

ArrayMethods classwork

```
//step 1: Write a method called printArray which takes as its parameter
// an array of doubles and prints it out using the Java array syntax
// e.g. { 2.3, 6.75, 8.234, 10.234 }.
```

```
public static void printArray(double[] data) {
    System.out.print("{");
    for(int i = 0; i < data.length; i++) {
        System.out.printf(" %f", data[i]);
        if (i != data.length-1) {
            System.out.print(",");
        }
    }
    System.out.println(" }");
}
```

```
//step2: Write a main method which declares and initializes
// an array of doubles and call the printArray method.
```

```
double[] dblData = {4.5, 6.7, 8.3};
printArray(dblData);
```

```
//Alternate approach with a problem
public static void printArray(double[] data) {
    System.out.printf("{ %f", data[0]);
    for(int i = 1; i < data.length; i++) {
        System.out.printf(", %f", data[i]);
    }
    System.out.println(" }");
}
```

```
double[] dblData2 = {};
printArray(dblData2);
```

Exception in thread "main" [java.lang.ArrayIndexOutOfBoundsException: Index 0 out of bounds for length 0](#)
at [ArrayMethods.printArray2\(ArrayMethods.java:43\)](#)
at [ArrayMethods.main\(ArrayMethods.java:14\)](#)

```
//step 3: Write a method called createRandomArray which takes two parameters  
// length and max. The method should return an array containing length  
// doubles randomly generated between 0 and max (excluding max).
```

```
public static double[] createRandomArray(int length, double max) {  
    double[] result = new double[length];  
  
    for(int i = 0; i < length; i++) {  
        result[i] = Math.random()*max;  
    }  
    return result;  
}
```

```
//step 4: Update the main method to test the createRandomArray method.  
double[] data = createRandomArray(10, 5);  
printArray(data);
```

//step 5: Write a method called sumValues which takes an array of doubles
// as its parameter and returns the sum of the values in the array.

```
public static double sumValues(double[] array) {  
    double result = 0;  
  
    for(int i = 0; i < array.length; i++) {  
        result += array[i];  
    }  
    return result;  
}
```

//step 6: Update your main method to test sumValues.

```
double sum = sumValues(data);  
System.out.printf("The sum is: %f\n", sum);
```

Test 2 Problems

Test 2: problem 3, version 1

```
int count = 0;
double sum = 0;

System.out.print("Enter a number: ");
double value = input.nextDouble();

while(value != 0) {
    count++;
    sum += value;
    //next
    System.out.print("Enter a number: ");
    value = input.nextDouble();
}

System.out.printf("Number = %d. Sum = %f\n", count, sum);
```

Test 2: problem 3, version 2

```
int words = 0;
int chars = 0;

System.out.print("Enter a word: ");
String w = input.next();

while(!w.equals("done")) {
    words++;
    chars += w.length();
    //next
    System.out.print("Enter a word: ");
    w = input.next();
}
System.out.printf("Words = %d. Characters = %d\n", words, chars);
```

Test 2: problem 4

```
public static boolean isBetween(double value, double min, double max) {  
    return value >= min && value <= max;  
}
```

Test 2: problem 4, version 1

```
public static int countHash(String s) {  
    int count = 0;  
  
    for(int i = 0; i < s.length(); i++) {  
        if(s.charAt(i) == '#') {  
            count++;  
        }  
    }  
  
    return count;  
}
```

Test 2: problem 4, version 2

```
public static int flipCoins(int n) {  
    int count = 0;  
  
    for(int i = 0; i < n; i++) {  
        int flip = (int)(Math.random()*2);  
        if(flip == 0) {  
            count++;  
        }  
    }  
    return count;  
}
```

Test 2: problem 6

```
// Version 1
```

```
for(int i = 1; i <= n; i++) {  
    for(int j = 1; j <= i; j++) {  
        System.out.printf("%d ", j);  
    }  
    System.out.println();  
}
```

```
1  
1 2  
1 2 3  
1 2 3 4  
1 2 3 4 5
```

```
// Version 2
```

```
for(int i = 1; i <= n; i++) {  
    for(int j = i; j >= 1; j--) {  
        System.out.printf("%d ", j);  
    }  
    System.out.println();  
}
```

```
1  
2 1  
3 2 1  
4 3 2 1  
5 4 3 2 1
```