

Day04 (H1)

20150817

Class and Objects
Method
Static Method
Constructor

Copyright (c) 2015 Young W. Lim.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

class 0/2
12

```
class Rect {
```

```
int w;
```

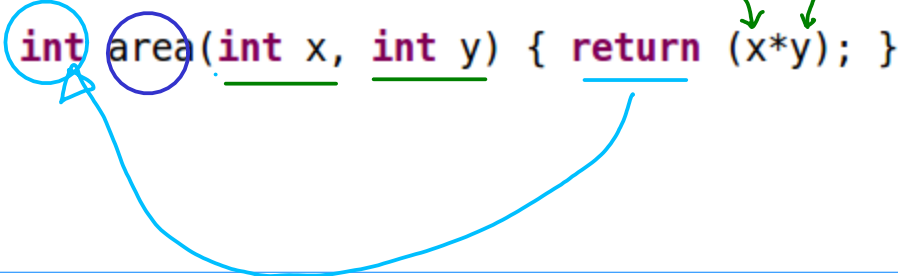
```
int h;
```

```
Rect() { w=0; h=0; }
```

```
int area(int x, int y) { return (x*y); }
```

```
}
```

첫 번째 인자
두 번째 인자



```
public class RectTest {
```

```
/**
```

```
 * @param args
```

```
 */
```

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

```
    // TODO Auto-generated method stub
```

```
    Rect R1 = new Rect();  
    Rect R2 = new Rect();  
    Rect R3 = new Rect();
```

객체를 생성하는 방법: **new**

class type

객체 참조 변수

```
    R1.w = 10;
```

```
    R1.h = 20;
```

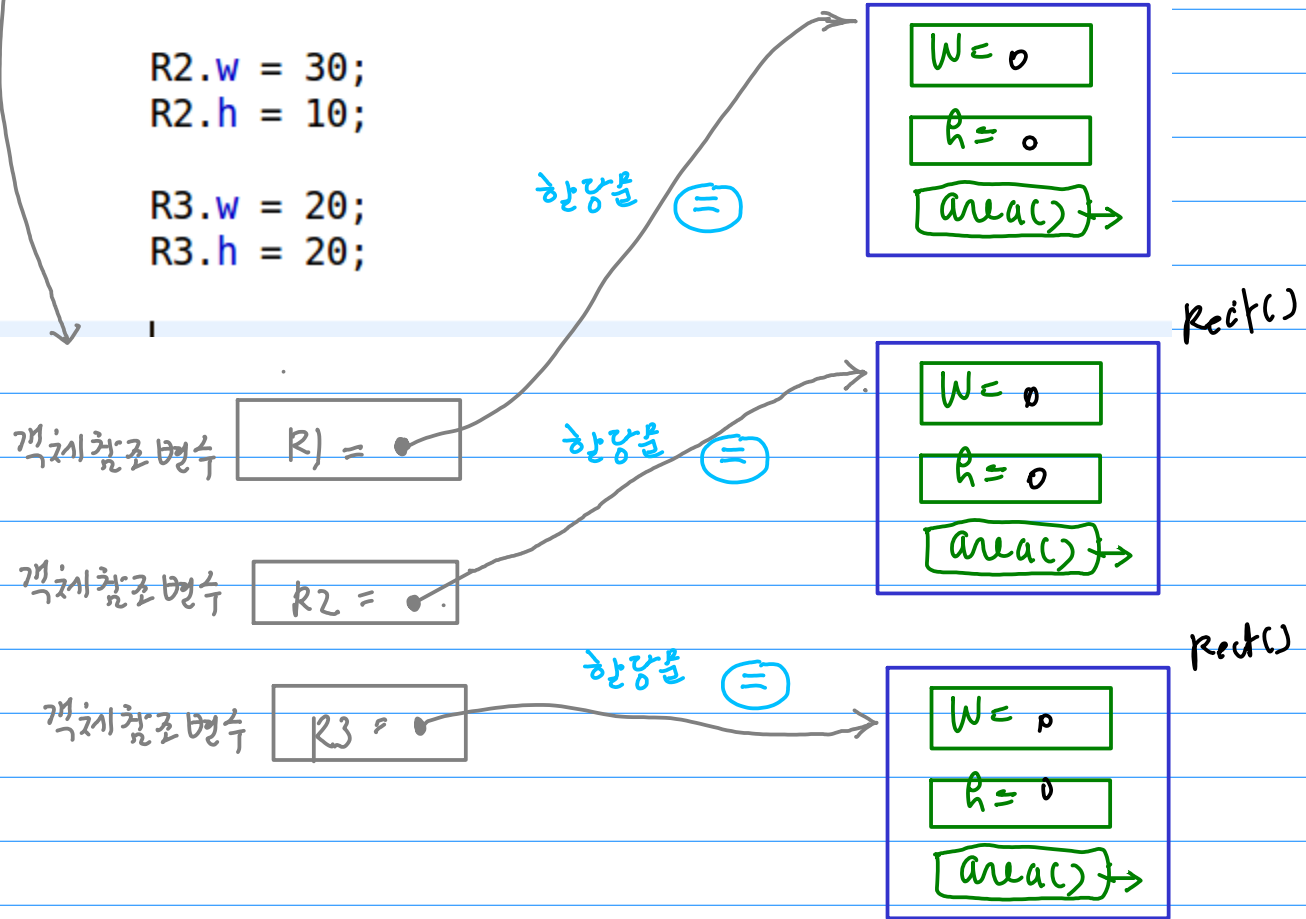
```
    R2.w = 30;
```

```
    R2.h = 10;
```

```
    R3.w = 20;
```

```
    R3.h = 20;
```

생성자 호출 **call**
Rect()



```
public class RectTest {
```

```
/**
```

```
 * @param args
```

```
 */
```

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

```
 // TODO Auto-generated method stub
```

```
Rect R1 = new Rect();
Rect R2 = new Rect();
Rect R3 = new Rect();
```

객체를 생성하는 방법

class type

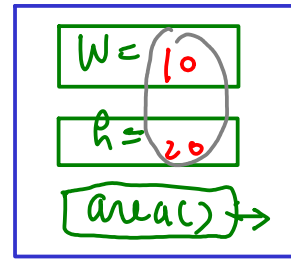
객체 참조명

```
R1.w = 10;
R1.h = 20;
```

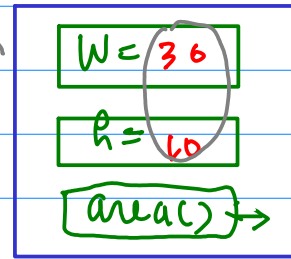
```
R2.w = 30;
R2.h = 10;
```

```
R3.w = 20;
R3.h = 20;
```

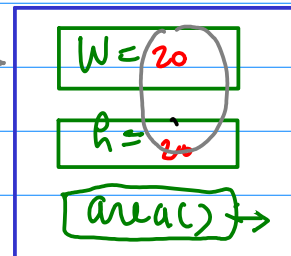
생성과 함께
Rect()



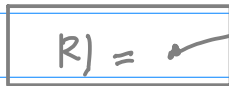
Rect()



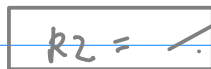
Rect()



R1 이 가리키는 객체의
w & h 수정



R2 가 가리키는 객체의
w & h 수정



R3 가 가리키는 객체의
w & h 수정



생성자 함수

```

class Rect {
    int w;
    int h;
    Rect() { w=0; h=0; }
    void display() {
        System.out.print("w=");
        System.out.println(w);
        System.out.print("h=");
        System.out.println(h);
    }
}
    
```

class 이름

class 이름과 같은 함수 = 생성자 함수

⇒ 초기화

⇒ return 값 X

⇒ 그러나 ~~void~~

일반 method

member data (field)는
이러한 변수를 받을 필요 없이
직접 사용 가능

Rect R

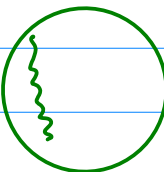
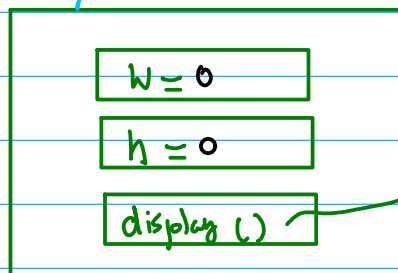
R = 참조변수

(어떤 객체를 가리키는)

= new Rect();

Rect 클래스의 객체를 memory에 생성함.

생성자 함수 Rect()를 부름



```

class Rect {
    int w;
    int h;

    Rect() { w=0; h=0; }

    int area() { return w*h; }

    void display() {
        System.out.print("w=");
        System.out.println(w);
        System.out.print("h=");
        System.out.println(h);

        int A;
        A = area();
    }
}

```

$$A = 20 * 30;$$

① 계산

$$A = 600 \quad \text{② assign}$$

$$A = \underline{\text{area()}};$$

① A에 할당할 값을 계산

: area() 함수가 돌려주는 값

```
class Circle {  
    int r;
```

```
    Circle( ) { r = 0; }   생성자 ①   no input  
    Circle(int n) { r = n; }   생성자 ②   one input
```

```
    double area() { return 3.14*r*r; }
```

```
    void display() {  
        System.out.println( "radius=" + r );  
        System.out.println( "Area=" + area() );  
    };
```

```
}|
```

```
public class CircleTest {
```

```
    /**  
     * @param args  
     */
```

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

```
        Circle C1 = new Circle( );   생성자 ① call  
        Circle C2 = new Circle( 20 );   생성자 ② call  
        Circle C3 = new Circle( 30 );   생성자 ② call
```

```
        C1.display();  
        C2.display();  
        C3.display();
```

```
    }
```

```
}
```