

객체와 클래스

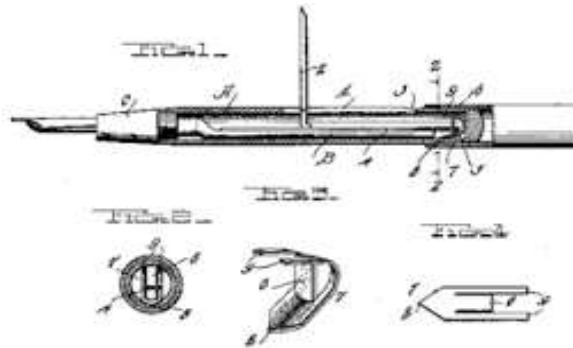


객체와 클래스

- 자바에서 클래스는 객체를 만들기 위한 틀입니다.
- 클래스의 변수는 객체의 속성이고, 클래스의 메서드는 객체의 기능입니다.



Instance(생산)



Class(설계도)



Object

자 다음 3개의 계산기를 보자

```
public class ObjectBasic1 {  
  
    public static void main(String[] args) {  
        계산기 사용..  
    }  
  
    //1번 계산기  
    static int result1 = 0;  
    static int add1(int n) {  
        result1 += n;  
        return result1;  
    }  
  
    //2번 계산기  
    static int result2 = 0;  
    static int add2(int n) {  
        result2 += n;  
        return result2;  
    }  
  
    //3번 계산기  
    static int result3 = 0;  
    static int add3(int n) {  
        result3 += n;  
        return result3;  
    }  
}
```

같은 기능을 가지고 있다
하지만 계산기가 또 필요하다면
계속해서 아래에 만들어야 한다..



```
public class Calculator {  
  
    int result = 0;  
    int add(int n) {  
        result += n;  
        return result;  
    }  
}
```

그래서!
클래스 라는 틀 안에
계산기 기능을 옮겨 놓고
필요할 때 생산 해서 쓰겠다!

자바는 공장이다

왜 우리가 클래스를 사용하는가?

- 공장에서 물건을 생산할 때 대량생산을 하기 위해 틀을 만들고 사용한다.
- 클래스는 이 대량생산을 위한 틀인 것이다.

객체생성 명령

```
ClassName instanceName = new Constructor();
```

Pen.java

```
1: public class Pen {  
2:     //클래스 안의 코드 작성  
3: }
```

PenInstanceExample.java

```
1: public class PenInstanceExample {  
2:     public static void main(String[] args) {  
3:         Pen redPen = new Pen();  
4:         System.out.println(redPen);  
5:     }  
6: }
```

Console ✕

✕ ✕ ✕ | ✕ ✕ ✕

<terminated> PenInstanc

Pen@15db9742

클래스 안에는 뭐가 존재하나?

멤버변수

-대량 생산물의 속성을 나타냄

메서드

-대량 생산물의 기능을 나타냄

Pen.java

```
1: public class Pen {  
2:     public String color;  
3:     int price;  
4:  
5:     public void write() {  
6:         System.out.println("Hello World!");  
7:     }  
8: }
```

팬이 생성되면,
가격과, 색깔이 있을 것이다.
팬은 글을 쓰는 기능이 있을 것이다.

멤버변수와 지역변수

멤버변수(Member Variable)

- 클래스 안에서 선언된 변수
- 객체를 생성할 때 각 데이터 타입의 기본 값으로 자동 초기화
- 기본 값 : 정수형 0, 실수형 0.0, 논리형 false, 객체 null

Variable.java

```
1: public class Variable {
2:
3:     int a;
4:
5:     public void printNum(int c) {
6:         int b = 1;    //반드시 초기화해야 함
7:         System.out.println("멤버변수 : " + a);
8:         System.out.println("지역변수 : " + b);
9:         System.out.println("매개변수 : " + c);
10:    }
11: }
```

지역변수(Local Variable)

- 메서드 안에서 선언된 변수
- 사용하기 전에 반드시 초기화 해야 함

VariableExample.java

```
1: public class VariableExample {
2:
3:     public static void main(String args[]) {
4:         Variable var = new Variable();
5:         var.printNum(2);
6:     }
7: }
```

Console

<terminated> Variable

멤버변수 : 0
지역변수 : 1
매개변수 : 2

생성자란 무엇인가? (자바 공장의 직원이다)

생성자(Constructor)

- 생성자는 객체가 생성될 때 객체의 상태(멤버변수)를 초기화하는데 사용됩니다.
- 객체가 생성될 때 **new 키워드 다음에 오는 것을** 생성자 라고합니다.
- 생성자는 메서드와 모양이 비슷하게 생겼지만, **반환 타입을 적지 않고 클래스 이름과 대/소문자까지 똑같이** 만들어야 합니다.
- 생성자는 반드시 클래스 안에 하나 이상 선언해야 하며, 만일 생성자를 만들지 않을 경우 자바 가상 머신(JVM: Java Virtual Machine)이 대신해서 기본 생성자를 자동으로 하나 만들어 줍니다.
- 기본 생성자는 매개 변수를 가지지 않는 기본 형태의 생성자 입니다.

- 생성자를 만드는 규칙

1. 생성자의 이름은 클래스 이름과 대/소문자까지 정확히 같아야 합니다. 만약 클래스 이름과 다른 이름으로 만들면 메서드로 간주합니다.
2. 생성자는 메서드와는 달리 **반환 유형을 적지 않습니다.**
3. 생성자는 다른 유형으로 **여러개 생성** 가능합니다 (생성자 오버로딩)

생성자 예시

Pen.java

```
1: public class Pen {
2:     public String color;
3:     int price;
4:
5:     public Pen() {
6:         color = "black";
7:     }
8:
9:     public Pen(String init_color) {
10:        color = init_color;
11:        System.out.println("생성자를 이용하여 color 값 초기화 함");
12:    }
13:
14:    public void write() {
15:        System.out.println("Hello World!");
16:    }
17: }
```

생성자는 클래스의 생성을
담당하는 직원이다!

생성자는 여러 개가 있을 수도 있다!

PenConstructorExample.java

```
1: public class PenConstructorExample {
2:     public static void main(String[] args) {
3:         Pen redPen = new Pen("red");
4:         System.out.println("펜의 색은 " + redPen.color + "입니다.");
5:     }
6: }
```

Console

<terminated> PenConstructorExample [Java Appl

생성자를 이용하여 **color** 값 초기화 함
펜의 색은 **red**입니다.



Chapter 9

수고하셨습니다