















Dashboard > Tutorials > 30 Days of Code > Day 19: Interfaces

# Day 19: Interfaces ■



Tutorial	Problem	Submissions	Leaderboard	Discussions	Editorial 🔒
ratorial	riobiem	3451113510113	Leaderboard	Discussions	Editorial

### Objective

Today, we're learning about Interfaces. Check out the Tutorial tab for learning materials and an instructional video!

#### Task

The AdvancedArithmetic interface and the method declaration for the abstract int divisorSum(int n) method are provided for you in the editor below. Write the Calculator class, which implements the AdvancedArithmetic interface. The implementation for the divisorSum method must be public and take an integer parameter, n, and return the sum of all its divisors.

**Note:** Because we are writing multiple classes in the same file, do not use an access modifier (e.g.: public) in your *class declaration* (or your code will not compile); however, you must use the *public* access modifier before your *method declaration* for it to be accessible by the other classes in the file.

#### **Input Format**

A single line containing an integer, n.

# Constraints

•  $1 \le n \le 1000$ 

# **Output Format**

You are not responsible for printing anything to stdout. The locked Solution class in the editor below will call your code and print the necessary output.

# Sample Input

6

### **Sample Output**

I implemented: AdvancedArithmetic
12

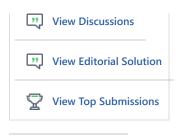
# **Explanation**

The integer  $\mathbf{6}$  is evenly divisible by  $\mathbf{1}$ ,  $\mathbf{2}$ ,  $\mathbf{3}$ , and  $\mathbf{6}$ . Our *divisorSum* method should return the sum of these numbers, which is  $\mathbf{1} + \mathbf{2} + \mathbf{3} + \mathbf{6} = \mathbf{12}$ . The Solution class then prints  $\mathbf{I}$  **implemented: AdvancedArithmetic** on the first line, followed by the sum returned by *divisorSum* (which is  $\mathbf{12}$ ) on the second line.



# Need Help?





Download problem statement

Download sample test cases

**Suggest Edits** 

f ⊌ in

```
Current Buffer (saved locally, editable) &
                                                                                       Java 8
                                                                                                                       *
 1 ▼ import java.io.*;
  import java.util.*;
 3
 4 ▼ interface AdvancedArithmetic{
 5
       int divisorSum(int n);
 6
 7
    //write your code here
  ▼ class Calculator implements AdvancedArithmetic {
 9
10
         int sum = 0;
11
         @override
12
         public int divisorSum(int n) {
13
             for (int i = 1; i < n+1; i++) {
                  if (n % i == 0) {
14
15
                      sum += i;
16
17
             }
18
19
             return sum;
20
         }
21
22 ▼ class Solution {
23
24 ▼
        public static void main(String[] args) {
25
            Scanner scan = new Scanner(System.in);
26
            int n = scan.nextInt();
27
            scan.close();
28
            AdvancedArithmetic myCalculator = new Calculator();
29
30
            int sum = myCalculator.divisorSum(n);
            System.out.println("I implemented: " + myCalculator.getClass().getInterfaces()[0].getName() );
31 ▼
32
            System.out.println(sum);
33
        }
34
                                                                                                             Line: 13 Col: 23
```

<u>♣ Upload Code as File</u> Test against custom input

Run Code

Submit Code

Testcase 0 ✓ Testcase 1 ✓ Testcase 2 ✓

Congratulations, you passed the sample Click the Submit Code button to run your code aga		
Input (stdin)		
6		
Your Output (stdout)		
I implemented: AdvancedArithmetic 12		
Expected Output		
I implemented: AdvancedArithmetic 12		

Copyright © 2017 HackerRank. All Rights Reserved

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Interview Prep | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature