

4장. 반복문



❖ 3절. 반복문

- for문
- while문
- do-while문

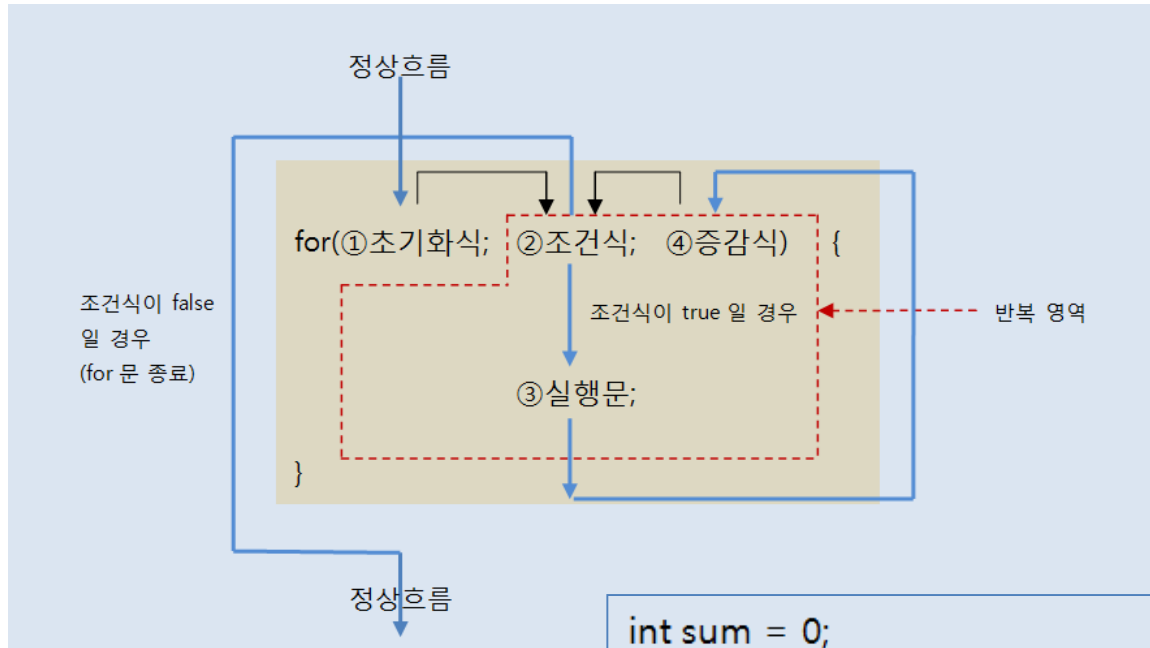
3절. 반복문(for문, while문, do-while문)

❖ 반복문

- 중괄호 블록 내용을 반복적으로 실행할 때 사용
- 종류: for문, while문, do-while문

3절. 반복문(for문, while문, do-while문)

❖ for문: 반복 횟수를 알고 있을 때 주로 사용

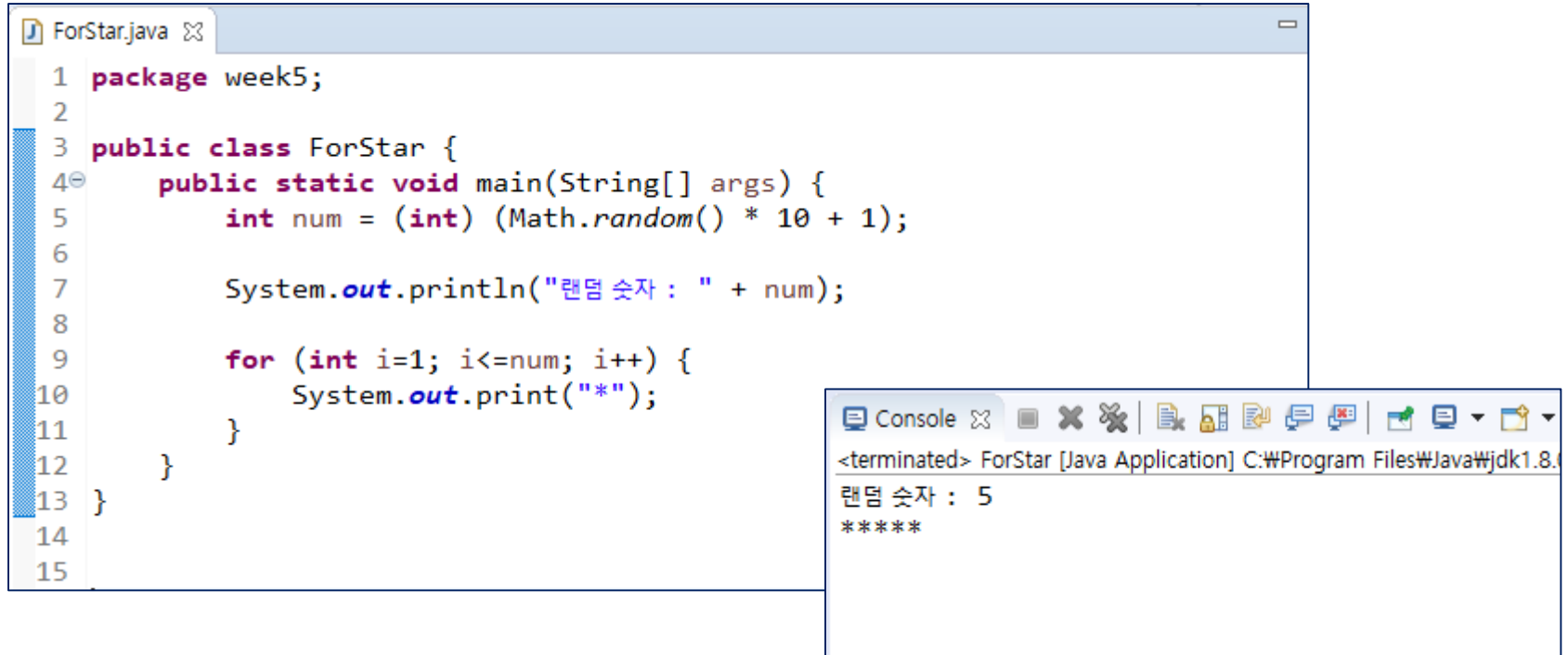


```
int sum = 0;
for (int i=1; i<=100; i++) {
    sum = sum + i; ●----- 100 번 반복
}
System.out.println("1~100 까지의 합:" + sum);
```

for 문 실습 (ForStar.java)

```
ForStar.java
1 package week5;
2
3 public class ForStar {
4     public static void main(String[] args) {
5         int num = (int) (Math.random() * 10 + 1);
6
7         System.out.println("랜덤 숫자 : " + num);
8
9         for (int i=1; i<=num; i++) {
10             System.out.print("*");
11         }
12     }
13 }
14
15
```

for 문 실습 (ForStar.java)



The screenshot shows an IDE window titled 'ForStar.java' with the following Java code:

```
1 package week5;
2
3 public class ForStar {
4     public static void main(String[] args) {
5         int num = (int) (Math.random() * 10 + 1);
6
7         System.out.println("랜덤 숫자 : " + num);
8
9         for (int i=1; i<=num; i++) {
10             System.out.print("*");
11         }
12     }
13 }
14
15
```

Below the code editor is a console window titled 'Console'. It displays the output of the program:

```
<terminated> ForStar [Java Application] C:\Program Files\Java\jdk1.8.0_101\bin\java.exe
랜덤 숫자 : 5
*****
```

for 문 실습 (Gugudan.java)

❖ 구구단 계산기 만들기

- Scanner 클래스 사용하기
- 입력된 숫자를 이용하여 구구단을 아래와 같이 출력한다

구구단 몇 단을 계산할까?

5

구구단 5단을 계산한다.

$5 \times 1 = 5$

$5 \times 2 = 10$

⋮

$5 \times 8 = 40$

$5 \times 9 = 45$

for 문 실습 (Gugudan.java)

❖ 구구단 계산기 만들기

- Scanner 클래스 사용하기
- 입력된 숫자를 이용하여 구구단을 아래와 같이 출력한다

구구단 몇 단을 계산할까?

5

구구단 5단을 계산한다.

5 × 1 = 5

5 × 2 = 10

⋮

5 × 8 = 40

5 × 9 = 45

```
package week4;

import java.util.Scanner;

public class Gugudan {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("구구단 몇 단을 계산할까?");

        int scanData = sc.nextInt();

    }
}
```


for 문 실습 (Gugudan.java)

❖ 구구단 계산기 만들기

- Scanner 클래스 사용하기
- 입력된 숫자를 이용하여 구구단을 아래와 같이 출력한다

구구단 몇 단을 계산할까?

5

구구단 5단을 계산한다.

5 × 1 = 5

5 × 2 = 10

⋮

5 × 8 = 40

5 × 9 = 45

```
package week4;

import java.util.Scanner;

public class Gugudan {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("구구단 몇 단을 계산할까?");

        int scanData = sc.nextInt();

        System.out.println("구구단 "+scanData+"단을 계산한다.");
        for (int i=1; i<=9; i++) {
            System.out.println(scanData+" * "+i+" = "+(scanData*i));
        }
    }
}
```

중첩 for 문

❖ for문 안에 또 다른 for문이 있는 경우

바깥 for문이 한 번
실행될 때마다

```
for (int i=0; i<3; i++) {  
    for (int j=0; j<3; j++) {  
        System.out.println("i="+i+", j="+j);  
    }  
    명령문;  
}
```

안쪽 for문은
지정된 횟수만큼
반복한다

중첩 for 문 실습 (ForGugu.java)

❖ 구구단 계산기 만들기 #2

- 2단 ~ 9단까지 모두 출력한다

```
Console [Java Application] C:\Program Files\Java
<terminated> ForGugu2 [Java Application] C:\Program Files\Java

=====
구구단 : 2단
=====
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
=====
구구단 : 3단
=====
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
```

중첩 for 문 실습 (ForGugu.java)

❖ 구구단 계산기 만들기 #2

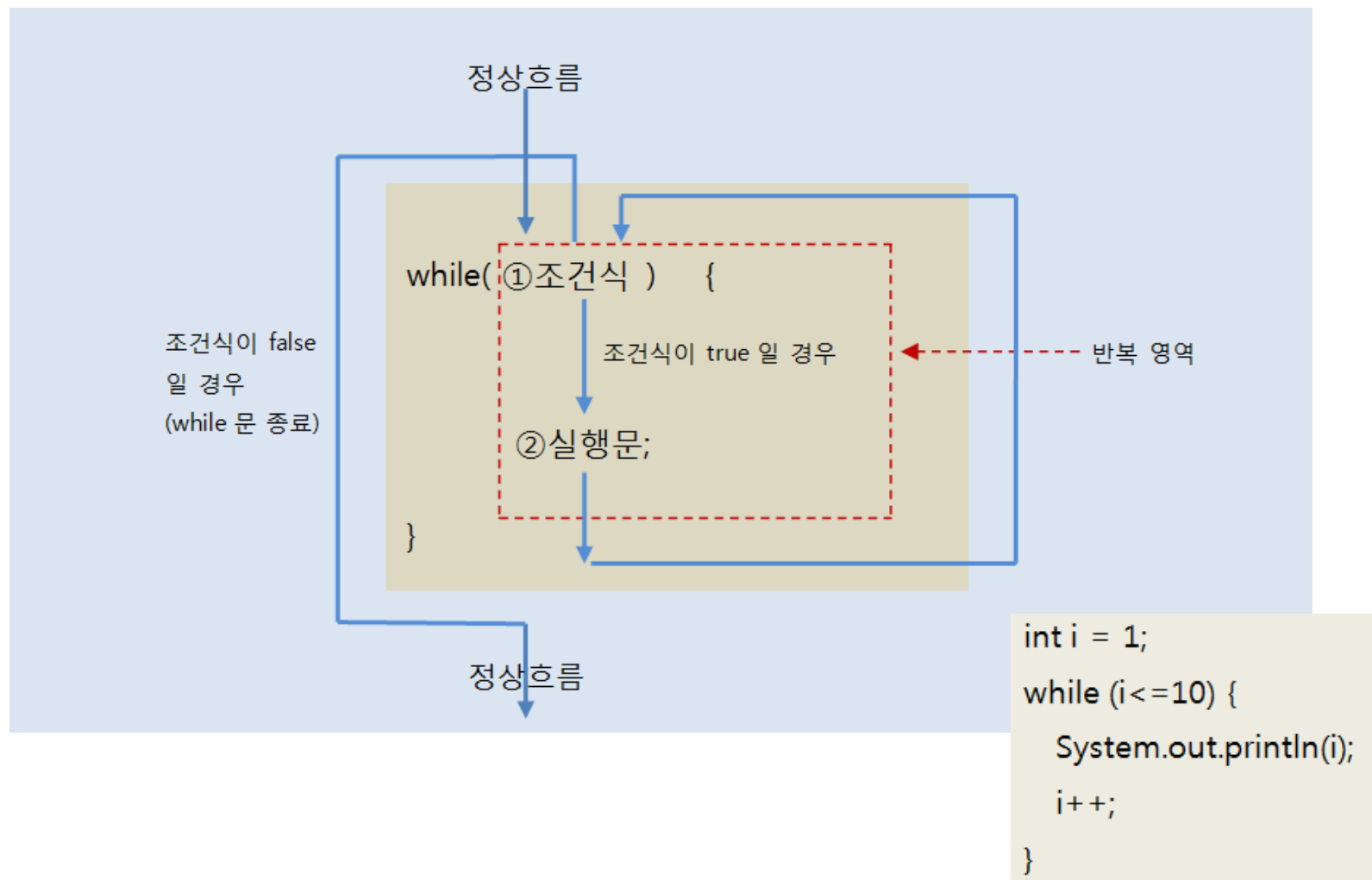
- 2단 ~ 9단까지 모두 출력한다

```
Console
<terminated> ForGugu2 [Java Application] C:\Program Files\Java\
=====
구구단 : 2단
=====
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
=====
구구단 : 3단
=====
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
```

```
ForGugu.java
1 package week5;
2
3 public class ForGugu {
4     public static void main(String[] args) {
5
6         for (int i=2; i<=9; i++) {
7             System.out.println("=====");
8             System.out.println("구구단 : " + i + "단");
9             System.out.println("=====");
10
11             for (int j=1; j<=9; j++) {
12                 System.out.println(i + " * " + j + " = " + (i*j));
13             }
14         }
15     }
16 }
17 }
```

3절. 반복문(for문, while문, do-while문)

❖ while문: 조건에 따라 반복을 계속할지 결정할 때 사용



키보드 값 입력 예제 (KeyControl.java)

```
KeyControl.java
1 package week5;
2
3 import java.io.IOException;
4
5 public class KeyControl {
6     public static void main(String[] args) throws IOException {
7         int keyCode = 0;
8
9         System.out.println("값을 입력하세요");
10
11         keyCode = System.in.read();
12
13         System.out.println("입력된 값 : " + keyCode);
14
15     }
16 }
17
```

키보드 값 입력 예제 (KeyControl.java)

```
KeyControl.java
1 package week5;
2
3 import java.io.IOException;
4
5 public class KeyControl {
6     public static void main(String[] args) throws IOException {
7         int keyCode = 0;
8
9         System.out.println("값을 입력하세요");
10
11         keyCode = System.in.read();
12
13         System.out.println("입력된 값 : " + keyCode);
14
15     }
16 }
17
```

Console

<terminated> KeyControl [Java Application] C:\Program Files\Java\j

값을 입력하세요

1

입력된 값 : 49

while 문 예제 (KeyControl2.java)

```
KeyControl2.java
1 package week5;
2
3 import java.io.IOException;
4
5 public class KeyControl2 {
6     public static void main(String[] args) throws IOException {
7         int    keyCode = 0;
8         int     count  = 0;
9         boolean run    = true;
10
11         while (run) {
12             System.out.println("값을 입력하세요");
13
14             keyCode = System.in.read();
15
16             System.out.println("입력된 값 : " + keyCode);
17
18             //키보드로 값을 입력받을 때마다 카운트해서 5번을 입력받으면 빠져나가도록 한다
19             count++;
20             System.out.println("count = " + count);
21             if (count > 5)
22                 run = false;
23         }
24
25         System.out.println("프로그램 종료");
26     }
27 }
```


while 문 예제 (KeyControl2.java)

KeyControl2.java

```
1 package week5;
2
3 import java.io.IOException;
4
5 public class KeyControl2 {
6     public static void main(String[] args) throws IOException {
7         int keyCode = 0;
8         int count = 0;
9         boolean run = true;
10
11         while (run) {
12             System.out.println("값을 입력하세요");
13
14             keyCode = System.in.read();
15
16             System.out.println("입력된 값 : " + keyCode);
17
18             //키보드로 값을 입력받을 때마다 카운트해서 5번을 입력받으면 빠져나가도록 한다
19             count++;
20             System.out.println("count = " + count);
21             if (count > 5)
22                 run = false;
23         }
24
25         System.out.println("프로그램 종료");
26     }
27 }
```

Console

<terminated> KeyControl2 [Java Application]

값을 입력하세요

1

입력된 값 : 49

count = 1

값을 입력하세요

입력된 값 : 13

count = 2

값을 입력하세요

입력된 값 : 10

count = 3

값을 입력하세요

2

입력된 값 : 50

count = 4

값을 입력하세요

입력된 값 : 13

count = 5

값을 입력하세요

입력된 값 : 10

count = 6

프로그램 종료

while 문 예제 (KeyControl2.java)

```
1 package week5;
2
3 import java.io.IOException;
4
5 public class KeyControl2 {
6     public static void main(String[] args) throws IOException {
7         int keyCode = 0;
8         int count = 0;
9         boolean run = true;
10
11         while (run) {
12             System.out.println("값을 입력하세요");
13
14             keyCode = System.in.read();
15
16             System.out.println("입력된 값 : " + keyCode);
17
18             //키보드로 값을 입력받을 때마다 카운트해서 5번을 입력받으면 빠져나가도록 한다
19             count++;
20             System.out.println("count = " + count);
21             if (count > 5)
22                 run = false;
23         }
24
25         System.out.println("프로그램 종료");
26     }
27 }
```

Console Output:

```
<terminated> KeyControl2 [Java Application]
값을 입력하세요
1
입력된 값 : 49
count = 1
값을 입력하세요
입력된 값 : 13
count = 2
값을 입력하세요
입력된 값 : 10
count = 3
값을 입력하세요
2
입력된 값 : 50
count = 4
값을 입력하세요
입력된 값 : 13
count = 5
값을 입력하세요
입력된 값 : 10
count = 6
프로그램 종료
```

Annotations:

- Carriage return 값 (points to the '\n' character in the code and the '13' input in the console)
- New line 값 (points to the '\n' character in the code and the '10' input in the console)

while 문 예제 (KeyControl2.java)

```
WhileKeyControl3.java
1 package week5;
2
3 import java.io.IOException;
4
5 public class WhileKeyControl3 {
6     public static void main(String[] args) throws IOException {
7         int    keyCode = 0;
8         int    count   = 0;
9         boolean run     = true;
10
11         System.out.println("숫자를 입력하세요(0~9)");
12
13         while(run) {
14             keyCode = System.in.read();
15
16             if (keyCode != 13 && keyCode != 10) {
17                 System.out.println("입력한 값은 " + (char)keyCode + "입니다");
18                 System.out.println("숫자를 입력하세요(0~9)");
19
20                 count++;
21                 if (count > 5)
22                     run = false;
23             }
24         }
25
26         System.out.println("프로그램 종료");
27     }
28 }
```

while 문 예제 (KeyControl2.java)

WhileKeyControl3.java

```
1 package week5;
2
3 import java.io.IOException;
4
5 public class WhileKeyControl3 {
6     public static void main(String[] args) throws
7         int    keyCode = 0;
8         int    count  = 0;
9         boolean run    = true;
10
11     System.out.println("숫자를 입력하세요(0~9)");
12
13     while(run) {
14         keyCode = System.in.read();
15
16         if (keyCode != 13 && keyCode != 10) {
17             System.out.println("입력한 값은 " + (char)keyCode + "입니다");
18             System.out.println("숫자를 입력하세요(0~9)");
19
20             count++;
21             if (count > 5)
22                 run = false;
23         }
24     }
25
26     System.out.println("프로그램 종료");
27 }
28 }
```

Console

WhileKeyControl3 [Java Application] C:\Program Files\Java\jdk1.

숫자를 입력하세요(0~9)

2

입력한 값은 2입니다

숫자를 입력하세요(0~9)

5

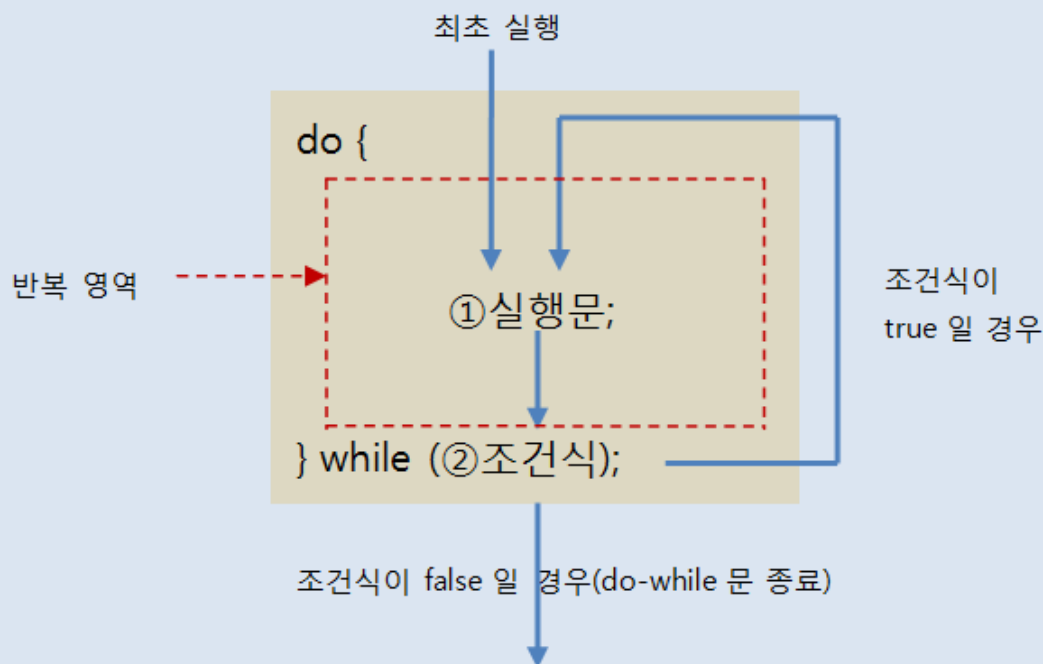
입력한 값은 5입니다

숫자를 입력하세요(0~9)

3절. 반복문(for문, while문, do-while문)

❖ do-while문

- 조건 따라 반복 계속할지 결정할 때 사용하는 것은 while문과 동일
- 무조건 중괄호 { } 블록을 한 번 실행한 후, 조건을 검사해 반복 결정



do while 문 예제 (DoWhile.java)

```
DoWhile.java
1 package week5;
2
3 import java.util.Scanner;
4
5 public class DoWhile {
6     public static void main(String[] args) {
7         System.out.println("메시지를 입력하세요");
8         System.out.println("프로그램을 종료하려면 q를 입력하세요");
9
10        Scanner scanData = new Scanner(System.in);
11        String inputStr;
12
13        do {
14            System.out.print(">");
15            inputStr = scanData.nextLine();
16            System.out.println(inputStr);
17        } while(!inputStr.equals("q"));
18
19        System.out.println();
20        System.out.println("프로그램 종료");
21
22        scanData.close();
23    }
24 }
```

do while 문 예제 (DoWhile.java)

```
1 package week5;
2
3 import java.util.Scanner;
4
5 public class DoWhile {
6     public static void main(String[] args) {
7         System.out.println("메시지를 입력하세요");
8         System.out.println("프로그램을 종료하려면 q를 입력하세요");
9
10        Scanner scanData = new Scanner(System.in);
11        String inputStr;
12
13        do {
14            System.out.print(">");
15            inputStr = scanData.nextLine();
16            System.out.println(inputStr);
17        } while (!inputStr.equals("q"));
18
19        System.out.println();
20        System.out.println("프로그램 종료");
21
22        scanData.close();
23    }
24 }
```

Scanner 객체 생성

키보드로 입력한 문자열을 얻음

문자열을 비교할 때는 equals() 메소드를 이용한다

While 안의 조건식이 true 인 경우에는 do 안을 실행한다

do while 문 예제 (DoWhile.java)

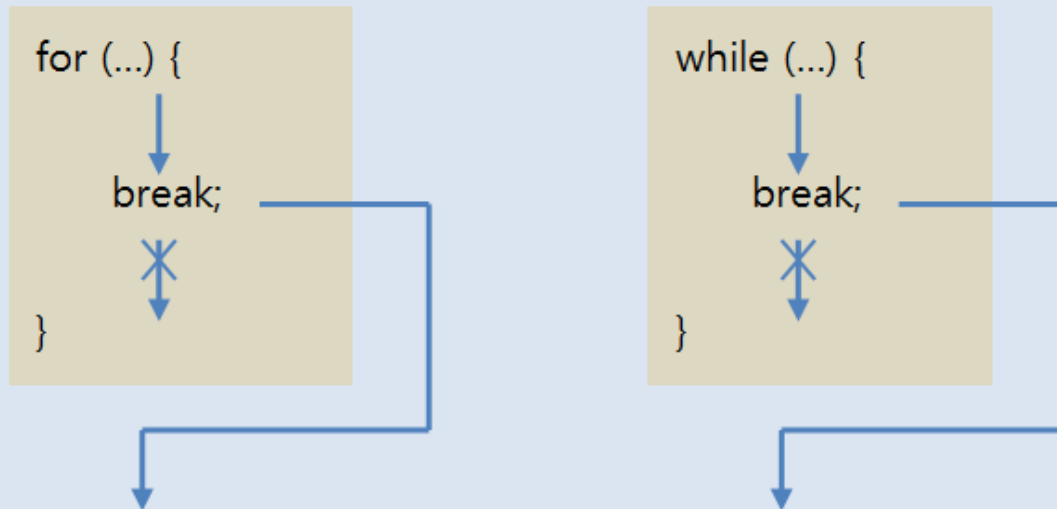
```
DoWhile.java
1 package week5;
2
3 import java.util.Scanner;
4
5 public class DoWhile {
6     public static void main(String[] args) {
7         System.out.println("메시지를 입력하세요");
8         System.out.println("프로그램을 종료하려면 q를 입력하세요");
9
10        Scanner scanData = new Scanner(System.in);
11        String inputStr;
12
13        do {
14            System.out.print(">");
15            inputStr = scanData.nextLine();
16            System.out.println(inputStr);
17        } while (!inputStr.equals("q"));
18
19        System.out.println();
20        System.out.println("프로그램 종료");
21
22        scanData.close();
23    }
24 }
```

```
Console
<terminated> DoWhile [Java Application] C:\Program Files
메시지를 입력하세요
프로그램을 종료하려면 q를 입력하세요
>안녕하세요
안녕하세요
>반갑습니다
반갑습니다
>q
q
프로그램 종료
```


3절. 반복문(for문, while문, do-while문)

❖ break 문

- for문, while문, do-while문 종료 (반복 취소)
- Switch문 종료
- 대개 if문과 같이 사용
 - if문 조건식에 따라 for문과 while문 종료할 때 사용

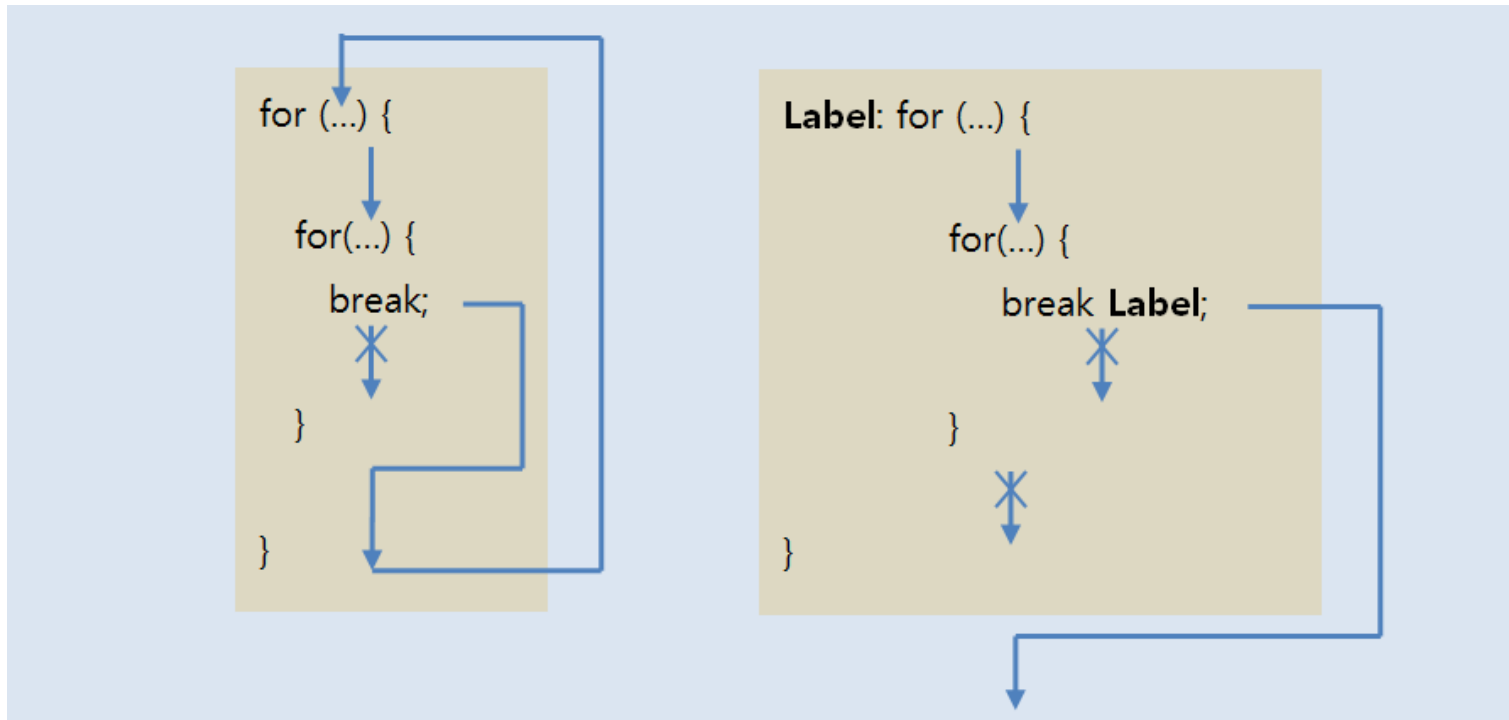


3절. 반복문(for문, while문, do-while문)

❖ break 문

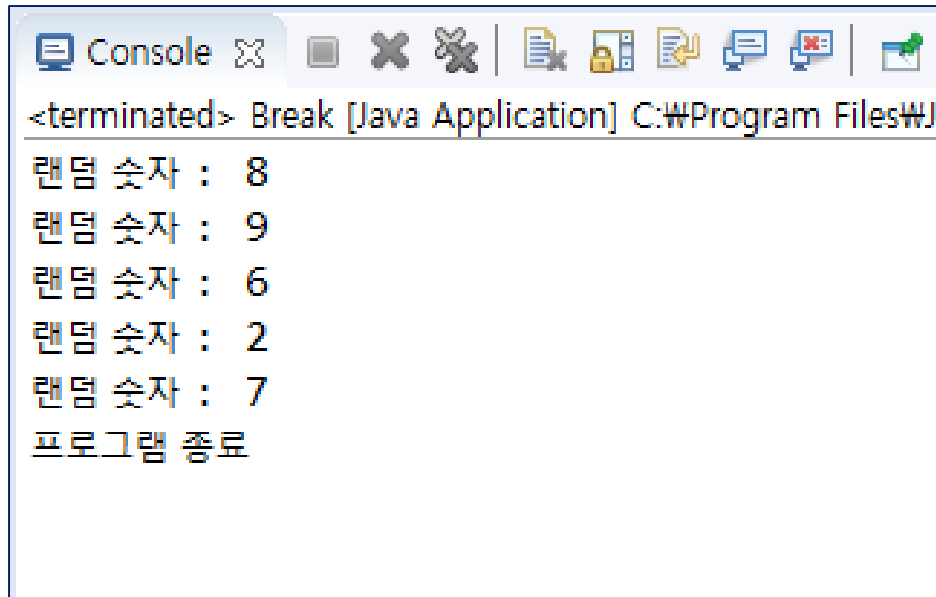
■ 반복문이 중첩된 경우

- 반복문이 중첩되어 있을 경우 `break;` 문은 가장 가까운 반복문만 종료
- 바깥쪽 반복문까지 종료시키려면 반복문에 이름(라벨)을 붙이고, “`break 이름;`” 사용



break 문 실습 (Break.java)

❖ 0 ~ 9까지의 랜덤 숫자 중 7이 나오면 프로그램을 종료



```
<terminated> Break [Java Application] C:\Program Files\J...
랜덤 숫자 : 8
랜덤 숫자 : 9
랜덤 숫자 : 6
랜덤 숫자 : 2
랜덤 숫자 : 7
프로그램 종료
```

break 문 예제 (Break.java)

❖ 0 ~ 9까지의 랜덤 숫자 중 7이 나오면 프로그램을 종료

```
Break.java ✖
1 package week5;
2
3 public class Break {
4     public static void main(String[] args) {
5
6         while (true) {
7             int num = (int) (Math.random() * 10);
8
9             System.out.println("랜덤 숫자 : " + num);
10
11             if (num == 7)
12                 break;
13         }
14
15         System.out.println("프로그램 종료");
16     }
17 }
18
```

break 문 예제 (Break.java)

❖ 0 ~ 9까지의 랜덤 숫자 중 7이 나오면 프로그램을 종료

```
Break.java
1 package week5;
2
3 public class Break {
4     public static void main(String[] args) {
5
6         while (true) {
7             int num = (int) (Math.random() * 10);
8
9             System.out.println("랜덤 숫자 : " + num);
10
11             if (num == 7)
12                 break;
13         }
14
15         System.out.println("프로그램 종료");
16     }
17 }
18
```

Console Break [Java Application] C:\Program Files\J...

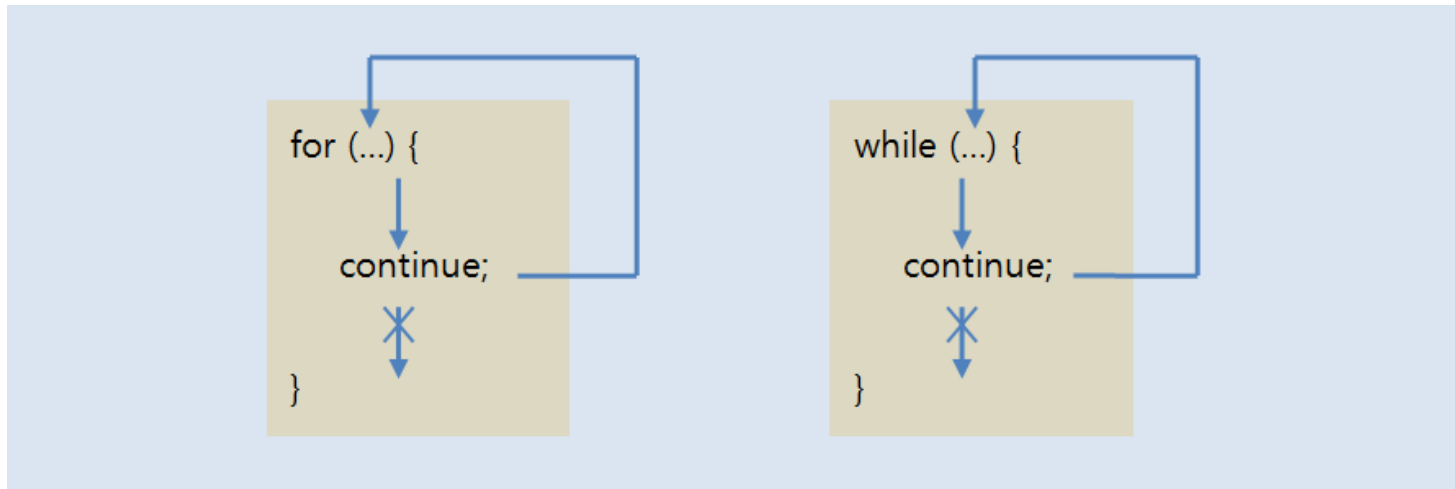
랜덤 숫자 : 8
랜덤 숫자 : 9
랜덤 숫자 : 6
랜덤 숫자 : 2
랜덤 숫자 : 7
프로그램 종료

3절. 반복문(for문, while문, do-while문)

❖ continue 문

■ for문, while문, do-while문에서 사용

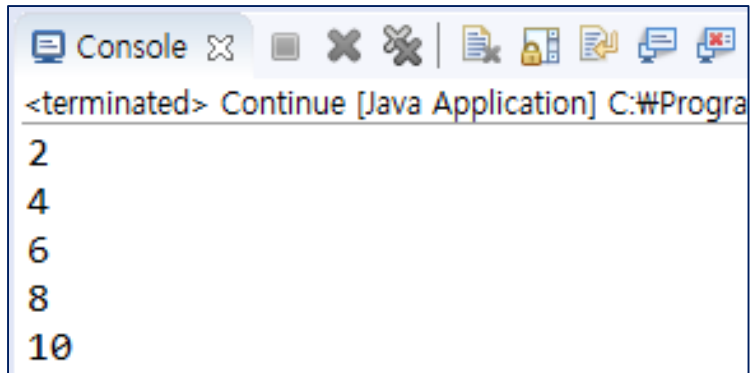
- for문: 증감식으로 이동
- while문, do-while문: 조건식으로 이동



```
for(int i=1; i<=10; i++) {  
    if(i%2 != 0) { ●----- 2로 나눈 나머지가 0 이 아닐 경우  
                        즉 홀수인 경우  
        continue;  
    }  
    System.out.println(i); ●----- 홀수는 실행되지 않는다.  
}
```

continue 문 예제 (Continue.java)

❖ 1 ~ 100까지의 숫자 중 짝수만 출력하는 프로그램



```
<terminated> Continue [Java Application] C:\#Progra
2
4
6
8
10
```

continue 문 예제 (Continue.java)

❖ 1 ~ 100까지의 숫자 중 짝수만 출력하는 프로그램

```
Continue.java ✖
1 package week5;
2
3 public class Continue {
4     public static void main(String[] args) {
5
6         for (int i=1; i<=100; i++) {
7             if (i % 2 != 0) {
8                 continue;
9             }
10
11             System.out.println(i);
12         }
13     }
14 }
15
16
```


continue 문 예제 (Continue.java)

❖ 1 ~ 100까지의 숫자 중 짝수만 출력하는 프로그램

```
Continue.java ✕
1 package week5;
2
3 public class Continue {
4     public static void main(String[] args) {
5
6         for (int i=1; i<=100;i++ ) {
7             if (i % 2 != 0) {
8                 continue;
9             }
10
11             System.out.println(i);
12         }
13     }
14 }
15
16
```

Console ✕

<terminated> Continue [Java Application] C:\#Progra

2
4
6
8
10