

# Java

## 제 4 강 조건문

### 1. 조건문

1.1 조건문(if, switch)

1.7 Math.random()

1.2 if문

1.3 중첩 if문

1.4 switch문

1.5 중첩 switch문

1.6 if문과 switch문의 비교

### 1. 조건문(if, switch)

### 1.1 조건문 – if, switch

```
if (조건식) { 문장들 }
```

- 조건문은 조건식과 실행될 하나의 문장 또는 블록{}으로 구성
- Java에서 조건문은 if문과 switch문 두 가지 뿐이다.
- if문이 주로 사용되며, 경우의 수가 많은 경우 switch문을 사용할 것을 고려한다.
- 모든 switch문은 if문으로 변경이 가능하지만, if문은 switch문으로 변경할 수 없는 경우가 많다.

```
if(num==1) {  
    System.out.println("SK");  
} else if(num==6) {  
    System.out.println("KTF");  
} else if(num==9) {  
    System.out.println("LG");  
} else {  
    System.out.println("UNKNOWN");  
}
```



```
switch(num) {  
    case 1:  
        System.out.println("SK");  
        break;  
    case 6:  
        System.out.println("KTF");  
        break;  
    case 9:  
        System.out.println("LG");  
        break;  
    default:  
        System.out.println("UNKNOWN");  
}
```

### 1.2 if문

- if문은 if, if-else, if-else if의 세가지 형태가 있다.
- 조건식의 결과는 반드시 true 또는 false이어야 한다.

```
if(조건식) {  
    // 조건식의 결과가 true일 때 수행될 문장들  
}
```

```
if(조건식) {  
    // 조건식의 결과가 true일 때 수행될 문장들  
} else {  
    // 조건식의 결과가 false일 때 수행될 문장들  
}
```

```
if(조건식1) {  
    // 조건식1의 결과가 true일 때 수행될 문장들  
} else if(조건식2) {  
    // 조건식2의 결과가 true일 때 수행될 문장들  
    // (조건식1의 결과는 false)  
} else if(조건식3) {  
    // 조건식3의 결과가 true일 때 수행될 문장들  
    // (조건식1과 조건식2의 결과는 false)  
} else {  
    // 모든 조건식의 결과가 false일 때 수행될 문장들  
}
```

```
if(score > 60) {  
    System.out.println("합격입니다.");  
}
```

```
if(score > 60)  
    System.out.println("합격입니다.");
```

```
if(score > 60) {  
    System.out.println("합격입니다.");  
} else {  
    System.out.println("불합격입니다.");  
}
```

```
if(num > 0) {  
    System.out.println("양수입니다.");  
} else if(num < 0) {  
    System.out.println("음수입니다.");  
} else {  
    System.out.println("영입니다.");  
}
```

```
if(score >= 90) {  
    System.out.println("A등급");  
} else if(score >= 80 && score < 90) { // 80<=score<90  
    System.out.println("B등급");  
} else if(score >= 70 && score < 80) { // 70<=score<80  
    System.out.println("C등급");  
} else { // score < 70  
    System.out.println("F등급");  
}
```

### 1.2 if문 – 조건식의 예(example)

```
int i = 0;  
if(i%2==0) { }  
if(i%3==0) { }
```

```
if(i=0) { }  
if(i==0) { }
```

```
String str = "";  
char ch = ' ';  
if(ch==' ' || ch=='\t') { }  
if(ch=='c' || ch=='C') { }  
if(str=="c" || str=="C") { }  
if(str.equals("c") || str.equals("C")) { }  
if(str.equalsIgnoreCase("c")) { }  
  
if(ch>='0' && ch<='9') { }  
if(!(ch>='0' && ch<='9')) { }  
if(ch<'0' || ch>'9')) { }
```

```
if(('a'<=ch && ch<='z')||  
   ('A'<=ch && ch<='Z')) { }
```

```
if( i<-1 || i>3 && i<5 ) { }
```

```
str="3";  
if(str!=null && !str.equals("")) {  
    ch = str.charAt(0);  
}
```

문자열 "3" → 문자 '3'

```
boolean powerOn=false;  
if(!powerOn) {  
    // 전원이 꺼져있으면...  
}
```

### 1.3 중첩 if문

- if문 안에 또 다른 if문을 중첩해서 넣을 수 있다.
- if문의 중첩횟수에는 거의 제한이 없다.

```
if (조건식1) {  
    // 조건식1의  
    if (조건식2)  
        // 조건식2의  
    } else {  
        // 조건식2의  
    }  
} else {  
    // 조건식1의  
}
```

```
if (score >= 90) { // score가 90점 보다 같거나 크면 A학점 (grade)  
    grade = "A";  
  
    if ( score >= 98) { // 90점 이상 중에서도 98점 이상은 A+  
        grade += "+"; // grade = grade + "+";  
    } else if ( score < 94) {  
        grade += "-";  
    }  
} else if (score >= 80) { // score가 80점 보다 같거나 크면 B학점 (grade)  
    grade = "B";  
  
    if ( score >= 88) {  
        grade += "+";  
    } else if ( score < 84) {  
        grade += "-";  
    }  
} else { // 나머지는 C학점 (grade)  
    grade = "C";  
}
```

### 1.4 switch문

- if문의 조건식과 달리, 조건식의 계산결과가 int범위 이하의 정수만 가능
- 조건식의 계산결과와 일치하는 case문으로 이동 후 break문을 만날 때까지 문장들을 수행한다.(break문이 없으면 switch문의 끝까지 진행한다.)
- 일치하는 case문의 값이 없는 경우 default문으로 이동한다.  
(default문 생략가능)
- case문의 값으로 변수를 사용할 수 없다.(리터럴, 상수만 가능)

```
switch (조건식) {  
    case 값1 :  
        // 조건식의 결과가 값1과 같을 경우 수행될 문장들  
        //...  
        break;  
    case 값2 :  
        // 조건식의 결과가 값2와 같을 경우 수행될 문장들  
        //...  
        break;  
    //...  
    default :  
        // 조건식의 결과와 일치하는 case문이 없을 때 수행될 문장들  
        //...  
}
```

```
switch(num) {  
    case 1:  
    case 7:  
        System.out.println("SK");  
        break;  
    case 6:  
    case 8:  
        System.out.println("KTF");  
        break;  
    case 9:  
        System.out.println("LG");  
        break;  
    default:  
        System.out.println("UNKNOWN");  
        break;  
}
```



### 1.4 switch문 – 사용예(examples)

```
int level = 3;

switch(level) {
    case 3 :
        grantDelete(); // 삭제권한을 준다.
    case 2 :
        grantWrite();  // 쓰기권한을 준다.
    case 1 :
        grantRead();   // 읽기권한을 준다.
}
```

```
switch(score) {
    case 100: case 99: case 98: case 97: case 96:
    case 95: case 94: case 93: case 92: case 91:
    case 90 :
        grade = 'A';
        break;
    case 89: case 88: case 87: case 86:
    case 85: case 84: case 83: case 82: case 81:
    case 80 :
        grade = 'B';
        break;
    case 79: case 78: case 77: case 76:
    case 75: case 74: case 73: case 72: case 71:
    case 70 :
        grade = 'C';
        break;
    case 69: case 68: case 67: case 66:
    case 65: case 64: case 63: case 62: case 61:
    case 60 :
        grade = 'D';
        break;
    default :
        grade = 'F';
} // end of switch
```

```
char op = '+';


switch(op) {
    case '+':
        result = num1 + num2;
        break;
    case '-':
        result = num1 - num2;
        break;
    case '*':
        result = num1 * num2;
        break;
    case '/':
        result = num1 / num2;
        break;
}
```

```
switch(score/10) {
    case 10:
    case 9 :
        grade = 'A';
        break;
    case 8 :
        grade = 'B';
        break;
    case 7 :
        grade = 'C';
        break;
    case 6 :
        grade = 'D';
        break;
    default :
        grade = 'F';
}
```

### 1.5 중첩 switch문

- switch문 안에 또 다른 switch문을 중첩해서 넣을 수 있다.
- switch문의 중첩횟수에는 거의 제한이 없다.

```
switch(num) {  
    case 1:  
    case 7:  
        System.out.println("SK");  
        switch(num) {  
            case 1:  
                System.out.println("1");  
                break;  
            case 7:  
                System.out.println("7");  
                break;  
        }  
        break;  
    case 6:  
        System.out.println("KTF");  
        break;  
    case 9:  
        System.out.println("LG");  
        break;  
    default:  
        System.out.println("UNKNOWN");  
}
```



```
switch(num) {  
    case 1:  
    case 7:  
        System.out.println("SK");  
        if(num==1) {  
            System.out.println("1");  
        } else if(num==7) {  
            System.out.println("7");  
        }  
        break;  
    case 6:  
        System.out.println("KTF");  
        break;  
    case 9:  
        System.out.println("LG");  
        break;  
    default:  
        System.out.println("UNKNOWN");  
}
```

### 1.6 if문과 switch문의 비교

- if문이 주로 사용되며, 경우의 수가 많은 경우 switch문을 사용할 것을 고려한다.
- 모든 switch문은 if문으로 변경이 가능하지만, if문은 switch문으로 변경할 수 없는 경우가 많다.
- if문 보다 switch문이 더 간결하고 효율적이다.

```
if(num==1) {  
    System.out.println("SK");  
} else if(num==6) {  
    System.out.println("KTF");  
} else if(num==9) {  
    System.out.println("LG");  
} else {  
    System.out.println("UNKNOWN");  
}
```



```
switch(num) {  
    case 1:  
        System.out.println("SK");  
        break;  
    case 6:  
        System.out.println("KTF");  
        break;  
    case 9:  
        System.out.println("LG");  
        break;  
    default:  
        System.out.println("UNKNOWN");  
}
```

### 1.7 Math.random()

- Math클래스에 정의된 난수(亂數) 발생함수
- 0.0과 1.0 사이의 double값을 반환한다.( $0.0 \leq \text{Math.random()} < 1.0$ )

예) 1~10범위의 임의의 정수를 얻는 식 만들기

1. 각 변에 10을 곱한다.

```
0.0 * 10 <= Math.random() * 10 < 1.0 * 10  
0.0 <= Math.random() * 10 < 10.0
```

2. 각 변을 int형으로 변환한다.

```
(int)0.0 <= (int)(Math.random() * 10) < (int)10.0  
0 <= (int)(Math.random() * 10) < 10
```

3. 각 변에 1을 더한다.

```
int score = (int)(Math.random() * 10) + 1;
```

```
0 + 1 <= (int)(Math.random() * 10) + 1 < 10 + 1  
1 <= (int)(Math.random() * 10) + 1 < 11
```

## 1.8 Random 클래스

```
2
3 import java.util.Random;
4
5 public class RandomTest {
6
7     public static void main(String[] args) {
8         // TODO Auto-generated method stub
9         // Random() 객체 생성
10        Random generator = new Random();
11
12        int num1;
13        float num2;
14
15        // int 값의 범위에서 정수 랜덤값을 추출한다.
16        // (-2,147,483,648 ~ 2,147,483,647)
17        num1= generator.nextInt();
18        System.out.println("A random integer: "+num1);
19
20        // 0~9 사이의 정수 랜덤값을 추출한다.
21        num1= generator.nextInt(10);
22        System.out.println("Form 0 to 9: " + num1);
23
24        // 1~10 사이의 정수랜덤값을 추출한다.
25        num1= generator.nextInt(10) + 1;
26        System.out.println("Form 1 to 10 : " + num1);
27
28    }
29
30 }
31
```

감사합니다.