

AP CS: Lesson 15: Arrays

Name: _____ Period: _____

Java Syntax:

Arrays:

Required Library for these: (placed above the Class creation

```
import java.util.*; // required for Arrays
```

Array Creation and use:

```
type[] name = new type[length];  
type[] name = {value0, value1, ..., valueN}; // initializing an array explicitly  
name [index] = value; // Assigning a value to an array's indexed location  
public static type methodName(type[] name) { // using an array as a parameter  
public static type[] methodName(parameters) { // returning an array  
methodName(arrayName); // calling a method with an array as its parameter  
type[] name = methodName(parameters); // assigning an array as a returned result  
Examples:  
double[] numbers = double int[8]; // creates an array of doubles with 8 elements  
int[] values = {11, 42, -5, 27, 0, 89}; // creates an array of integers with the  
// specified values of length 6  
public static void actOnIt(double[] input) {} // a double array as a parameter  
public static boolean[] figureItOut(int x, int y) {} // returns a boolean array  
actOnIt(numbers); // calling a method with an array as its parameter  
boolean[] values = figureItOut(x, y); // assigning an array as a returned result
```

Array Length Field:

```
name.length // returns the integer length of the array, note no parenthesis
```

Examples:

```
int arrayLength = numbers.length; // Assigns length of the array to a variable  
for (int i = 0; i < numbers.length; i++) { // for loop going through an array  
    <statements acting on numbers[i]> // one by one element at a time  
}
```

Array Methods: (for the Arrays class)

```
Arrays.equals(array1, array2) // returns true if the two arrays contain same  
// elements in the same order  
Arrays.toString(array) // returns a string representing the array, such as  
// "[10, 30, -25, 17]"  
Arrays.copyOf(array, length) // returns a new copy of an array  
Arrays.fill(array, value) // sets every element to the given value  
Arrays.sort(array) // arranges the elements into sorted order  
Arrays.binarySearch(array, value) // returns the index of the given value in a  
// sorted array (or < 0 if not found)
```

AP CS: Lesson 15: Arrays

Class Notes: