

# 컴퓨터 프로그래밍 (Computer Programming)

이 선 순



# 10. 배열



# 목차

1. 배열의 이해

2. 2차원 배열

3. 배열의 활용 : 스택

**02**

## **2차원 배열**

### ■ 2차원 배열의 개념

- 1차원 배열을 여러 개 모아 놓은 것
- 수학에서 평면을 나타내기 위하여  $x, y$  좌표를 쓰는 것처럼 프로그램에서도 평면을 구현하기 위해 이차원 배열 사용
- 예. 바둑, 체스 게임, 네비게이션 지도 등을 구현할 때 이차원 배열 활용
- JAVA는 3차원, 4차원, ... 배열도 지원 : 다차원 배열
- 다차원 배열은 평면이나 공간개념 구현

## 02 2차원 배열

### ■ 2차원 배열의 개념

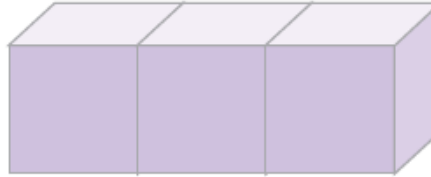
- 2개의 첨자 사용

가로줄 수

- `int aa[3][4]`

세로줄 수

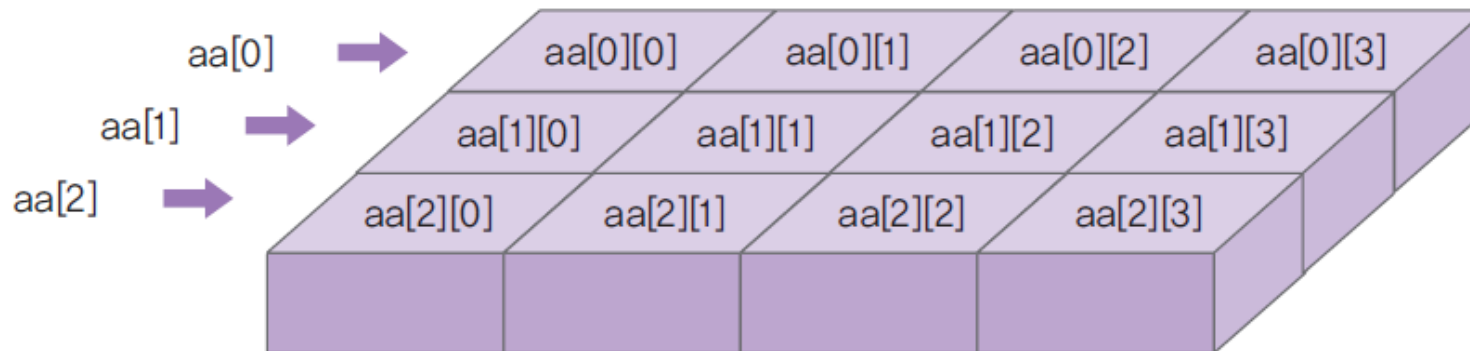
```
int[] aa = new int[행]
```



aa[0] aa[1] aa[2]

그림 8-12 1차원 배열의 개념

```
int[][] aa = new int[행][열]
```



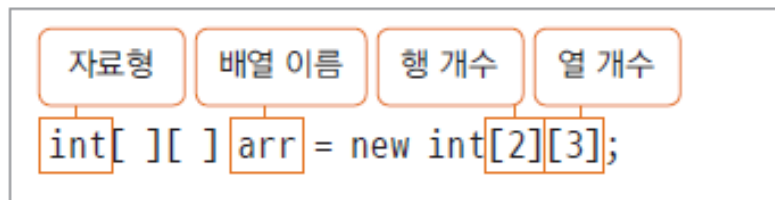
전체 배열 이름 : aa

그림 8-13 2차원 배열의 개념

## 02 2차원 배열

### ■ 2차원 배열의 선언

- 예. 2행3열의 이차원배열



arr[0][0]	arr[0][1]	arr[0][2]
arr[1][0]	arr[1][1]	arr[1][2]

- 배열의 모든 요소를 참조하려면 각 행을 기준으로 열 값을 순회함

### ■ 이차원 배열의 선언과 초기화

```
int[ ][ ] arr = {{1, 2, 3}, {4, 5, 6}};
```



arr[0][0]	arr[0][1]	arr[0][2]
1	2	3
4	5	6
arr[1][0]	arr[1][1]	arr[1][2]

## 02 2차원 배열

### ■ 2차원 배열 사용

- 예. 3행 4열의 배열을 선언하고, 총 12개의 배열에 1~12의 숫자 채우기

```
1 //실습8-7 2차원 배열 사용 예1
2
3 public class Ex08_07 {
4     public static void main(String[] args) {
5
6         int[][] aa = new int[3][4];    2차원 배열을 선언
7
8         aa[0][0] = 1; aa[0][1] = 2; aa[0][2] = 3; aa[0][3] = 4;
9         aa[1][0] = 5; aa[1][1] = 6; aa[1][2] = 7; aa[1][3] = 8;    각 요소에 값을 대입-각 배열의 요소 1~12 입력
10        aa[2][0] = 9; aa[2][1] = 10; aa[2][2] = 11; aa[2][3] = 12;
11
12        System.out.printf("aa[0][0]부터 aa[2][3]까지 출력 \n");
13
14        System.out.printf("%3d %3d %3d %3d\n", aa[0][0], aa[0][1], aa[0][2], aa[0][3]);
15        System.out.printf("%3d %3d %3d %3d\n", aa[1][0], aa[1][1], aa[1][2], aa[1][3]);    배열의 내용 출력
16        System.out.printf("%3d %3d %3d %3d\n", aa[2][0], aa[2][1], aa[2][2], aa[2][3]);
17    }
18 }
```

Problems @ Javadoc Declaration Console

<terminated> Ex08\_07 [Java Application] C:\Program File

aa[0][0]부터 aa[2][3]까지 출력

1	2	3	4
5	6	7	8
9	10	11	12



## 02 2차원 배열

### ■ 2차원 배열 사용

- 예. 중첩 for문을 이용 : 3행 4열의 배열을 선언하고, 총 12개의 배열에 1~12의 숫자 채우기

```
1 //실습8-8 2차원 배열 사용 예2
2
3 public class Ex08_08 {
4     public static void main(String[] args) {
5
6         int[][] aa = new int[3][4];    2차원 배열 선언
7
8         int i, k;                      배열의 첨자로 사용할 변수 선언
9
10        int val = 1;                   배열에 들어갈 값을 초기화
11
12        for (i = 0; i < 3; i++) {
13            for (k = 0; k < 4; k++) {
14                aa[i][k] = val;
15                val++;
16            }
17        }
```

바깥 for문을 세번 반복한다. 즉, 앞 첨자가 변경됨(행 단위 변경)

안쪽 for문을 네번 반복한다. 즉, 뒤 첨자가 변경됨(열 단위 변경)

배열에 val값을 입력한 후 1을 증가시킨다

## 02 2차원 배열

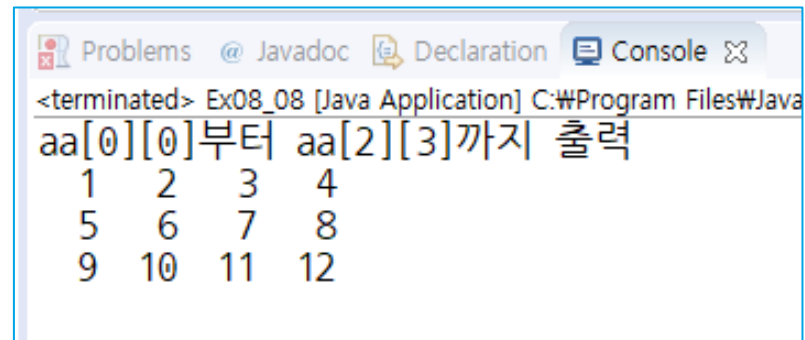
### ■ 2차원 배열 사용

- 예. for문을 이용 : 3행 4열의 배열을 선언하고, 총 12개의 배열에 1~12의 숫자 채우기

```
18
19      System.out.printf("aa[0][0]부터 aa[2][3]까지 출력 \n");
20
21
22      for (i = 0; i < 3; i++) {
23          for (k = 0; k < 4; k++) {
24              System.out.printf("%3d ", aa[i][k]);
25          }
26
27          System.out.printf("\n");
28      }
29
30 }
31 }
32 }
```

바깥 for문 : 입력과 동일한 개념으로 12회 출력

한 행을 출력한 다음 줄바꿈을 한다



```
Problems @ Javadoc Declaration Console
<terminated> Ex08_08 [Java Application] C:\Program Files\Java
aa[0][0]부터 aa[2][3]까지 출력
 1  2  3  4
 5  6  7  8
 9 10 11 12
```

## 02 2차원 배열

### ■ 2차원 배열의 초기화

- 예. 2차원 배열을 선언하는 동시에 값을 초기화하는 방법

```
1 //실습8-9 2차원 배열의 초기화
2
3 public class Ex08_09 {
4     public static void main(String[] args) {
5
6         int[][] aa = { { 1, 2, 3, 4 },
7                        { 5, 6, 7, 8 },
8                        { 9, 10, 11, 12 } };
9
10
11         int i, k;
12
13         System.out.printf("aa[0][0]부터 aa[2][3]까지 출력 \n");
14
15         for (i = 0; i < 3; i++) {
16             for (k = 0; k < 4; k++) {
17                 System.out.printf("%3d", aa[i][k]);
18             }
19
20             System.out.printf("\n");
21         }
22     }
23 }
```

2차원 배열을 초기화한다

2차원 배열에 저장된 값을 출력한다.

## 02 2차원 배열

```
Problems @ Javadoc Declaration Console
<terminated> Ex08_09 [Java Application] C:\Program Files\Java\jdk-11.0.6\bin\java
aa[0][0]부터 aa[2][3]까지 출력
1 2 3 4
5 6 7 8
9 10 11 12
```

- 6행~8행은 가장 기본적인 배열의 초기화 방법을 보여줌.
- 3행 4열의 배열이므로 다음 그림과 같이 초기화하고 네모칸은 2차원 배열의 공간으로 봄

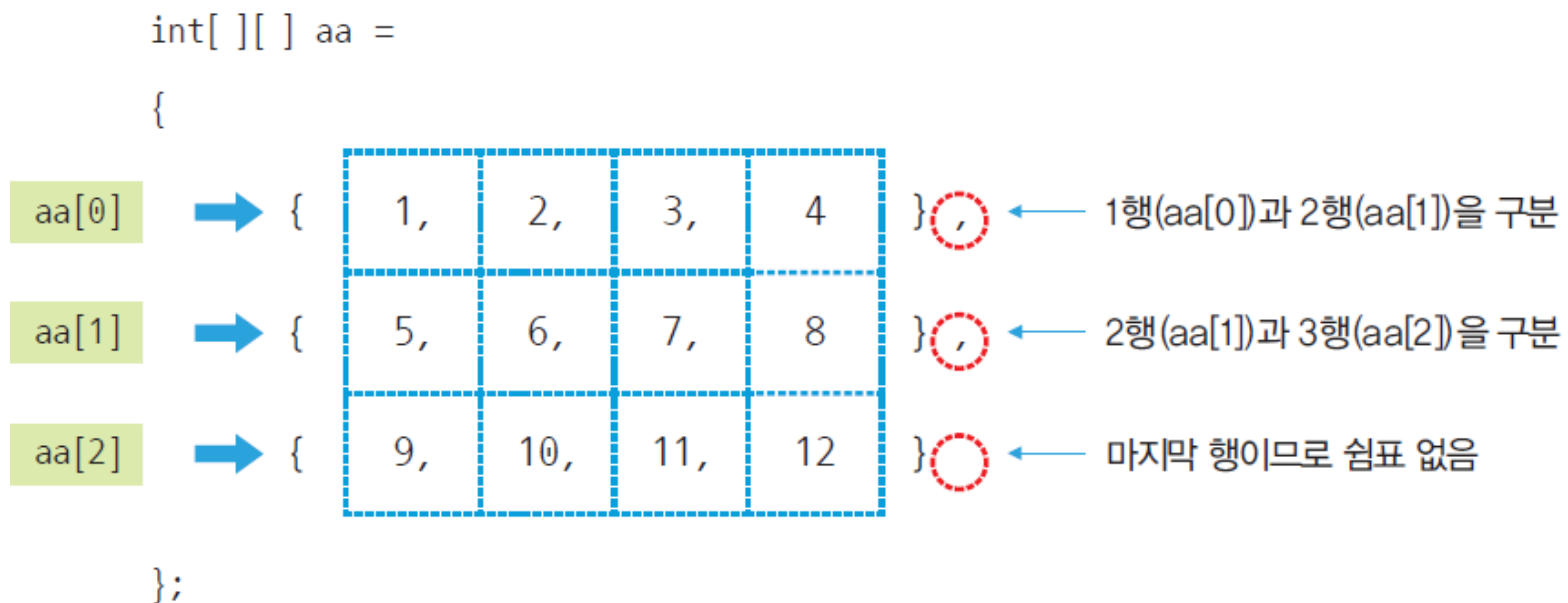


그림 8-17 2차원 배열의 초기화

### ■ 배열 크기의 동적할당

- 동적할당 : 배열을 생성하면서 크기를 지정하지 않고 사용자의 입력에 따라 배열의 크기를 지정하는 것

```
1 //실습8-10 2차원 배열의 동적할당
```

```
2  
3 import java.util.Scanner;
```

```
4  
5 public class Ex08_10 {
```

```
6     public static void main(String[] args) {
```

```
7         Scanner s = new Scanner(System.in);
```

```
8  
9         int row, col;
```

행의 개수와 열의 개수를 입력받을 변수를 선언

```
10  
11         System.out.print("행 개수를 입력 : ");
```

행의 개수 입력

```
12         row = s.nextInt();
```

```
13         System.out.print("열 개수를 입력 : ");
```

열의 개수 입력

```
14         col = s.nextInt();
```

```
15  
16         int [][] aa = new int[row][col];
```

배열을 선언할 때 변수로 첨자를 사용할 수 있음  
즉, 동적으로 크기를 할당함

```
17         int i, k;
```

```
18         int val = 1;
```

16행 : 배열의 크기를 변수로 선언하기 때문에 동적으로 크기가 변경되는 배열의 사용이 가능함. 즉, 프로그램을 진행하는 도중에 배열의 크기를 결정할 수 있다는 의미.

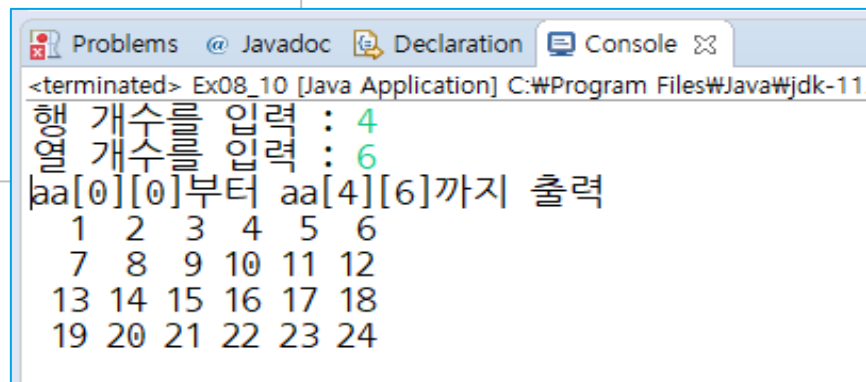
## 02 2차원 배열의 이해

### ■ 배열 크기의 동적할당

```
20
21     for (i = 0; i < row; i++) {
22         for (k = 0; k < col; k++) {
23             aa[i][k] = val;
24             val++;
25         }
26     }
27
28     System.out.printf("aa[0][0]부터 aa[%d][%d]까지 출력 \n", row, col);
29
30     for (i = 0; i < row; i++) {
31         for (k = 0; k < col; k++) {
32             System.out.printf("%3d", aa[i][k]);
33         }
34         System.out.printf("\n");
35     }
36
37     s.close();
38 }
39 }
```

배열의 크기만큼 값을 입력한다

2차원 배열에 저장된 값을 출력한다



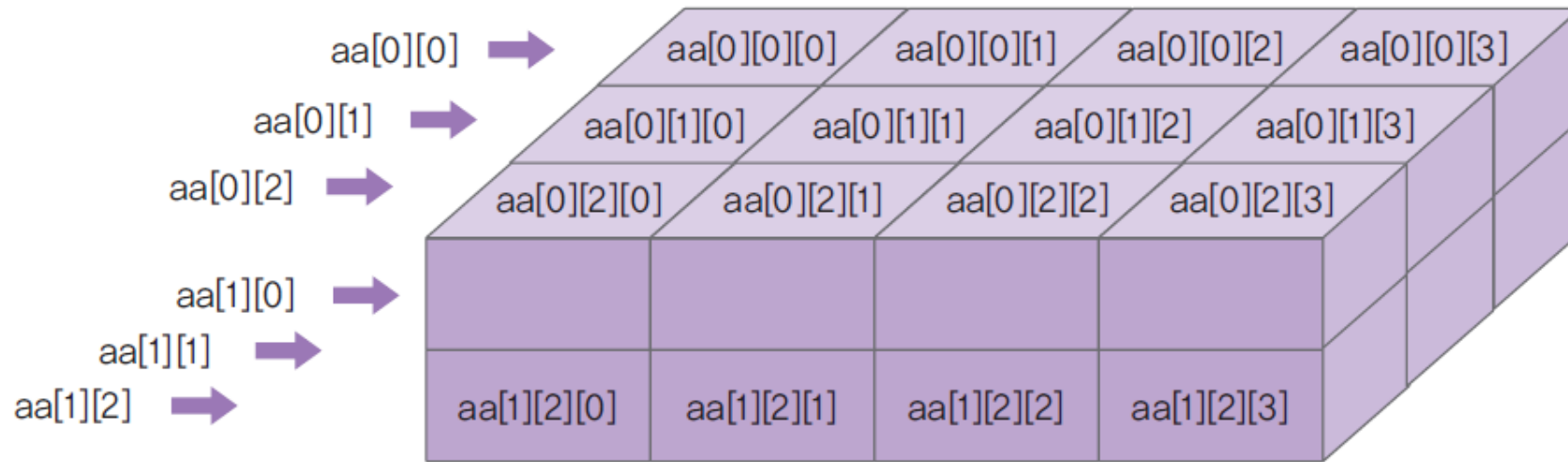
```
Problems @ Javadoc Declaration Console
<terminated> Ex08_10 [Java Application] C:\Program Files\Java\jdk-11
행 개수를 입력 : 4
열 개수를 입력 : 6
aa[0][0]부터 aa[4][6]까지 출력
 1  2  3  4  5  6
 7  8  9 10 11 12
13 14 15 16 17 18
19 20 21 22 23 24
```

## 02 3차원 이상의 배열

### ■ 3차원 배열

- 2차원 배열 위에 또 다른 2차원 배열을 쌓아놓은 것
- 2차원 배열 하나를 한면으로 취급하여 첨자를 추가함

```
int[][][] aa = new int[면][행][열]
```



전체 배열 이름 : aa

그림 8-19 3차원 배열의 개념

## 02 3차원 이상의 배열

### ■ 3차원 배열

- [그림 8-19]의 3차원 배열 초기화
- 2면 3행 4열의 3차원 배열의 초기화는 그림 8-20 참고
- 2차원 배열의 초기화를 한번 더한다고 생각하고, 각 2차원을 쉼표로 분리한 다음 전체를 다시 중괄호({})로 묶음

```
int[ ][ ][ ] aa =  
{  
    {  
        { 1, 2, 3, 4 },  
        { 5, 6, 7, 8 },  
        { 9, 10, 11, 12 }  
    }  
    ,  
    {  
        { 13, 14, 15, 16 },  
        { 17, 18, 19, 20 },  
        { 21, 22, 23, 24 }  
    }  
};
```

→ 윗면의 2차원 배열

← 면 사이 분리

→ 아랫면의 2차원 배열

그림 8-20 3차원 배열의 초기화



## Self study 10-2

1. [실습8-8]을 수정하여 배열 aa에 홀수 1,3,5,7,9, ...를 입력하고 a[2][3]. a[2][2],a[2][1], ...과 같이 뒤집힌 순서대로 출력해보자.
2. 2차원 배열 `int aa[][] = new int[5][10];` 을 선언하면 배열의 요소는 몇 개인가?
3. 알파벳 소문자를 2글자씩 13줄(13행 2열)로 출력하는 프로그램을 이차원 배열로 작성해보자.

질문은이메일을 이용해주세요.  
[ds.june2@gmail.com](mailto:ds.june2@gmail.com)

**감사합니다**