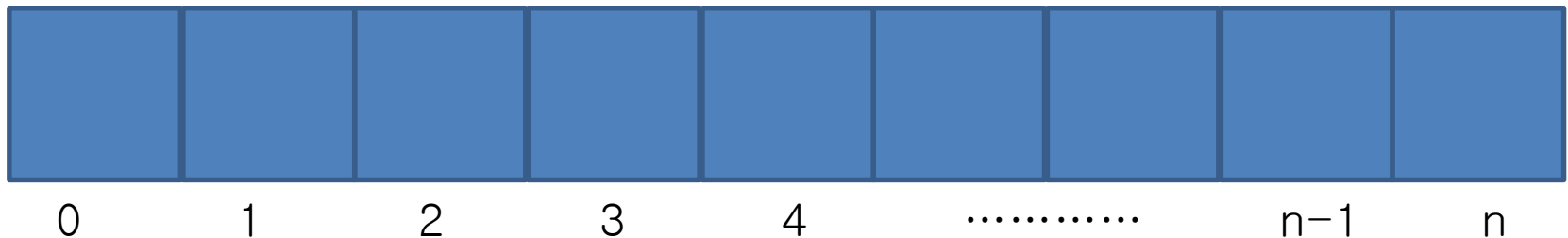


# 배열(Array)

동일 자료형의 묶음을 의미한다.

- 동일한 물품을 보관하는 서랍장
- 아파트, 오피스텔, 기차

☞ 배열은 저장된 값마다 **인덱스 번호**가 설정됨.(0부터 시작)



# 배열 선언과 할당

배열 선언 : 주소값을 가지지 않은 변수생성(Stack)

- 자료형[] 변수이름;
- 자료형 변수이름[];

```
예) int[] arr;  
    int arr[];
```

배열의 할당: 배열객체 생성후 변수에 주소값할당(Heap)

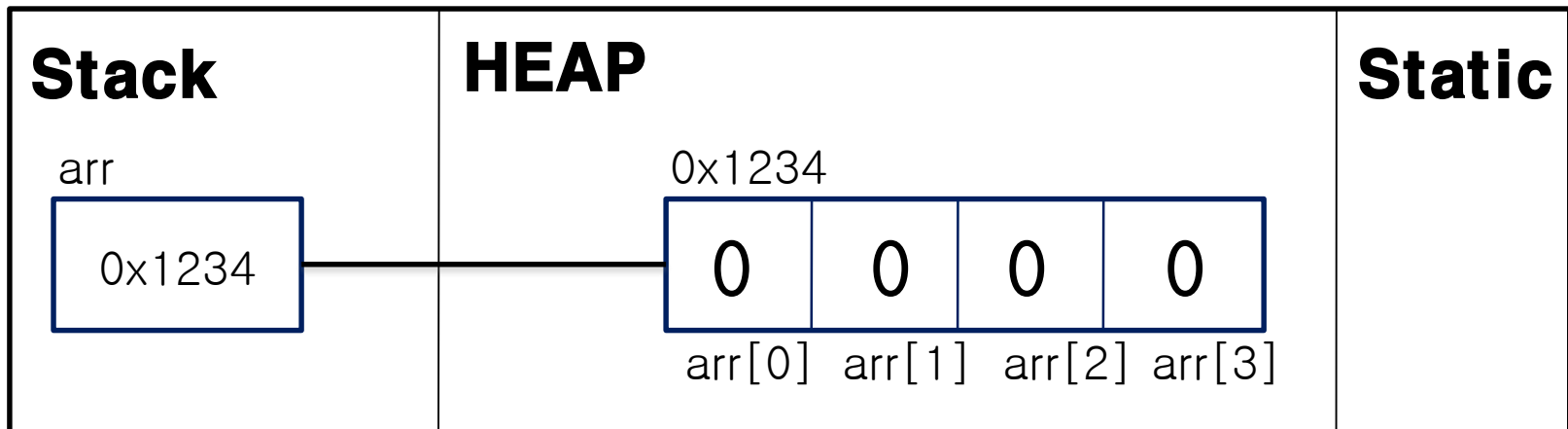
- 변수이름 = **new** 자료형[저장데이터개수];

```
예) arr = new int[5];
```

# 배열 저장 구조

```
int[] arr;
```

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



참조형 : 주소값을 가지고 다른 공간에 있는 데이터를  
확인하고 처리

선언과 동시에 값기록

예) `int[] arr = {1, 2, 3, 4, 5};`

`String arr[] = {"홍길동", "이순신", "세종대왕"};`

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

※ 할당과 동시에 자동적으로 기본값으로 초기화됨

예) `boolean : false, char : ` ``

인덱스를 이용한 직접 값기록

예) `arr[0] = 1;`

`arr[1] = 2;`

`arr[2] = 3;`

.....

**for문을 이용**

```
예) for(int i = 0; i < arr.length; i++){  
    arr[i] = i + 1;  
}
```

**\*\* 초기화 할 리터럴이 일정 규칙으로 증가한다면,  
반복문의 index를 통해 배열에 값을 기록 할 수 있다.**

**for문을 이용 사용자 입력 데이터 기록**

**예) int[] arr=new int[5];**

**for(int i = 0; i < arr.length; i++){**

**System.out.print("저장 데이터 : ");**

**arr[i] = sc.nextInt();**

**}**



배열에 저장된 값을 호출(출력)하려면 인덱스를 이용  
**arr[인덱스번호]**

**\*\* arr[인덱스번호]**가 하나의 변수명이라고 생각하면 된다.

**출력 : System.out.println(arr[인덱스번호]);**

# 실습문제1

길이가 100인 배열을 선언하고 1부터 100까지의 값을 순서대로 배열 인덱스에 넣어 그 값을 출력하는 코드를 작성하시오.

인덱스를 이용하는 방법 / for문을 이용하는 방법을 한번씩 사용하여 작성하세요.

## 실습문제2

길이가 5인 String 배열을 선언하고,  
"딸기", "복숭아", "키위", "사과", "바나나" 로  
초기화를 한 후,  
배열 인덱스를 활용해서 바나나를 출력해 보세요

**얕은 복사** : 객체의 주소값만 가져와 참조형변수에  
저장하고 하나의 객체를 두 변수가  
참조하고 있는 것

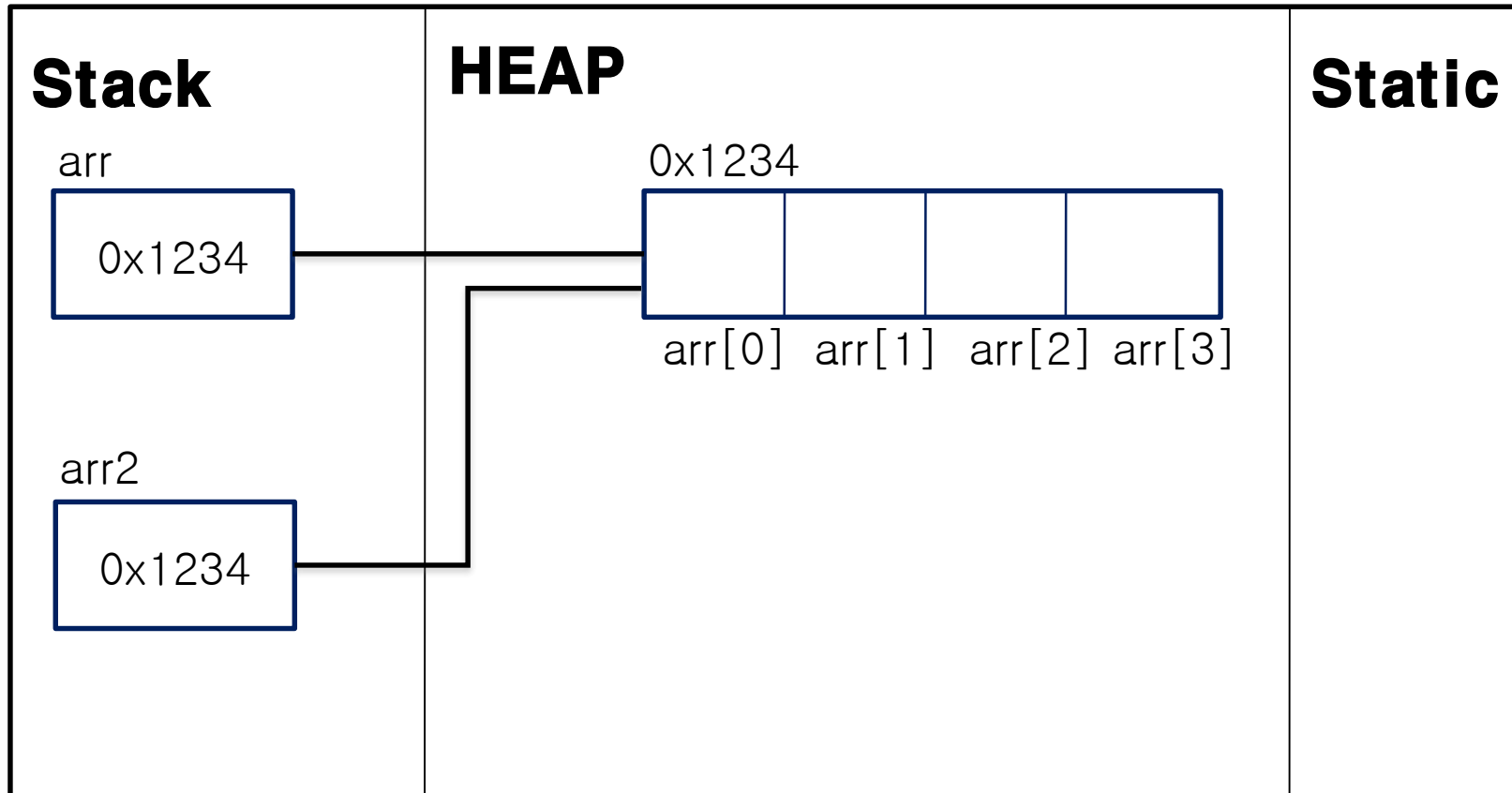
- 방법 : `int[] arr1 = new int[4]`  
`int[] arr2 = arr1;`

**깊은 복사** : 다른 객체를 생성하여 새로운 객체에  
데이터 내용을 복사하는 것

- 방법 : for문을 이용한 1:1복사

`System.arraycopy()` 메소드를 이용한 복사  
`clone()` 메소드를 이용한 복사

## 얕은 복사



예제

```
1 public class Shallow_copy{
2     public static void main(String[] args){
3         char[] carr1 = {'a', 'b', 'c', 'd'};
4         char[] carr2 = carr1;
5
6         for(int i = 0; i < carr1.length; i++){
7             System.out.print(carr1[i] + ", ");
8         }
9         System.out.println("주소값 : " + carr1.hashCode());
10
11        for(int i = 0; i < carr2.length; i++){
12            System.out.print(carr2[i] + ", ");
13        }
14        System.out.println("주소값 : " + carr2.hashCode());
15    }
16 }
```

실행결과

a, b, c, d, 주소값 : 148659610  
a, b, c, d, 주소값 : 148659610

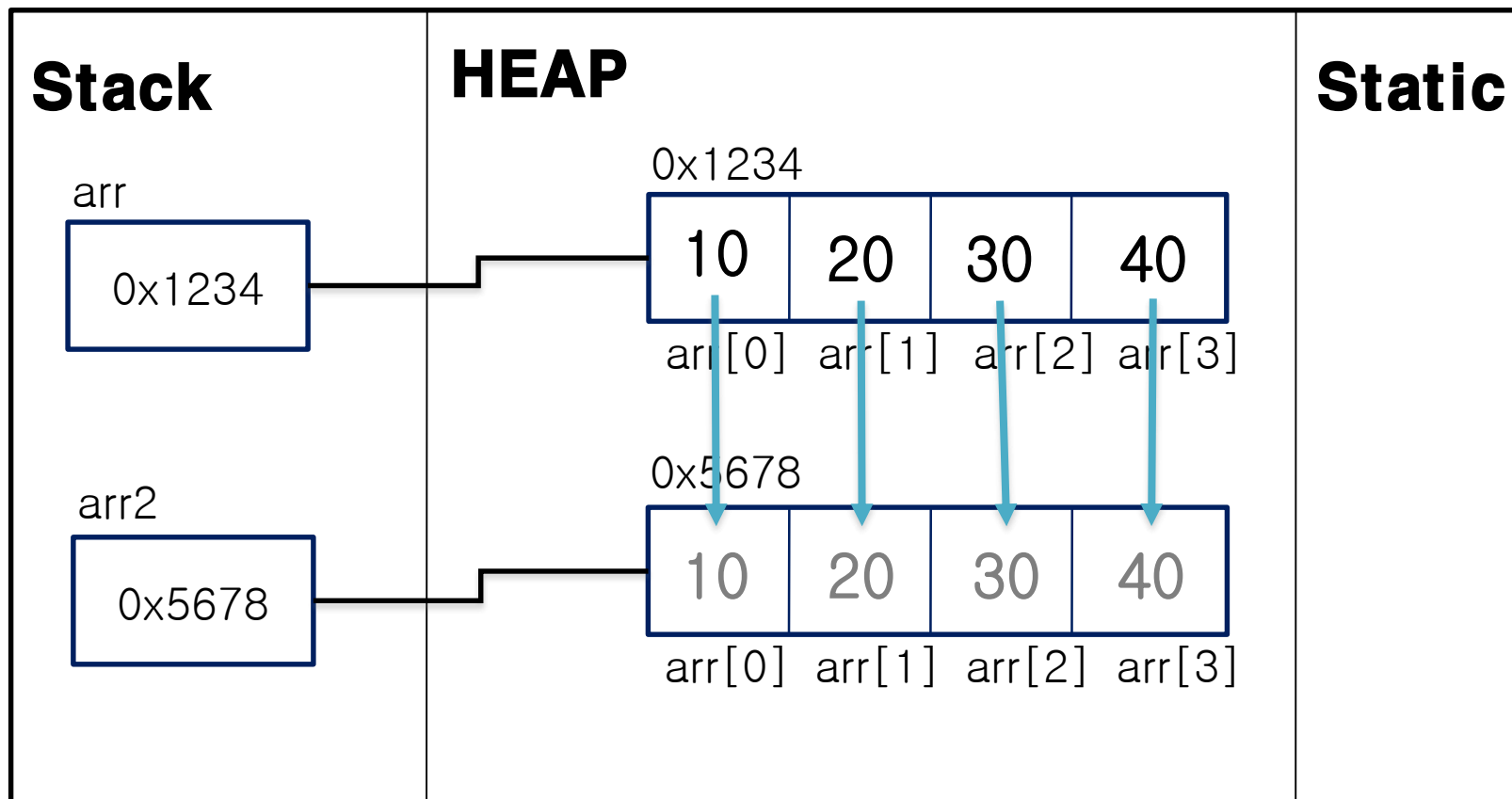
예제

```
1 public class Shallow_copy{
2     public static void main(String[] args){
3         char[] carr1 = {'a', 'b', 'c', 'd'};
4         char[] carr2 = carr1;
5
6         carr2[0] = 'f';
7
8         for(int i = 0; i < carr1.length; i++){
9             System.out.print(carr1[i] + ", ");
10        }
11        System.out.println("주소값 : " + carr1.hashCode());
12
13        for(int i = 0; i < carr2.length; i++){
14            System.out.print(carr2[i] + ", ");
15        }
16        System.out.println("주소값 : " + carr2.hashCode());
17    }
18 }
```

실행결과

```
f, b, c, d, 주소값 : 148659610
f, b, c, d, 주소값 : 148659610
```

## 깊은 복사





예제

```
3 public class for_copy{
4     public static void main(String[] args){
5         int[] arr1 = {1, 2, 3, 4, 5};
6         int[] arr2 = new int[5];
7
8         for(int i = 0; i < arr1.length; i++){
9             arr2[i] = arr1[i];
10        }
11
12        for(int i = 0; i < arr1.length; i++){
13            System.out.print(arr1[i] + ", ");
14        }
15        System.out.println("주소값 : " + arr1.hashCode());
16
17        for(int i = 0; i < arr2.length; i++){
18            System.out.print(arr2[i] + ", ");
19        }
20        System.out.println("주소값 : " + arr2.hashCode());
    }
```

## 실행결과

1, 2, 3, 4, 5, 주소값 : 148659610  
1, 2, 3, 4, 5, 주소값 : 154856980

# for문 이용복사

예제

```
3      public class for_copy{
4          public static void main(String[] args){
5              int[] arr1 = {1, 2, 3, 4, 5};
6              int[] arr2 = new int[5];
7
8              for(int i = 0; i < arr1.length; i++){
9                  arr2[i] = arr1[i];
10             }
11
12             arr2[0] = 0;
13
14             for(int i = 0; i < arr1.length; i++){
15                 System.out.print(arr1[i] + ", ");
16             }
17             System.out.println("주소값 : " + arr1.hashCode());
18
19             for(int i = 0; i < arr2.length; i++){
20                 System.out.print(arr2[i] + ", ");
21             }
22             System.out.println("주소값 : " + arr2.hashCode());
23         }
24     }
```

## 실행결과

1, 2, 3, 4, 5, 주소값 : 148659610  
0, 2, 3, 4, 5, 주소값 : 154856980

# arraycopy() 메소드 이용복사

예제

```
public class Arraycopy{
    public static void main(String[] args){
        int[] arr1 = {1, 2, 3, 4, 5};
        int[] arr2 = new int[5];
        System.arraycopy(arr1, 0, arr2, 0, arr1.length);

        for(int i = 0; i < arr1.length; i++){
            System.out.print(arr1[i] + ", ");
        }
        System.out.println("주소값 : " + arr1.hashCode());

        for(int i = 0; i < arr2.length; i++){
            System.out.print(arr2[i] + ", ");
        }
        System.out.println("주소값 : " + arr2.hashCode());
    }
}
```

실행결과

1, 2, 3, 4, 5, 주소값 : 148659610  
1, 2, 3, 4, 5, 주소값 : 154856980

# arraycopy() 메소드 이용복사

예제

```
public class Arraycopy{
    public static void main(String[] args){
        int[] arr1 = {1, 2, 3, 4, 5};
        int[] arr2 = new int[5];
        System.arraycopy(arr1, 0, arr2, 0, arr1.length);
        arr2[0] = 0;
        for(int i = 0; i < arr1.length; i++){
            System.out.print(arr1[i] + ", ");
        }
        System.out.println("주소값 : " + arr1.hashCode());

        for(int i = 0; i < arr2.length; i++){
            System.out.print(arr2[i] + ", ");
        }
        System.out.println("주소값 : " + arr2.hashCode());
    }
}
```

실행결과

1, 2, 3, 4, 5, 주소값 : 148659610  
0, 2, 3, 4, 5, 주소값 : 154856980

# clone() 메소드 이용복사

예제

```
1 public class Clone_copy{
2     public static void main(String[] args){
3         char[] carr1 = {'1', '2', '3', '4', '5'};
4         char[] carr2 = new char[5];
5
6         carr2 = carr1.clone();
7
8         for(int i = 0; i < carr1.length; i++){
9             System.out.print(carr1[i] + ", ");
10        }
11        System.out.println("주소값 : " + carr1.hashCode());
12
13        for(int i = 0; i < carr2.length; i++){
14            System.out.print(carr2[i] + ", ");
15        }
16        System.out.println("주소값 : " + carr2.hashCode());
17    }
18 }
```

실행결과

```
1, 2, 3, 4, 5, 주소값 : 148659610
1, 2, 3, 4, 5, 주소값 : 154856980
```

# clone() 메소드 이용복사

예제

```
3 public class Clone_copy{
4     public static void main(String[] args){
5         char[] carr1 = {1, 2, 3, 4, 5};
6         char[] carr2 = new int[5];
7
8         carr2 = carr1.clone();
9
10        carr2[0] = 0;
11
12        for(int i = 0; i < carr1.length; i++){
13            System.out.print(carr1[i] + ", ");
14        }
15        System.out.println("주소값 : " + carr1.hashCode());
16
17        for(int i = 0; i < carr2.length; i++){
18            System.out.print(carr2[i] + ", ");
19        }
20        System.out.println("주소값 : " + carr2.hashCode());
21    }
22 }
```

실행결과

1, 2, 3, 4, 5, 주소값 : 148659610  
0, 2, 3, 4, 5, 주소값 : 154856980

문자열을 하나 입력 받아 배열에 넣고 문자를 하나 입력 받아 입력 받은 문자가 문자열에 몇 개인지 개수를 출력하는 프로그램을 작성하세요.

예) 입력값 : helloworld

검색값 : l

출력 : 입력하신 문자열 helloworld에서  
찾으시는 문자 l은 3개 입니다.

사용자의 전화번호를 입력받고, 13자리의  
문자형배열에 저장한후, 가운데 4자리를 \*로  
가리기.

단, 원본 배열값은 변경 없이 배열 복사본으로  
변경하세요

힌트) 복사방법

- for문이용 1:1대입
- System.arraycopy() 매소드 이용
- clone() 매소드 이용



주민등록번호 성별자리 이후부터 \*로 가리기.  
단, 원본 배열값은 변경 없이 배열 복사본으로  
변경하세요

힌트) 복사방법

- for문이용 1:1대입
- System.arraycopy() 매소드 이용
- clone() 매소드 이용

배열의 크기로 홀수인 양의 정수를 입력 받아 배열을 만드세요.

배열의 중간까지는 1부터 1씩 증가하여 오름차순으로 값을 넣고, 중간 이후부터 끝까지는 1씩 감소하여 내림차순으로 값을 넣어 출력하는 프로그램을 작성하세요.

출력예) 7을 입력하면,  
[1, 2, 3, 4, 3, 2, 1]

로또번호 자동 생성기 프로그램을 작성하세요.  
단, 중복 값이 있으면 안되고, 오름차순으로  
정렬하여 출력하세요.

힌트) Random클래스 사용 혹은  
Math.random() 사용