

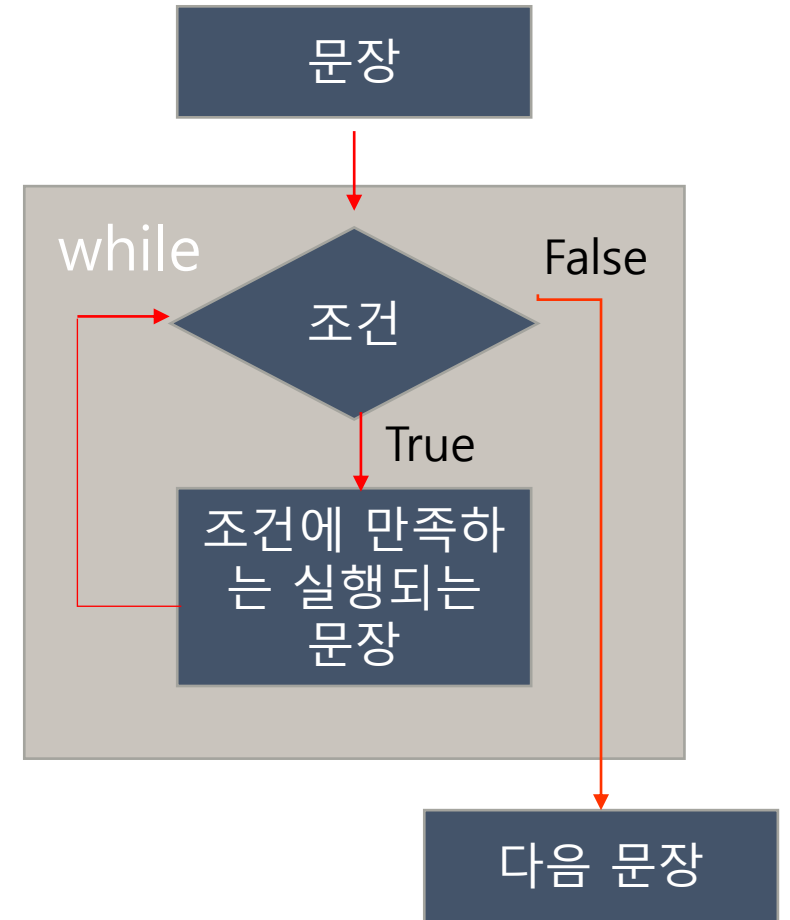


JAVA

while 문

주어진 **조건을 만족하는 경우**에만 특정 문장을 **반복수행** 하도록 하는 제어문
조건식에 **참(true)**을 지정하면 **무한반복(루프)**를 수행한다.

```
while (조건식)
{
    반복할 코드
}
```



while 문

```
1 package controlSet;
2
3 public class Ex05 {
4     public static void main(String[] args) {
5         int num = 0;
6
7         while(num < 5) {
8             System.out.println("반복 내용 : " + num);
9             num++;
10        }
11
12        System.out.println("반복 종료 후 : " + num);
13    }
14 }
15
```

while 문

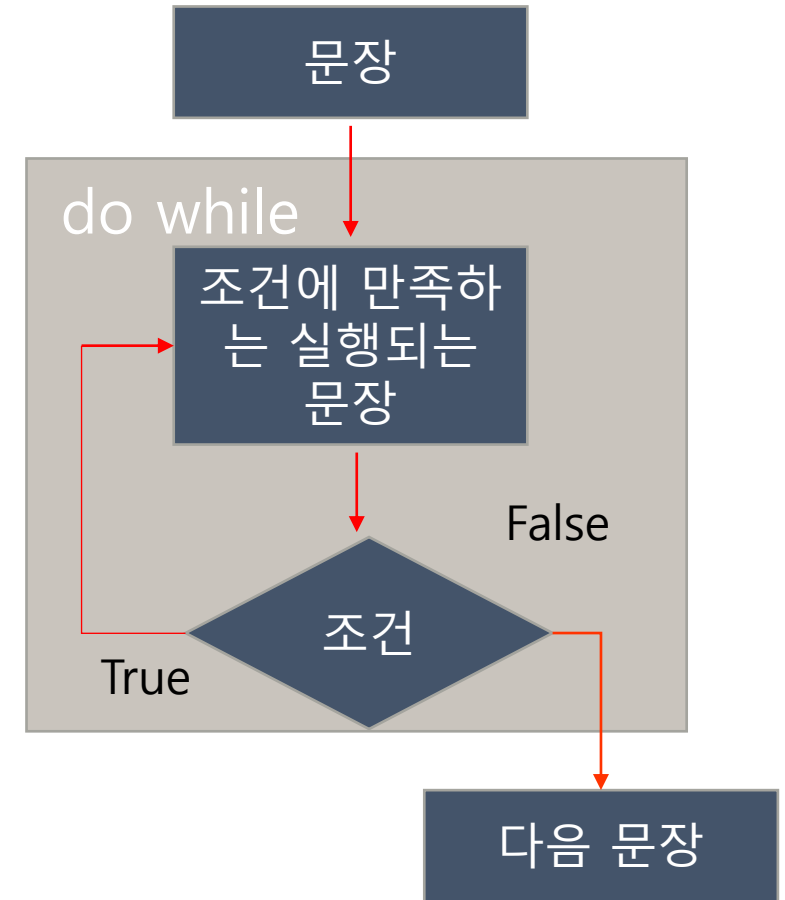
```
1 package controlSet;
2
3 import java.util.Scanner;
4
5 public class Ex06 {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         int num = 1;
9
10        while(num != 0) {
11            System.out.print("입력(0을 입력시 종료) : ");
12            num = sc.nextInt();
13
14            System.out.println("입력받은 값 : " + num);
15        }
16        System.out.println("종료~~");
17
18        sc.close();
19    }
20 }
21
```

먼저 한 번 수행하고, 주어진 조건을 만족하는 경우에만 특정 문장을 반복 수행 하도록 하는 제어문.

조건식에 참(1)을 지정하면 무한 반복(루프)를 수행한다.

do while 문

```
do  
{  
  반복할 코드  
}while(조건식);
```



do while 문

```
1 package controlSet;
2
3 import java.util.Scanner;
4
5 public class Ex07 {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         int num;
9
10        do {
11            System.out.print("양수 입력 : ");
12            num = sc.nextInt();
13        } while(num < 0);
14
15        System.out.println("입력받은 값 : " + num);
16
17        sc.close();
18    }
19 }
20
```



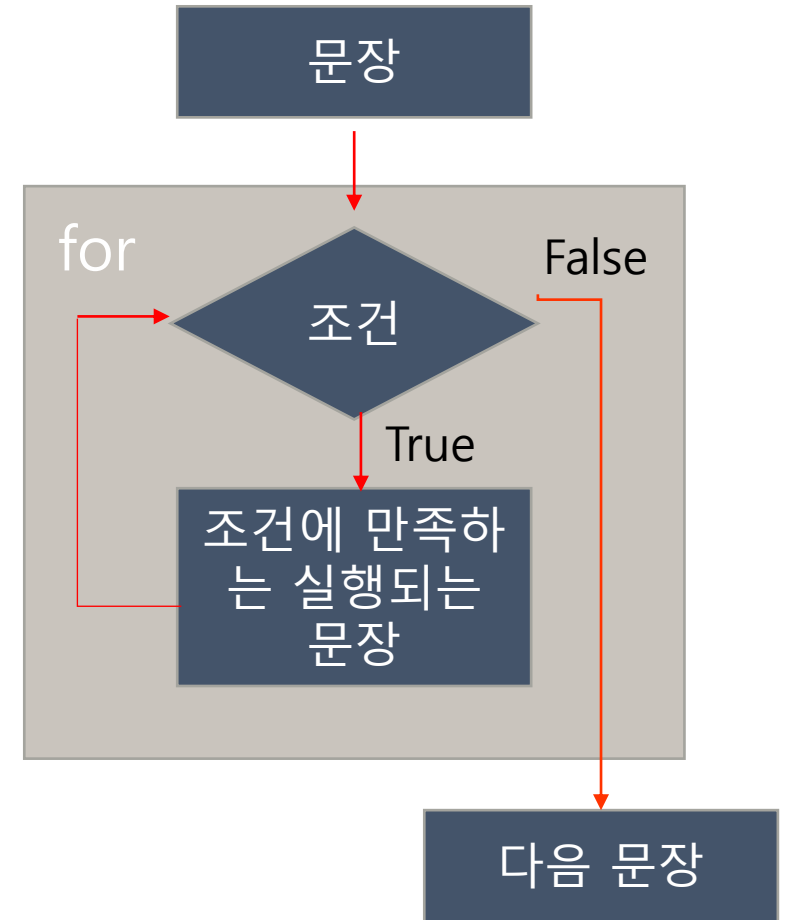
문제

1. 정수를 입력 받아 그 수 만큼의 배수를 출력
2. 정수를 입력 받고, 입력 받은 정수부터 1까지 역순으로 출력
3. 입력된 세 자리 정수를 거꾸로 출력

for 문

주어진 **조건을 만족하는 경우**에만 특정 문장을 **반복수행** 하도록 하는 제어문
조건식에 **참(true)**을 지정하면 **무한반복(루프)**를 수행한다.

```
for (초기식; 조건식; 변화식)
{
    반복할 코드
}
```



for 문

```
1 package controlSet;
2
3 import java.util.Scanner;
4
5 public class Ex08 {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         int num = sc.nextInt();
9
10
11         for(int i = 1; i <= num; i++) {
12             System.out.println("i = " + i);
13         }
14
15
16         sc.close();
17     }
18 }
19
```

보조 제어문

break는 for, while, do while, switch문법에서 **제어 흐름을 벗어나기 위해** 사용한다.

continue는 제어 흐름(반복)을 유지한 상태에서 **코드의 실행만 건너뛰는 역할**을 한다.



break

```
1 package controlSet;
2
3 public class Ex09 {
4     public static void main(String[] args) {
5
6         for(int i = 1; i <= 10; i++) {
7             if(i == 5) {
8                 break;
9             }
10
11             System.out.println("i = " + i);
12         }
13
14     }
15 }
16
```



continue

```
1 package controlSet;
2
3 public class Ex09 {
4     public static void main(String[] args) {
5
6         for(int i = 1; i <= 10; i++) {
7             if(i == 5) {
8                 //break;
9                 continue;
10            }
11
12            System.out.println("i = " + i);
13        }
14
15    }
16 }
17
```

while 문

```
1 package controlSet;
2
3 import java.util.Scanner;
4
5 public class Ex10 {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         int num;
9
10        while(true) {    // 무한 반복
11            System.out.print("입력(0을 입력시 종료) : ");
12            num = sc.nextInt();
13
14            if(num == 0)    break;
15
16            System.out.println("입력받은 값 : " + num);
17
18        }
19
20        sc.close();
21    }
22 }
23
```

문제

1. 1 ~ 1000까지의 합을 구하시오
2. 1 ~ 1000까지의 합을 구하시오(단, 3의 배수는 제외)
3. 1 ~ 1000까지의 합을 구하시오(단, 3의 배수는 제외, 3의 배수이면서 5의 배수는 합에 포함)
4. 입력 받은 값의 약수와 약수 개수를 출력
5. 입력 받은 수가 소수인지 판별하여 출력
6. Factorial을 구현
ex) 입력 : 5 => $5 * 4 * 3 * 2 * 1 = 120$



문제

10. 첫날에는 1원을 예금하고, 다음날에는 전날의 2배의 금액을
예금하는 방식으로 한달(30일) 동안 저축한 금액을 출력

Ex) 1일) 예금액 : 1원, 총액 : 1원

2일) 예금액 : 2원, 총액 : 3원

3일) 예금액 : 4원, 총액 : 7원

...

한달 동안 저축한 금액 : 1073741823원

중첩 반복문

반복문 안에 반복문이 들어가는 형태를 **중첩 루프(다중 루프)**라고 한다.

※ 반복문의 루프 인덱스 변수는 index의 첫 글자를 따서 i를 사용하는데
다중 루프를 사용할 때 변수 이름은 i,j,k처럼 i 부터 알파벳 순서대로 짓는다.

```
for(초기식i; 조건식i; 변화식i){  
    ...  
    for(초기식j; 조건식 j; 변화식j){  
        ...  
    }  
    ...  
}
```


중첩 반복문

```
1 package controlSet;
2
3 public class Ex11 {
4     public static void main(String[] args) {
5         for(int i = 1; i <= 9; i++) {
6             System.out.println("====" + i + "단====");
7
8             for(int j = 1; j <= 9; j++) {
9                 System.out.println(i + " x " + j + " = " + i * j);
10            }
11
12            System.out.println();
13        }
14    }
15 }
16
```

문제

1. 아래와 같이 출력

출력>

찍을 별의 줄 수 : 5

*

**

문제

2. 아래와 같이 출력

출력>

찍을 별의 줄 수 : 5

*

**

문제

3. 아래와 같이 출력

출력>

찍을 별의 줄 수 : 5

*

문제

4. 아래와 같이 출력

단을 입력 (1:모두, 0:종료) : 1

구구단 전체 출력

2 x 1 = 2	3 x 1 = 3	4 x 1 = 4	5 x 1 = 5
2 x 2 = 4	3 x 2 = 6	4 x 2 = 8	5 x 2 = 10
2 x 3 = 6	3 x 3 = 9	4 x 3 = 12	5 x 3 = 15
2 x 4 = 8	3 x 4 = 12	4 x 4 = 16	5 x 4 = 20
2 x 5 = 10	3 x 5 = 15	4 x 5 = 20	5 x 5 = 25
2 x 6 = 12	3 x 6 = 18	4 x 6 = 24	5 x 6 = 30
2 x 7 = 14	3 x 7 = 21	4 x 7 = 28	5 x 7 = 35
2 x 8 = 16	3 x 8 = 24	4 x 8 = 32	5 x 8 = 40
2 x 9 = 18	3 x 9 = 27	4 x 9 = 36	5 x 9 = 45
6 x 1 = 6	7 x 1 = 7	8 x 1 = 8	9 x 1 = 9
6 x 2 = 12	7 x 2 = 14	8 x 2 = 16	9 x 2 = 18
6 x 3 = 18	7 x 3 = 21	8 x 3 = 24	9 x 3 = 27
6 x 4 = 24	7 x 4 = 28	8 x 4 = 32	9 x 4 = 36
6 x 5 = 30	7 x 5 = 35	8 x 5 = 40	9 x 5 = 45
6 x 6 = 36	7 x 6 = 42	8 x 6 = 48	9 x 6 = 54
6 x 7 = 42	7 x 7 = 49	8 x 7 = 56	9 x 7 = 63
6 x 8 = 48	7 x 8 = 56	8 x 8 = 64	9 x 8 = 72
6 x 9 = 54	7 x 9 = 63	8 x 9 = 72	9 x 9 = 81