



[Course](#) [Progress](#)

[Course](#) > [Readings/Videos](#) > [Reading 5: Designing Specifications](#) > [Questions](#)


[< Previous](#)





















[Next >](#)

Questions

 [Bookmark this page](#)

joint declaration

1/1 point (graded)

Given this specification:

```
static String join(String delimiter, String[] elements)
    effects: append together the strings in elements, but at each step,
             if there are more elements left, insert delimiter
```

Rewrite the spec so it is declarative, not operational.

- ☐ effects: returns the result of adding all elements to a new `StringJoiner(delimiter)`
- ☐ effects: returns the result of looping through elements and alternately appending an element and the delimiter
- ☒ effects: returns elements joined together with copies of delimiter, i.e.
`elements[0] + delimiter + elements[1] + delimiter`
`+ ... + delimiter + elements[elements.length-1]`



Explanation

The first and second options clearly talk about implementation.

`join` is, in fact, a static method of class `String`.

`StringJoiner` is also part of the standard API.

Submit

 Show Answer

 Answers are displayed within the problem

out of joint

1/1 point (graded)

Which are valid criticisms of the declarative spec from the previous problem? Check all that apply.

☐ Be more clear about the empty delimiter special case☒ Be more clear about the empty elements special case☐ Be less deterministic, implementors need more freedom☐ Be more deterministic, clients need more specific results**Explanation**

Behavior with the empty array is one aspect where none of the specs are immediately clear, including the third one.

The behavior with the empty delimiter, however, doesn't seem in doubt.

The spec was deterministic to begin with, and an underdetermined spec seems unlikely to be nearly as useful.

Submit Show Answer Answers are displayed within the problem

< Previous

Next >

 Some Rights Reserved[Open Learning Library](#)[About](#)[Accessibility](#)[All Courses](#)[Why Support MIT Open Learning?](#)[Help](#)[Connect](#)[Contact](#)[Twitter](#)[Facebook](#)[Privacy Policy](#) [Terms of Service](#)

© Massachusetts Institute of Technology, 2024