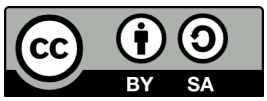
# Arrays

Java Fundamentals

Libre Education



# Arrays

An array is a data structure that can hold multiple values of the same data type. It is therefore considered as a container object.

# Declaring An Array

When declaring an array, like any variable the data type has to be specified as well as the maximum number of values that the array can contain.

### Syntax

```
data_type[] array_name = new data_type[number_of_values];
```

### Example

```
int[] arr = new int[5]; // An array that can hold 5 int values
```

# Indexes

Each position in the array is called an index. Indexes differ from regular counting as they start from 0. So an array that can hold 5 values would have a starting index of 0 and an ending index of 4.

# Assigning Values

## Assigning values to a specific index

### Syntax

```
array_name[index] = value;
```

### Example

```
int[] arr = new int[5];
arr[0] = 4; // Assigning a value to the first (0) index
arr[1] = 8; // Assigning a value to the second (1) index
arr[2] = 7; // Assigning a value to the third (2) index
```

### Assigning values at declaration

When assigning values at declaration, the number of values assigned becomes the number of values the array can contain.

#### Syntax

```
data_type[] array_name = {value, value, value};
```

#### Example

```
int[] arr = {4, 8, 7 , 2, 3};
// An array that can hold 5 values, with 5 values assigned to each index
```

## Accessing Values

To access a specific value in an array, the value is referred to by stating the array name and the index at which the value is located.

#### Syntax

```
array_name[index];
```

#### Example

```
int[] arr = {4, 8, 7 , 2, 3};
System.out.println(arr[2]);
// Output: 7
```

## Manipulating Arrays

There are many manipulations that can be applied to arrays as well many methods (functions). The following is a list of the most common methods and manipulations made to arrays.

### Convert an array to string

```
int[] arr = {4, 8, 7};
String stringArr = Arrays.toString(arr);
System.out.println(stringArr);
// Output: [4, 8, 7]
```

### Sort in ascending order

```
int[] arr = {8, 2, 5, 1};
Arrays.sort(arr);
System.out.println(Arrays.toString(arr));
// Output: [1, 2, 5, 8]
```

### Check if an array contains a value

```
String[] stringArr = {"a", "b", "c"};
boolean containsValue = Arrays.asList(stringArr).contains("a");
System.out.println(containsValue);
// Output: true
```

## Multi-Dimensional Arrays

Multi-dimensional arrays can be considered as an array of arrays where elements can themselves be an array.

### Two dimensional arrays

When using two dimensional arrays it can often be easier to think of the array as a grid where the first bracket specifies the row and the second bracket specifies the column thus giving the "coordinates" of a specific value.

#### Syntax

```
data_type[][] array_name = {1D_Array, 1D_Array, 1D_Array};
```

#### Example

```
int[][] twoDimArr = { {7, 4, 1} ,
                      {2, 5, 9, 3} };

System.out.println(twoDimArr[1][0]);
// Output: 2
System.out.println(Arrays.toString(twoDimArr[1]));
// Output: [2, 5, 9]
```

#### Note

Using just one bracket will refer the whole array. E.g. array_name[index]

## Other Resources

1. The Java Tutorials (docs.oracle.com/javase/tutorial/java/nutsandbolts/ arrays.html)
2. TutorialsPoint (www.tutorialspoint.com/java/java_arrays.htm)
3. Java Programming - WikiBooks (en.wikibooks.org/wiki/Java_Programming/ Arrays)
4. Princeton (introcs.cs.princeton.edu/java/14array)
5. ZetCode (zetcode.com/lang/java/arrays)
6. w3Resource (www.w3resource.com/java-tutorial/java-arrays.php)