

## Chapter 3. 연산자

### 19장. 대입연산자

▷ 오른쪽에 있는 값을 왼쪽에 있는 값에 넣어준다.

```
package ch03;
public class OperEx01{
    public static void main(String[] args){
        int num = 10; // 대입연산자
    }
}
```

### 20장. 사칙연산자

▷ 사칙연산자 : +, -, \*, /

```
package ch03;
public class OperEx02{
    public static void main(String[] args){
        int n1 = 3;
        int n2 = 4;
        int sum = n1 + n2;
        System.out.println(sum);
        String s1 = "반가워 나의 나이는 ";
        int age = 25;
        System.out.println(s1 + age); // 더하기가 아닌 결합이 된다.

        System.out.println(5-3);
        System.out.println(10/2);
        System.out.println(20*2);
    }
}
```

## 21장. 비교연산자

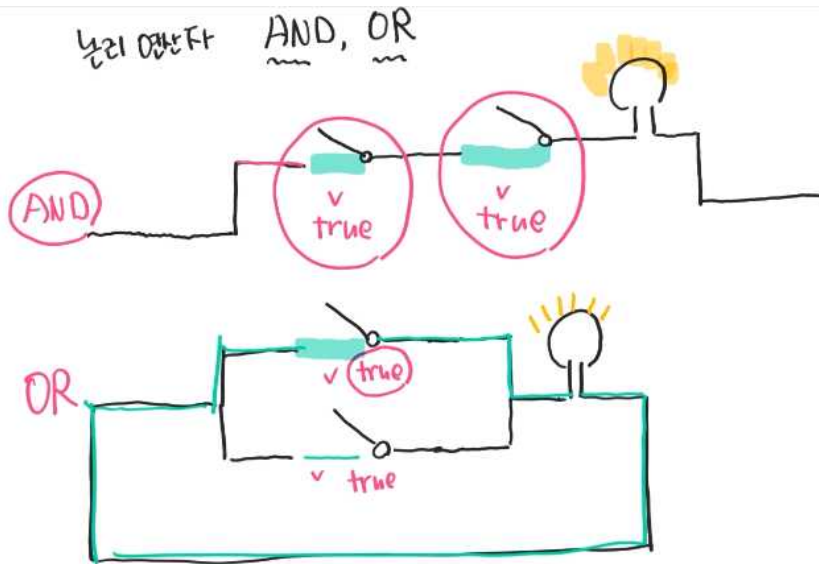
▷ 관계연산자를 이용한 비교 판단 : 결과는 항상 true or false 이다.

```
package ch03;
public class OperEx03{
    public static void main(String[] args){
        System.out.println( 1 == 1 );
        System.out.println( 1 == 2 );
        System.out.println( 5 != 3 );
        System.out.println( 3 > 1 );
        System.out.println( 3 < 1 );
        System.out.println( 2 <= 2 );
        System.out.println( 2 <= 2 );
        System.out.println( 2 >= 2 );
        System.out.println( 2 >= 1 );
    }
}
```

## 22장. 논리연산자

▷ and == 『 && 』

▷ or == 『 || 』



```
package ch03;
public class OperEx04{
    public static void main(String[] args){
        System.out.println(1==1 && 1==2);
        System.out.println(1==1 || 1==2 );
        int n1 = 1;
        int n2 = 2;
        System.out.println(n1==n1 || n1==n2 );
        // 상수 : 변하지 않는 수
        // 변수 : 변하는 수
    }
}
```

## 23장. 조건연산자(삼항연산자)

▷ 조건(비교연산자) ? true : false;

```
package ch03;
public class OperEx05{
    public static void main(String[] args){
        // 삼항 연산자 = 조건 연산자
        int n1 = 5;
        // 조건(비교연산자)? true : false;
        System.out.println(n1==5 ? "5가 맞습니다." : "5가 아닙니다." );
        int point = 59;
        System.out.println(point >= 60 ? "시험을 통과하였습니다." : "내년에 재수강해야합니다." );

        boolean result2 = 1==1;
        System.out.println("결과 : " + result2);

        int result3 = point < 60 ? 0 : 1;
        System.out.println("결과 : " + result3);
    }
}
```

## 24장. 문제풀기(나머지연산자)

▷ 나머지 연산자 : %

```
package ch03;
public class OperEx06{
    public static void main(String[] args){
        System.out.println(130/100); // 몫 = 1
        System.out.println(130%100); // 나머지 = 30
        // 조건(비교연산자)? true : false;
        int money = 130;
        System.out.println("100원 : " + money%100); // 100원 갯수
        System.out.println("10원 : " + (money%100)/10; // 10원 갯수
    }
}
```

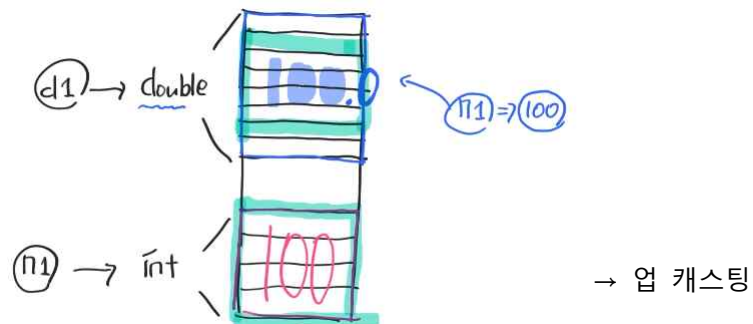
## 25장. 문제풀기(형변환)

▷ 다운캐스팅, 업캐스팅

```
package ch03;
public class CastEx01{
    public static void main(String[] args){
        // 캐스팅(다운캐스팅, 업캐스팅)
        int n1 = 100; //4Byte
        double d1 = n1; // 8Byte(8Byte ← 4Byte)
        System.out.println(d1); // 100.0 → 업캐스팅(묵시적)

        double d2 = 100.8;
        // 명시적 형변환을 해야 가능
        int n2 = (int)d2; // double 을 int 로 변환할 수 없어요...
        System.out.println(n2) // 단점 : 데이터가 유실될 수 있다.

        int money = 130;
        System.out.println(money/100);
        System.out.println(money/100.0); // int / double → double
    }
}
```



## 26장. 문제풀기(최소동전구하기)

▷ 2,680원을 500원, 100원, 50원, 10원짜리 각 동전이 몇 개씩 필요한가?

```
package ch03;
public class CoinExample{
    public static void main(String[] args){
        int money = 3680;
        int restMoney = money;
        // Step 1
        System.out.println("500원 : " + (restMoney/500)); // 몫 : 7
        int restMoney = money % 500; // 나머지 : 180
        System.out.println("남은금액 : " + restMoney);

        // Step 2
        System.out.println("100원 : " + (restMoney/100));
        int restMoney = money % 100;
        System.out.println("남은금액 : " + restMoney);

        // Step 3
        System.out.println("50원 : " + (restMoney/50));
        int restMoney = money % 50;
        System.out.println("남은금액 : " + restMoney);

        // Step 4
        System.out.println("10원 : " + (restMoney/10));
        int restMoney = money % 10;
        System.out.println("남은금액 : " + restMoney);

        // 그런데, 맨 마지막에
        // 마무리
        // System.out.println("3680의 최소 동전 갯수는?");
        // System.out.println("500원 7개");
        // System.out.println("100원 1개");
        // System.out.println("50원 1개");
        // System.out.println("10원 3개");
        // 위의 문제를 해결하기 위해 어떻게 해야할까?
    }
}
```

※ 동전 교환을 함수로 변환해 보기

## 27장. 문제풀기(해답)

▷

### ▶ Chapter 3 연습문제

▷ 문제 : 가솔린 8.86L를 충전한 A 자동차는 총 182.736km를 운행할 수 있다고 한다. 이차의 연비는?  
(단, 소수이하를 버리고 정수치만 표기하시오.)

```
package ch03;
public class ExampleEx01{
    static double calc(double gasoline, double distance){
        int e = (int) (distance / gasoline);
        return e;
    }
    public static void main(String[] args){
        double gasoline = 8.86;
        double distance = 182.736;
        double efficiency = calc(gasoline, distance);
        System.out.println("연비 : " + efficiency + "km/L");
    }
}
```