

배열

융합프로그래밍

학습 목표

- 배열이 어떻게 동작하는지 이해한다.
- 배열이 필요한 곳에 쓸 수 있는 능력을 배운다.

실습 #1

10 개의 숫자를 입력받아서 그 합을 구하고, 합과 입력된 숫자를 출력하는 프로그램을 작성하시오.

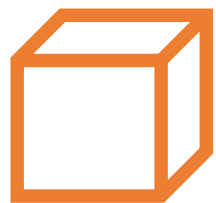
이전 프로그램의 단점

- 10개의 숫자를 입력받기 위해서, 10개의 변수를 뒤야 하나?
 - `int a, b, c, d, e, f, g, h, i, j;`

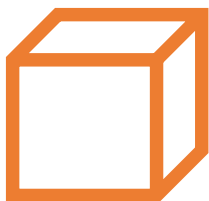
배열

- 같은 데이터 타입 여러 개를 나란히 두어 하나의 이름으로 사용하는 것

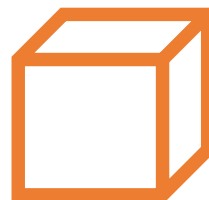
따로 따로 변수를 선언할 때



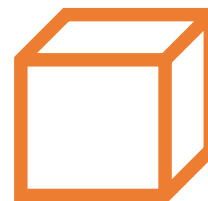
`int a;`



`int b;`



`int c;`

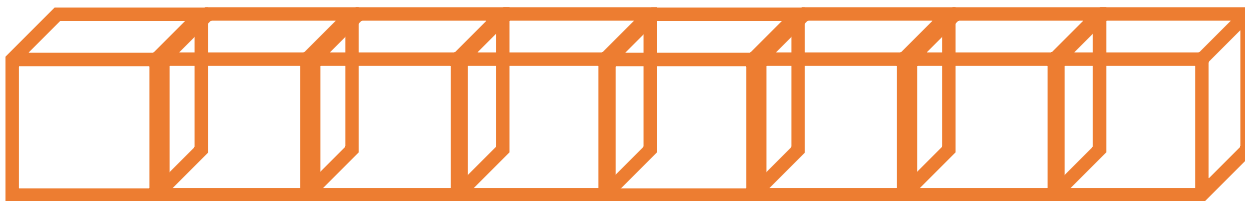


`int d;`

배열

- 같은 데이터 타입 여러 개를 나란히 두어 하나의 이름으로 사용하는 것

배열로 선언할 때



```
int a[] = new int[10];
```

배열

- 배열은 0번부터 시작해서 $a[N]$ 로 선언했다면 $N-1$ 까지로 지정할 수 있다.

배열로 선언할 때



```
int a[] = new int[8];
```

$a[6] = 1;$

실습 #2 - Array.java

```
import java.util.Scanner;

public class Array {
    public static void main(String...args) {
        Scanner sc = new Scanner(System.in);
        int a[] = new int[10];

        for (int i = 0; i < 10; i++) {
            a[i] = sc.nextInt();
        }
    }
}
```

```
int sum = 0;
for (int i = 0; i < 10; i++) {
    sum = sum + a[i];
}

System.out.println("Sum ==> " + sum);
for (int i = 0; i < 10; i++) {
    System.out.println(i + " ==> "
                       + a[i]);
}
}
```


배열의 선언

```
int [] array = new int[10];
```

```
int array[] = new int[10];
```

```
int array[];
```

```
array = new int[10];
```

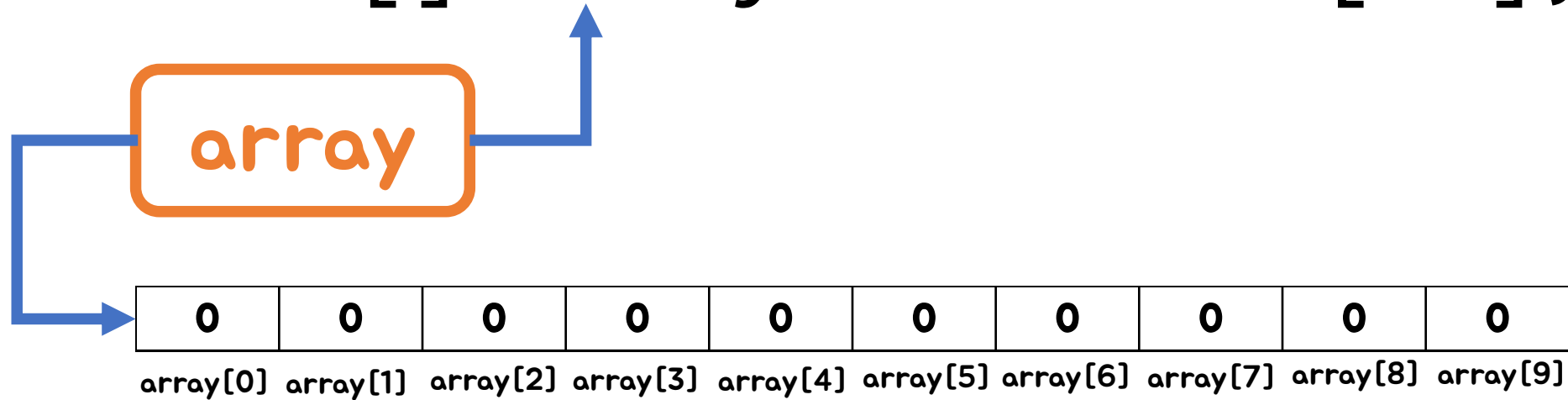
배열의 선언

`int [] array = new int[10];`

데이터 타입 배열이라는 표시 변수명 할당 명령 데이터 타입 배열 크기

배열의 선언

```
int [] array = new int[10];
```



배열의 초기화 #1

- 처음 배열에 값을 할당하는 방법

```
int [] array = new int [5];  
array[0] = 0;  
array[1] = 1;  
array[2] = 2;  
array[3] = 3;  
array[4] = 4;
```

배열의 초기화 #2

- 처음 배열에 값을 할당하는 방법

```
int [] array = {0, 1, 2, 3, 4};
```

```
int [] array = new int [] {0, 1, 2, 3, 4};
```

```
int [] array;
```

```
array = new int [] {0, 1, 2, 3, 4};
```

배열의 초기화 #3

- 처음 배열에 값을 할당하는 방법

```
int [] array;  
array = {0, 1, 2, 3, 4};
```



초기화 오류
컴파일 오류

실습 #3 Array.java

```
import java.util.Scanner;

public class ArrayLength {
    public static void main(String...args) {
        Scanner sc = new Scanner(System.in);
        int a[] = new int[10];

        for (int i = 0; i < a.length; i++) {
            a[i] = sc.nextInt();
        }
    }
}
```

```
int sum = 0;
for (int i = 0; i < a.length; i++) {
    sum = sum + a[i];
}

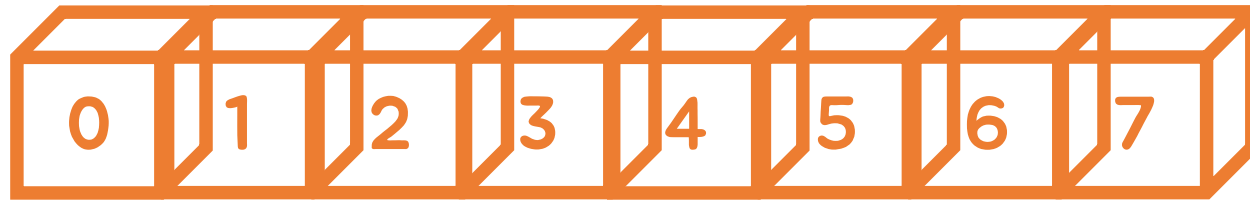
System.out.println("Sum ==> " + sum);
for (int i = 0; i < a.length; i++) {
    System.out.println(i + " ==> "
                        + a[i]);
}
}
```

배열의 길이 length #1

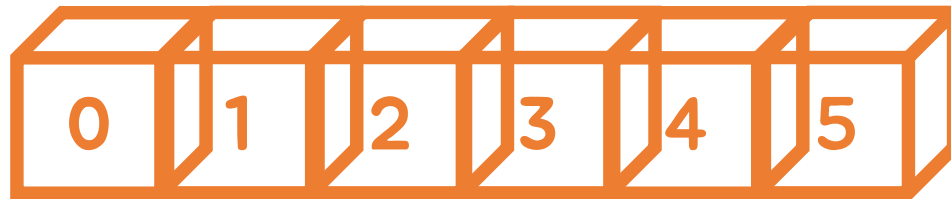
- 배열은 length 라는 내부 변수를 가진다.

```
public class ArrayLength {  
    public static void main(String...args) {  
        int a[] = {0, 1, 2, 3, 4};  
        System.out.println("a.length ==> " + a.length);  
    }  
}
```


배열의 길이 length #2



```
int a[] = new int[8];  
a.length //8
```



```
int a[] = new int[6];  
a.length //6
```

실습 #4 - CopyArray.java

```
public class CopyArray {  
    public static void main(String...args) {  
        int a[] = new int[]{1,2,3,4,5};  
        int b[] = new int[10];  
  
        for (int i = 0; i < a.length; i++) {  
            b[i] = a[i];  
        }  
  
        for (int i = 0; i < b.length; i++) {  
            System.out.println(i + " ==> " + b[i]);  
        }  
    }  
}
```

실습 #5 - CopyArray2.java

```
int a[] = new int[]{1,2,3,4,5};  
int b[] = new int[]{2,4,6,8,10};  
int c[] = new int[10];
```

을 넣고 다음과 같은 결과가 나오도록 작성하시오.

a[0]	b[0]	a[1]	b[1]	a[2]	b[2]	a[3]	b[3]	a[4]	b[4]
------	------	------	------	------	------	------	------	------	------

실습 #6 - CopyArray3.java

```
public class CopyArray3 {  
    public static void main(String...args) {  
        int a[] = new int[]{1,2,3,4,5};  
        int b[] = new int[10];  
  
        System.arraycopy(a, 0, b, 0, a.length);  
  
        for (int i = 0; i < b.length; i++) {  
            System.out.println(i + " ==> " + b[i]);  
        }  
    }  
}
```

System.arraycopy #1

```
System.arraycopy(a, 0, b, 0, a.length);
```

복사할 원본

원본에서의 복사할
시작위치

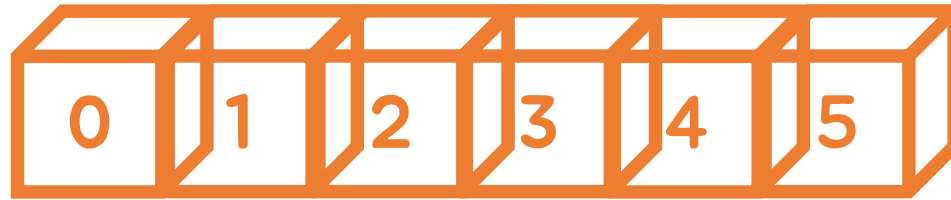
데이터를 복사받을
대상

복사받을 시작 위치

복사할 크기

System.arraycopy #2

a



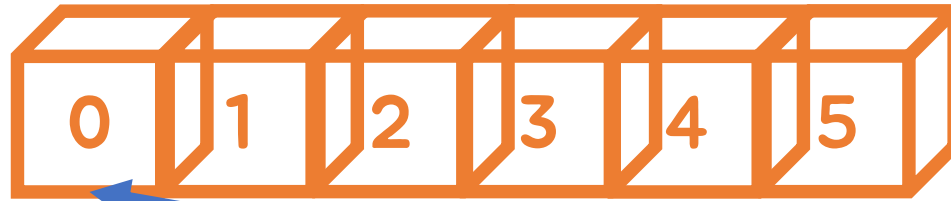
```
System.arraycopy(a, 0, b, 0, a.length);
```

b



System.arraycopy #3

a



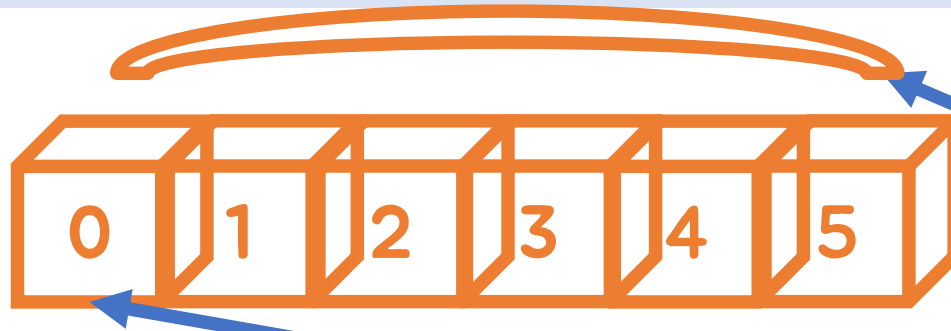
```
System.arraycopy(a, 0, b, 0, a.length);
```

b



System.arraycopy #4

a



5

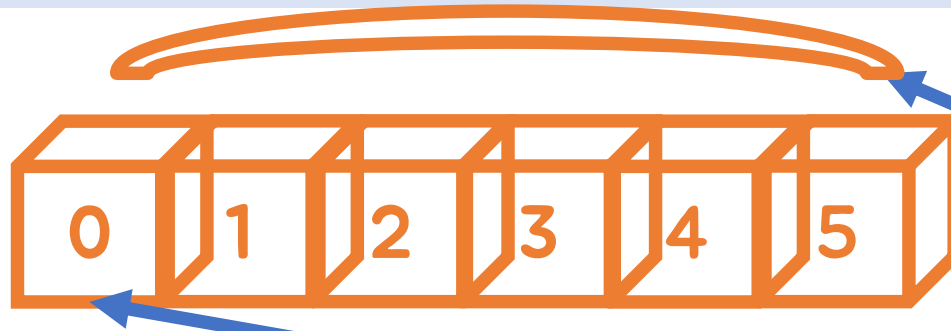
```
System.arraycopy(a, 0, b, 0, a.length);
```

b



System.arraycopy #5

a



6

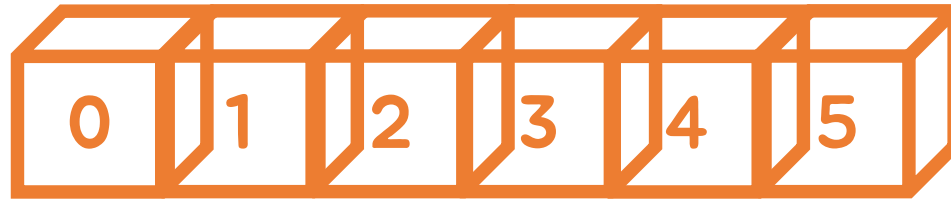
```
System.arraycopy(a, 0, b, 0, a.length);
```

b



System.arraycopy #6

a



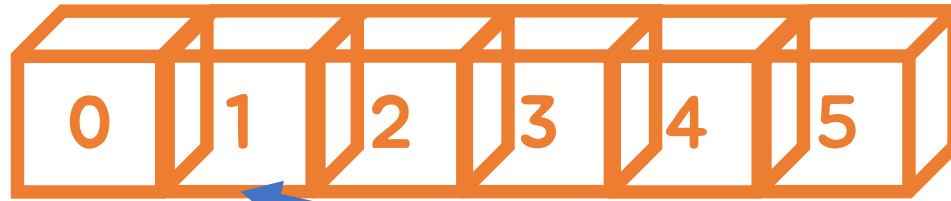
```
System.arraycopy(a, 1, b, 2, 4);
```

b



System.arraycopy #7

a



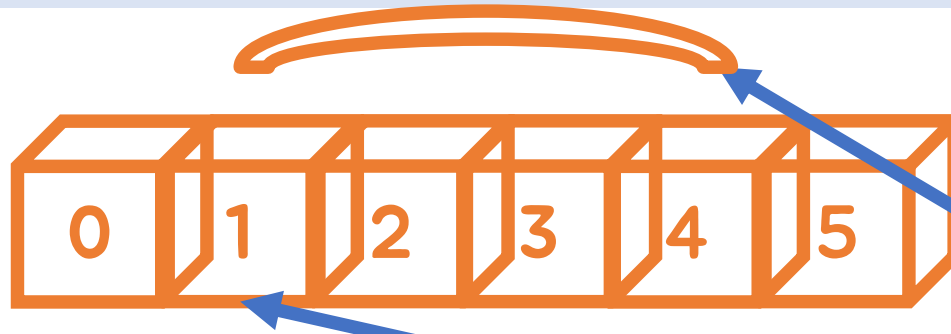
`System.arraycopy(a, 1, b, 2, 4);`

b



System.arraycopy #8

a



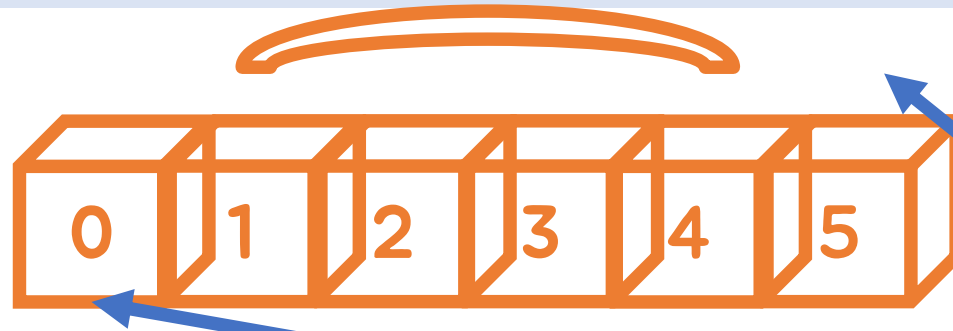
`System.arraycopy(a, 1, b, 2, 4);`

b



System.arraycopy #9

a



`System.arraycopy(a, 1, b, 2, 4);`

b



실습 #5 - CopyArray4.java

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

을 넣고 System.arraycopy를 이용하여
다음과 같은 결과가 나오도록 작성하시오.

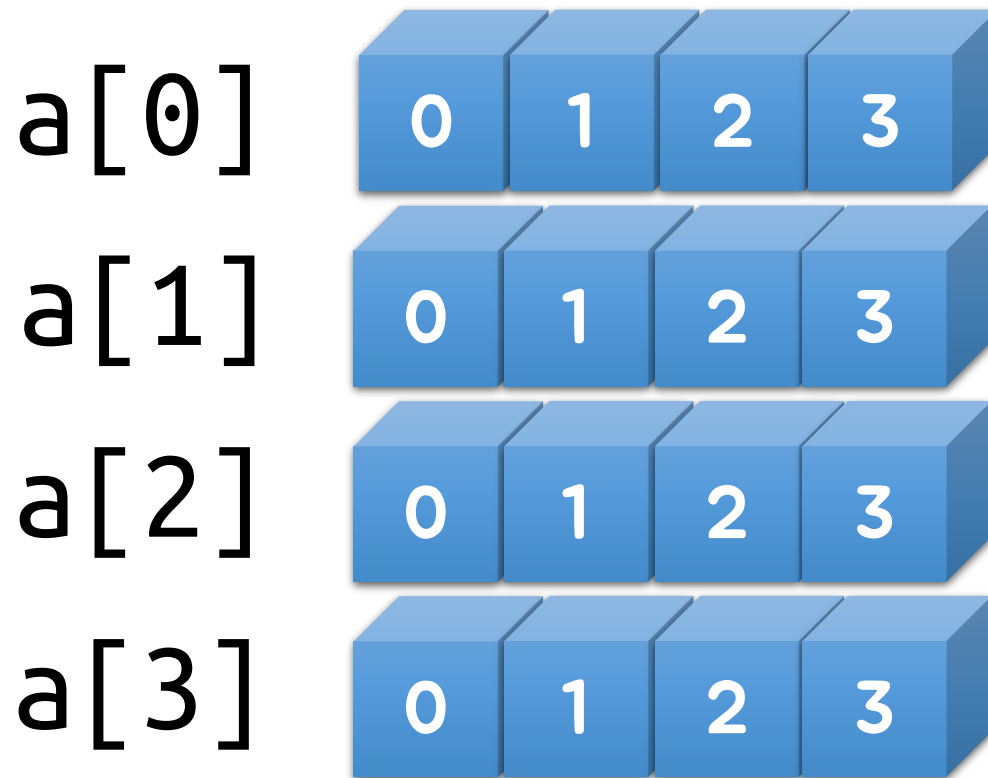
0	0	0	0	0	a[0]	a[1]	a[2]	a[3]	a[4]
---	---	---	---	---	------	------	------	------	------

실습 #6 - TwoDimensionArray.java

```
public class TwoDimensionArray {  
    public static void main(String...args) {  
        int a[][] = {  
            {1, 1, 1, 1},  
            {2, 2, 2, 2},  
            {3, 3, 3, 3},  
            {4, 4, 4, 4}  
        };  
        for (int i = 0; i < a.length; i++) {  
            for (int j = 0; j < a[i].length; j++) {  
                System.out.print(a[i][j] + ", ");  
            }  
            System.out.println("");  
        }  
    }  
}
```

이차원 배열

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



실습 #7 - For.java

```
public class For{  
    public static void main(String...args) {  
        int a[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};  
  
        for (int n : a) {  
            System.out.println(n);  
        }  
    }  
}
```

배열이나 Collection을 위한 For 의 사용법

```
for (int n : array) {  
  
}
```

배열의 데이터 타입

배열의 값을 가져올 변수

배열

For 사용법

```
public class For{  
    public static void main(String...args) {  
        int a[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};  
  
        for (int n : a) { ①  
            System.out.println(n); ②  
        }  
    }  
}
```

1,2 과 배열의 크기 만큼 반복된다. 해당 문법에서 n은 첫번째 반복에는 a[0], 두번째 반복에는 a[1], 세번째는 a[2] 등, N번째 반복에서 a[N-1]이 된다.

배열이나 Collection을 위한 For 의 사용법

```
for (int n : array) {  
    .....  
}
```

서로 비슷

```
for (int i = 0; i < array.length; i++) {  
    int n = array[i];  
    .....  
}
```

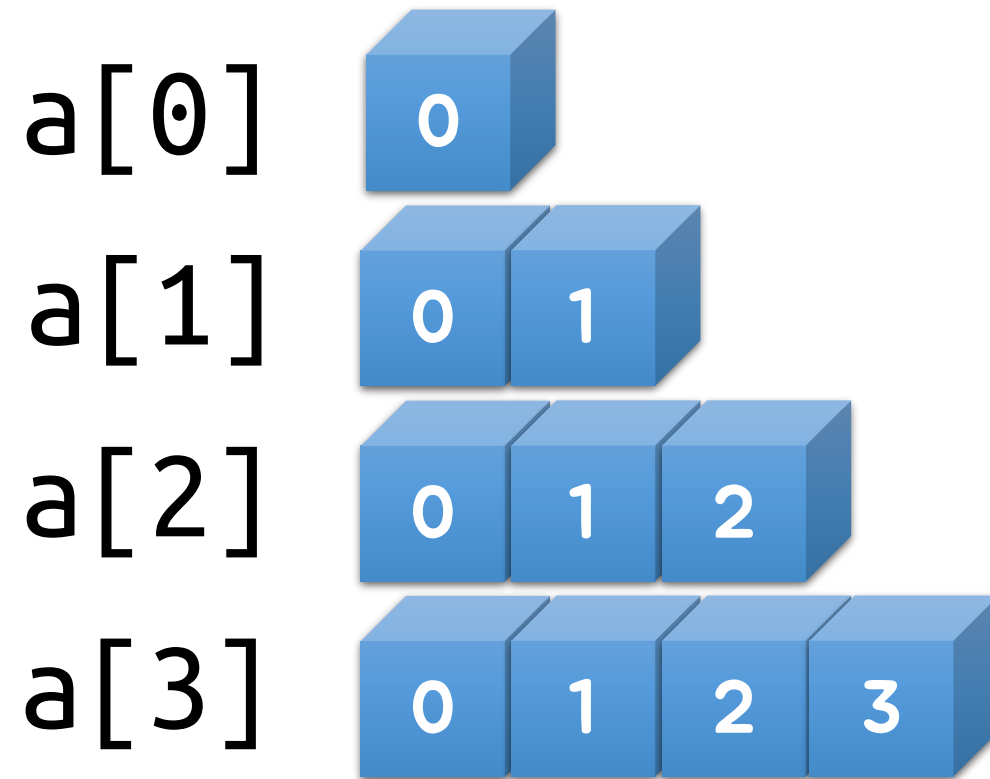
실습 #8 - VariableArray.java

```
public class VariableArray{
    public static void main(String...args) {
        int a[][] = new int[4][];
        a[0] = new int[1];
        a[1] = new int[2];
        a[2] = new int[3];
        a[3] = new int[4];
        for (int i = 0; i < a.length; i++) {
            System.out.println(
                "a[" + i + "] length ==> " + a[i].length);
        }
    }
}
```

가변 배열

```
int a[][] = new int [4][];  
a[0] = new int[1];  
a[1] = new int[2];  
a[2] = new int[3];  
a[3] = new int[4];
```

가변 배열



실습 #9 - FindMin.java

- `int a[] = {1,9,7,4,3,2,8,5,6,0};`
- 위의 배열에서 최소값을 구하는 프로그램을 작성하시오.

실습 #10 - FindMax.java

- `int a[] = {1,9,7,4,3,2,8,5,6,0};`
- 위의 배열에서 최대값을 구하는 프로그램을 작성하시오.

정렬

정렬에는 여러가지 방법이 있다.



오름 차순



내림 차순



선택정렬

- 배열에서 최소값을 찾는다.
- 최소값을 첫번째 자리와 변경



최소값

1이 최소값이지만 제일 앞자리에 있으므로 변경할 필요 없음

선택정렬

- 두번째 자리부터 최소값을 찾는다.
- 최소값을 두번째 자리와 변경



최소값



선택정렬

- 세번째 자리부터 최소값을 찾는다.
- 최소값을 세번째 자리와 변경



최소값



3이 최소값이지만 변경할 위치에 있으므로 변경할 필요 없음

선택정렬

- 네번째 자리부터 최소값을 찾는다.
- 최소값을 네번째 자리와 변경



↑
최소값



실습 #11 - SelectSort.java

- 주어진 수 10개를 받고, 선택정렬하는 프로그램을 작성하시오.