

컴퓨터 프로그래밍1

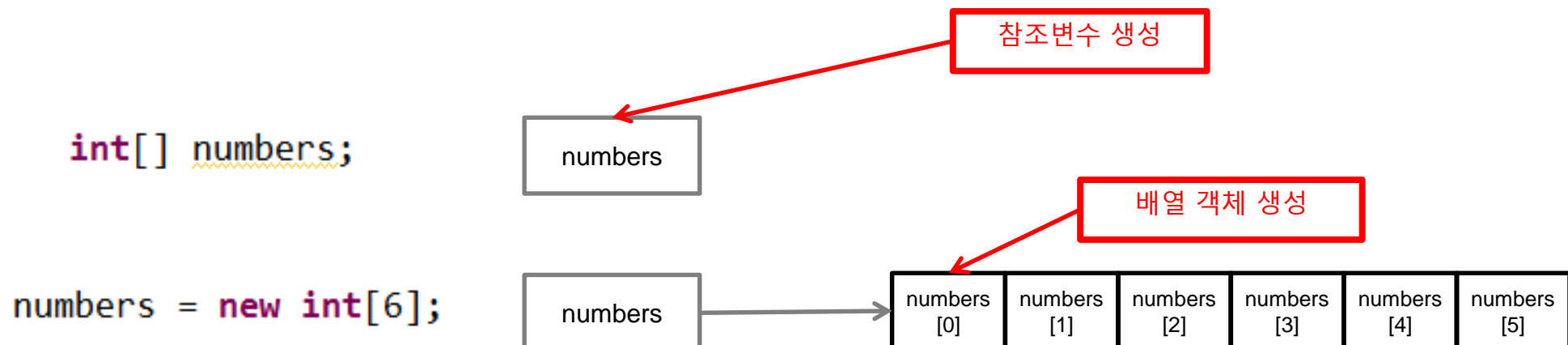
실습 12주차

- 배열 -

배열 (1/9)

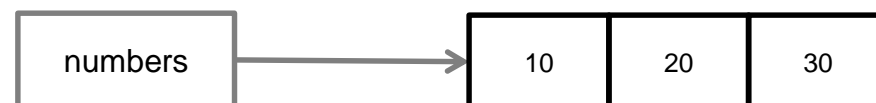
배열

- 배열은 같은 타입의 변수들의 모임
- 자바에서 배열은 객체이므로 다른 객체를 생성하는 것과 같이 참조변수를 선언하고 객체를 생성해야 함



```
int[] numbers = new int[6];  
float[] distances = new float[20];  
char[] letters = new char[50];
```

```
int[] numbers = {10, 20, 30};
```



배열 (2/9)

프로그래밍 예제

```
public static void main(String[] args) {  
    final int STUDENTS = 5;  
    int total = 0;  
    Scanner scan = new Scanner(System.in);  
    int[] scores = new int[STUDENTS];  
  
    for(int i=0;i< STUDENTS; i++) {  
        System.out.print("성적을 입력하시오: ");  
        scores[i] = scan.nextInt();  
        total += scores[i];  
    }  
    System.out.println("평균 성적은" + (float)total/STUDENTS + "입니다.");  
}
```

```
성적을 입력하시오: 30  
성적을 입력하시오: 40  
성적을 입력하시오: 50  
성적을 입력하시오: 83  
성적을 입력하시오: 24  
평균 성적은45.4입니다.
```

배열 (3/9)

인덱스 검사와 for-each 루프

- 배열의 인덱스에 대한 한도 검사
- 허용되는 인덱스 값보다 작거나 큰 값이 인덱스로 사용되면 실행 오류가 발생

```
int[] numbers = new int[5];  
numbers[5] = 10;
```

<terminated> Hello [Java Application] C:\Program Files\Java\jre\bin\javaw.exe (2013. 11. 13. 오후 9:44:40)

Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 5

```
for (자료형 변수 : 배열이름)  
{  
    // 반복 문장들  
}
```

```
int[] numbers = { 10, 20, 30 };  
for (int value : numbers)  
    System.out.println(value);
```

<terminated>

10
20
30

배열 (4/9)

사용자에 의한 배열 크기 지정

■ 사용자의 입력에 따라 배열의 크기 지정

```
public static void main(String[] args) {  
    int total = 0;  
    int size;  
    Scanner scan = new Scanner(System.in);  
    System.out.print("배열의 크기를 입력하시오:");  
    size = scan.nextInt();  
    int[] scores = new int[size];
```

변수로 배열의 크기를
지정할 수 있다.

```
    for (int i = 0; i < scores.length; i++) {  
        System.out.print("성적을 입력하시오:");  
        scores[i] = scan.nextInt();  
    }
```

배열 객체는 배열의 크기를 나타내는 length라는
필드를 가진다.

```
    for (int i = 0; i < scores.length; i++)  
        total += scores[i];
```

```
    System.out.println("평균 성적은 " + total / scores.length + "입니다");  
}
```

배열 (5/9)

메소드의 매개변수로 배열 전달

- 배열도 객체이기 때문에 매개변수로서 배열을 전달하는 것은 배열 참조 변수를 전달하는 것임

```
public class ArrayTest5 {  
    final static int STUDENTS = 5;  
  
    public static void main(String[] args) {  
        int[] scores = new int[STUDENTS];  
        getValues(scores);  
        getAverage(scores);  
    }  
  
    private static void getValues(int[] array) {  
        Scanner scan = new Scanner(System.in);  
        for (int i = 0; i < array.length; i++) {  
            System.out.print("성적을 입력하십시오:");  
            array[i] = scan.nextInt();  
        }  
    }  
  
    private static void getAverage(int[] array) {  
        int total = 0;  
        for (int i = 0; i < array.length; i++)  
            total += array[i];  
        System.out.println("평균 성적은 " + total / array.length + "입니다");  
    }  
}
```

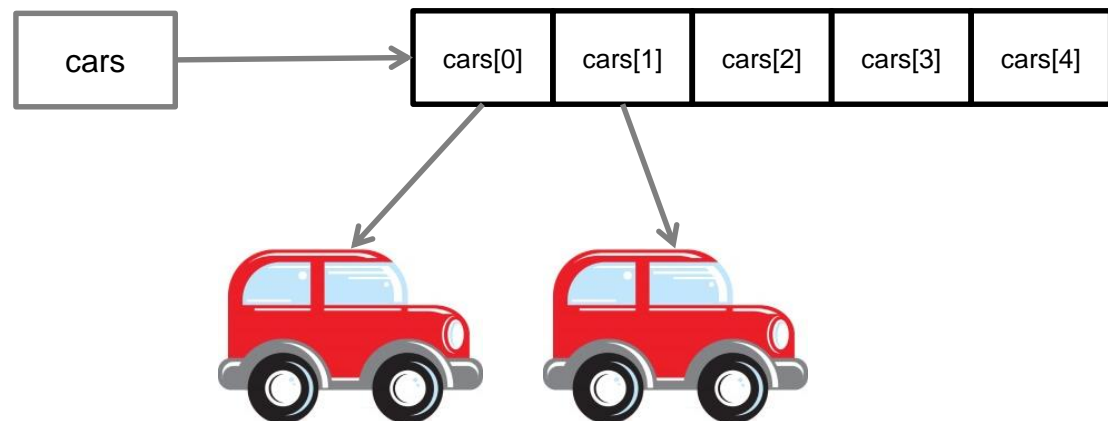
배열 객체의 참조 값이 전달된다.

배열 (6/9)

객체들의 배열

- 배열은 변수들의 모임이기 때문에 참조형 변수로 이루어진 배열도 생성 가능. 변수를 전달하는 것임

```
Car[] cars = new Car[5];  
  
cars[0] = new Car();  
cars[1] = new Car();
```



- 반복 루프를 이용하면 한번에 모든 객체 생성 가능

```
for(int i=0; i<cars.length; i++)  
    cars[i] = new Car();
```

배열 (7/9)

프로그래밍 예제 - P.231

```
class Car {
    public int speed; // 속도
    public int gear; // 주행거리
    public String color; // 색상
    // 생성자
    public Car() {
        speed = 0;
        gear = 1;
        color = "red";
    }
    // 속도 증가 메소드
    public void speedUp() {
        speed += 10;
    }
    // 객체의 상태를 문자열로 반환하는 메소드
    public String toString() {
        return "속도: " + speed
            + " 기어: " + gear
            + " 색상: " + color;
    }
}
```

```
public class CarArrayTest {
    public static void main(String[] args) {
        final int NUM_CARS = 5;
        Car[] cars = new Car[NUM_CARS];
        for (int i = 0; i < cars.length; i++)
            cars[i] = new Car();
        for (int i = 0; i < cars.length; i++)
            cars[i].speedUp();
        for (int i = 0; i < cars.length; i++)
            System.out.println(cars[i]);
    }
}
```

<terminated> Hello [Java Application] C:\Program File:

```
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
```


배열 (8/9)

2차원 배열

- 배열은 2차원이 될 수 있음

```
double[][] s = new double[3][5];
```

두개의 대괄호가
2차원 배열을 나타낸다.

행의 개수

열의 개수

s[0][0]	s[0][1]	s[0][2]	s[0][3]	s[0][4]
s[1][0]	s[1][1]	s[1][2]	s[1][3]	s[1][4]
s[2][0]	s[2][1]	s[2][2]	s[2][3]	s[2][4]

- 2차원 배열의 초기화도 중괄호 이용 가능

```
int[][] testArray = { {10, 20, 30}, {40, 50, 60}, {70, 80, 90} };
```

10	20	30
40	50	60
70	80	90

배열 (9/9)

2차원 배열, 다차원 배열

- 2차원 배열은 전체적으로 하나의 length 필드가 있고 이것은 행의 개수를 나타냄. 각 행마다 별도의 length 필드가 있고 이것은 각 행이 가지는 열의 개수를 나타냄

```
public static void main(String[] args) {  
    int[][] array = { { 10, 20, 30, 40 },  
                      { 50, 60, 70, 80 },  
                      { 90, 100, 110, 120 } };  
  
    for (int r = 0; r < array.length; r++) {  
        for (int c = 0; c < array[r].length; c++) {  
            System.out.println(r+"행"+c+"열 : "+array[r][c]);  
        }  
    }  
}
```

전체 행의 개수

각 행이 가지는
열의 개수

<terminated> Hello Jav

```
0행0열 : 10  
0행1열 : 20  
0행2열 : 30  
0행3열 : 40  
1행0열 : 50  
1행1열 : 60  
1행2열 : 70  
1행3열 : 80  
2행0열 : 90  
2행1열 : 100  
2행2열 : 110  
2행3열 : 120
```

- 다차원 배열도 생성 가능

```
double[][][] sales = new double[3][2][12];
```

실습 (1/2)

프로그래밍 실습 (I)

- 사용자로부터 정수를 입력받아 배열에 저장하고 출력하는 프로그램 작성

```
public static void main(String[] args) {  
  
    Scanner scan = new Scanner(System.in);  
    // 10개의 정수를 저장할 수 있는 배열을 선언하고 생성하라.  
    ;  
    // 사용자로부터 정수를 받아서 배열에 저장하는 반복 루프  
    ;  
    ;  
    // 배열에 저장된 정수를 출력하는 반복 루프  
    ;  
    ;  
}
```

- 위의 프로그램의 빈칸을 채우고 실행하라.
- 배열에 저장된 정수를 출력할 때, for-each 반복 구조를 사용해 보자.
- 배열에 저장된 정수를 역순으로 출력해 보라.

실습 (2/2)

프로그래밍 실습 (2)

- 회사에서 직원을 관리하는 프로그램 작성
- 직원을 나타내는 Employee 클래스는 이름, 주소, 연봉, 전화번호를 가짐
- 회사 전체 직원은 Employee 배열 안에 저장

1) 먼저 Employee 클래스를 작성하자.

2) Employee 클래스를 테스트하기 위하여 EmployeeTest 클래스를 작성한다. 주석에 기술된 대로, 객체 배열을 생성하고 새로운 데이터를 객체로 만들어서 객체 배열에 추가해보자. 또 배열에 저장된 모든 데이터를 출력해 본다.

```
class Employee{  
    String name;    // 직원의 이름  
    ; // 주소  
    ; // 연봉  
    ; // 전화 번호  
  
    // 각 필드에 대해 접근자와 설정자를 작성한다.  
}
```

```
public class EmployeeTest {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        // 크기가 3인 Employee의 배열 employees를 생성한다.  
        // ...  
        // 3명의 사원 정보를 받아서 각각 Employee 객체를 생성한 후에 배열에 추가해 본다.  
        // 반복 루프를 사용한다.  
        // ...  
        // employees 배열에 저장된 모든 데이터를 출력한다. 반복 루프를 사용한다.  
        // ...  
    }  
}
```

hw08 과제

과제I. 극장 예약 시스템 -P.237 I번

- 배열을 이용하여 간단한 극장 예약 시스템을 작성하여 보자. 아주 작은 극장이라서 좌석이 10개밖에 안 된다. 사용자가 예약을 하려고 하면 먼저 좌석 배치표를 보여준다. 즉 예약이 끝난 좌석은 1로, 예약이 안된 좌석은 0으로 나타낸다

```
좌석을 예약하시겠습니까? (1 또는 0) 1
현재의 예약 상태는 다음과 같습니다.
-----
1 2 3 4 5 6 7 8 9 10
-----
0 0 0 0 0 1 1 1 0 0
몇번째 좌석을 예약하시겠습니까? 1
예약되었습니다.
좌석을 예약하시겠습니까? (1 또는 0) 0
```

Q&A



수고하셨습니다