



# JAVA

GU, DA HAE

1 차 원 배 열

2 차 원 배 열

f o r - e a c h

J A V A

배

배

# 배열

| Array

같은 자료형으로 메모리 공간을 연속하여 할당하는 문법이다.

```
자료형[] 배열명 = new 자료형[크기];  
자료형[] 배열명 = new 자료형[] {값1, 값2, 값3, ...};  
자료형[] 배열명 = {값1, 값2, 값3, ...};
```

가

가

배열에 저장된 값을 **원소(element; 요소)**라 한다.

원소는 0부터 시작하여 차례대로 **인덱스(index)**를 부여 받는다.

배열을 **초기화** 할 경우 **크기를 생략**한다.

배열을 초기화 할 때 **{0}**을 사용하면 **모든 원소가 0으로 초기화**된다.

**반복문과 함께 사용**하면 편리하게 각 원소에 접근할 수 있다.

## 예제

```
public class Ex01{  
    public static void main(String[] args){  
        int[] arr1 = new int[3]; // 길이가 3인 int형 배열  
        double[] arr2 = new double[5]; // 길이가 5인 double형 배열  
        char[] arr3 = new char[10]; // 길이가 10인 char형 배열  
  
        arr1[0] = 10;  
        arr1[1] = 20;  
        arr1[2] = 30;  
  
        int result = arr1[0] + arr1[1] + arr1[2];  
        System.out.println(result);  
    }  
}
```

J A V A

## 예제

```
public class Ex02{  
    public static void main(String[] args){  
        int[] arr = new int[3];  
  
        for(int i = 0; i<3; i++){  
            arr[i] = i+1;  
        }  
  
        for(int i = 0; i<3; i++){  
            System.out.println(arr[i]);  
        }  
    }  
}
```

J A V A

## 예제

```
public class Ex03{  
    public static void main(String[] args){  
        int[] arr = new int[] {1, 2, 3};  
  
        for(int i = 0; i<arr.length; i++){  
            System.out.println(arr[i]);  
        }  
    }  
}
```

J A V A

## 예제

```
public class Ex04{  
    public static void main(String[] args){  
        int[] arr = {1, 2, 3};  
  
        for(int i = 0; i<arr.length; i++){  
            System.out.println(arr[i]);  
        }  
    }  
}
```

## 2차원 배열

| Array

```
자료형[] 배열명 = new 자료형[크기][크기];  
자료형[] 배열명 = new 자료형[][] {  
    {값, 값, 값, ...},  
    {값, 값, 값, ...}  
};  
자료형[] 배열명 = {  
    {값, 값, 값, ...},  
    {값, 값, 값, ...}  
};
```



J A V A

## 예제

```
public class Ex01{  
    public static void main(String[] args){  
        int[][] arr = new int[2][3];  
  
        for(int i=0; i<2 ; i++)  
            for(int j=0; j<3; j++)  
                arr[i][j] = i*3 + j + 1;  
  
        for(int i = 0; i<arr.length; i++)  
            for(int j = 0; j<arr[i].length; j++)  
                System.out.print(arr[i][j] + " ");  
    }  
}
```

J A V A

## 예제

```
public class Ex02{  
    public static void main(String[] args){  
        int[][] arr = new int[][] {  
            {1,2,3},  
            {4,5,6}  
        };  
  
        for(int i = 0; i<arr.length; i++)  
            for(int j = 0; j<arr[i].length; j++)  
                System.out.println(arr[i][j] + " ");  
    }  
}
```

J A V A

## 예제

```
public class Ex03{  
    public static void main(String[] args){  
        int[][] arr = {  
            {1,2,3},  
            {4,5,6}  
        };  
  
        for(int i = 0; i<arr.length; i++)  
            for(int j = 0; j<arr[i].length; j++)  
                System.out.println(arr[i][j] + " ");  
    }  
}
```

J A V A

## 예제

```
public class Ex01{  
    public static void main(String[] args){  
        int[][] arr = {// 각 행의 길이를 다르게 선언할 수 있다.  
            {1,},  
            {1, 2},  
            {1, 2, 3},  
            {1, 2, 3, 4}  
        };  
  
        for(int i = 0; i<arr.length; i++)  
            for(int j = 0; j<arr[i].length; j++)  
                System.out.println(arr[i][j] + " ");  
    }  
}
```

## 예제

```
public class Ex01{  
    public static void main(String[] args){  
        int[][] arr = new int[3][]; // 가로 길이 표시 x  
  
        arr[0] = new int[3]; // 첫 번째 행 생성  
        arr[1] = new int[4]; // 두 번째 행 생성  
        arr[2] = new int[5]; // 세 번째 행 생성  
  
    }  
}
```

//2차원 배열은 둘 이상의 1차원 배열을 묶어서 선언하는 형태이기 때문에, 세로를 담당하는 배열과 가로를 담당하는 배열 생성을 따로 할 수도 있다.

## 해보기

다음과 같은 2차원 배열이 있다.

1	1	1
2	2	2
3	3	3
4	4	4

위의 2차원 배열을 아래와 같이 변경하는 프로그램을 짜보자. 마지막 행을 첫 번째 행으로, 나머지 행은 한 줄 씩 뒤로 미뤄져야 한다.

4	4	4
1	1	1
2	2	2
3	3	3

# for-each

enhanced for

변수에 연속으로 저장되어 있는 모든 원소를 꺼내어 활용할 때 사용한다.

```
for(자료형 식별자 : 변수){  
    코드  
}
```

for-each 문은 대상의 값을 변경할 수 없다.

J A V A

## 예제

```
public class Ex01{  
    public static void main(String[] args){  
        int[] arr = {10, 20, 30};  
  
        for(int e : arr){  
            System.out.println(e);  
        }  
    }  
}
```



## 해보기

다음과 같은 2차원 배열이 있다. for-each 문을 사용하여 출력해보자.

10	20	30
40	50	60
70	80	90