

















Dashboard > Tutorials > 30 Days of Code > Day 21: Generics

Day 21: Generics ■



Tutorial Problem Submissions Leaderboard Discussions Editorial

Objective

Today we're discussing Generics; be aware that not all languages support this construct, so fewer languages are enabled for this challenge. Check out the Tutorial tab for learning materials and an instructional video!

Task

Write a single generic function named *printArray*; this function must take an array of generic elements as a parameter (the exception to this is C++, which takes a *vector*). The locked *Solution* class in your editor tests your function.

Note: You must use generics to solve this challenge. Do not write overloaded functions.

Input Format

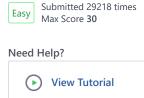
The locked Solution class in your editor will pass different types of arrays to your printArray function.

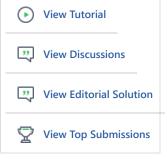
Constraints

• You must have exactly **1** function named *printArray*.

Output Format

Your printArray function should print each element of its generic array parameter on a new line.







Download problem statement

Download sample test cases

Suggest Edits

f ¥ in

```
Current Buffer (saved locally, editable) & 40
                                                                                          Java 8
                                                                                                                           Ö
 1 ▶ import ↔;
 2
    class Printer <T> {
 3
          /**
 4
 5
               Method Name: printArray
 6
               Print each element of the generic array on a new line. Do not return anything.
          *
 7
               Oparam A generic array
 8
 9
10
          // Write your code here
11
12
          public <T> void printArray(T[] array) {
13
              for (T element:
14
                        array) {
15
                   System.out.println(element);
              }
16
17
18
         }
19
20
    }
21
22 ▼ public class Generics {
23
24 🔻
        public static void main(String args[]){
25
             Scanner scanner = new Scanner(System.in);
26
             int n = scanner.nextInt();
27 ▼
             Integer[] intArray = new Integer[n];
28 ▼
             for (int i = 0; i < n; i++) {
29 🔻
                 intArray[i] = scanner.nextInt();
30
31
32
            n = scanner.nextInt();
33 ▼
             String[] stringArray = new String[n];
34 ▼
             for (int i = 0; i < n; i++) {
35 ▼
                 stringArray[i] = scanner.next();
36
37
38
            Printer<Integer> intPrinter = new Printer<Integer>();
39
            Printer<String> stringPrinter = new Printer<String>();
             intPrinter.printArray( intArray );
40
41
             stringPrinter.printArray( stringArray );
42 ▼
             if(Printer.class.getDeclaredMethods().length > 1){
43
                 System.out.println("The Printer class should only have 1 method named printArray.");
44
45
        }
46
    }
                                                                                                                  Line: 19 Col: 1
                      Test against custom input
                                                                                                       Run Code
                                                                                                                    Submit Code
1 Upload Code as File
 Testcase 0 ✓
  Congratulations, you passed the sample test case.
  Click the Submit Code button to run your code against all the test cases.
  Input (stdin)
```

3

```
3
 2
 Hello
 World
Your Output (stdout)
 1
 2
 3
 Hello
 World
Expected Output
 1
 2
 3
 Hello
 World
```

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