

# 자바 기초 문법

연산자,조건문

# 형 변환 (type conversion, type casting)

참 조: <https://docs.oracle.com/javase/specs/jls/se7/html/jls-5.html>

## ✚ 확대 형 변환 (자동 수행됨)

● byte → short, char → int → long → float → double

## ✚ 축소 형 변환 (강제 수행 필요)

```
public class Test {  
    public static void main(String[] args) {  
        int    v = 123; // 4바이트  
        long   w = v; // 확대 형 변환 (4바이트 => 8바이트)  
        int    x = (int) w; // 축소 형 변환 (8바이트 => 4바이트)  
  
        double n = 3.14;  
        int    m = (int) n; // 축소 형 변환 (double => int)  
        System.out.println(m);  
    }  
}
```

# 형 변환 (type conversion, type casting)

참 조 : <https://docs.oracle.com/javase/specs/jls/se7/html/jls-5.html>

```
public class Test {  
    public static void main(String[] args) {  
        int m = 95;  
  
        double n1 = (double)m;    // int => double  
        double n2 = m;           // int => double  
        System.out.println(n1);   // 95.0  
        System.out.println(n2);   // 95.0  
  
        double x = 3.14;  
        int y1 = (int)x;           // double => int  
        // int y2 = x;             // 오류 발생  
        System.out.println(y1);    // 3  
    }  
}
```

# 연산자(operator)

유형	연산자	비고
증감	++ --	n++ ++n
산술	+ - * / %	
쉬프트	>> << >>>	
비교	> < >= <= == !=	
비트	&   ^ ~	AND, OR, XOR, NOT
논리	&&    ^ !	AND, OR, XOR, NOT
삼항	? :	(x>0)? x : -x
대입	= += -= *= /= &= ^=  = <<= >>= >>>=	

# 정수 나눗셈

```
public class Test {  
    public static void main(String[] args) {  
        System.out.println( 7 / 2 );  
        System.out.println( 7.0 / 2.0 );  
        System.out.println( 7.0 / 2 );  
        System.out.println( 7 / 2.0 );  
        System.out.println( (double)7 / (double)2 );  
        System.out.println( (double)7 / 2 );  
        System.out.println( 7 / (double)2 );  
    }  
}
```

← 분자, 분모가 모두 정수이면  
그 결과도 정수(나머지 버림)

## 분모 0 정수 나눗셈: division by zero, divide by zero

```
public class Test {  
    public static void main(String[] args) {  
        System.out.println( 2 / 0. ); // Infinity  
        System.out.println( 2. / 0 ); // Infinity  
        System.out.println( 2.0 / 0.0 ); // Infinity  
        System.out.println( 2 / 0 ); // divide by zero error  
    }  
}
```

### division by zero error

- java.lang.ArithmeticException: / by zero
- 분모가 0인 나눗셈 수행 시 divide by zero 에러 발생

# 모듈로(modulo) 연산자

```
public class Test {  
    public static void main(String[] args) {  
  
        System.out.println( 7 % 2 );      // 7을 2로 나눈 나머지(정수)  
  
        int  n = 15;  
  
        boolean  check3m = (n%3 == 0);  
        System.out.println(n+"은(는) 3의 배수인가? "+check3m);  
  
        boolean  check5m = (n%5 == 0);  
        System.out.println(n+"은(는) 5의 배수인가? "+check5m);  
  
        boolean  check15m = check3m && check5m;  
        System.out.println(n+"은(는) 3의 배수이면서 5의 배수인가? "+check5m);  
  
        boolean  checkEven = (n%2==0);  
        System.out.println(n+"은(는) 짝수인가? "+checkEven);  
    }  
}
```

## **n+=1, n++, n--**

```
public class Test {  
    public static void main(String[] args) {  
        int n=1;  
  
        n = n + 1;  
        n += 1;      // n=n+1과 동일  
        n++;         // n=n+1과 동일  
        ++n;         // n=n+1과 동일  
  
        System.out.println("n="+n);      // 실행결과: n=5  
  
        int x = n++;      // n의 값 사용 후에 n의 값 1 증가  
        System.out.println("n="+n+", x="+x); // 실행결과: n=6, x=5  
        int y = ++n;      // n의 값 사용 전에 n의 값 1 증가  
        System.out.println("n="+n+", y="+y); // 실행결과: n=7, y=7  
    }  
}
```



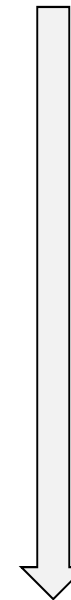
# 연산자 우선순위 (precedence)

참 조: <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/operators.html>

참 조: <https://introcs.cs.princeton.edu/java/11precedence/>

유형	연산자	결합성(associativity)
Postfix 증감	n++ n--	
Prefix 증감, 양/음 부호, NOT	++n --n +n -n ~ !	R to L
곱셈, 나눗셈, 모듈로	* / %	L to R
덧셈, 뺄셈	+ -	L to R
쉬프트	<< >> >>>	L to R
비교	< > <= >= instanceof	
동등 비교	== !=	L to R
비트 단위 AND	&	L to R
비트 단위 XOR	^	L to R
비트 단위 OR		L to R
논리 AND	&&	L to R
논리 OR		L to R
삼항	? :	L to R
대입	= += -= *= /= %= &= ^=  = <<= >>= >>>=	R to L

높음



낮음

# 조건문

## if조건문

- 조건식의 참/거짓에 따라 문장의 1회 실행 여부가 결정되는 문장

```
if(조건식) {  
    조건식이 참일 때 1회 실행될 문장(들)  
}
```

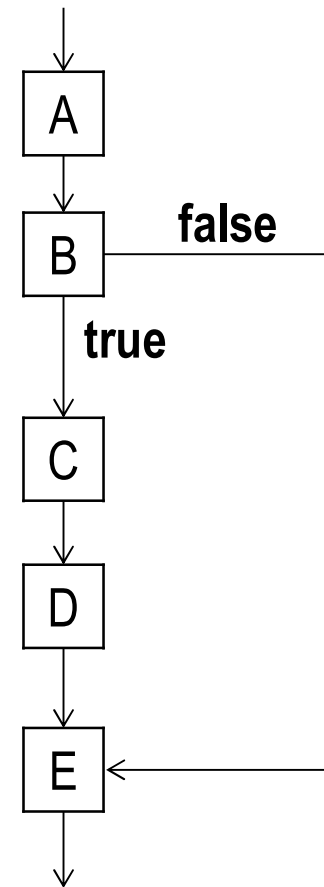
```
public class Test {  
    public static void main(String[] args) {  
        int age=18;  
        if(age<19) {  
            System.out.println("미성년자");  
        }  
    }  
}
```

```
if(조건식) {  
    조건식이 참일 때 1회 실행될 문장(들)  
} else {  
    조건식이 거짓일 때 1회 실행될 문장(들)  
}
```

```
public class Test {  
    public static void main(String[] args) {  
        int age=20;  
        if(age<19) {  
            System.out.println("미성년자");  
        } else {  
            System.out.println("성인");  
        }  
    }  
}
```

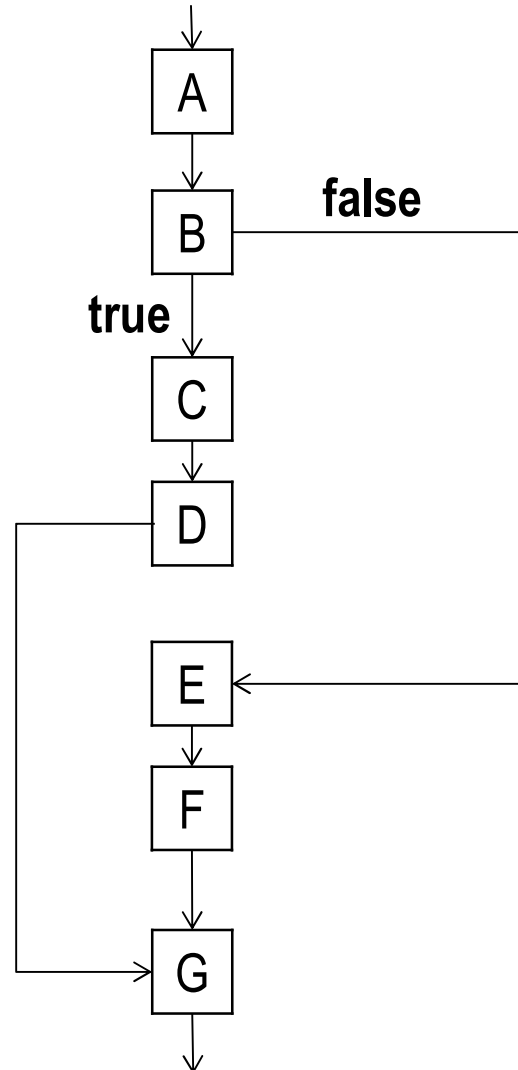
## if 조건문 실행 흐름

```
A;  
if(B) {  
    C;  
    D;  
}  
E;
```



## if-else 조건문 실행 흐름

```
A;  
if(B) {  
    C;  
    D;  
}  
else {  
    E;  
    F;  
}  
G;
```



# 조건식

## + 조건식

- 참, 거짓이 계산될 수 있는 식

## + 예

- `age<20`
  - ◆ 변수 `age`에 저장된 값이 20보다 적으면 `true` 그렇지 않으면 `false`
- `age<20 || age>=65`
  - ◆ `age` 값이 20 미만이거나 `age` 값이 65이상이면 `true`, 그렇지 않으면 `false`
- `age>=20 && age<65`
  - ◆ `age` 값이 20이상으면서 65 미만이면 `true`, 그렇지 않으면 `false`
- `score==100`
  - ◆ `score` 값이 100이면 `true`, 그렇지 않으면 `false`
- `score!=100`
  - ◆ `score` 값이 100이 아니면 `true`, 그렇지 않으면 `false`

## if-else 문장과 삼항연산자(ternary operator)

```
public class Test {  
    public static void main(String[] args) {  
        int x=5, y=8;  
        int max=0;  
  
        if(x>y) max=x;  
        else max=y;  
  
        System.out.println(max);  
    }  
}
```

```
public class Test {  
    public static void main(String[] args) {  
        int x=5, y=8;  
        int max=0;  
  
        max = (x>y)? x : y ;  
  
        System.out.println(max);  
    }  
}
```

# dangling else problem

## ✚ dangling else problem

- else를 이전의 어떤 if에 연결할 것인가의 문제
- 자바의 경우 이전의 가장 가까운 if에 연결하는 방식으로 문제 해결

```
public class Test {  
    public static void main(String[] args) {  
        // 기본 할인율 20%  
        // 35세 이상, 운전경력 5년이상 => 할인율 30%  
        // 35세 미만 => 할인율 10%  
        int age=20, drivingYears=2;  
        int discountRate=20;  
  
        if(age>=35)  
            if(drivingYears>=5) discountRate=30;  
            else discountRate=10;  
  
        System.out.println("할인율:"+discountRate+"%");  
    }  
}
```

```
if(age>=35) {  
    if(drivingYears>=5) discountRate=30;  
}  
else discountRate=10;
```

```
if(age>=35) {  
    if(drivingYears>=5) discountRate=30;  
    else discountRate=10;  
}
```

# 조건식: null 검사

```
import javax.swing.JOptionPane;
```

```
public class Test {
```

```
    public static void main(String[] args) {
```

```
        String s=JOptionPane.showInputDialog("임의의 문자열을 입력하세요");
```

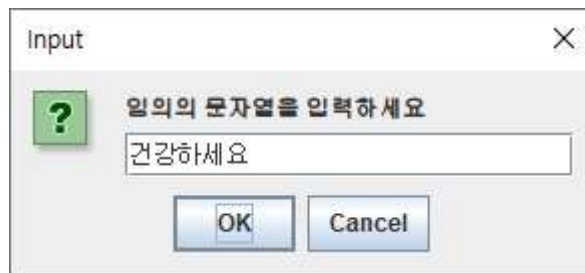
```
        if(s==null) System.out.println("Cancel 버튼을 클릭하셨습니다.");
```

```
        else System.out.println("당신이 입력한 값은 '"+s+"'입니다.");
```

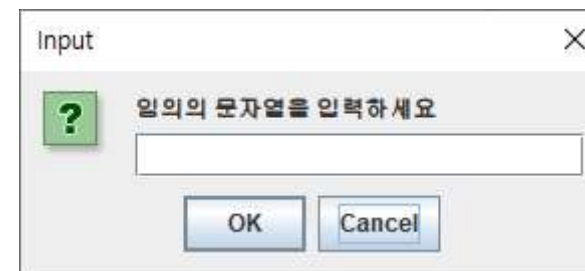
```
    }
```

```
}
```

JOptionPane.showInputDialog("...")에  
서 Cancel 버튼 클릭 시 null 반환됨



당신이 입력한 값은 '건강하세요'입니다.



Cancel 버튼을 클릭하셨습니다.



## 조건문 - 실습 A0

✚ 변수 score에 저장된 값의 절대값(absolute value)을 출력하는 자바 프로그램을 작성하시오

- Math.abs() 함수를 사용하지 마시오
- 삼항연산자 대신 if문(혹은 if-else문)으로 작성하시오
- score의 값은 프로그램 내에 그 값을 입력해 두시오
  - ◆ score 값은 -8, 7, 0에 대해 프로그램이 정상 동작하는지 테스트하시오

## 조건문 - 실습 A1

- ✚ 변수 score에 저장된 값의 절대값(absolute value)을 출력하는 자바 프로그램을 작성하시오
  - Math.abs() 함수를 사용하지 마시오
  - if문(혹은 if-else문) 대신 삼항연산자로 작성하시오
  - score의 값은 프로그램 내에 그 값을 입력해 두시오
    - ◆ score 값은 -8, 7, 0에 대해 프로그램이 정상 동작하는지 테스트하시오

## 조건문 – 실습 B

✚ 다음 입장료 계산법과 프로그램 내용에 따라 입장료를 계산하는 자바 프로그램을 작성하시오.

✚ 입장료 계산법

- 기준 1: 기본 입장료 1000원
- 기준 2: 6세 이하 50% 할인
- 기준 3: 65세 이상 30% 할인

✚ 프로그램 내용

- 기본입장료를 저장하는 변수 fee를 선언
- 나이를 저장하는 변수 age를 선언
- age의 값은 프로그램 내에 그 값을 입력해 두시오
  - ◆ age 값은 5, 6, 7, 25, 64, 65, 66에 대해 테스트할 것
- 변수 age에 저장된 값에 해당하는 입장료를 위 기준에 맞게 계산하여 출력할 것

## 조건문 - 실습 C

✚ 다음 입장료 계산법과 프로그램 내용에 따라 입장료를 계산하는 자바 프로그램을 작성하시오.

✚ 입장료 계산법

- 기준 1: 기본 입장료 1000원
- 기준 2: 6세 이하 50% 할인
- 기준 3: 65세 이상 30% 할인
- 기준 3: 7세~20세 미만 20% 할인

✚ 프로그램 내용

- 기본입장료를 저장하는 변수 fee를 선언
- 나이를 저장하는 변수 age를 선언
- age의 값은 프로그램 내에 그 값을 입력해 두시오
  - ◆ age 값은 5, 6, 7, 8, 19, 20, 21, 64, 65, 66에 대해 테스트할 것
- 변수 age에 저장된 값에 해당하는 입장료를 위 기준에 맞게 계산하여 출력할 것

## 조건문 - 실습 D0

- ✚ 변수 score에 저장된 성적 값이 0~100 사이일 경우 **정상**을 출력하고, 그렇지 않은 경우 **비정상**을 출력하는 자바프로그램을 작성하시오
  - 삼항연산자 대신 if문(혹은 if-else문)으로 작성하시오
  - score의 값은 프로그램 내에 그 값을 입력해 두시오
    - ◆ score 값은 -1, 0, 1, 99, 100, 101에 대해 테스트할 것

## 조건문 - 실습 D1

✚ 변수 score에 저장된 성적 값이 0~100 사이일 경우 **정상**을 출력하고, 그렇지 않은 경우 **비정상**을 출력하는 자바프로그램을 작성하시오

- if문(혹은 if-else문) 대신 삼항연산자로 작성하시오
- score의 값은 프로그램 내에 그 값을 입력해 두시오
  - ◆ score 값은 -1, 0, 1, 99, 100, 101에 대해 테스트할 것

## 조건문 - 실습 E

✚ 다음 예와 같이 두 변수  $x, y$ 에 저장된 값들에 대해 큰 수를 먼저 출력하고 작은 수를 그 다음 줄에 출력하는 자바프로그램을 작성하시오

- 실행 결과 예

- ◆  $x=5, y=9$ 일 경우 큰 수인 9를 먼저 출력하고 다음 줄에 5를 출력

9
5

- $x, y$ 의 값은 프로그램 내에 그 값을 입력해 두시오

- ◆ 다음 값들에 대해 테스트하시오

- $x=34, y=90$
    - $x=23, y=0$

## 조건문 - 실습 F

✚ 두 변수  $x, y$ 에 저장된 값 중 큰 수를 출력하는 자바프로그램을 작성하시오

● 실행 결과 예

◆  $x=5, y=9$ 일 경우 9를 출력

9

●  $x, y$ 의 값은 프로그램 내에 그 값을 입력해 두시오

◆ 다음 값들에 대해 테스트하시오

-  $x=23, y=89$

-  $x=5, y=-4$



## 조건문 - 실습 G

✚ 세 변수  $x, y, z$ 에 저장된 값 중 가장 큰 수를 출력하는 자바프로그램을 작성하시오

● 실행 결과 예

◆  $x=5, y=9, z=1$ 일 경우 9를 출력

9

●  $x, y, z$ 의 값은 프로그램 내에 그 값을 입력해 두시오

◆ 다음 값들에 대해 테스트하시오

- $x=237, y=89, z=-6$
- $x=-109, y=-3, z=-5$
- $x=9, y=0, z=45$

## 조건문 - 실습 H

✚ 변수 score에 성적 값(0~100)이 저장되어 있다고 가정하고 score에 저장된 값에 해당하는 학점을 출력하는 자바 코드를 작성하시오

- 학점 기준

- ◆ 90~100 → A, 80~89 → B, 70~79 → C, 60~69 → D, 0~59: F

- 실행 결과 예

- ◆ score=90일 경우 A를 출력

- score의 값은 프로그램 내에 그 값을 입력해 두시오

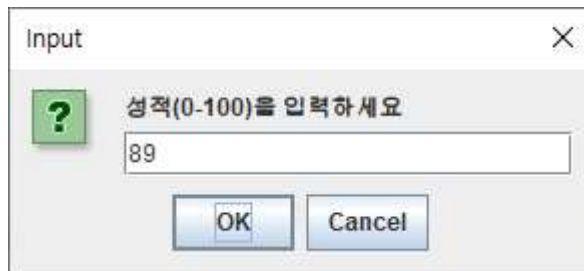
- ◆ score 값은 0, 59, 60, 69, 70, 79, 80, 89, 90, 100에 대해 테스트하시오

## 조건문 - 실습 I

다음 실행 예시와 같이 성적 값(0~100)을 입력 받아 대응하는 학점을 출력하는 자바 코드를 작성하시오

- 학점 기준

- ◆ 90~100 → A, 80~89 → B, 70~79 → C, 60~69 → D, 0~59: F



Input dialog box titled "Input" with a close button (X). It contains a green question mark icon, the text "성적(0-100)을 입력하세요", a text input field containing "89", and "OK" and "Cancel" buttons.




Message dialog box titled "Message" with a close button (X). It contains a blue information icon, the text "성적 89.0의 학점은 B입니다.", and an "OK" button.

## References

 <http://docs.oracle.com/javase/7/docs/api/>

 <https://docs.oracle.com/javase/tutorial/java/>

 김윤명. (2008). 뇌를 자극하는 Java 프로그래밍. 한빛미디어.

 남궁성. 자바의 정석. 도우출판.

 황기태, 김효수 (2015). 명품 Java Programming. 생능출판사.