

4강 JAVA



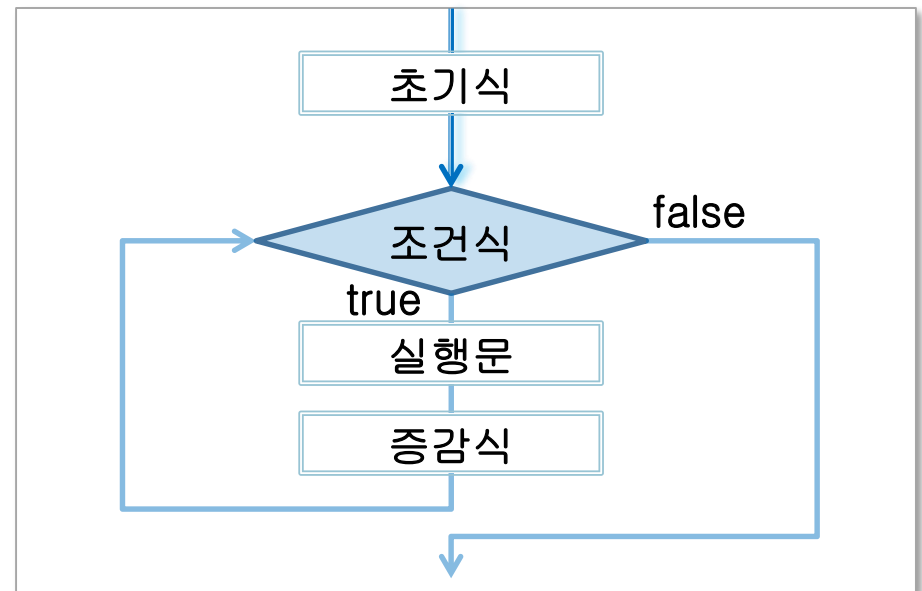
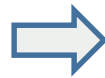
반복문

for 문

❖ for문

- 지정된 위치에 식을 구성하여 가독성 높음
- 정해진 수 만큼 반복
- 식은 일부 또는 전부를 생략 가능

```
for(초기식; 조건식; 증감식){  
    실행문;  
    실행문;  
}
```



제어문 – for

```
public class Ex01 {  
    public static void main(String[] args) {  
        for(int i = 0; i < 5; i++)  
        {  
            System.out.print(i + " ");  
        }  
        for(int i = 5; i > 0; i--)  
        {  
            System.out.print(i + " ");  
        }  
    }  
}
```

제어문 – for

```
public class Ex02 {  
    public static void main(String[] args) {  
        int i, sum = 0;  
        for(i = 1; i <= 10; i+=2) {  
            sum += i;  
        }  
        System.out.println("합계: "+ sum);  
    }  
}
```

제어문 - for

```
public class Ex03 {  
    public static void main(String[] args) {  
        int i, sum;  
        i = 1; sum = 0;  
        sum = sum + i;    i = i+1;  
        sum = sum + i;    i = i+1;  
        sum = sum + i;    i = i+1;  
        sum = sum + i;    i = i+1;  
        sum = sum + i;    i = i+1;  
        sum = sum + i;    i = i+1;  
        sum = sum + i;    i = i+1;  
        sum = sum + i;    i = i+1;  
        sum = sum + i;  
        System.out.println(sum);  
    }  
}
```

제어문 – for

```
public class Ex04 {  
    public static void main(String[] args) {  
        int i, evenSum = 0, oddSum = 0;  
        for(i = 1; i <= 10; i++) {  
            if(i % 2 == 0)  
                evenSum += i;  
            else  
                oddSum += i;  
        }  
        System.out.println("1~10 짝수 합: " + evenSum);  
        System.out.println("1~10 홀수 합: " + oddSum);  
    }  
}
```

제어문 – for

```
public class Ex05 {  
    public static void main(String[] args) {  
        Scanner input = new Scanner(System.in);  
        int repet;  
        System.out.print("반복 횟수 : ");  
        repet = input.nextInt();  
        for(int i = 0; i < repet; i++){  
            System.out.println(i+1+"번 반복");  
        }  
    }  
}
```


제어문 – for

```
public class Ex06 {  
    public static void main(String[] args) {  
        int i = 1, sum = 0;  
        for( ; i <= 10; ){  
            sum += i;  
            i++;  
        }  
        System.out.println("합계: " + sum);  
    }  
}
```

이중 반복문

제어문 - for

```
public class Ex07 {  
    public static void main(String[] args) {  
        int i, j, sum = 0;  
        for(i = 1; i <= 10; i++) {  
            for(j = 1; j <= 10; j++)  
                sum += 1;  
        }  
        System.out.println("sum: " + sum);  
    }  
}
```

제어문 – for

```
public class Ex08 {  
    public static void main(String[] args) {  
        int i, j;  
        for(i = 1; i <= 5; i++) {  
            for(j = 1; j <= 5; j++)  
                System.out.println("i: "+i+", j: "+j);  
        }  
    }  
}
```

제어문 – for

```
public class Ex09 {  
    public static void main(String[] args) {  
        int i, j, line, star, space;  
        line = 5;  
        star = 1;  
        space = line-1;  
        for(i = 0; i < line; i++) {  
            for(j = 0; j < space; j++)  
                System.out.print(" ");  
            for(j = 0; j < star; j++)  
                System.out.print("*");  
            System.out.println();  
            star+=2;  
            space--;  
        }  
    }  
}
```

while 문

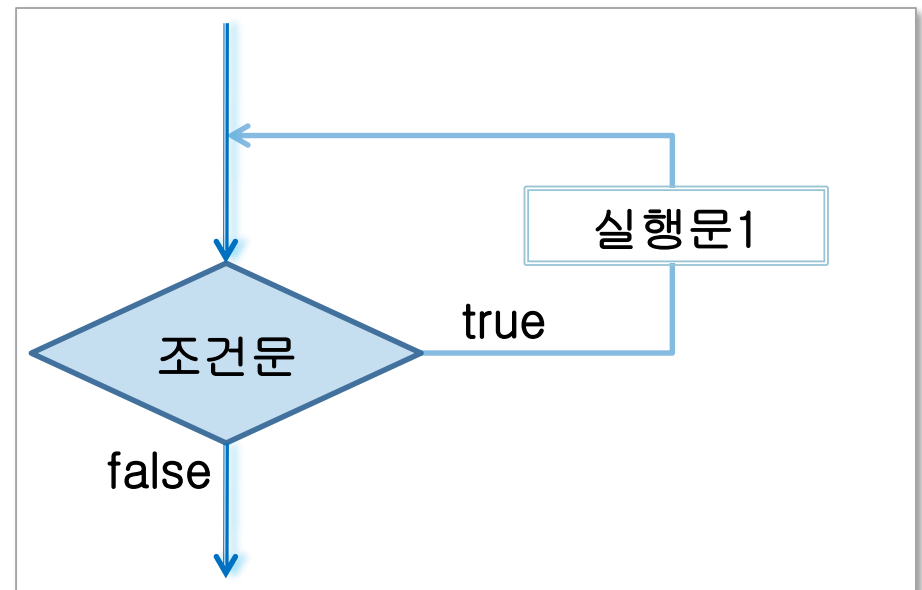
❖ while문

- 사용자가 원하는 만큼 명령들을 여러 번 실행하는 제어구조.
- 특정 조건이 만족되는 동안 명령을 계속적으로 실행한다.

❖ while문의 문법

- 식이 참일 경우에만 루프 처리를 반복한다.

```
while( 조건문 )  
{  
    실행문1;  
}
```



Loop – while

```
public static void main(String[] args) {  
    int i = 1, sum = 0;  
  
    while(i <= 10){  
        sum+=i;  
        i++;  
    }  
    System.out.println("1~10까지의 합 : "+sum);  
}
```

Loop – while

```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
    int i = 0, sum = 0;  
    boolean loopFlag = true;  
    while(loopFlag) {  
        System.out.print("수입력(1~100) : ");  
        i = input.nextInt();  
        if(i >= 1 && i <= 100)  
            loopFlag = false;  
        else  
            System.out.println("잘못 입력, 다시");  
    }  
    for(int j = 1; j <= i; j++)  
        sum += j;  
    System.out.println("입력 받은 수까지의 합 : "+sum);  
}
```


Loop – while

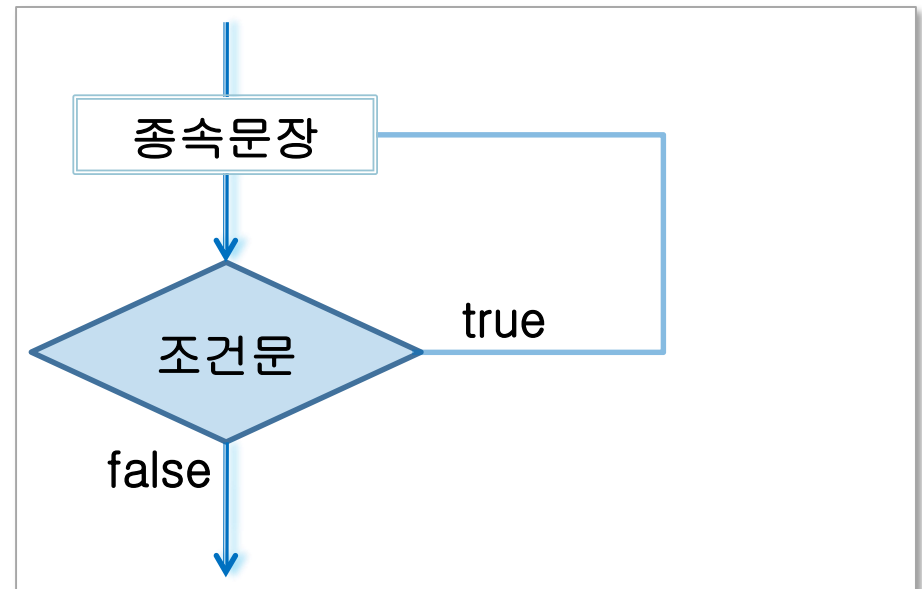
```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
    int i,tmp;  
    System.out.print("수 입력 : ");  
    i = input.nextInt();  
    while(true) {  
        tmp = i % 10;  
        i = i / 10;  
        System.out.print(tmp+" ");  
        if(i == 0)  
            break;  
    }  
}
```

do while 문

❖ 정의

- 다음의 종속 문장을 먼저 실행한 후에 조건식을 평가.

```
do  
{  
    실행문1;  
} while( 조건문 );
```



제어문 (do while문)

```
public static void main(String[] args) {  
    int su = 0, sum = 0;  
    do {  
        su++;  
        sum += su;  
    }while(su <= 9);  
    System.out.println("sum = "+ sum);  
}
```

기타 제어문 [반복문의 흐름제어]

❖ break문

- 반복문이 실행되고 있는 시점에서 가장 근접한 반복문(루프) 탈출
- 여러 개의 루프가 중첩된 경우 현재 위치한 루프 하나만 탈출

❖ continue문

- 반복도중 처리를 중단하고 반복문의 시작 위치로 이동
- 정밀한 제어 구조를 만들 때 사용

continue

```
public static void main(String[] args) {  
    int i = 0;  
    while(i < 5) {  
        i++;  
        if(i == 3)  
            continue;  
        System.out.println("i : " + i);  
    }  
}
```

break

```
public static void main(String[] args) {  
    int i = 0;  
    while(i < 5){  
        i++;  
        if(i == 3){  
            System.out.println("i : "+i+" => break");  
            break;  
        }  
        System.out.println("i : " + i);  
    }  
}
```

boolean

```
public static void main(String[] args) {  
    int i = 0;  
    boolean bool = true;  
    while(bool){  
        i++;  
        if(i == 3){  
            bool = false;  
        }  
        System.out.println("i : " + i);  
    }  
    System.out.println("다음 문장 실행!!");  
}
```

boolean

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


예제 [문자열 길이]

```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
    String n1,n2;  
    System.out.print("첫번째 문자열 입력 : ");  
    n1 = input.next();  
    System.out.print("두번째 문자열 입력 : ");  
    n2 = input.next();  
    System.out.println("n1 길이: "+n1.Length());  
    System.out.println("n2 길이: "+n2.Length());  
}
```

예제 [문자열 비교]

```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
    String n1,n2;  
    System.out.print("첫번째 문자열 입력 : ");  
    n1 = input.next();  
    System.out.print("두번째 문자열 입력 : ");  
    n2 = input.next();  
    System.out.println("n1==n2 : "+(n1 == n2));  
    System.out.println("n1.equals(n2) : "+n1.equals(n2));  
}
```

주의

```
public static void main(String[] args) {  
    int i = 0;  
    String s = null;  
    System.out.println(i);  
    System.out.println(s);  
}
```

Random

```
public static void main(String[] args) {  
    System.out.println(Math.random());  
    System.out.println(Math.random());  
    System.out.println(Math.random());  
}
```

Random

```
public static void main(String[] args) {  
    int begin, end;  
    begin = 1;  
    end = 10;  
    for(int i = 0; i < 5; i++)  
        System.out.println((int)(Math.random() * (end-begin+1) + begin));  
}
```