PREPARE ^NEW   CERTIFY   COMPETE   CAREER FAIR   Search   jordi480 ∨

Array Of N Elements

Problem	Submissions	Leaderboard	Discussions

Create an array of n integers, where the value of n is passed as an argument to the pre-filled function in your editor. This challenge uses a custom checker, so you can create *any* array of n integers. For example, if $n = 4$, you could return $[1, 1, 1, 1]$, $[1, 2, 3, 4]$, or any other array of equal length.

Note: Code stubs are provided for almost every language in which you must either fill in a blank (i.e., ____) or write your code in the area specified by comments.

Method Signature

```
Number Of Parameters: 1
Parameters: [n]
Returns: List or Vector
```

Input Format

A single integer, n.

Constraints

- $1 \le n \le 100$

- The members returned by the list/vector/array must be integers.

Output Format

The function must return an array, list, or vector of n integers. Stub code in the editor prints this to stdout as a space, comma, or semicolon-separated list (depending on your submission language).

Note: Your output need not match the *Expected Output* exactly; the size of your printed list is confirmed by a custom checker, which determines whether or not you passed each test case.

Sample Input 0

```
10
```

Sample Output 0

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Sample Input 1

```
3
```

Sample Output 1

```
[1, 2, 3]
```

| Scala | ▼ |

```scala
1  object Solution extends App {
2      import scala.io.StdIn.{readLine, readInt}
3
4      def f(num:Int) : List[Int] = {
5          var list = List.range(0,num)// Write your code here
6
7          return list
8
9
10     }
11     println(f(readInt))
12  }
```

Line: 6 Col: 1

⬆ Upload Code as File   ☐ Test against custom input   Run Code   Submit Code

Testcase 0 ✔   Testcase 1 ✔

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Score: 0

Compile Message

```
warning: 1 deprecation (since 2.13.3); re-run with -deprecation for details
```

Compile Time

Input (stdin)

```
10
```

Run Time

Your Output (stdout)

```
List(0, 1, 2, 3, 4, 5, 6, 7, 8, 9)
```

Expected Output

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Compiler Message

```
Success
```