

if, equals

if문이란 비교 연산자를 이용해서 프로그램을 제어할 수 있는 '제어문'입니다.

이 if문을 배울 때부터 본격적인 프로그래밍이 시작됩니다.

정확히, if문으로 들어온 순간 '초급' -> '중급' 구간으로 들어왔다고 봐도 좋습니다.

```
public class test {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.println("첫 번째 숫자 입력 : ");  
        int a = sc.nextInt();  
        System.out.println("두 번째 숫자 입력 : ");  
        int b = sc.nextInt();  
  
        if(a < b){  
            System.out.println("b가 a보다 큼니다.");  
        }  
        if(a > b){  
            System.out.println("a가 b보다 큼니다.");  
        }  
        if(a == b){  
            System.out.println("값이 같습니다.");  
        }  
        if(a != b){  
            System.out.println("값이 다릅니다.");  
        }  
        if(a >= b){  
            System.out.println("a가 b보다 크거나, 둘이 같습니다.");  
        }  
        if(a <= b){  
            System.out.println("b가 a보다 크거나 둘이 같습니다.");  
        }  
    }  
}
```

<위 코드를 그대로 따라서 써주세요.>

아래로

실행 결과까지 확인이 되시면, 아래를 이어가주세요.(위 코드는 지우지마세요)

단, 위 문장은 최대한 다양하게 test해보시고(이거저거 지우거나, 혹은 추가해보는 식.) 넘어가시길 권장합니다.

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.println("첫 번째 숫자 입력 : ");
    int a = sc.nextInt();
    System.out.println("두 번째 숫자 입력 : ");
    int b = sc.nextInt();

    if(a == 1 && b == 1){
        System.out.println("a에 1이 들어있고, b에도 1이 들어있습니다.(and)");
    }
    if(a != 1 || b != 1){
        System.out.println("a에 1이 들어있거나, 혹은 b에 1이 들어있지 않습니다(or)");
    }
}
```

위 `a!=1 || b != 1`에서 a에 1이 들어있지 않거나, 혹은 b에 1이 들어있지 않습니다.입니다 잘못 적은거예요.

이번에는 여러 가지 조건을 다룰 수 있는 것들에 대해서 알아보시다.

or : 둘중에 하나만 참이어도 참이다 하지만 둘 다 참이어도 실행한다.

and : 동시에 참이어야 한다.

참고로 위 ||을 쓰는 방법은,

우리가 \n을 많이 사용해왔죠?

\를 왼쪽 쉬프트 누르고 두 번 눌러보세요.

||라고 나올 겁니다.

그게 or입니다.

다 작성하셨다면 이제 다양하게 test도 해보시고, 응용도 바꿔보면서 해보세요.

아래로

일단 하나만 더 배워볼게요. equals() 함수인데요. 이 녀석은 Java에서 객체 비교를 위해 사용되는 함수예요. 이 함수는 Object 클래스(Object라고 해서, java의 최상위 단위가 있습니다. 예를 들어 Hello라는 문자가 결국 영문 단어에서 파생했다시피, Object는 최 상위 부모라고 이해하시면 됩니다. 자세한 건 나중에 다시 다루겠습니다.)에서 상속받아 사용하며, 모든 Java 클래스에서 사용할 수 있어요. equals() 함수는 매개변수(일반 변수)로 다른 객체를 받아 그 객체와 자신의 값을 비교하게 됩니다.

예를 들면 String 클래스에서 equals() 함수를 사용하여 문자열을 비교할 수 있어요.

```
public static void main(String[] args) {
    String str1 = "Hello";
    String str2 = "Hello";
    String str3 = "World";

    if(str1.equals(str2)) {
        System.out.println("str1과 str2는 같습니다.");
    } else {
        System.out.println("str1과 str2는 다릅니다.");
    }

    if(str1.equals(str3)) {
        System.out.println("str1과 str3은 같습니다.");
    } else {
        System.out.println("str1과 str3은 다릅니다.");
    }
}
```

위에서는 str1과 str2를 equals() 함수로 비교하여 같은 문자열인지를 비교해요. 그리고 str1과 str3도 비교하여 다른 문자열임을 비교하죠. 이렇게 문자열을 비교할 때 equals() 함수를 사용하는 것은 매우 흔한 패턴이기 때문에 꼭 기억하셔야해요.

아래 문장에서, 사람 이름, km수, 답:은 입력을 받아주세요.

규민과 철수와 민영이 있다.

셋의 집은 각각 10km마다 떨어져 있는 거리에 존재한다.

세 명의 집을 차례대로 방문하려면 몇 km를 걸어가야 하는가.

답 :

정답일 경우

정답입니다.

오답일 경우

오답입니다.

답은 30km입니다.