



# Week 3. 배열, 상속

## ❖ CarArrayTest

```
for(int i; i < cars.length; i++)  
    cars[i] = new Cars();
```

### CarArrayTest.java

```
01  class Car {  
02      public int speed;           // 속도  
03      public int gear;           // 기어  
04      public String color;       // 색상  
05  
06      public Car() {  
07          speed = 0;  
08          gear = 1;  
09          color = "red";  
10      }  
11      public void speedUp() {     // 속도 증가 메소드  
12          speed += 10;  
13      }  
14      public String toString() {  // 객체의 상태를 문자열로 반환하는 메소드  
15          return "속도: " + speed + " 기어: " + gear + " 색상: " + color;  
16      }  
17  }  
18
```

- java.lang.Object 클래스에 정의 됨.
- 객체가 가지고 있는값들을 문자열로 출력



```
19 public class CarArrayTest {
20     public static void main(String[] args) {
21         final int NUM_CARS = 5;
22         Car[] cars = new Car[NUM_CARS];
23         for (int i = 0; i < cars.length; i++)
24             cars[i] = new Car();
25         for (int i = 0; i < cars.length; i++)
26             cars[i].speedUp();
27         for (int i = 0; i < cars.length; i++)
28             System.out.println(cars[i]);
29     }
30 }
```

배열 객체 생성

배열의 각 원소에 Car 객체를  
생성하여 대입한다.

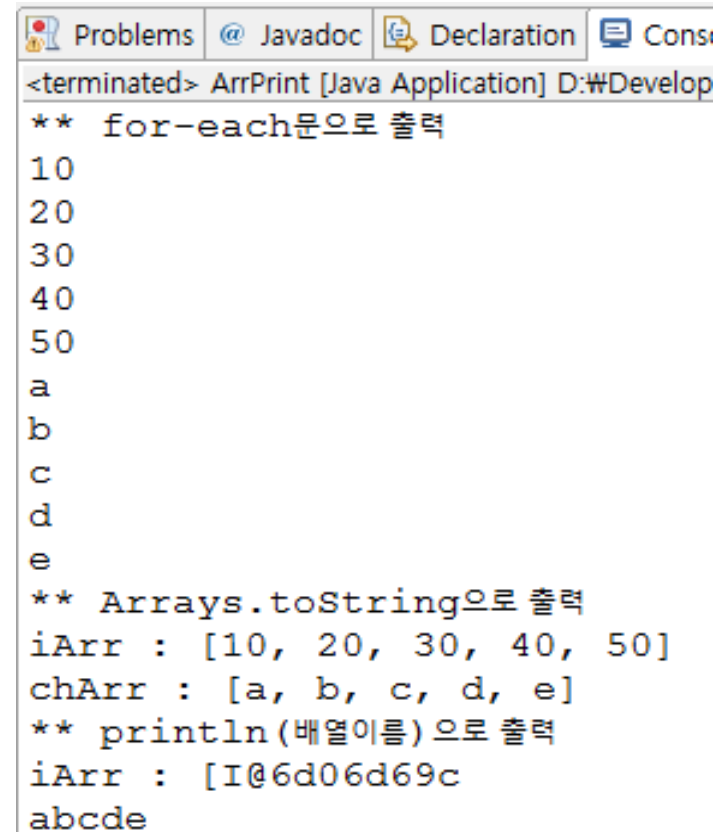
- 배열을 출력하면 toString() 이 자동 호출
- toString() 을 오버라이딩 하지 않으면 object 클래스의 toString() 이 출력됨

### 실행결과

```
속도: 10 기어: 1 색상: red
속도: 10 기어: 1 색상: red
속도: 10 기어: 1 색상: red
속도: 10 기어: 1 색상: red
속도: 10 기어: 1 색상: red
```

## ○ Week3\_1. 배열 출력

- ❖ `Arrays.toString(배열 이름) : [ ... ]` 형태로 배열 원소가 출력됨
- ❖ 다음 두개의 배열을 초기화
  - ❑ 크기가 5인 정수배열 - {10, 20, 30, 40, 50}
  - ❑ 크기가 5인 문자 배열 - {'a', 'b', 'c', 'd', 'e'}
- ❖ 각 배열을 다음 방법으로 출력한다.
  - ❑ for-each문 이용
  - ❑ `Arrays.toString()` 을 이용  
ex) `int[] iArr = new int[5];`  
`System.out.println(Arrays.toString(iArr));`
  - ❑ `println(배열이름)`을 이용



```
<terminated> ArrPrint [Java Application] D:\#Develop
** for-each문으로 출력
10
20
30
40
50
a
b
c
d
e
** Arrays.toString으로 출력
iArr : [10, 20, 30, 40, 50]
chArr : [a, b, c, d, e]
** println(배열이름)으로 출력
iArr : [I@6d06d69c
abcde
```

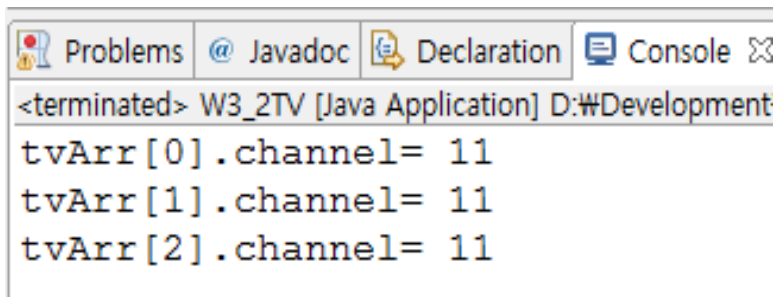
## ○ Week3\_2. 객체 배열

```
❖ class TV {  
    String color;  
    boolean power;  
    int channel;  
    public void power() { power을 toggle 한다 }  
    public void channelUp() { 채널을 높인다 }  
    public void channelDown() { 채널을 낮춘다 }  
}
```

### ❖ main()

- ❑ 길이가 3인 TV타입의 참조 변수 배열을 생성
- ❑ 객체를 생성해서 배열의 각 요소에 저장
- ❑ 배열의 각 channel 값에 +10을 저장
- ❑ 객체 배열의 channelUp()을 호출해 channel을 증가시킨다

### ❖ 실행 결과



```
<terminated> W3_2TV [Java Application] D:\#Development  
tvArr[0].channel= 11  
tvArr[1].channel= 11  
tvArr[2].channel= 11
```

## ○ Week3\_3. 상속 & 생성자

❖ 다음 코드를 수행한 후 결과를 알아보세요

```
❖ class ParentClass {  
    public void print() { “슈퍼클래스 메소드” 출력 }  
    생성자 { “parent 생성자” 출력 }  
}  
public class ChildClass extends ParentClass {  
    public void print() { “서브클래스 메소드” 출력 }  
    생성자 { “child 생성자” 출력 }  
    main() {  
        ChildClass a = new ChildClass();  
        ①  
    }  
}
```

❖ ①에 a.print() 을 추가해 본다.

❖ 서브 클래스의 기본 생성자에 대해 슈퍼 클래스의 기본 생성자가 묵시적으로 호출

## ○ Week3\_4

❖ 다음 프로그램의 결과를 알아보세요.

```
1 class A {  
2     public A() {  
3         System.out.println("생성자A");  
4     }  
5     public A(int x) {  
6         System.out.println("매개변수 생성자A");  
7     }  
8 }  
9  
10 class B extends A {  
11     public B() {  
12         System.out.println("생성자B");  
13     }  
14     public B(int x) {  
15         System.out.println("매개변수 생성자B");  
16     }  
17 }  
18  
19 public class W3_5Constructor {  
20  
21     public static void main(String[] args) {  
22         // TODO Auto-generated method stub  
23         B b = new B();  
24     }  
25 }
```

super(x); 를 삽입하여 실행해 본다

B b = new B(5); 로 바꾸어 실행해 본다

## ○ Week3\_5

❖ 다음 코드의 출력 결과를 예상해 보세요.

```
1  class Animal
2  {
3      String name;
4  public Animal() {
5      name = "UNKNOWN";
6      System.out.println("동물입니다 : " + name);
7  }
8  public Animal(String name) {
9      this.name = name;
10     System.out.println("동물입니다. : " + name);
11 }
12 }
13
14 class Lion extends Animal
15 {
16 public Lion() {
17     System.out.println("사자입니다.");
18 }
19 public Lion(String name) {
20     super(name);
21     System.out.println("매개변수-사자입니다.");
22 }
23 }
24
25 public class W3_6Cons {
26
27 public static void main(String[] args) {
28     // TODO Auto-generated method stub
29     Lion lion = new Lion("Brave");
30     Lion lion2 = new Lion();
31 }
32 }
```

Problems @ Javadoc

<terminated> W3\_6Cons [Java A

동물입니다. : Brave  
매개변수-사자입니다.  
동물입니다 : UNKNOWN  
사자입니다.



## ○ Week3\_6

❖ 상속받은 클래스의 생성자에서는 슈퍼클래스의 기본 생성자() 가 자동호출됨.

```
class Animal
{
    String name;
    public Animal() {
        name = "UNKNOWN";
        System.out.println("동물입니다 : " + name);
    }
    public Animal(String name) {
        this.name = name;
        System.out.println("동물입니다. : " + name);
    }
}

class Lion extends Animal
{
    public Lion() {
        System.out.println("사자입니다.");
    }
    public Lion(String name) {
        // super(name);
        System.out.println("매개변수-사자입니다.");
    }
}
```

Problems @ Javadoc

<terminated> W3\_6Cons [Java A

동물입니다 : UNKNOWN  
매개변수-사자입니다.  
동물입니다 : UNKNOWN  
사자입니다.

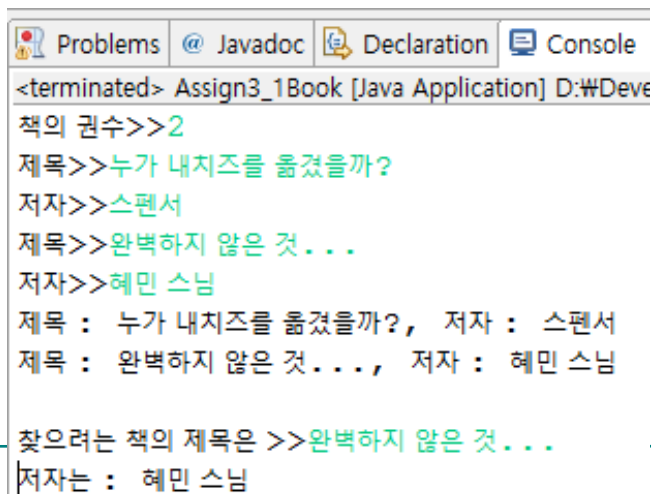
```
25 public class W3_6Cons {
26
27     public static void main(String[] args) {
28         // TODO Auto-generated method stub
29         Lion lion = new Lion("Brave");
30         Lion lion2 = new Lion();
31     }
32 }
```

## ○ Assignment3\_1

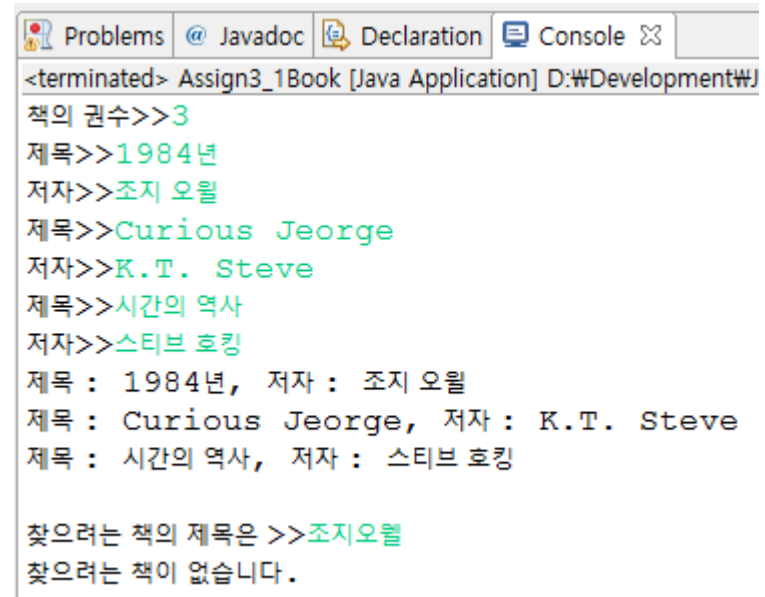
❖ Week3\_3 을 이용하여 다음을 완성하시오. (equals() 이용해 책을 찾는다)

```
❑ class Book {  
    getter 설정  
    toString()  
}  
main() {  
    배열의 개수를 입력받아 처리  
    입력받은 배열 출력  
    찾으려는 책의 제목을 입력하고, 존재하면 getter를 이용하여 저자를 출력한다.  
    없으면 없음을 출력  
}
```

❖ 실행 결과



```
<terminated> Assign3_1Book [Java Application] D:\Development\WJ  
책의 권수>>2  
제목>>누가 내치즈를 훔겼을까?  
저자>>스펜서  
제목>>완벽하지 않은 것...  
저자>>헤민 스님  
제목 : 누가 내치즈를 훔겼을까?, 저자 : 스펜서  
제목 : 완벽하지 않은 것..., 저자 : 헤민 스님  
찾으려는 책의 제목은 >>완벽하지 않은 것...  
저자는 : 헤민 스님
```

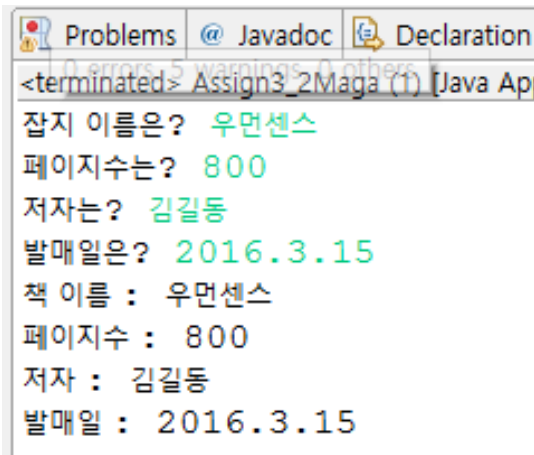


```
<terminated> Assign3_1Book [Java Application] D:\Development\WJ  
책의 권수>>3  
제목>>1984년  
저자>>조지 오웰  
제목>>Curious George  
저자>>K.T. Steve  
제목>>시간의 역사  
저자>>스티브 호킹  
제목 : 1984년, 저자 : 조지 오웰  
제목 : Curious George, 저자 : K.T. Steve  
제목 : 시간의 역사, 저자 : 스티브 호킹  
찾으려는 책의 제목은 >>조지오웰  
찾으려는 책이 없습니다.
```

## ○ Assignment3\_2

- ❖ 책을 나타내는 **Book** 클래스를 상속받아서 잡지를 나타내는 **Magazine** 클래스를 작성한다.
  - ❑ Book클래스는 제목, 페이지수, 저자등의 정보를 가진다.
  - ❑ Magazine클래스는 추가로 발매일 정보를 가진다.
  - ❑ 생성자, 접근자, 설정자를 포함하여 각각의 클래스를 작성하고 각 객체의 모든 정보를 출력하는 테스트 클래스를 작성

### ❖ 실행 결과



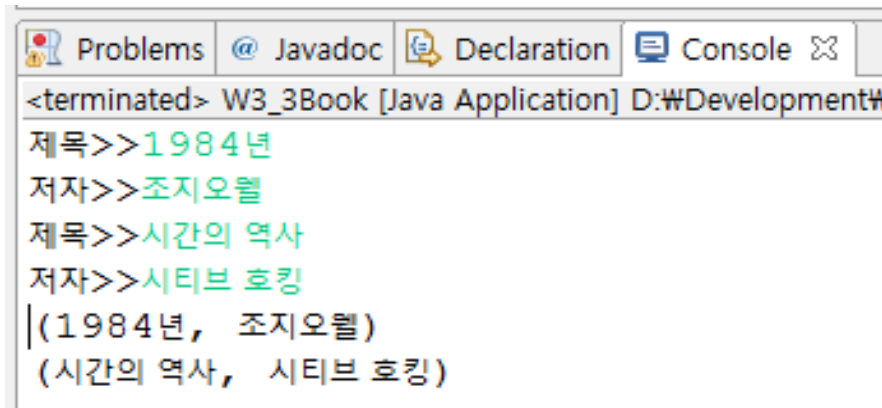
```
Problems Javadoc Declaration
<terminated> Assign3_2Maga (1) [Java Ap
잡지 이름은? 우먼센스
페이지수는? 800
저자는? 김길동
발매일은? 2016.3.15
책 이름 : 우먼센스
페이지수 : 800
저자 : 김길동
발매일 : 2016.3.15
```

## ○ Assignment3\_3

- ❖ **Book class** 를 이용하여 크기가 2인 **Book** 객체 배열을 만들고, 사용자로부터 책의 제목과 저자를 입력받아 배열을 출력하시오.

```
❑ class Book {  
    String title, author;  
    public Book(  ...  ) {      }    //생성자  
}  
main() {  
    Book객체 배열 선언  
    제목, 저자를 입력받아 생성자를 이용하여 객체배열 원소를 생성  
    배열의 값을 출력  
}
```

- ❖ 실행 결과



```
<terminated> W3_3Book [Java Application] D:\Development\
제목>>1984년
저자>>조지오웰
제목>>시간의 역사
저자>>시티브 호킹
|(1984년, 조지오웰)
(시간의 역사, 시티브 호킹)
```