Array Variable / Defining an Array / Length of an Array / Accessing Array Elements / Reusing Array Variable / Initializing Arrays.

Using Arrays

1. Declare an int array named ***nums*** of size 10.
2. Initializing ***nums*** with random integer values from [1-100] inclusive.
3. Write a method named ***output***, this method takes an int array as a parameter and displays all elements of the array in a column as shown below:
   ```
   public static void output(int[] n){...
   output
   arr[0] : 17
   arr[1] : 42
   arr[2] : 20
   arr[3] : 63
   arr[4] : 43
   ```
4. Write a method named ***average***, this method takes an int array as a parameter and returns the average of all the elements stored in this array.
5. Write a method named ***min***, this method takes an int array as a parameter and returns the minimum value stored in the array.
6. Write a method named ***max***, this method takes an int array as a parameter and returns the maximum value stored in the array.
7. Write a method named **swap**, this method takes 3 parameters: an int array, an int value first index and another int value second index. This methods switches the elements located at indexes first and second. Finally this method should return updated int array back. Example:
   ```
   public static int[] swap(int[] val, int fIndex, int sIndex){...}
   swap(x, 4, 8);
   ```
   Switches the value stored in elements x[4] and x[8]. NOTE: this method should check to see if the max of two indexes passed is less than the length of the array.
8. Write a method named **maxFirstMinLast**, this method sets the max value stored in the array as the first value and the min value stored in the array as the last value. This is done by finding the max value and its index, then swapping this element with the element stored at index 0. Do the same for the min value. This method takes an int array as a parameter and returns updated array.
9. Write a method named **countEven's**, this method accepts an int array as a parameter and returns an int value representing total number of the even values stored in this array.
10. Write a method named **countOdd's**, this method accepts an int array as a parameter and returns an int value representing total number of the odd values stored in this array.

NOTE: Your program should display the result of execution of each method as it is performing these operations.