

Everyone plays a role in preventing cyberattacks.

FREE Cyber

## Java Stream reuse – traverse stream multiple times?

📅 Last Updated: August 30, 2020    👤 By: Lokesh Gupta    📁 Java 8    💎 Java Stream Basics

Is it possible to reuse streams in Java? Learn the alternative of Java stream reuse.

### 1. Can we reuse stream? No.

Java streams, once consumed, can not be reused by default. As [Java docs](#) say clearly,

"A stream should be operated on (invoking an intermediate or terminal stream operation) only once. This rules out, for example, "forked" streams, where the same source feeds two or more pipelines, or multiple traversals of the same stream. A stream implementation may throw `IllegalStateException` if it detects that the stream is being reused."

So the simple answer is : NO, **we cannot reuse the streams** or traverse the streams multiple times. Any attempt to do so will result in error : **Stream has already been operated on or closed.**

### 2. Solution

First of all, any implementation code which require to traverse the stream multiple times – isn't efficient code and need to be refactored.

Only usecase where you might want to create a source and get stream multiple times is – [unit testing](#). In that case, we can always use **stream()** method or just **create a fresh stream**.

Create a fresh stream everytime

```
import java.util.Arrays;
import java.util.List;
import java.util.Optional;

public class Main
{
    public static void main(String[] args)
    {
        List<Integer> tokens = Arrays.asList(1, 2, 3, 4, 5);

        //first use
        Optional<Integer> result = tokens.stream().max(Integer::compareTo);
        System.out.println(result.get());

        //second use
        result = tokens.stream().min(Integer::compareTo);
        System.out.println(result.get());

        //third use
        long count = tokens.stream().count();
        System.out.println(count);
    }
}
```

Program output.

Console

```
5
1
5
```

Reference :

[Java Stream Interface](#)

**Was this post helpful?**

Let us know if you liked the post. That's the only way we can improve.

Yes

No

## Recommended Reading:

1. [String repeat\(\) – Repeat string N times in Java](#)
2. [Sorting a Stream by Multiple Fields in Java](#)
3. [Getting Distinct Stream Items by Comparing Multiple Fields](#)
4. [Chaining Multiple Predicates in Java](#)
5. [Applying Multiple Conditions on Java Streams](#)
6. [Java Stream max\(\)](#)
7. [Java Stream limit\(\)](#)
8. [Collecting Stream Items into Map in Java](#)
9. [Java 8 Stream concat\(\)](#)
0. [Convert Iterable or Iterator to Stream in Java](#)

## Join 7000+ Awesome Developers


Get the latest updates from industry, awesome resources, blog updates and much more.

## Email Address

Subscribe

*\* We do not spam !!*

## Leave a Comment



☐ Add me to your newsletter and keep me updated whenever you publish new blog posts

# Post Comment

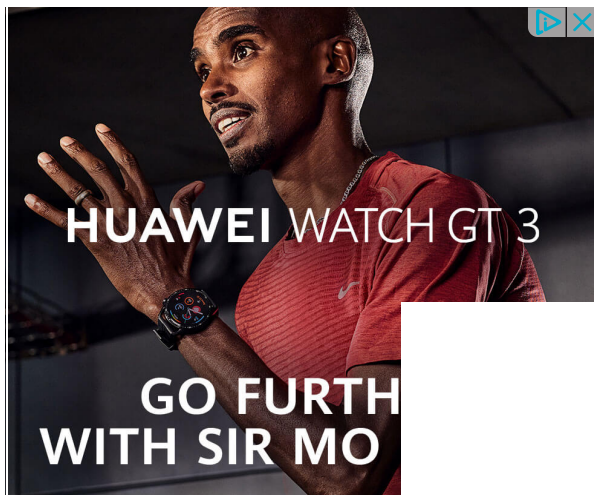
Search ...



Promoted by **usertesting.com**  
Sponsored



A message from our sponsor



## HowToDoInJava

A blog about Java and related technologies, the best practices, algorithms, and interview questions.

### Meta Links

➤ [About Me](#)

- [Contact Us](#)
- [Privacy policy](#)
- [Advertise](#)
- [Guest Posts](#)

## Blogs

[REST API Tutorial](#)



Copyright © 2022 · Hosted on [Cloudways](#) · [Sitemap](#)