

Chapter

04

조건문과 반복문



04-1. 조건문 : if문, switch문

혼자 공부하는 자바 (신용권 저)

- 0. 시작하기 전에
- 1. if문
- 2. if-else문
- 3. if-else if-else문
- 4. switch문
- 5. 키워드로 끝내는 핵심 포인트



0. 시작하기 전에

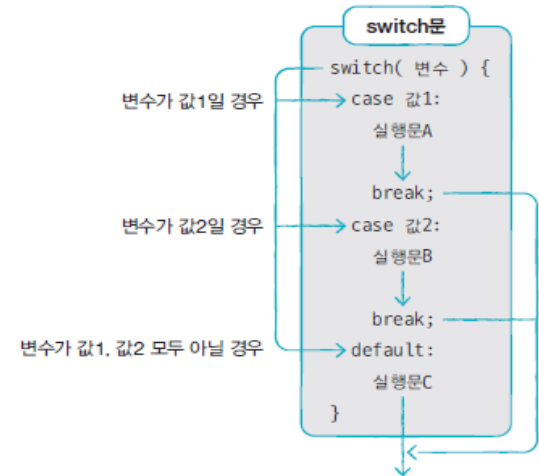
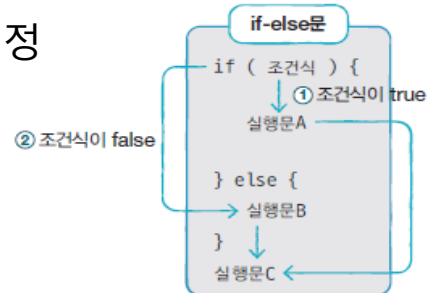
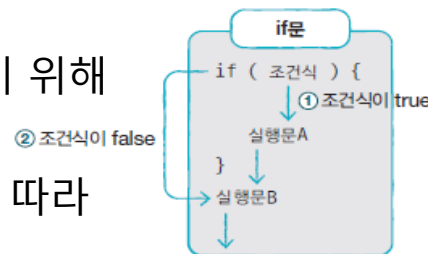
[핵심 키워드] : if문, if-else문, if-else if-else문, switch문

[핵심 포인트]

- 자바 프로그램은 main() 메소드의 시작 중괄호 {에서 끝 중괄호 }까지 위에서부터 아래로 실행하는 흐름을 가지고 있다. 이러한 실행 흐름을 개발자가 원하는 방향으로 바꿀 수 있도록 해주는 것을 흐름 제어문 혹은 제어문이라고 한다.
- 제어문의 종류에는 조건문과 반복문이 있다.

❖ 조건문

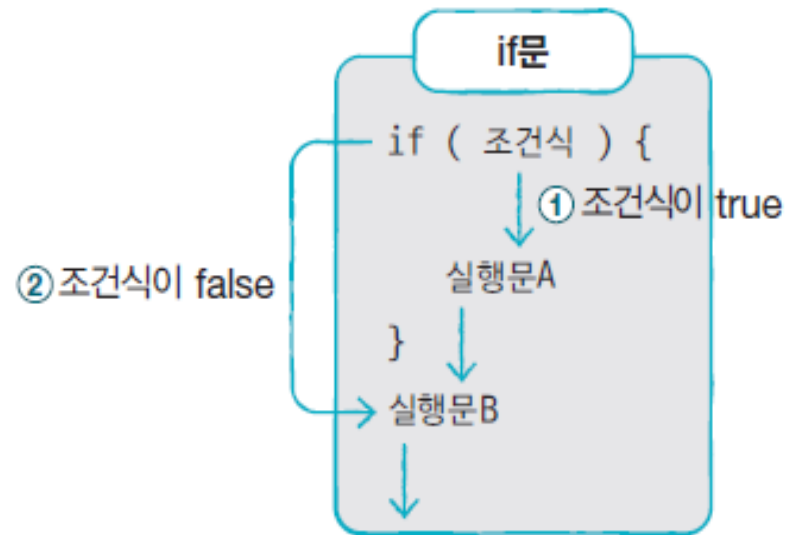
- 조건식에 따라 다른 실행문을 실행하기 위해 사용
- if문** : 조건식 결과의 true, false 여부에 따라 실행문 결정
- switch문** : 변수의 값에 따라 실행문 결정



1. if문

❖ if문

- 조건식 결과에 따라 블록 실행 여부가 결정
- 조건식에 올 수 있는 요소
 - true / false 값을 산출하는 연산식
 - boolean 타입 변수
- 중괄호 블록은 조건식이 true가 될 때 실행
 - 실행할 문장 하나뿐인 경우 생략 가능



```
if ( 조건식 ) {  
    실행문;  
    실행문;  
    ...  
}
```

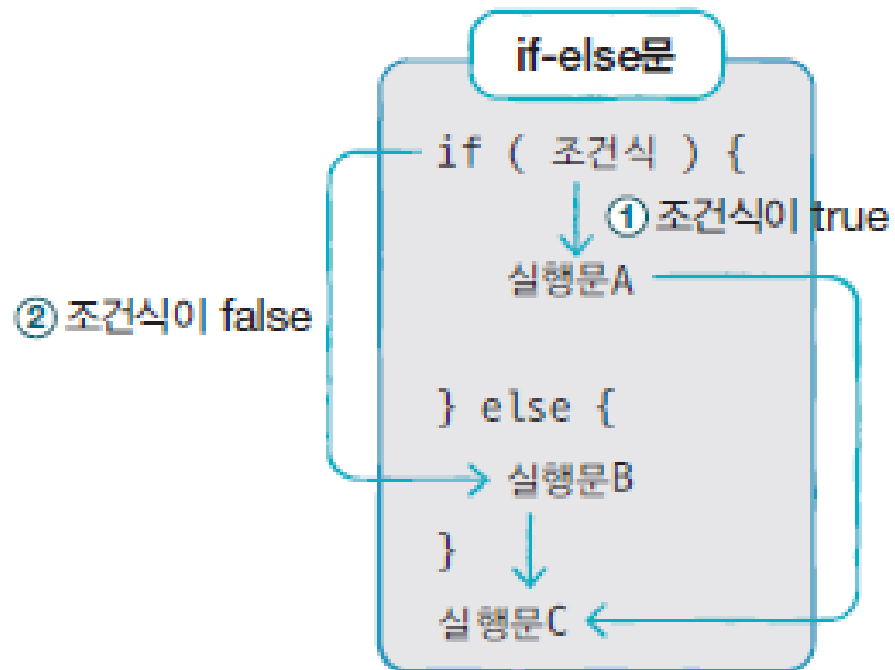
```
if ( 조건식 )  
    실행문;
```



2. if-else문

❖ if-else문

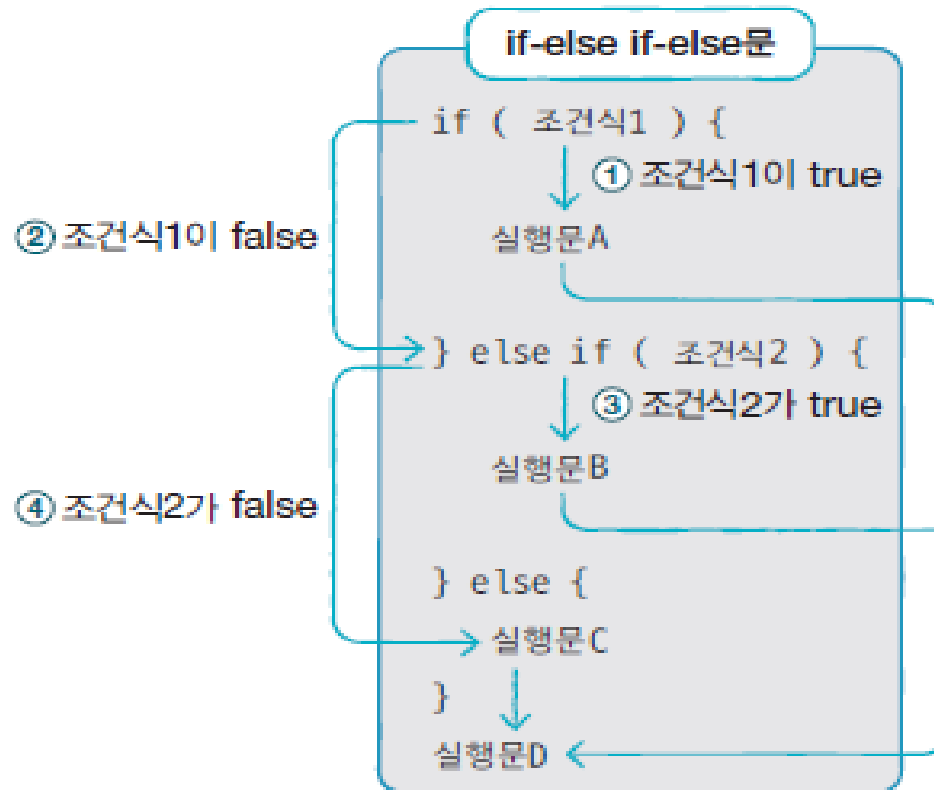
- if문을 else 블록과 함께 사용
- 조건식의 결과에 따라 실행 블록 선택
 - if문 조건식 true이면 if문 블록 실행
 - if문 조건식 false이면 else 블록 실행



3. if-else if-else문

❖ if-else if-else문

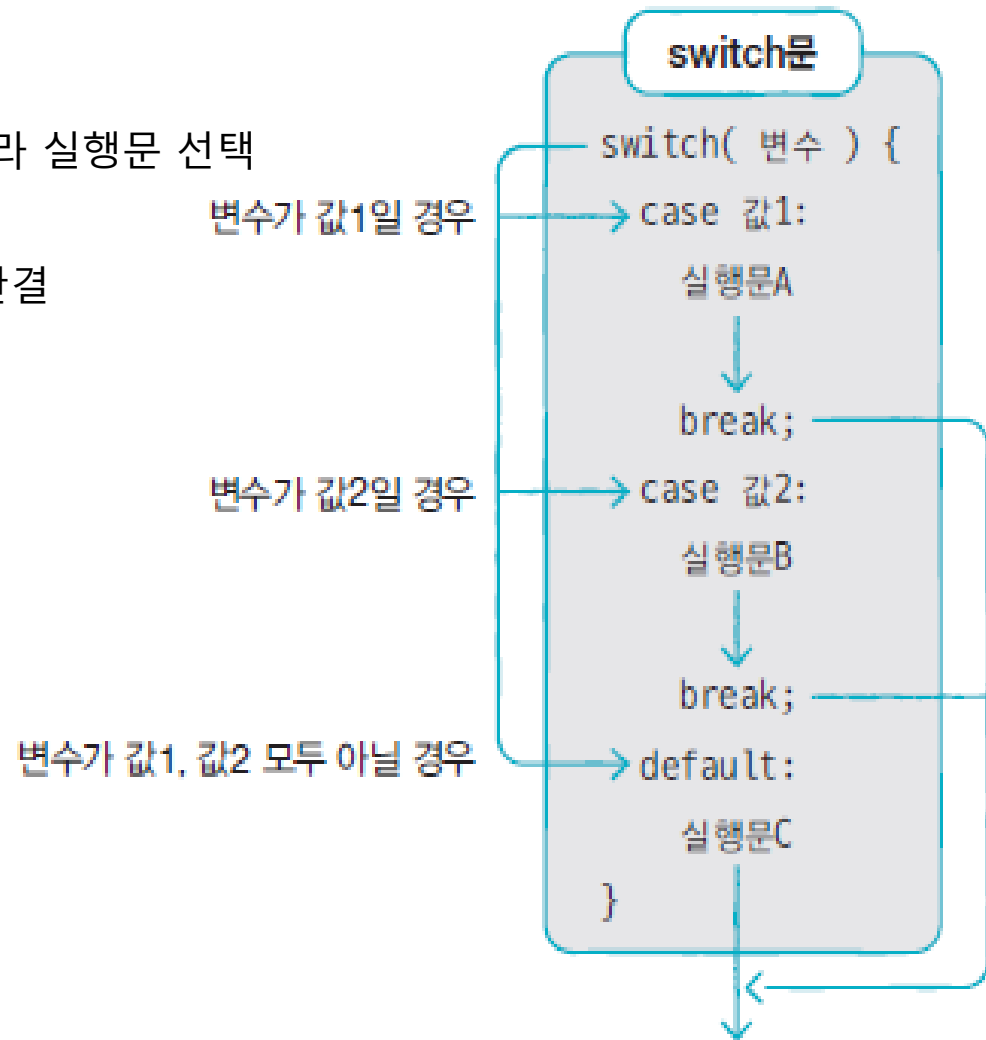
- 조건식이 여러 개인 if문
- 처음 if문의 조건식이 false일 경우 다른 조건식의 결과에 따라 실행 블록 선택
 - if 블록 끝에 else if문 추가
 - else if문 개수는 제한 없음



4. switch문

❖ switch문

- 변수가 어떤 값을 갖는가에 따라 실행문 선택
- 같은 기능의 if문보다 코드가 간결



5. 키워드로 끝내는 핵심 포인트

- **If문**: `if(조건식) { ... }`을 말하며 조건식이 true가 되면 중괄호 내부를 실행
- **If-else문**: `if(조건식) { ... } else { ... }`를 말하며 조건식이 true가 되면 if 중괄호 내부를 실행하고, false가 되면 else 중괄호 내부를 실행
- **if-else if-else문**: `if(조건식1) { ... } else if(조건식2) { ... } else { ... }`를 말하며 조건식1이 true가 되면 if 중괄호 내부를 실행하고, 조건식2가 true가 되면 else if 중괄호 내부를 실행한다. 조건식1과 조건식2가 모두 false가 되면 else 중괄호 내부가 실행된다.
- **switch문** : `switch(변수) { case 값1: ... case 값2: ... default: ... }`를 말하며 변수의 값이 값1이면 첫 번째 case 코드를 실행하고, 값2이면 두 번째 case 코드를 실행한다. 값1과 값2가 모두 아니면 default 코드를 실행한다.



Chapter

04

조건문과 반복문



04-2. 반복문: for문, while문, do-while문

혼자 공부하는 자바 (신용권 저)

- 시작하기 전에
- for문
- while문
- do-while문
- break문
- continue문
- 키워드로 끝내는 핵심 포인트



시작하기 전에

[핵심 키워드] : for문, while문, do-while문, break문, continue문

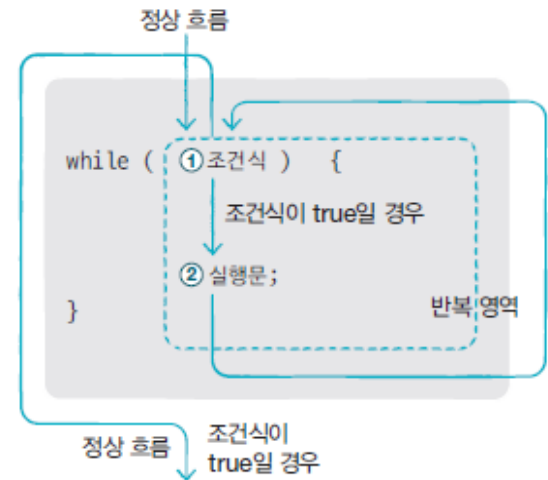
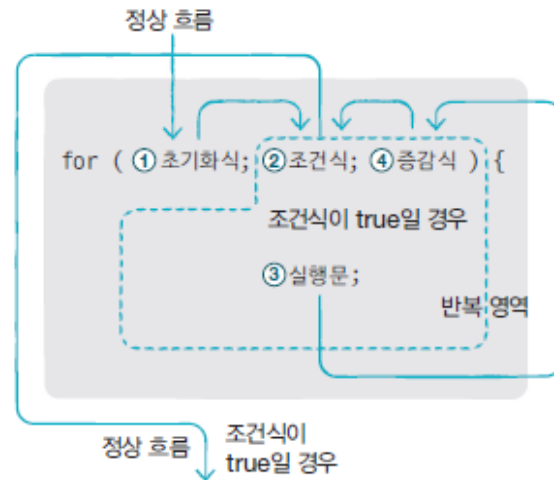
[핵심 포인트]

반복문에는 for문, while문, do-while문 있다.

반복문은 제어문 처음으로 되돌아가 반복 실행하는데 이것을 루핑(looping)이라고 한다.

❖ 반복문

- 어떤 작업을 반복적으로 실행하고 싶을 때 사용



- for문, while문, do-while문
 - for문은 반복 횟수 알고 있을 때 주로 사용
 - while문은 조건에 따라 반복할 때 주로 사용
 - do-while문은 while 문과 유사하나 조건을 나중에 검사



for문

❖ for문

- 주어진 횟수만큼 반복하고 싶을 경우 사용

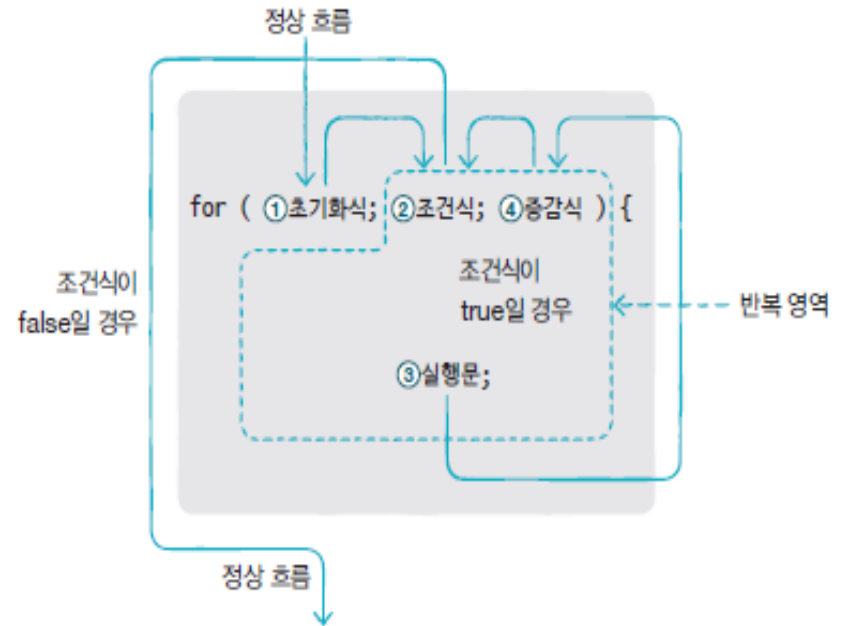
```
int sum = 0;
sum = sum + 1;
sum = sum + 2;
sum = sum + 3;
sum = sum + 4;
sum = sum + 5;
System.out.println("1~5의 합:" + sum);
```

5개의 실행문



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

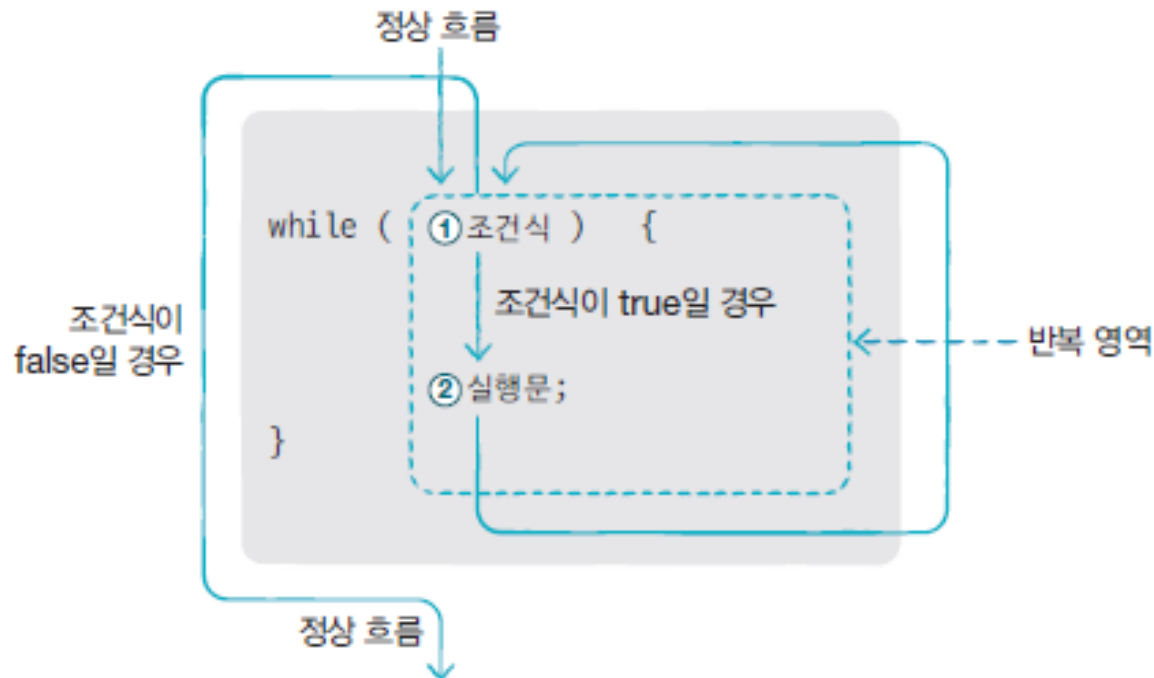
100번 반복



while문

❖ while문

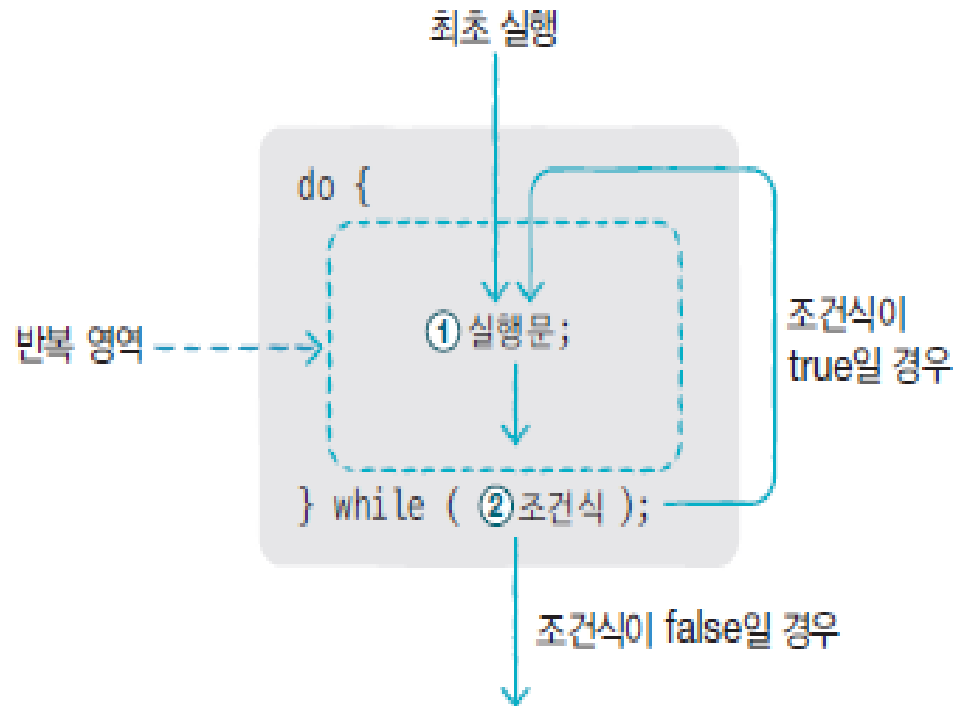
- 조건식에 따라 반복 여부를 결정할 경우
 - true일 경우 계속해서 반복
 - false일 경우 반복 종료
- 조건식에는 주로 비교 연산식, 논리 연산식



do-while문

❖ do-while문

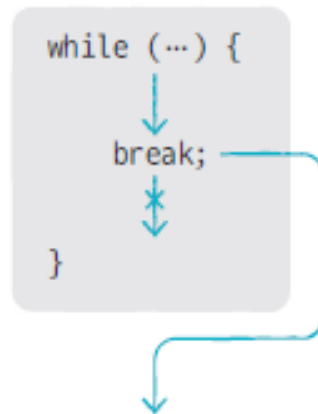
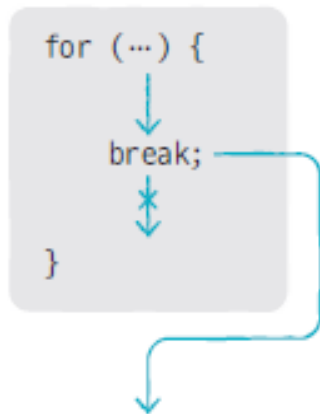
- 조건식에 의해 반복 실행하는 점에서 while문과 동일
- 블록 내부 실행문을 우선 실행하고 그 결과에 따라 반복 실행 여부를 결정



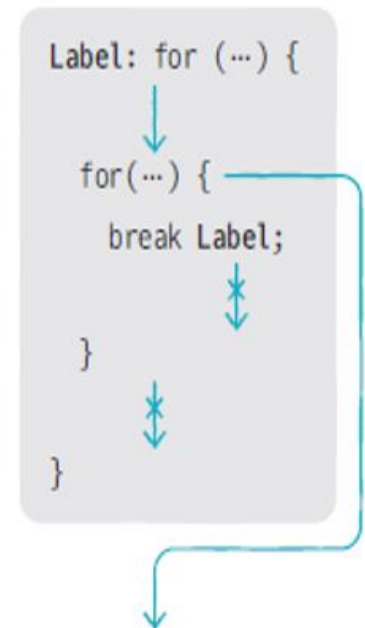
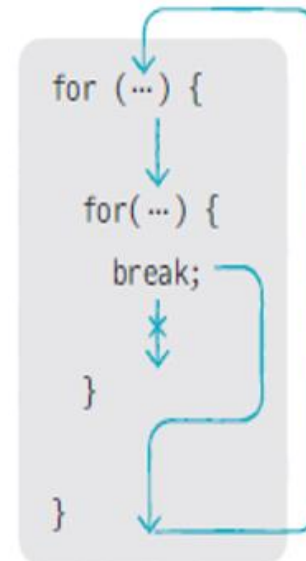
break문

❖ break문

- for, while, do-while, switch문의 실행을 중지할 때 사용
- 주로 if문과 함께 사용



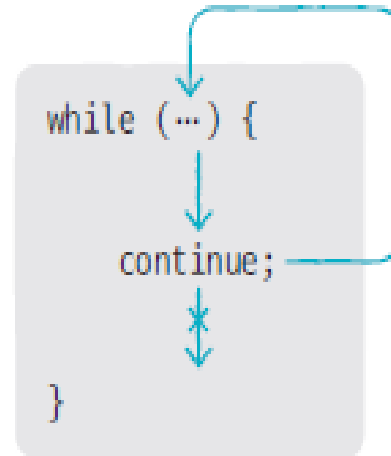
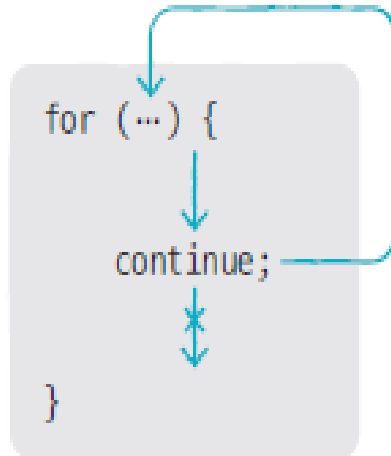
- 반복문이 중첩되어 있을 경우
 - Label을 이용해서 바깥 반복문을 빠져나감



continue문

❖ continue문

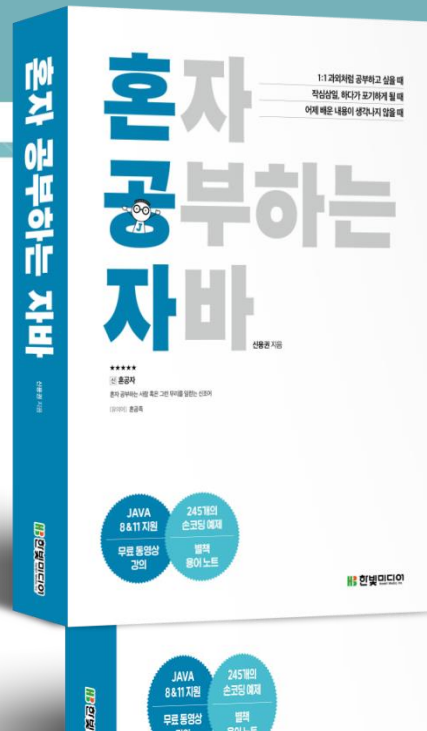
- for, while, do-while문에서만 사용
- for문의 증감식이나 while, do-while문의 조건식으로 이동
- 주로 if문과 함께 사용



키워드로 끝내는 핵심 포인트

- **for문**: for(초기화식; 조건식; 증감식;) ~ 을 말하며 지정한 횟수만큼만 반복할 때 주로 사용
- **while문**: while(조건식;) ~ 을 말하며 조건식이 true가 될 때까지 반복 실행
- **do-while문** : do ~ while(조건식;) 을 말하며 while문과 동일하나 조건식이 뒤에 있음
- **break문** : for문, while문, do-while문의 반복을 종료할 때 사용
- **continue문** : for문, while문, do-while문의 증감식 또는 조건식으로 돌아감





Thank You!

혼자 공부하는 자바 (신용권 저)

