

제어문

- 조건문 (선택제어문)

if ~ else / if ~ else if

switch ~ case ~ default

- 반복문 (반복제어문)

while

for

- 조건문

조건을 만족할 때만 {} 안에 있는 문장들을 수행

조건에 따라 경우의 수를 나누기 / 선택지를 주기

if문

```
if(조건식1) {  
    조건식1의 연산결과가 true일때 실행  
} else if(조건식2) {  
    조건식2의 연산결과가 true일때 실행  
} else if(조건식3) {  
    조건식3의 연산결과가 true일때 실행  
}  
...  
else {  
    위의 조건식들이 모두 false일 때 실행  
}
```

조건식 중 하나가 참(true)인 경우 블록 {} 안의 내용을 모두 실행한 후
아래 내용들은 하지 않고 세트를 탈출한다.

※ else if와 else문은 생략 가능.

switch문

변수에 담긴 값에 따라 알맞은 case로 이동하고 문장을 실행한다.

그리고 break문 이나 switch문의 끝을 만나면 switch문 전체를 빠져나간다.

만약 break를 써주지 않으면, 다음 case문으로 내려가 그 안에 있는 문장들도 실행된다.

따라서 각 케이스들을 나누어 주려면 알맞게 break; 를 사용해야 한다.

```
switch(변수) {  
  case 값1:  
    변수의 값이 값1 일 때 실행  
    break;  
  case 값2:  
    변수의 값이 값2 일 때 실행  
    break;  
  ...  
  default:  
    변수의 값이 위의 값들이 아닐 때 실행  
}
```

- switch문의 제약조건

1. switch문의 조건식 결과는 정수 또는 문자열이어야 한다.
2. case문의 값은 중복될 수 없다.
3. switch문은 항상 if문으로 바꿀 수 있다.

누적 대입 연산자 (복합 대입 연산자, 누적 연산자)

`+=, -=, *=, /=, ...`

```
int data = 10;
```

```
data = data + 5;      data += 5;    // data에 5만큼 더해서 넣어라!
```

```
data *= 10 + x;      >>    data = data * (10 + x);
```

※ 문자열은 '뒤'에 누적연결인 경우에는 사용가능

'앞'의 누적연결인 경우는 직접 써주어야 한다.

```
String result = "A";      result += "B"
```

```
>> result = result + "B"  // "AB"
```

-증감 연산자

`++, --` (1씩 증가, 1씩 감소)

```
data += 1;
```

```
data++;
```

-전위형

`++data` : 증감이 해당 줄부터 적용

```
int data = 10;
```

```
sysout(++data);      //11
```

-후위형

`data++` : 증감이 다음 줄부터 적용

```
int data = 10;
```

```
sysout(data++);      //10
```

```
sysout(data);        //11
```

반복문

특정한 코드나 행위를 반복 해야할 때 사용하는 문장

- 조건에 의한 반복문 ▶ while, do ~ while
- 횟수에 의한 반복문 ▶ for

for문

반복 횟수가 특정된 경우에 사용

값이 일정하게 변하는 반복일 경우에 높은 확률로 for문 사용

ex) 2 4 6 8 10

```
for(초기식;조건식;증감식) {  
    반복할 문장  
}  
  
for(int i=0; i < 10; i++) {  
    System.out.println("안녕");  
}
```

1. 초기식에 변수를 초기화 시켜서 몇부터 시작할건지 시작값을 정해준다.
2. 조건식에서 true인지 false인지 판단하여 문장을 반복할건지 안할건지 결정한다.
3. 조건식이 true일 경우 블럭안에 있는 문장을 실행 시킨다.
4. 문장을 실행 시킨 후 다시 증감식으로 올라와서 해당 수만큼 증감 시킨다.
5. 다시 조건식으로 가서 2-3-4 를 반복한다.

만약 조건식이 false라면 문장을 반복하지 않고 for문을 탈출한다.

초기식에서는 새로운 변수를 하나 선언 해 준다.(for문 내부에서만 해당 변수 사용가능)

만약 조건식이 false일 경우 for문은 실행되지 않는다.