

StringJoiner Class in Java

Difficulty Level : Easy Last Updated : 24 Jan, 2022

StringJoiner is a class in [java.util package](#) is used to construct a sequence of characters(strings) separated by a delimiter and optionally starting with a supplied prefix and ending with a given suffix. Though this can also be done with the help of the StringBuilder class to append delimiter after each string, StringJoiner provides an easy way to do that without much code to write.

Constructors of StringJoiner Class

1. StringJoiner(CharSequence delimiter): It constructs a StringJoiner with no characters, no prefix or suffix, and a copy of the supplied delimiter.

Syntax:

```
public StringJoiner(CharSequence delimiter)
```

Parameters: The sequence of characters to be used between each element added to the StringJoiner value

Exception Thrown: [NullPointerException](#) if the delimiter is null

2. StringJoiner(CharSequence delimiter, CharSequence prefix, CharSequence suffix): It constructs a StringJoiner with no characters using copies of the supplied prefix, delimiter, and suffix. If no characters are added to the StringJoiner and methods accessing the string value are invoked, it will return the prefix + suffix (or properties thereof) in the result unless *setEmptyValue* has first been called.

Syntax:

```
public StringJoiner(CharSequence delimiter, CharSequence prefix, CharSequence
```



Parameters:

- The sequence of characters to be used between each element added to the StringJoiner value
- The sequence of characters to be used at the beginning
- The sequence of characters to be used at the end

Exception Thrown: [NullPointerException](#) if prefix, delimiter, or suffix is null.

Methods of StringJoiner Class

Method	Action Performed
<u>add()</u>	Adds a copy of the given CharSequence value as the next element of the StringJoiner value. If newElement is null, then “null” is added.
<u>length()</u>	Returns the length of the String representation of this StringJoiner.
<u>merge()</u>	Adds the contents of the given StringJoiner without prefix and suffix as the next element if it is non-empty. If the given StringJoiner is empty, the call has no effect. Suppose the other StringJoiner is using a different delimiter. In that case, elements from the other StringJoiner are concatenated with that delimiter, and the result is appended to this StringJoiner as a single element.
<u>toString()</u>	Returns the String object of this StringJoiner
<u>setEmptyValue()</u>	Sets the string to be used when determining the string representation of this StringJoiner, and no elements have been added yet; that is when it is empty

Example:

```
// Java program to Demonstrate Methods
// of StringJoiner class

// Importing required classes
import java.util.ArrayList;
import java.util.StringJoiner;

// Main class
public class GFG {

    // Main driver method
    public static void main(String[] args)
    {
        // Creating an empty ArrayList of string type
        ArrayList<String> al = new ArrayList<>();

        // Adding elements to above List
        al.add("Ram");
        al.add("Shyam");
        al.add("Alice");
        al.add("Bob");
```

```
// Creating object of class inside main()
StringJoiner sj1 = new StringJoiner(",");

// Using setEmptyValue() method
sj1.setEmptyValue("sj1 is empty");
System.out.println(sj1);

// Using add() method
sj1.add(al.get(0)).add(al.get(1));
System.out.println(sj1);

// Using length() method
System.out.println("Length of sj1 : "
                  + sj1.length());

StringJoiner sj2 = new StringJoiner(":");
sj2.add(al.get(2)).add(al.get(3));

// Using merge() method
sj1.merge(sj2);

// Using toString() method
System.out.println(sj1.toString());

System.out.println("Length of new sj1 : "
                  + sj1.length());
    }
}
```

Output

```
sj1 is empty
Ram,Shyam
Length of sj1 : 9
Ram,Shyam,Alice:Bob
Length of new sj1 : 19
```

This article is contributed by **Gaurav Miglani**. If you like GeeksforGeeks and would like to contribute, you can also write an article using [write.geeksforgeeks.org](https://www.geeksforgeeks.org/write-geeksforgeeks/) or mail your article to review-team@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.