 4 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์ เพื่อรับค่าตัวเลขจำนวนเต็มหนึ่งจำนวน เข้ามาทางแป้นพิมพ์ แล้วทำการแสดงค่าตัวเลขจำนวนเต็มตั้งแต่ 0 จนถึงตัวเลขที่รับเข้ามานั้นออกทาง จอภาพทีละบรรทัดให้สวยงาม (10 คะแนน)

[illegible]

Diagram illustrating a stack data structure with four slots. The slots are labeled 1, 2, 3, and 4 from top to bottom. The stack contains the values 1, 2, 3, and 4. To the right of the stack, there are three input/output boxes showing the sequence of operations:

- Enter number: -1  
-1
- Enter number: 4  
4 \*\*\*\*
- Enter number: 6  
6 \*\*\*\*\*

[illegible]



|  | 1 | 2 | 3 | 4 |
|--|---|---|---|---|
|  |   |   |   |   |

[illegible]



## ตัวอย่าง

```

1 for(int i = 1; i <= 5; i++) {
2     System.out.print(i);
3 }

```

กำหนดค่าเริ่มต้นให้ตัวแปร  $i$  เท่ากับ 1 แล้วทำการวนซ้ำเพื่อแสดงค่า  $i$  ตั้งแต่ 1 ถึง 5 โดยในแต่ละรอบการวนซ้ำจะเพิ่มค่า  $i$  ขึ้นทีละ 1 ดังนั้นผลลัพธ์ที่ได้คือ 12345

โจทย์ข้อที่ 12 [ระดับง่าย] จงแสดงผลลัพธ์จากส่วนของโปรแกรมตามค่า  $n$  ที่กำหนดให้ต่อไปนี้ (4 คะแนน)

```

1 double c = 3.0;
2 for(c = 1.0; c <= n; c += 0.1)
3     System.out.println(c++);
4 c++;
5 for(int i = 6; i > n; i--) {
6     System.out.print(i);
7     System.out.print(n++);
8 }

```

| $n = 3$ | $n = 0$ | $n = 1$ | $n = 4$ |
|---------|---------|---------|---------|
|         |         |         |         |

โจทย์ข้อที่ 13 [ระดับง่าย] จงเขียนเฉพาะคำสั่ง `for` ตามเงื่อนไขต่อไปนี้ (10 คะแนน)

1) แสดงคำว่า "Get A" ขึ้นบนจอภาพ 1000 ครั้ง (5 คะแนน)

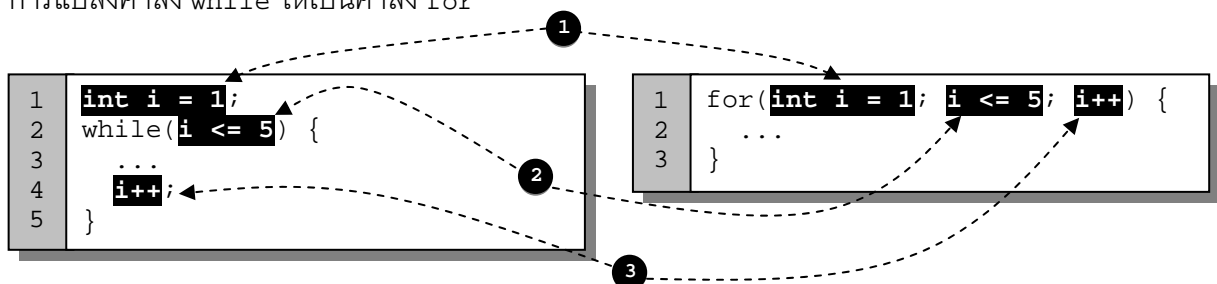
|  |  |  |
|--|--|--|
|  |  |  |
|--|--|--|

2) แสดงเลขคู่ 900 ตัวแรกออกทางจอภาพ (เริ่มที่ 78) (5 คะแนน)

|  |  |  |
|--|--|--|
|  |  |  |
|--|--|--|

## 2. เปรียบเทียบคำสั่ง `while` และคำสั่ง `for`

- คำสั่ง `while` และคำสั่ง `for` มีลักษณะการทำงานที่วนซ้ำเหมือนกัน แต่มีรูปแบบการเขียนที่แตกต่างกัน
- คำสั่ง `while` ใช้วนซ้ำเมื่อไม่ทราบจำนวนรอบที่แน่ชัด แต่คำสั่ง `for` ใช้วนซ้ำเมื่อทราบจำนวนรอบที่แน่ชัด
- การแปลงคำสั่ง `while` ให้เป็นคำสั่ง `for`





โจทย์ข้อที่ 14 [ระดับง่าย] จงเขียนส่วนของโปรแกรมภาษาจาวาเพื่อเปลี่ยนคำสั่ง while ในโจทย์ข้อที่ 6 ถึง 9 ให้เป็นคำสั่ง for โดยไม่ต้องเขียนโปรแกรมในส่วนของการรับค่าและแสดงผล และสามารถเรียกใช้ตัวแปรเดิมของการรับค่าได้เลย แต่ตัวแปรอื่นๆ จะต้องประกาศใหม่ทั้งหมด (20 คะแนน)

- 1) แปลงเฉพาะคำสั่ง while ในโจทย์ข้อที่ 6 ให้เป็นคำสั่ง for (5 คะแนน)

|  |  |  |
|--|--|--|
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

- 2) แปลงเฉพาะคำสั่ง while ในโจทย์ข้อที่ 7 ให้เป็นคำสั่ง for (5 คะแนน)

|  |  |  |
|--|--|--|
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

- 3) แปลงเฉพาะคำสั่ง while ในโจทย์ข้อที่ 8 ให้เป็นคำสั่ง for (5 คะแนน)

|  |  |  |
|--|--|--|
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

- 4) แปลงเฉพาะคำสั่ง while ในโจทย์ข้อที่ 9 ให้เป็นคำสั่ง for (5 คะแนน)

|  |  |  |
|--|--|--|
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

โจทย์ข้อที่ 15 [ระดับง่าย - ระดับยาก] จงเขียนเฉพาะคำสั่ง for เพื่อหาคำตอบของสมการต่อไปนี้ พร้อมทั้งประกาศตัวแปรทุกตัวที่เรียกใช้งาน และห้ามเรียกใช้เมทอดในคลาส Math (30 คะแนน)

- 1) [ระดับง่าย]  $5 + 10 + 15 + \dots + 40$  (5 คะแนน)

---

|  |  |  |
|--|--|--|
|  |  |  |
|--|--|--|

---

- 2) [ระดับง่าย]  $1 + 8 + 27 + \dots + 20^3$  (5 คะแนน)

---

|  |  |  |
|--|--|--|
|  |  |  |
|--|--|--|

---

- 3) [ระดับปานกลาง]  $1 + 3 + 7 + 15 + 31 + \dots + (2^{15} - 1)$  (5 คะแนน)

---

|  |  |  |
|--|--|--|
|  |  |  |
|--|--|--|

---

- 4) [ระดับปานกลาง]  $1 + \frac{1}{2} - \frac{1}{3} + \frac{1}{4} - \frac{1}{5} + \dots + \frac{1}{30}$  (5 คะแนน)

---

|  |  |  |
|--|--|--|
|  |  |  |
|--|--|--|

---



Diagram illustrating the iterative process of building a number pyramid. The table shows the state of the pyramid at each step:

| 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|
| * |   |   |   |   |
| * | * |   |   |   |
| * | * | * |   |   |
| * | * | * | * |   |
| * | * | * | * | * |

Number: 1  
\*

Number: 10  
\*  
\*\*  
\*\*\*  
\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

The diagram consists of five vertical dashed lines extending from a grey header bar. The header bar contains the numbers 1, 2, 3, 4, and 5 above each respective line. To the right of these lines are two speech bubbles. The first bubble contains the text "Number: 17" followed by "17 is a prime number". The second bubble contains the text "Number: 45" followed by "45 is not a prime number".