

웹 개발자 부트캠프 과정

CodingOn

표준 배열

배열

- 같은 타입의 여러 변수를 하나의 묶음으로 다루는 것
- 처음 선언한 배열의 크기는 변경할 수 없음 (정적 할당)

```
// 배열 선언
int[] intArray;
char[] charArray;

// 배열 생성
int[] intArray = new int[10];
char[] charArray = new char[20];

// 배열 초기화
int[] intArray = { 1, 2, 3, 4, 5 }
```

배열 출력

- for문 활용

```
for( int i=0; i<intArray.length; i++ ) {  
    System.out.print( intArray[i] + " " );  
}
```

배열 출력

- Arrays 클래스의 toString 메소드 활용
 - Arrays 클래스는 배열을 다루기 위한 다양한 메소드가 포함되어 있음.

```
import java.util.Arrays;
```

```
System.out.println(Arrays.toString(intArray));
```

2차원 배열

```
int[][] intArray;  
char[][] charArray;  
double[][] doubleArray  
  
int [][] intArray = new int[2][5];  
char [][] charArray = new char[5][5];  
double [][] doubleArray = new double[5][2];  
  
int intArray[][] = {{0,1,2},{3,4,5},{6,7,8}};  
char charArray[][] = {{'a', 'b', 'c'},{'d', 'e', 'f'}};  
double doubleArray[][] = {{0.01, 0.02}, {0.03, 0.04}}
```

for-each 문

- 배열의 각 원소에 순차적으로 접근하는데 유용한 for 문

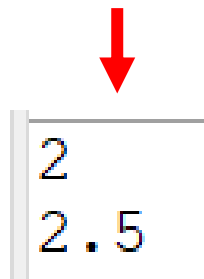
```
for( int value: intArray ) {  
    System.out.print( value + " ");  
}
```

실습. 배열 실습

- 숫자 5개의 정수를 입력 받아, 평균을 구하는 프로그램을 작성해주세요.

힌트! 강제 형 변환

```
System.out.println(5/2);  
System.out.println((double) 5/2);
```



```
2  
2.5
```

5개의 정수를 입력하세요

1 2 3 4 7

평균은 3.4

Arrays 메소드

- `copyOf(arr, copyArrayLength)` : 배열 전체를 복사해 복사할 길이 만큼 지정하여 복사한 새로운 배열로 반환
- `copyOfRange(arr, sIdx, eIdx)` : 배열 시작 인덱스와 끝 인덱스를 지정하여 복사한 새로운 배열 반환
- `fill(arr, n)` : 배열의 모든 요소를 동일한 값으로 채워주는 함수
- `toString(arr)` : 배열을 문자열로 변환하여 반환
- `sort(arr)` : 배열내의 요소들을 오름차순으로 정렬(원본 배열 순서 변경)

Arrays 메소드

- `equals(arr1, arr2)` : 두 배열의 각각의 요소 값을 비교하여 true/false 반환
- `deepEquals (arr1, arr2)` : 단일 차원 또는 다차원 배열인 두 배열이 같은지 비교하여 true/false 반환
- `binarySearch(arr, idx)` : 배열의 특정 인덱스 요소값을 이진 검색 알고리즘을 사용하여 검색한 후, 해당 위치 반환
 - 이진 검색 알고리즘을 사용하므로 매개변수로 전달되는 배열이 `sort()` 등을 사용하여 미리 정렬되어 있어야만 제대로 동작

ArrayList

ArrayList 클래스

- 표준 배열보다 조금 느릴 수 있지만, 배열에서 많은 조작이 필요할 때 유용하게 사용되는 클래스
- 크기를 미리 정하지 않아도 되는 배열 (동적 할당)

```
import java.util.ArrayList;
```

```
ArrayList<Integer> arrayList1 = new ArrayList<>();  
ArrayList<String> arrayList2 = new ArrayList<>();  
ArrayList<Double> arrayList3 = new ArrayList<>();
```

ArrayList 메소드

- `add(element)` : ArrayList 맨 뒤에 element 추가
- `add(index, element)` : index 위치에 element 삽입
- `addAll(ArrayList)` : ArrayList 뒤에 ArrayList추가
- `size()` : ArrayList의 길이 리턴
- `get(index)` : index에 해당하는 요소 리턴

ArrayList 메소드

- `indexOf(params)` : params와 같은 첫 번째 요소의 index 리턴, 없으면 -1 리턴
- `remove(index)` : index의 요소 삭제
- `clear()` : 모든 요소 삭제

실습. ArrayList 실습

- ArrayList를 만들어주세요.
- 사용자로부터 문자열을 입력받아 ArrayList에 넣어주세요.
- 사용자가 "exit"이라는 값을 입력하면, 배열 안에 있는 값을 모두 출력해주세요.

```
문자를 입력해주세요. :  
코딩온  
문자를 입력해주세요. :  
코딩온 안녕?  
문자를 입력해주세요. :  
응 반가워  
문자를 입력해주세요. :  
exit  
코딩온  
코딩온 안녕?  
응 반가워
```