

Arrays

🕒 작성일시	@2023년 4월 18일 오전 12:16
📄 강의 번호	자바의정석 Chapter11
📄 유형	
📎 자료	
☑ 복습	<input type="checkbox"/>
☰ Spring Framwork	

▼ 배열을 다루는데 유용한 메서드가 정의되어 있다.

▼ toString()

- 모두 기본형 배열과 참조형 배열 별로 하나씩 정의되어 있다.
-

```
static String toString(boolean[] a)
static String toString(byte[] a)
...
```

▼ 배열의 복사- copyOf(), copyOfRange()

- copyOf()는 배열 전체를, copyOfRange()는 배열의 일부를 복사해서 새로운 배열을 만들어 반환한다.
- 항상 그랬듯이 copyOfRange()는 범위에서 끝을 포함하지 않는다.

```
int[] arr={0,1,2,3,4};
int[] arr2 = Arrays.copyOf(arr, arr.length); //arr2=[0,1,2,3,4]
```

```
int[] arr3 = Arrays.copyOf(arr, 3); //arr3=[0,1,2]
int[] arr4 = arrays.copyOf(arr,7); //arr4 =[0,1,2,3,4,0,0]
int[] arr5 = arrays.copyOfRange(arr,2,4); //arr5= [2,3]
int[] arr6= arrays.copyOfRange(arr,0,7); //arr6=[0,1,2,3,4,0,0]
```

▼ 배열 채우기 - fill(), setAll()

- fill()은 배열의 모든 요소를 지정된 값으로 채운다.
- setAll()은 배열을 채우는데 사용할 함수형 인터페이스를 매개변수로 받는다.
- 매개변수 = 람다식 또는 함수형 인터페이스를 구현한 객체

```
int[] arr= new int[5];
Arrays.fill(arr,9); //arr= [9,9,9,9,9]
Arrays.setAll(arr,() -> (int)(Math.random()*5)+1); arr=[1,5,2,1,1]
```

▼ 배열의 정렬과 검색 - sort(), binarySearch()

- set()은 배열을 정렬할 때, 그리고 배열에 저장된 요소를 검색할 때는 binarySearch()를 사용
- binarySearch()는 배열에서 지정된 값이 저장된 위치를 찾아서 반환하는데, 반드시 정렬된 상태이어야 올바른 결과를 얻는다. 만일 검색한 값과 일치하는 요소들이 여러 개 있다면, 이 중에서 어떤 것의 위치가 반환될지는 알 수 없다.

```
int[] arr={3,2,0,4,5,1};
int idx = Arrays.bianrySearch(arr,2); //idx=-5 << 잘못된 결과

Arrays.sort(arr);
System.out.println(Arrays.toString(arr)); //[0,1,2,3,4,5]
int idx = Arrays.binarySearch(arr,2); // idx=2
```

- 순차검색 : 첫 번째 요소부터 순서대로 검색, 정렬은 필요없지만 시간이 많이 소요
- 이진 검색(binary Search) : 배열의 검색할 범위를 반복적으로 절반씩 줄여나간다, 정렬 필요

▼ 배열의 비교와 출력 - equals(), toString()

- toString()은 모든 요소를 문자열로 편하게 출력할 수 있다. 일차원 배열에서만 가능

- 다차원 배열 - `deepToString()`을 사용한다. 2차원뿐만 아니라 3차원 이상의 배열에서도 가능

```
int[] arr= {0,1,2,3,4,5};
int[][] arr2D= {{11,22}, {21,22}};

System.out.println(Arrays.toString(arr)); //[0,1,2,3,4,5]
System.out.println(Arrays.deepToString(arr2D)); //[11,12[, [21,22]]
```

▼ 배열을 List로 변환 - `asList(Object... a)`

- `asList()`는 배열을 list에 담아서 반환한다. 매개변수의 타입이 가변인수라서 배열 생성 없이 저장할 요소들만 나열하는 것도 가능하다.
-

```
List list= Arrays.asList(new Integer[]{1,2,3,4,5});
List list = Arrays.asList(1,2,3,4,5);

//둘 모두 list={1,2,3,4,5}
list.add(6); // UnsupportedOperationException 예외 발생
```

- `asList()`가 반환한 List의 크기를 변경할 수 없다. 추가, 삭제가 불가능하다.
- 저장된 내용은 변경 가능하다.
- 만일 크기를 변경할 수 있는 리스트가 필요하다면 아래와 같이 하면 된다.

```
List list= new ArrayList(Arrays.asList(1,2,3,4,5));
//list = [1,2,3,4,5]
```

▼ `parallelXXX()`, `splitterore()`, `stream()`

- 보다 빠른 결과를 얻기 위해 여러 쓰레드가 작업을 나누어 처리하도록 한다.
- `splitterore()`는 여러 쓰레드가 처리할 수 있게 하나의 작업을 여러 작업으로 나누는 `Splitter`를 반환한다.
- `steram()`은 컬렉션 스트림으로 변환한다.

- 14람다와 스트림과 관련된 내용이다.

▼ 예시 종합

```
package ch11;

import java.util.Arrays;

public class ArraysEx {
    public static void main(String[] args) {
        int[] arr= {0,1,2,3,4};
        int[][] arr2D = {{11,12},{21,22}};

        System.out.println("arr="+Arrays.toString(arr));
        System.out.println("arr2D="+Arrays.deepToString(arr2D));

        int[] arr2 = Arrays.copyOf(arr, arr.length);
        int[] arr3 = Arrays.copyOf(arr, 3);
        int[] arr4 = Arrays.copyOf(arr, 7);
        int[] arr5 = Arrays.copyOfRange(arr, 2, 7);

        System.out.println(arr2);
        System.out.println(arr3);
        System.out.println(arr4);
        System.out.println(arr5);

        int[] arr7 = new int[5];
        Arrays.fill(arr7, 9);
        System.out.println("arr7 : " + Arrays.toString(arr7));

        Arrays.setAll(arr7, i -> (int)(Math.random()*6)+1);
        System.out.println("arr7 : " + Arrays.toString(arr7));

        for (int i : arr7) {
            char[] graph = new char[i];
            Arrays.fill(graph, '*');
            System.out.println(new String(graph)+i);
        }

        String[][] str2D = new String[][] {{ "aaa", "bbb"}, {"AAA", "BBB"}};
        String[][] str2D2 = new String[][] {{ "aaa", "bbb"}, {"AAA", "BBB"}};

        System.out.println(Arrays.equals(str2D, str2D2));
        System.out.println(Arrays.deepEquals(str2D, str2D2));

        char[] chArr = {'A', 'B', 'C', 'B', 'E'};

        System.out.println("chArr="+Arrays.toString(chArr));
        System.out.println("index of B= " +Arrays.binarySearch(chArr, 'B'));
        System.out.println("= After sorting=");
        Arrays.sort(chArr);
        System.out.println("chArr="+ Arrays.toString(chArr));
```

```
        System.out.println("index of B= " +Arrays.binarySearch(chArr, 'B'));  
    }  
}
```