

자바 기초 문법

반복문

while 반복문

🚦 while 반복문

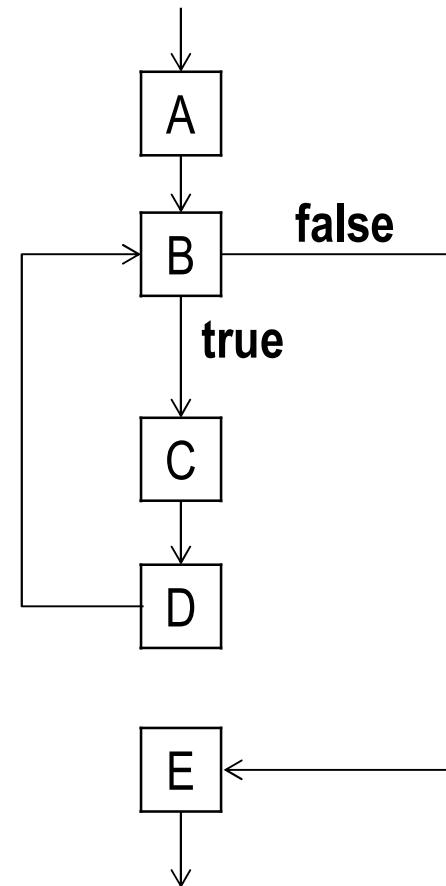
- 조건식이 참인 동안 반복적으로 실행되는 문장

```
while(조건식) {  
    조건식이 참일 동안 계속 반복적으로 실행될 문장(들)  
}
```

```
public class Test {  
    public static void main(String[] args) {  
        int    i=1;  
        int    sum=0;  
        while (i<=5){  
            sum=sum + i ;  
            i = i + 1 ;  
        }  
        System.out.println(sum);  
    }  
}
```

while 반복문 실행 흐름

```
A;  
while(B) {  
    C;  
    D;  
}  
E;
```



for 반복문

for 반복문

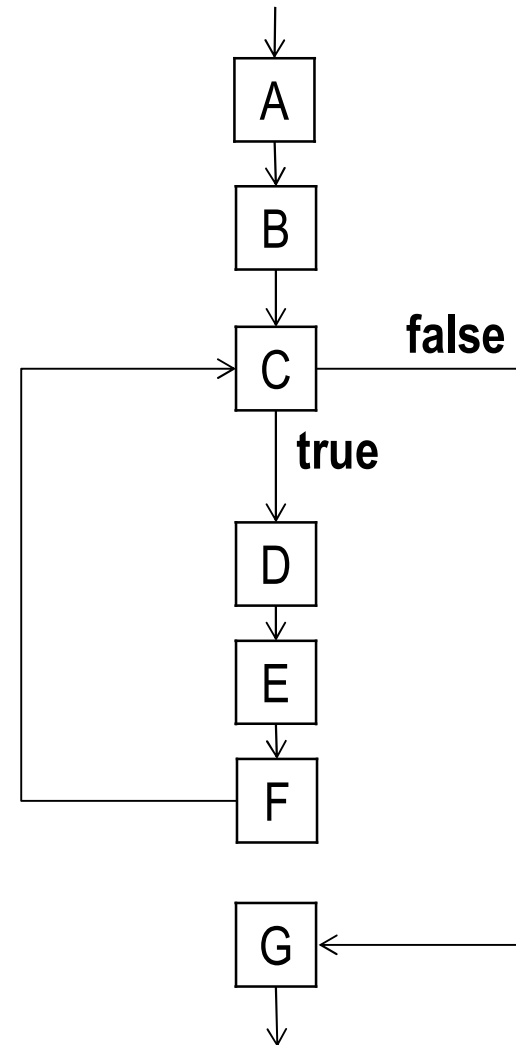
- 조건식이 참인 동안 반복적으로 실행되는 문장

```
for(최초 1회 실행될 문장 ; 조건식 ; 블록 실행 후 실행될 문장)
{
    조건식이 참일 동안 계속 반복적으로 실행될 문장(들)
}
```

```
public class Test {
    public static void main(String[] args) {
        int    sum=0;
        for (int i = 1; i <= 5; i++) {
            sum=sum + i;
        }
        System.out.println(sum);
    }
}
```

for 반복문 실행 흐름

```
A;  
for(B;C;F)  
{  
    D;  
    E;  
}  
G;
```



반복 탈출문: break

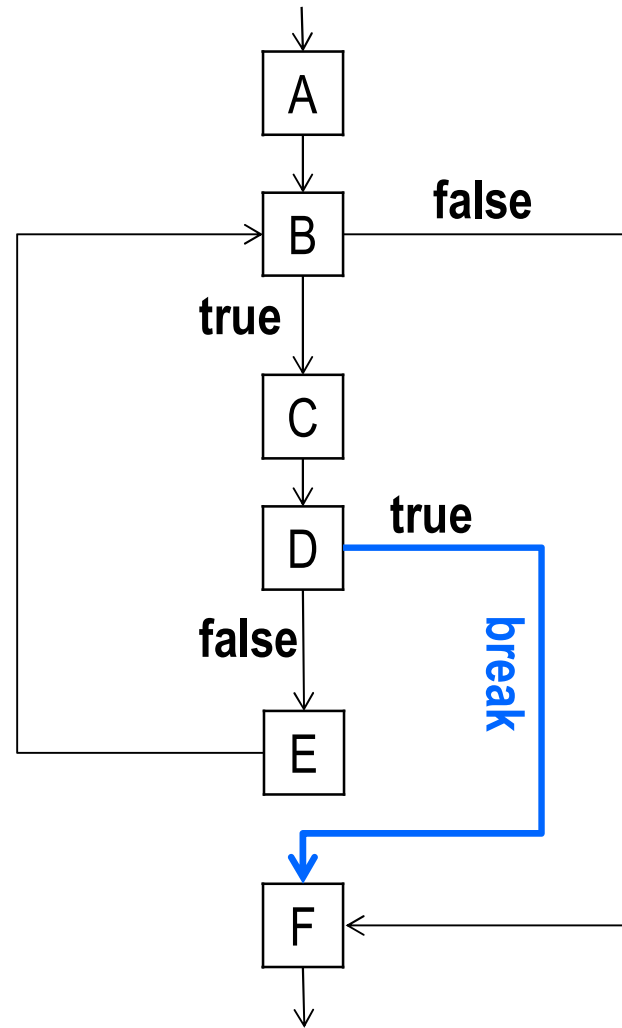
break

- break 문이 실행되면 이 break문을 포함하는 반복문을 탈출

```
public class Test {  
    public static void main(String[] args) {  
        int i=1;  
        int sum=0;  
        while(true){  
            if(i>5) break;  
            sum=sum + i;  
            i+=1;  
        }  
        System.out.println(sum);  
    }  
}
```

break 문 실행 흐름

```
A;  
while(B) {  
    C;  
    if(D) break;  
    E;  
}  
F;
```



반복 계속문: continue

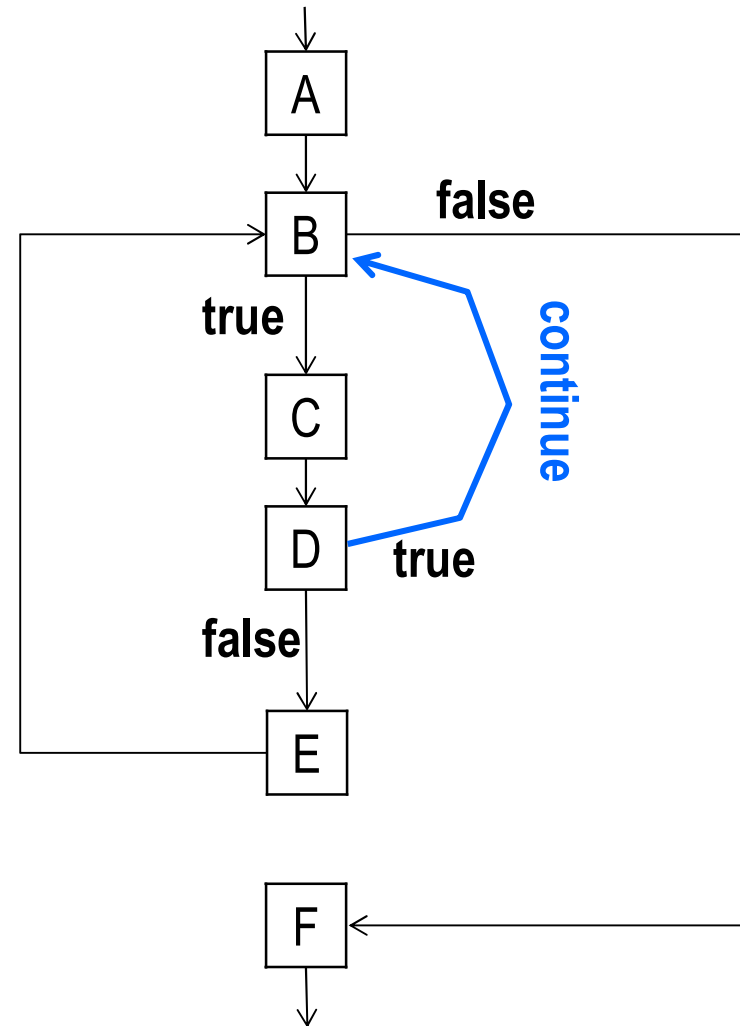
continue

- continue 문이 실행되면 이 continue문 이후 반복문을 무시하고 반복 절차를 다시 계속

```
public class Test {  
    public static void main(String[] args) {  
        int sum=0;  
        for (int i = 1; i <= 100; i++) {  
            if(i%2==1) continue;  
            sum=sum + i;  
        }  
        System.out.println(sum);  
    }  
}
```

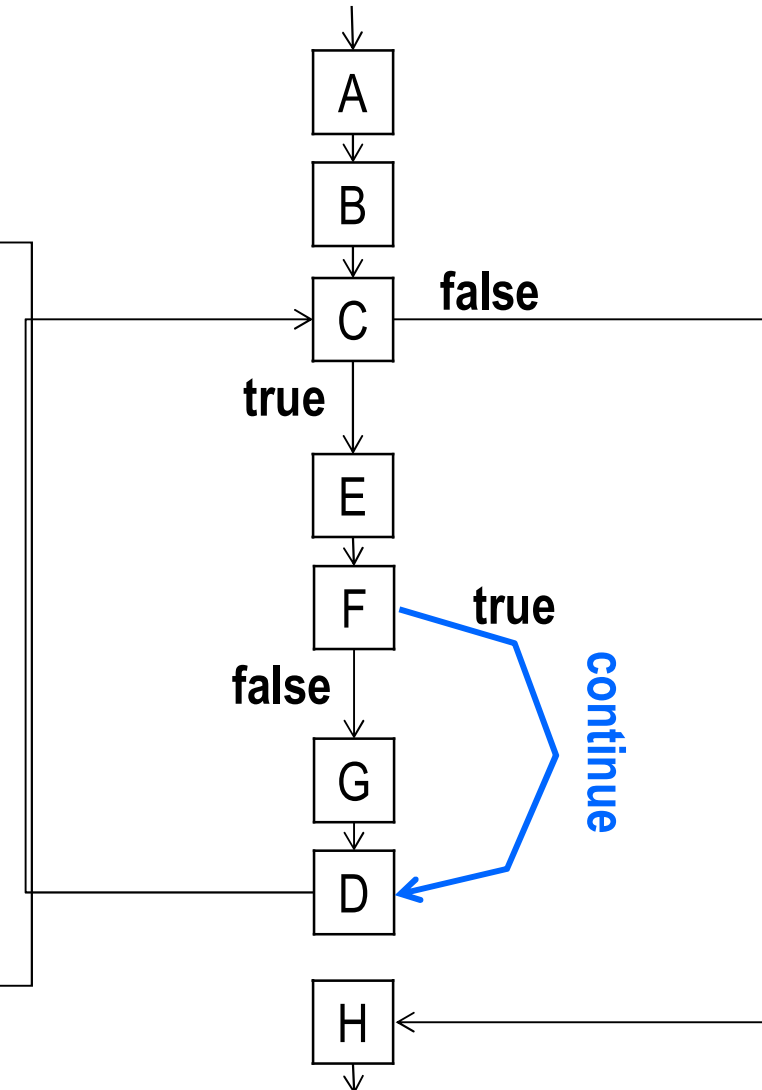

while 반복문의 continue 문 실행 흐름

```
A;  
while(B) {  
    C;  
    if(D) continue;  
    E;  
}  
F;
```



for 반복문의 continue 문 실행 흐름

```
A;  
for(B;C;D) {  
    E;  
    if(F) continue;  
    G;  
}  
H;
```



무한루프(infinite loop)

```
public class Test {  
    public static void main(String[] args) {  
        while(true) {  
            System.out.println("Hello");  
        }  
    }  
}
```

```
public class Test {  
    public static void main(String[] args) {  
        for (; true ;) {  
            System.out.println("Hello");  
        }  
    }  
}
```

반복문 - 실습 A

- ✚ 1부터 100까지의 모든 홀수들의 합을 계산하여 출력하는 자바 프로그램을 작성하시오
 - while문을 사용할 것

반복문 - 실습 B

✚ 1부터 10까지의 곱을 계산하여 출력하는 자바프로그램을 작성
하십시오

- for문을 사용할 것

반복문 - 실습 C

✚ 1부터 n 까지의 합을 계산하여 출력하는 자바프로그램을 작성하시오

- while문을 사용하여 구현하시오
- 정수 n 의 값은 프로그램 내에서 그 값을 입력해 두시오
 - ◆ n 값은 1, 5, 10, 100에 대해 테스트하시오

반복문 - 실습 D

✚ m부터 n까지의 합을 계산하여 출력하는 자바프로그램을 작성 하시오

- for문을 사용하여 구현하시오
- m, n은 프로그램 내에 그 값을 입력해 두시오
 - ◆ 다음 m, n값들에 대해 테스트하시오
 - m=1, n=100
 - m=-5, n=12
 - m=13, n=45

반복문 - 실습 E

- ✚ $2n+1$ ($n=-3 \sim 3$)로 만들어지는 수열을 출력하시오
 - while문을 사용하여 구현하시오

반복문 - 실습 F

- ✚ $2k-1$ ($k \geq 3$)로 만들어지는 수열 n 개를 출력하시오
 - for문을 사용하여 구현하시오
 - n 의 값은 프로그램 내에서 그 값을 입력해 두시오
 - ◆ n 값은 1, 5, 10, 100에 대해 테스트하시오

반복문 - 실습 G

✚ n개의 m의 배수들의 합을 계산하여 출력하는 자바프로그램을 작성하시오

- 예

- ◆ $n=5, m=3 \rightarrow 3+6+9+12+15 \rightarrow 45$ 를 출력

- n, m은 프로그램 내에서 그 값을 입력해 두시오

- 다음 m, n값들에 대해 테스트하시오

- ◆ $m=4, n=9$

- ◆ $m=5, n=12$

- ◆ $m=10, n=13$

반복문 - 실습 H

✚ n개의 피보나치 수열을 출력하는 자바프로그램을 작성하시오

- 피보나치 수열은 다음과 같이 정의된다

- ◆ 0번째 수는 0이고, 1번째 수는 1이며, 2번째 수부터는 이전 두 수의 합으로 정의한다

- ◆ 참조: <http://ko.wikipedia.org/wiki/피보나치수>

- n은 프로그램 내에 그 값을 입력해 두시오

- ◆ 다음 n 값들에 대해 테스트하시오

- n=1, 2, 3, 5, 10

- n의 값에 따른 피보나치 수열의 예

n=1 → 0

n=2 → 0, 1

n=3 → 0, 1, 1

n=4 → 0, 1, 1, 2

n=5 → 0, 1, 1, 2, 3

n=6 → 0, 1, 1, 2, 3, 5

n=7 → 0, 1, 1, 2, 3, 5, 8

반복문 - 실습 I

✚ $n!$ 을 계산하여 출력하는 자바프로그램을 작성하시오

● 예

◆ $n=5 \rightarrow 5!=5*4*3*2*1 \rightarrow 120$ 을 출력

● n 은 프로그램 내에서 그 값을 입력해 두시오.

◆ 다음 n 값에 대해 테스트하시오

- $n=0, 1, 3, 5, 10$

반복문 - 실습 J

✚ n^m 을 계산하여 출력하는 자바프로그램을 작성하시오

● 예

◆ $n=2, m=5 \rightarrow 2^5 = 32$

● n, m 은 프로그램 내에 그 값을 입력해 두시오.

◆ 다음 n, m 값에 대해 테스트하시오

- $n=2, m=5$

- $n=3, m=0$

- $n=4, m=3$

● Math.power()함수를 사용하지 말 것

반복문 - 실습 K

✚ 다음 예와 같이 구구단을 출력하는 자바프로그램을 작성하시오

- 중첩 for 문 (for문 안에 또 다른 for문을 갖는 구조)을 사용할 것
- 프로그램 실행 결과는 아래 예 같이 각 구구단 식을 하나의 라인에 분리하여 출력되도록 할 것

1x1=1

1x2=2

...

1x9=9

2x1=2

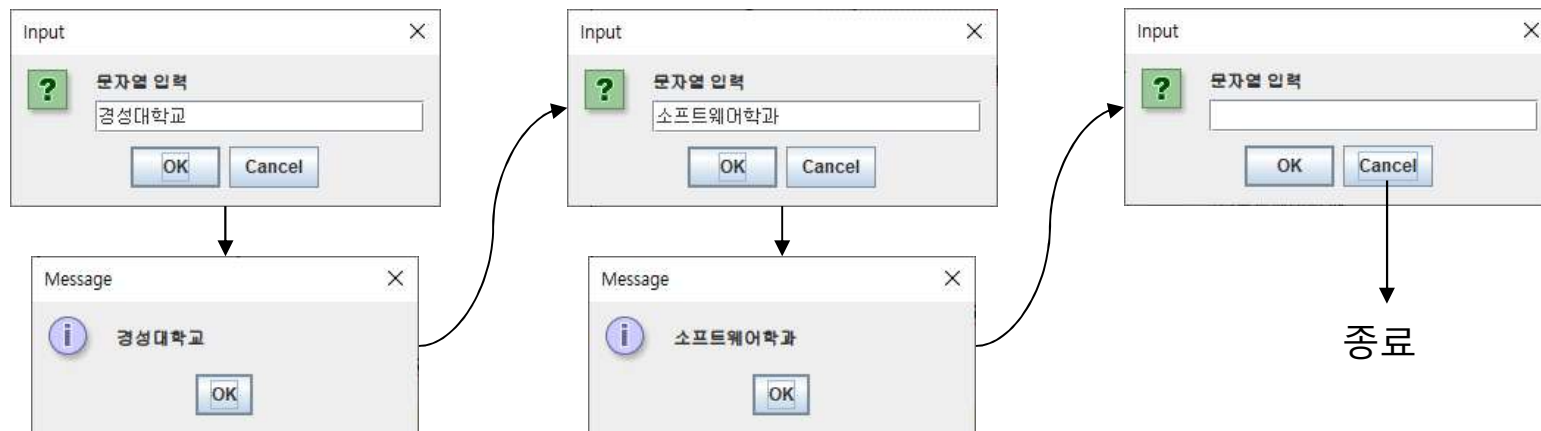
...

9x9=81

JOptionPane: showInputDialog(), showMessageDialog()

- 오른쪽 코드는 아래 실행 예시와 같이 입력창에 입력한 문자열을 메시지창에 출력하는 작업을 반복하는 코드이다
- CANCEL 버튼을 클릭하면 종료된다.

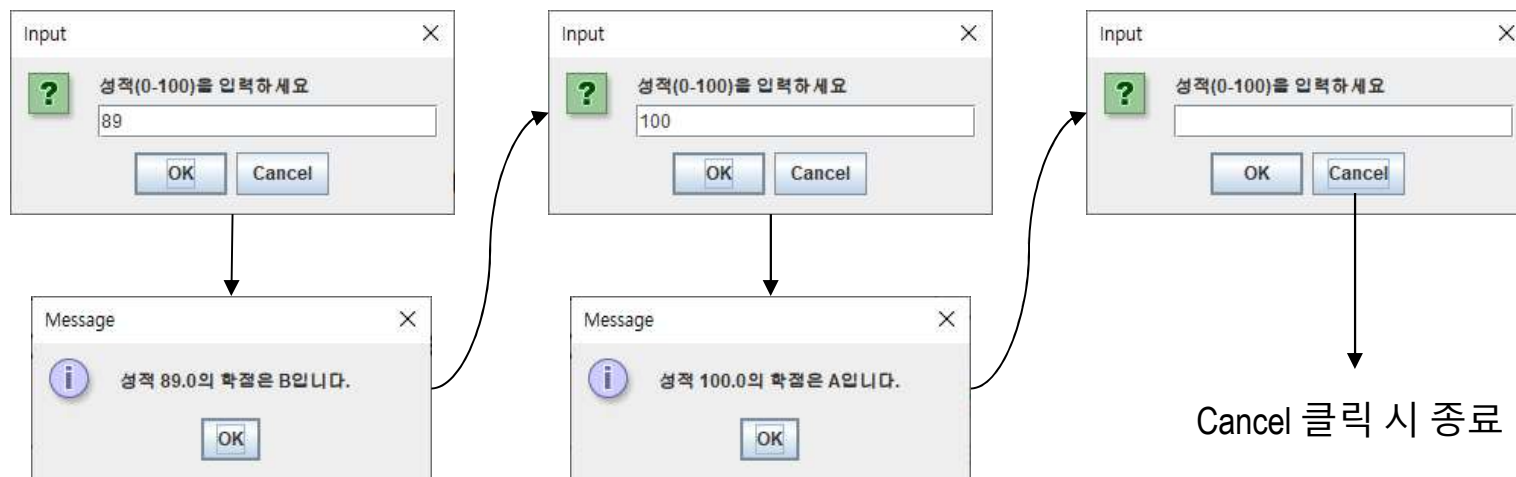
```
import javax.swing.JOptionPane;  
public class Test {  
    public static void main(String[] args) {  
        while(true) {  
            String s = JOptionPane.showInputDialog("문자열 입력");  
            if(s==null) break;  
            JOptionPane.showMessageDialog(null, s);  
        }  
    }  
}
```



반복문 - 실습 L

다음 실행 예시와 같이 Cancel 버튼 클릭하기 전까지 성적 값 (0~100)을 입력 받아 대응하는 학점을 출력하는 작업을 반복하는 자바 코드를 작성하시오


- 학점 기준: 90~100 → A, 80~89 → B, 70~79 → C, 60~69 → D, 0~59: F
- JOptionPane.showInputDialog(...)에서 Cancel 버튼 클릭 시 null 반환됨



References

 <http://docs.oracle.com/javase/7/docs/api/>

 <https://docs.oracle.com/javase/tutorial/java/>

 김윤명. (2008). 뇌를 자극하는 Java 프로그래밍. 한빛미디어.

 남궁성. 자바의 정석. 도우출판.

 황기태, 김효수 (2015). 명품 Java Programming. 생능출판사.