

Array

배열

Array

배열 : 같은 타입의 값 여러 개를 하나의 변수로 다루기 위한 자료구조
배열을 구성하는 각각의 값을 요소(Element)라 한다.

index는 각 요소의 위치를 나타내는 양의 정수이다.

index는 0부터 시작한다.

* 자료구조 : 여러 개의 데이터를 효율적으로 접근 및 수정하기 위한 데이터 저장 방식

int[] arr

| | | | | |
|--------|--------|--------|--------|--------|
| 1 | 2 | 3 | 4 | 5 |
| arr[0] | arr[1] | arr[2] | arr[3] | arr[4] |

Array

배열 선언

타입[] 배열명; (추천)

타입 배열명[];

배열 할당

타입[] 배열명 = new 타입[배열크기]; (추천)

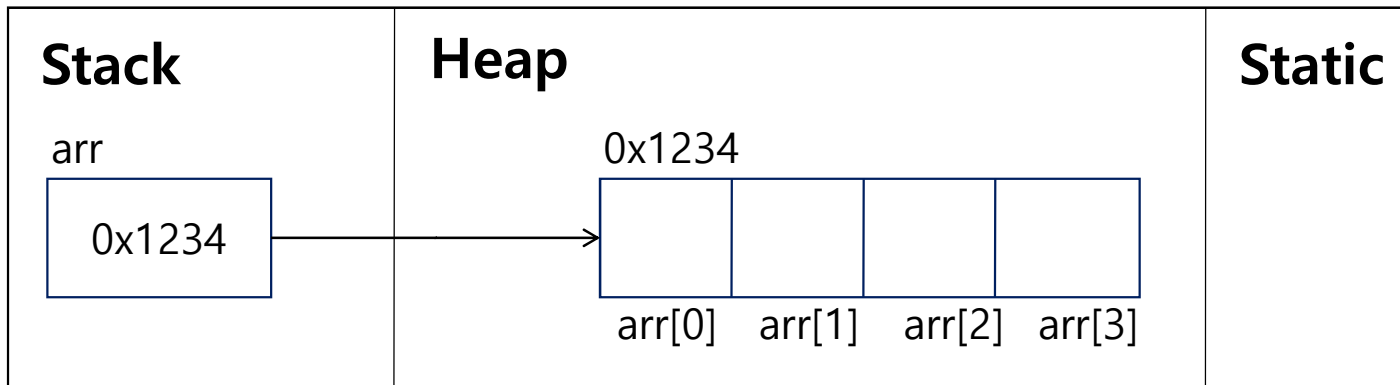
타입 배열명[] = new 타입[배열크기];

ex) **int**[] arr = **new int**[3];
 int arr[] = **new int**[3];

Array

배열은 참조 변수로 Heap 영역에 할당되며 배열 공간의 주소를 저장
배열 공간의 주소를 이용해 인덱스를 참조하는 방식으로 값 처리

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



Array

배열 초기화

1. 인덱스를 이용한 초기화

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

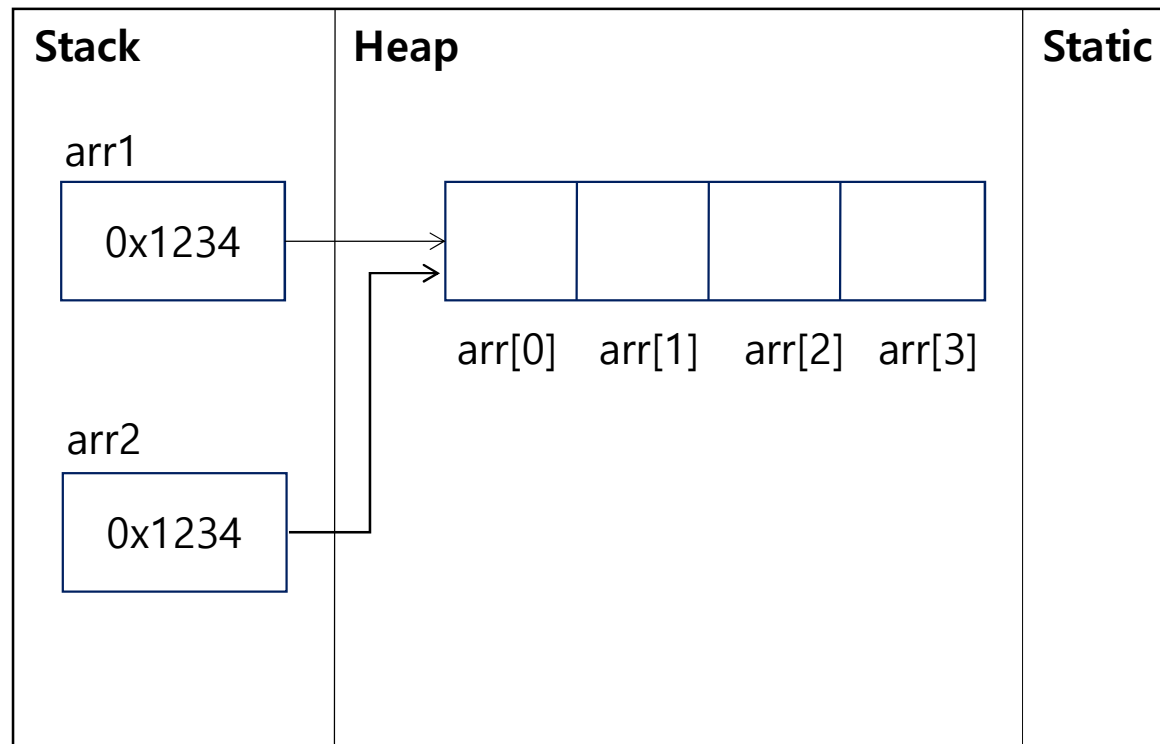
2. 선언과 동시에 초기화

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

Array

얕은 복사 : 하나의 객체를 두 레퍼런스가 참조하도록 객체의 주소를 복사하는 것

```
int[] arr1 = new int[4];  
int[] arr2 = arr1;
```



Array

깊은 복사 : 새로운 배열 객체를 생성하여 기존 배열의 데이터를 복사하는 것

```
for(int i = 0; i < arr1.length; i++) {  
    arr2[i] = arr1[i];  
}
```

| Stack | Heap | Static |
|-----------------------------------|--|--------|
| <div>arr1</div> <div>0x1234</div> | <div>0x1234</div> <div>arr[0] arr[1] arr[2] arr[3]</div> | |
| <div>arr2</div> <div>0x5678</div> | <div>0x5678</div> <div>arr2[0] arr2[1] arr2[2] arr2[3]</div> | |

Array - 2차원 배열

2차원 배열: 배열을 요소로 가지는 배열

2차원 배열 선언

타입[][] 배열명; (추천)

타입 배열명[][];

2차원 배열 할당

타입[][] 배열명 = new 타입[배열크기][요소(배열)크기]; (추천)

타입 배열명[][] = new 타입[배열크기][요소(배열)크기];

2차원 가변배열 : 요소인 배열의 크기가 가변적인 배열

2차원 가변배열 할당

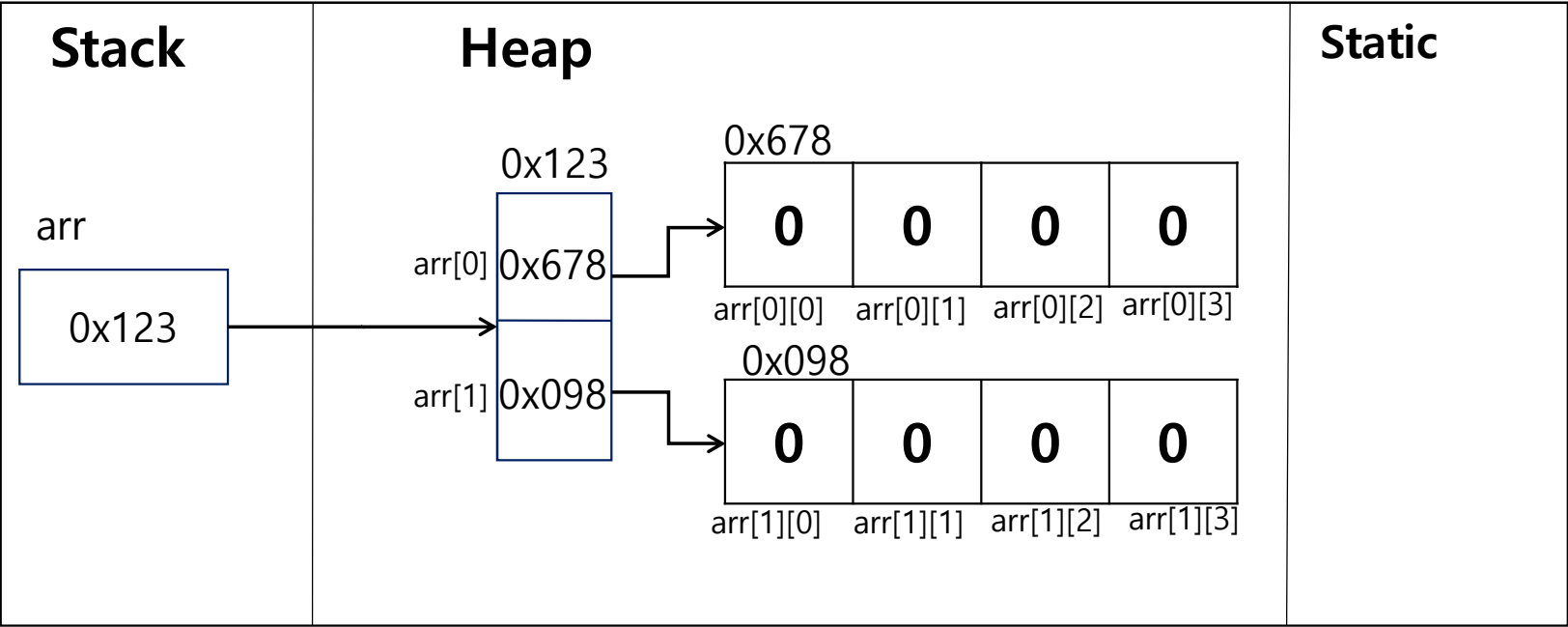
타입[][] 배열명 = new 타입[배열크기][]; (추천)

타입 배열명타입[][] 배열명 = new 타입[배열크기][];

Array - 2차원 배열

2차원 배열 구조

```
int [][] arr=new int[2][4];
```



Array - 2차원 배열

2차원 배열 초기화

1. 인덱스를 이용한 초기화

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

2. 선언과 동시에 초기화

```
int[][] arr = new int[][]{{1,2,3,4},{5,6,7,8}};  
int[][] arr = {{0,1},{2,3}};
```