

컴퓨터 프로그래밍

Array

배열의 필요성

- 5개의 정수를 받아서 역순으로 출력?

```
import java.util.Scanner;

public class Test {

    public static void main(String[] args) {
        int x0, x1, x2, x3, x4;
        Scanner input = new Scanner(System.in);

        x0 = input.nextInt();
        x1 = input.nextInt();
        x2 = input.nextInt();
        x3 = input.nextInt();
        x4 = input.nextInt();

        System.out.println(x4);
        System.out.println(x3);
        System.out.println(x2);
        System.out.println(x1);
        System.out.println(x0);

    }
}
```

1
2
3
4
5
5
4
3
2
1

배열

- 같은 자료형의 자료 여러 개를 하나로 묶어서 표현
- 대량의 변수를 편리하게 선언
- 반복적인 코드에서 반복문 사용 가능

```
import java.util.Scanner;

public class Test {

    public static void main(String[] args) {
        int[] x = new int[5];
        Scanner input = new Scanner(System.in);

        for (int i=0 ; i<5 ; i++)
            x[i] = input.nextInt();

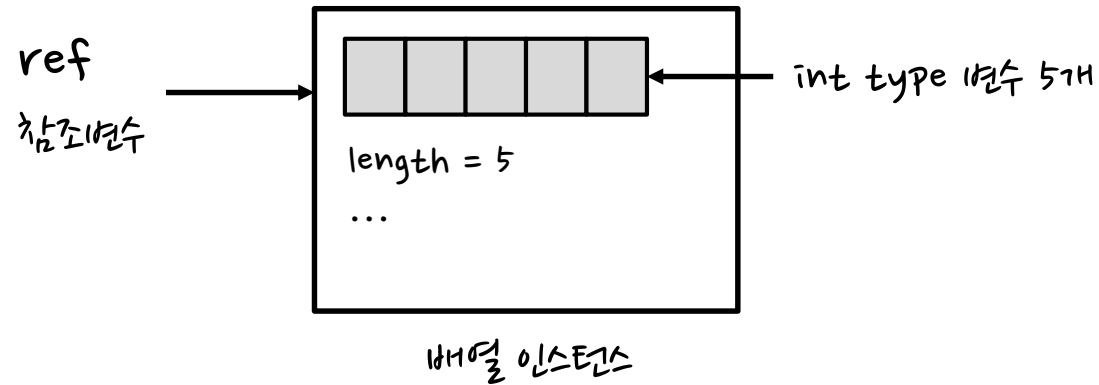
        for (int i=4 ; i>=0 ; i--)
            System.out.println(x[i]);

    }
}
```

배열 생성

- 배열도 인스턴스!

```
int[] ref = new int[5];
```



배열 접근

- 첨자 연산자(index operator) [] 사용
- 첨자 연산자 내에 0부터 시작하는 index 값으로 각 element에 접근
- 배열 인스턴스의 멤버 변수 length는 배열의 길이 정보 저장

```
int[] ref = new int[5];
```

ref : 배열 인스턴스의 주소를 가지고 있는 참조변수

ref[0] ~ ref[4] : int형 변수

ref.length == 5

Primitive vs. User-defined(class)

- 기본 자료형 배열

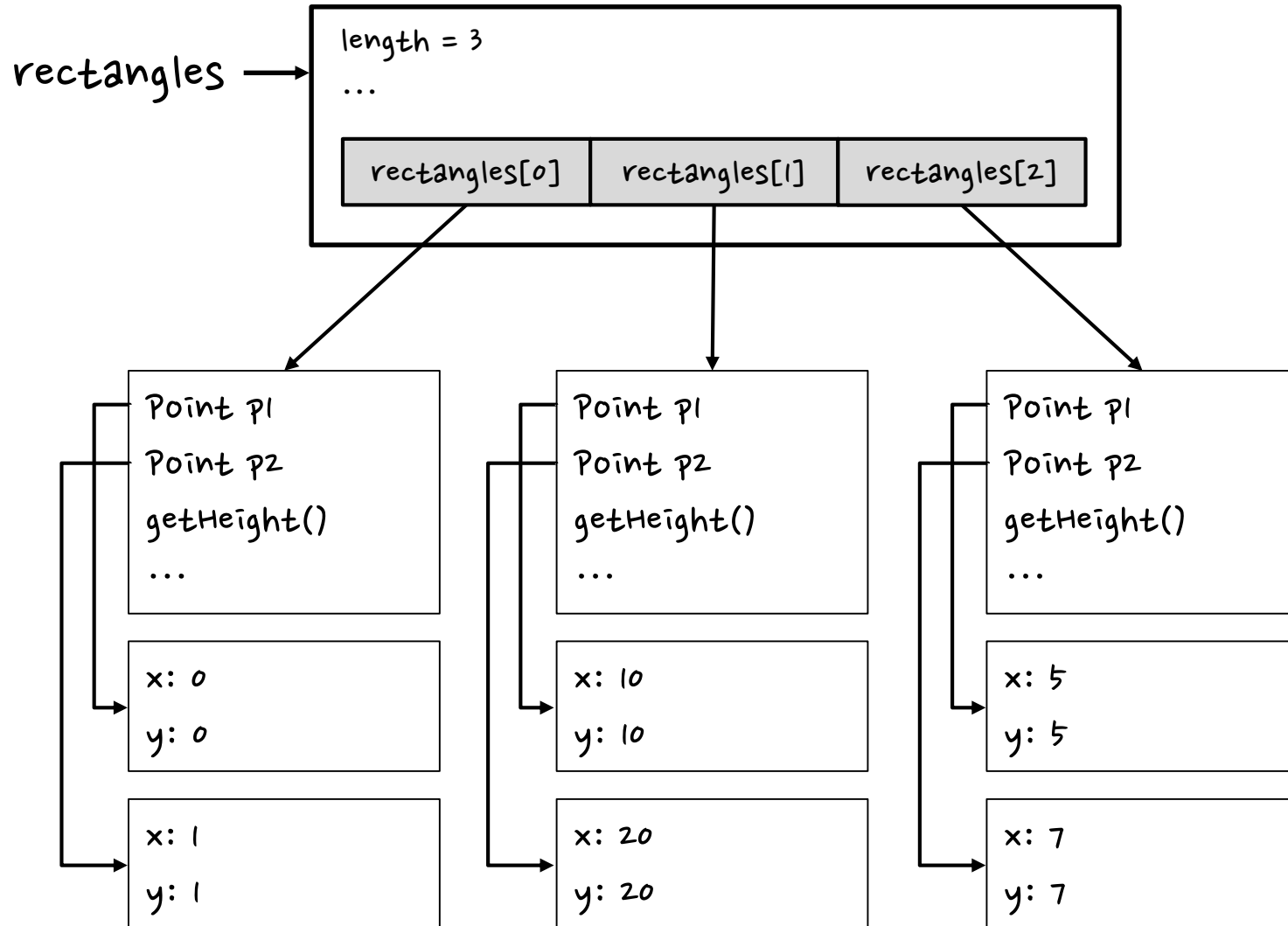
- `int[] arr = new int[3];`
`arr[0] = 1;`
`arr[1] = 2;`
`arr[2] = 3;`

- 객체 배열

- `Rectangle[] rectangles = new Rectangle[3];`
`rectangles[0] = new Rectangle();`
`rectangles[1] = new Rectangle(10, 10, 20, 20);`
`rectangles[2] = new Rectangle(new Point(5, 5),`
`new Point(7, 7));`

- 객체 배열에는 객체가 저장되는 것이 아니라 참조(주소)값이 저장

Primitive vs. User-defined(class)



Initialization

- 초기화 데이터들을 중괄호 안에 나열
- 초기화 데이터가 있는 경우 길이가 생략
 - 초기화 데이터의 수를 통해 길이를 자동 계산
- Example
 - 초기화 없는 경우
 - `int[] arr = new int[3];`
 - 초기화 데이터가 있는 경우
 - `int[] arr = new int[] {1, 2, 3};`
 - 줄여서 표현 가능
 - `int[] arr = {1, 2, 3};`
- 객체 배열은 어떻게 초기화 데이터를 나열할 수 있나?

Parameter Passing of Array

```
class ArrayAndMethods {  
    public static int[] addAllArray(int[] ar, int addval) {  
        for (int i=0 ; i<ar.length ; i++)  
            ar[i] += addval;  
        return ar;  
    }  
    public static void main(String arg[]) {  
        int[] arr = {1, 2, 3, 4, 5};  
        int[] ref;  
        ref = addAllArray(arr, 7);  
        for (int i=0 ; i<ref.length ; i++)  
            System.out.println(ref[i] + " ");  
        for (int i=0 ; i<arr.length ; i++)  
            System.out.println(arr[i] + " ");  
    }  
}
```

Result?

시스
즈

- Rectangle class 구현
- Rectangle의 배열을 생성하고 3개의 사각형 데이터 입력
- RectangleManager class 구현
 - Rectangle 배열 내 모든 사각형의 면적 합을 구하는 sumOfAreas() method 구현

Multi-Dimensional Array

- 2차원 배열

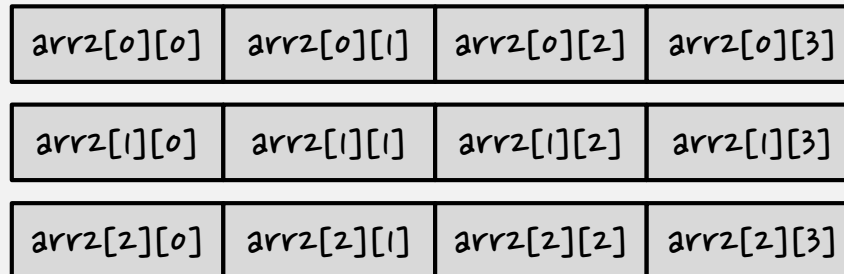
```
int[] arr1 = new int[4];
```

arr1



```
int[][] arr2 = new int[3][4];
```

arr2



Multi-Dimensional Array

- Example

```
public static void main(String arg[]) {  
    int[][] arr = new int[3][4];  
    for (int i=0 ; i<arr.length ; i++)  
        for (int j=0 ; j<arr[i].length ; j++)  
            arr[i][j] = i+j;  
  
    for (int i=0 ; i<arr.length ; i++) {  
        for (int j=0 ; j<arr[i].length ; j++)  
            System.out.println(arr[i][j]+" ");  
        System.out.println();  
    }  
}
```

Multi-Dimensional Array

- 2차원 배열의 초기화

```
int[][] arr = new int[][] { {1, 2, 3, 4}, {5, 6, 7, 8}, {9, 10, 11, 12} };
```

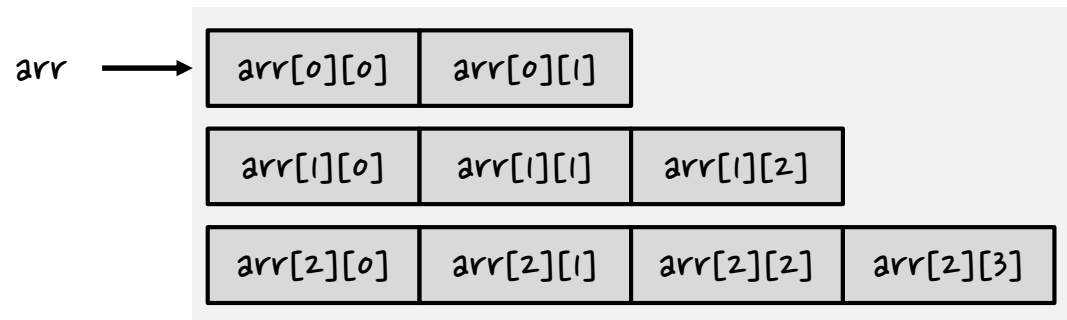
```
int[][] arr = { {1, 2, 3, 4}, {5, 6, 7, 8}, {9, 10, 11, 12} };
```

```
int[][] arr = { {1, 2}, {5, 6, 7}, {9, 10, 11, 12} };
```

```
arr[0].length = 2
```

```
arr[1].length = 3
```

```
arr[2].length = 4
```



시스템

- 배예를 이용하여 극장 예약 시스템 구현
 - 극장의 좌석은 10개씩 2줄로 되어있다
 - 앞 줄은 10000원, 뒤 줄은 20000원
 - 해당 자리가 예약되어 있지 않으면 0, 예약되어 있으면 1로 상태를 보여준다
 - 사용자는 자리 번호를 입력하여 예약하며, 이 때 금액을 보여준다

```
좌석을 예약하시겠습니까? (1/0) 1
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
몇 번 자리를 예약하시겠습니까?
0 5
예약되었습니다. 가격은 10000원 입니다.
좌석을 예약하시겠습니까? (1/0) 1
0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0
몇 번 자리를 예약하시겠습니까?
0 5
예약할 수 없습니다.
좌석을 예약하시겠습니까? (1/0) 1
0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0
몇 번 자리를 예약하시겠습니까?
1 1
예약되었습니다. 가격은 20000원 입니다.
좌석을 예약하시겠습니까? (1/0) 0
종료합니다
```

시작
종료

```
class Seat
    private int fee
    private boolean isReserved

    public Seat(int fee, boolean isReserved)
    public boolean isReserved()
    public void reserve()
    public int getFee()
```

```
class Reservation
    private Seat[][] s

    public Reservation()
    public void showStatus()
    public boolean reserve(int row, int col)
```

