

# Challenge: Implement Swap

Do you know how Swap Function works? Implement to prove yourself!

## We'll cover the following ^

- Swap Function \*
- Function Prototype:
- Output:
- Sample Input
- Sample Output
- Explanation
- Coding Exercise

## Swap Function #

A key step in many sorting algorithms (including selection sort) is swapping the location of two items in an array. Here's an Empty Function that needs to Swap the elements of a given array, as per the specified indices.

### Function Prototype: #

```
public static void swap(int[] array, int firstIndex, int secondIndex);
```

where *array* is the input int array and *firstIndex*/ *secondIndex* specify the integers to swap.

### Output: #

Returns the Updated Array

### Sample Input #

```
arr1 = [9,4,7,1,2,6,5]
```

```
firstIndex = 2  
secondIndex = 0
```

### Sample Output #

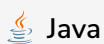
```
arr1 = [7,4,9,1,2,6,5]
```

### Explanation #

You are provided with an **array** of integers, along with the first and second index. You need to return the input array with the integers at the position of **firstIndex** and **secondIndex** *Swapped!*

## Coding Exercise #

Take a close look and design a step-by-step algorithm first before jumping on implementation. This problem is designed for your practice, so try to solve it on your own first. If you get stuck, you can always refer to the solution provided in the solution section. Good Luck!



Java

```
class Solution {  
    public static void swap(int[] array, int firstIndex, int secondIndex) {  
  
        int len = array.length;  
        if (firstIndex <= len && secondIndex <= len)  
        {  
            int temp = array[firstIndex];  
            array[firstIndex] = array[secondIndex];  
            array[secondIndex] = temp;  
        }  
        return;  
    }  
}
```



