

조건문

# 조건문의 종류

- if 문 : 특정 조건이 만족될 때에만 실행하는 문장.
- switch 문 : 조건에 따라서 실행시킬 코드를 구분.

# if 문

- 형식1

```
if(조건식)
{
    실행문장;
    ...
}
```

- 형식2

```
if(조건식)
{
    실행문장;
    ...
}
else
{
    실행문장;
    ...
}
```

- 형식3

```
if(조건식1)
{
    실행문장;
    ...
}
else if(조건식2)
{
    실행문장;
    ...
}
else
{
    실행문장;
    ...
}
```

# 조건연산자

- 구조

- true or false ? 숫자1 : 숫자2

- ex)

```
{  
    int num1 = 50, num2 = 100;  
    int big, diff;  
  
    big = (num1 > num2) ? num1 : num2;  
    diff = (num1 > num2) ? num1 - num2 : num2 - num1;  
}
```

# switch 문

- 형식

```
switch(수식(값)) // 수식값은 반드시 정수(or 문자)
{
    case 값1:
        실행문장;
        break;
    case 값2:
        실행문장;
        break;
    ...
    default:
        실행문장;
        break;
}
```

# 문제

- 아래의 예제를 switch문을 활용하여 변경해 보자.

```
1. class NumRange {  
2.     public static void main(String[] args) {  
3.         int n = 24;  
4.  
5.         if(n >= 0 && n < 10)  
6.             System.out.println("0이상 10미만의 수");  
7.         else if(n >= 10 && n < 20)  
8.             System.out.println ("10이상 20미만의 수");  
9.         else if(n >= 20 && n < 30)  
10.            System.out.println ("20이상 30미만의 수");  
11.        else  
12.            System.out.println ("음수 혹은 30이상의 수");  
13.    }  
14. }
```

반복문

# 반복문의 종류

- for 문
- while 문
- do {} while 문



# for 문

- 형식

```
for(초기식; 조건식; 증감식)
{
    실행문장;
    ...
}
```

- ex)

```
for(int num = 0; num < 3; num++)
{
    System.out.println("Hi");
}
```

# while 문

- 형식

```
while(조건식)
{
    실행문장;
    ...
}
```

- ex)

```
int num = 0;
while(num < 3)
{
    System.out.println("Hi");
    num++;
}
```

# do { } while 문

- 형식

```
do{  
    실행문장;  
    ...  
} while(조건식);
```

- ex)

```
int num = 0;  
  
do{  
    System.out.println("Hi");  
    num++;  
} while(num < 3);
```

# 문제

- 1. 1~100까지 2의 배수이면서 5의 배수인 숫자를 출력하고, 그 출력된 숫자들의 합을 구하는 프로그램을 for문을 사용하여 작성.
- 2. 1번 문제를 while문과 do~while문으로 바꾸어 작성.

# 기타 제어문 : break

```
1. class TestBreak {
2.     public static void main(String[] args) {
3.         int num = 1;
4.         boolean search = false;
5.
6.         while(num < 100) {
7.             if(num % 5 == 0 && num % 7 == 0) {
8.                 search = true;
9.                 break;
10.            }
11.            num++;
12.        }
13.
14.        if(search)
15.            System.out.println("찾는 정수 : " + num);
16.        else
17.            System.out.println("찾지 못했습니다.");
18.    }
```

# 기타 제어문 : continue

```
1. class TestContinue {  
2.     public static void main(String[] args) {  
3.         int num = 0, count = 0;  
4.  
5.         while(num++ < 100) {  
6.             if(num % 5 != 0 || num % 7 != 0) {  
7.                 continue;  
8.             }  
9.  
10.            count++;  
11.            System.out.println(num);  
12.        }  
13.  
14.        System.out.println("count : " + count);  
15.    }  
16. }
```