```java
public class ArrayOps {
    public static void main(String[] args) {


    }

    public static int findMissingInt (int[] array) {
        int[] array2 = new int[array.length + 1];
        int missing = 999;

        for(int i = 0 ; i < array2.length; i++){
            array2[i] = i;
            }

        for(int i = 0; i < array2.length; i++){
            Boolean didFind = false;

            for(int j = 0; j < array.length; j++){
                if(array[j]==array2[i]){
                    didFind = true;
                    }

                }
                if(!didFind){
                    missing = array2[i];
                    break;
                    }

                }
    return missing;
    }

    public static int secondMaxValue(int [] array) {
        int sec;
        int max;

        if(array[0] >= array[1]){
            sec = array[1];
            max = array[0];
        }

        else{
            sec = array[0];
            max = array[1];

        }

        for(int i = 0; i < array.length; i++){
```

```
            if(array[i] >= max){
                sec = max;
                max = array[i];

            }
        }
        return sec;
    }

    public static boolean containsTheSameElements(int []
array1,int [] array2) {

        for(int i =0; i < array1.length; i++){
            Boolean isContaining = false;
            for(int j=0; j < array2.length; j++){
                if(array1[i] == array2[j]){
                    isContaining = true;
                    break;
                }
            }
            if(!isContaining){return false;}
        }
        return true;
    }

    public static boolean isSorted(int [] array) {
        Boolean isSortedDecreasingly = true;
        Boolean isSortedIncreasingly = true;
        int min = array[0];
        for(int i =0; i < array.length; i++){
            if(array[i] >= min){
                min = array[i];
            }
                else{
                    isSortedIncreasingly = false;
                }

        }

        int max = array[0];
        for(int i =0; i < array.length; i++){
            if(array[i] <= max){
                max = array[i];
            }
                else{
                    isSortedDecreasingly = false;
                }

        }
```

```
        return (isSortedDecreasingly ||
isSortedIncreasingly);
    }

}
    public static void main(String[] args) {
        String s = "One two tHRee world";
        System.out.println(capVowelsLowRest(s));

    }
    private static boolean isVowel(char ch){
        //checking if the letter is vowel type
        if(ch == 'A' || ch == 'E' || ch == 'O' || ch ==
'U' ||ch == 'I' || ch == 'a' || ch == 'o' || ch == 'u' ||
ch == 'e' || ch == 'i'){
            return true;
        }
        else {return false;}
    }

    private static boolean lowVowel(char ch){
        //Checking if a char is a low vowel
        if(ch == 'a' || ch == 'e' || ch == 'o' || ch ==
'i' || ch == 'u'){
            return true;
        }
        else{return false;}
    }

    public static String capVowelsLowRest (String string)
{
        String output = "";
        for(int i = 0; i < string.length(); i++){
            if(isVowel(string.charAt(i))){
                if(lowVowel(string.charAt(i))){
                    output = output +
((char)(string.charAt(i) - 32));
                }
                else {

                    output = output + string.charAt(i);
                }
            }
            else{
                if(string.charAt(i) < 91 &&
string.charAt(i) > 64){
                    output = output +
((char)(string.charAt(i) + 32));
                }
```

```
                else{
                    output = output + string.charAt(i);
                }

            }
        }
        return output;
    }

    private static String lowerAll(String str){
        //lowering the letters in the whole string
        String out ="";
        for(int i = 0 ; i < str.length(); i++){
            if(str.charAt(i) > 64 && str.charAt(i) < 91)
{
                out = out+ ((char)(str.charAt(i) + 32));
            }
            else{out = out + str.charAt(i);}
        }
        return out;
    }

    private static String cutOuterSpaces(String str){
        //cutting spaces at the edges of the string
        while(str.charAt(0) == ' ' ||
str.charAt(str.length()-1) == ' '){
            if(str.charAt(0) == ' '){
                str = str.substring(1, str.length());
            }
            if(str.charAt(str.length()-1) == ' '){
                str = str.substring(0, str.length()-1);
            }
        }
        return str;

    }

    private static String upperFirstCharOfWord(String
str){
        //uppering first letters of words
        String out = "";
        for(int i = 0; i < str.length(); i++){
            if(str.charAt(i) == ' ' && (i+1) <
str.length()){
                if(str.charAt(i+1) > 96 &&
str.charAt(i+1) < 123){
                    out = out + " " +
(char)(str.charAt(i+1) -32);
```

```
                    i++;
                }
            }

            else{
                out = out + str.charAt(i);
            }

        }
        return out;
    }

    private static String removeSpace(String str){
        //removing spaces between words
        String out = "";
        for(int i = 0; i < str.length(); i++){
            if(!(str.charAt(i) == ' ')){
                out = out + str.charAt(i);
            }
        }
        return out;
    }

    public static String camelCase (String string) {
        String output = lowerAll(string);
        output = cutOuterSpaces(output);
        output = upperFirstCharOfWord(output);
        output = removeSpace(output);
        if(output.charAt(0) > 64 && output.charAt(0) <
91)//lowering the first letter
        {
            output = (char)(output.charAt(0) + 32) +
output.substring(1, output.length()-1);
        }
        return output;
    }

    public static int[] allIndexOf (String string, char
chr) {
        int counter =0;
        int counter2 = 0;
        for(int i = 0; i < string.length(); i++){
            if(string.charAt(i) == chr){
                counter++;
            }
        }

        int [] array = new int[counter];
        for(int i = 0; i < string.length(); i++){
```

```
            if(string.charAt(i) == chr){
                array[counter2] = i;
                counter2++;
            }
        }
        return array;
    }
}
```