Contents

[Preface 2](#_Toc61110857)

[Start 3](#_Toc61110858)

[Modulazation of the JDK under Project Jigsaw 3](#_Toc61110859)

[Provided Money and Currency API 3](#_Toc61110860)

[Try-with 3](#_Toc61110861)

[Annonymous Classes Improvement 4](#_Toc61110862)

[Java annotation @SafeVarags And Variables Argument (Varargs) 6](#_Toc61110863)

[Annotation SafeVarargs 6](#_Toc61110864)

[Variables Argument (Varargs) 9](#_Toc61110865)

[Provided java linker. 10](#_Toc61110866)

[Interface Private Methods 10](#_Toc61110867)

[HTTP 2 Client 11](#_Toc61110868)

[JShell REPL Tool 11](#_Toc61110869)

[Platform and JVM Logging 13](#_Toc61110870)

[Process API Updates 13](#_Toc61110871)

[Collection Factory Of Method 14](#_Toc61110872)

[Create immutable List 14](#_Toc61110873)

[Create immutable Set 14](#_Toc61110874)

[Create immutable Map 14](#_Toc61110875)

[Stream API Improvements 14](#_Toc61110876)

[Multi-Release JRA Files 15](#_Toc61110877)

[@Deprecated Tag Changes 15](#_Toc61110878)

[Reference 16](#_Toc61110879)

[My Github source for java 9: 16](#_Toc61110880)

# Preface

Java se9 was introduced in 21/09/2019 some features impove bellow:

Modulazation of the JDK under Project Jigsaw

Provided Money and Currency API

Try-with

Annonymous Classes Improvement

Java annotation @SafeVarags

Provided java linker.

Interface Private Methods

HTTP 2 Client

JShell REPL Tool

Platform and JVM Logging

Process API Updates

Collection Factory Of Method

Stream API Improvements

Multi-Release JRA Files

@Deprecated Tag Changes

# Start

## Modulazation of the JDK under Project Jigsaw

Java Module System is a major change in Java 9 version. Java added this feature to collect java packages and code into a single unit called module. The verions earlier of java, there was no concept of module to create modular java applications, that why size of application increased and difficult to move around. Even JDK itself was too heavy in size.

Modular Jar file is introduced. This Jar contains module-info.class file in its root folder.

To use **a module**, include the jar file into **modulepath** instead of the classPaht. A **module jar** file added to **classPath** is normal jar file and module-infor.class file will be ignored.

A class is container of fields and methods

A packages is a container of classes and interfaces

A module is a container of packages

Detail visit here: <https://sharecodefull.blogspot.com/2020/12/java-platform-module-system.html>

## Provided Money and Currency API

JSR 354 java Specification request Money and Currency API. The JSR did not make its way into JDK 9 but candidate for future JDK releases. It has version (moneta) implementation of version original (javax.money) is used a lot so i will introduced it.

Features:

To provide an API for handing and calculating monetary amounts.

To define classes representing currencies and monetary amounts as well as rounding, add, subtract, mulity, divide, exchange currencies, formating, parsing.

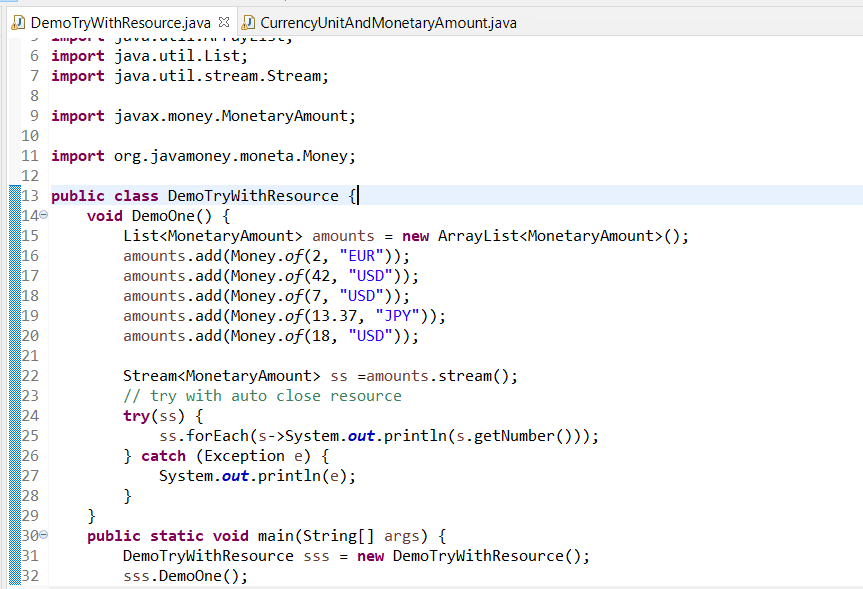
Detail visit here: <https://sharecodefull.blogspot.com/2021/01/provided-money-and-currency-api.html>

## Try-with

Java introduced try-with-resource feature in java 7 that helps to close resource automatically after being used.

In other words, we can say that we don’t need to close resource (file, connection, network, etc ) explicitly, try-with-resource close that automatically by using AutoClosable interface.

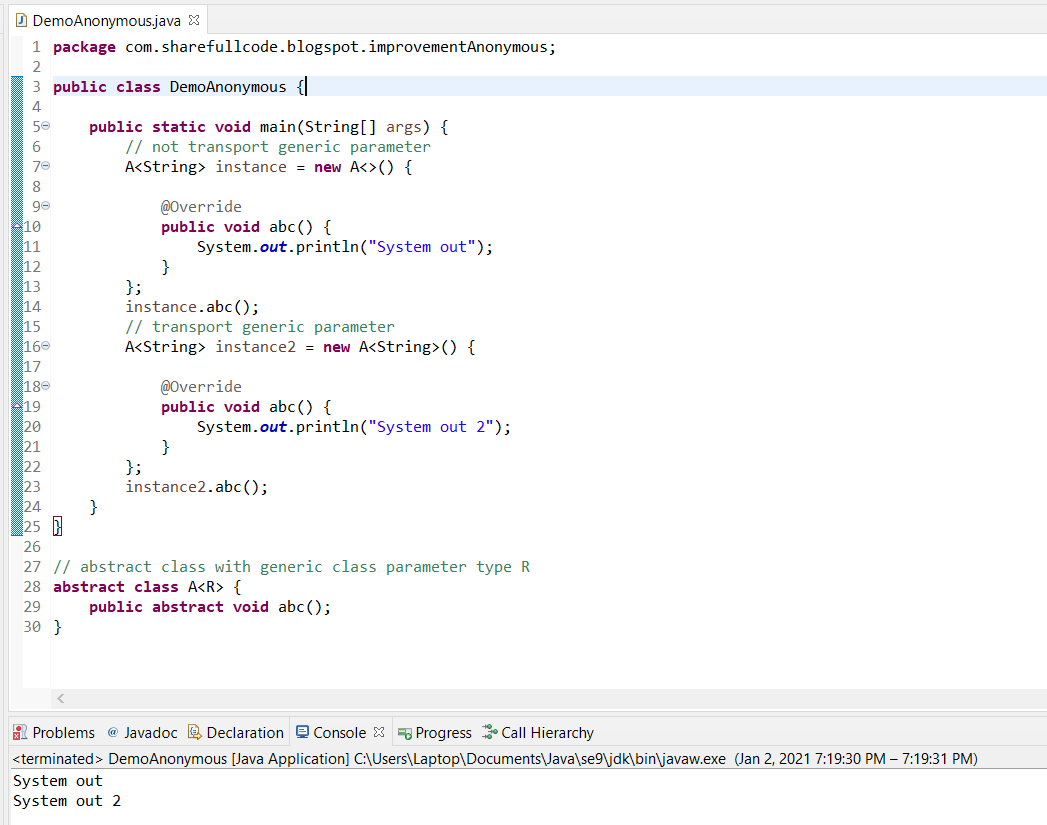
In java 7, try-with-resource has a limitation that requires resource to declare locally within its block.



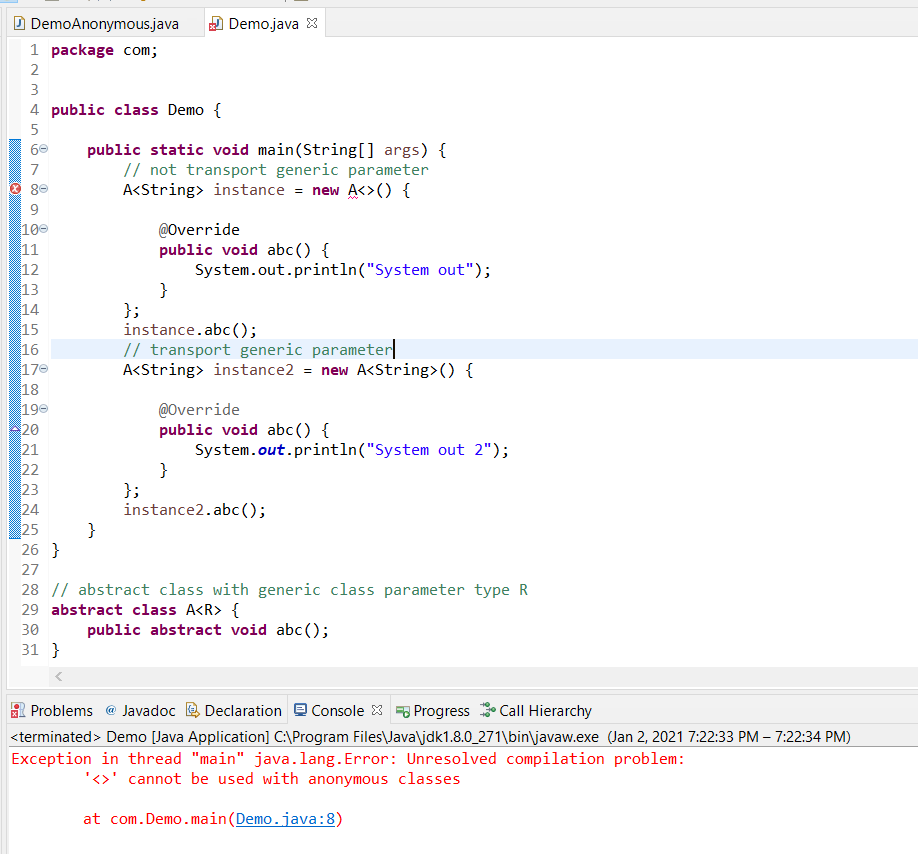
## Annonymous Classes Improvement

In java 9 introduced a new feature that allows us to use diamond operator with anonymous classes, In java 9 as long as the inferred type is denotable, we can use the diamond operator when we create an anonymous inner class.

In java 9 not transport parameter with generic still Ok



With java 8 error complier time



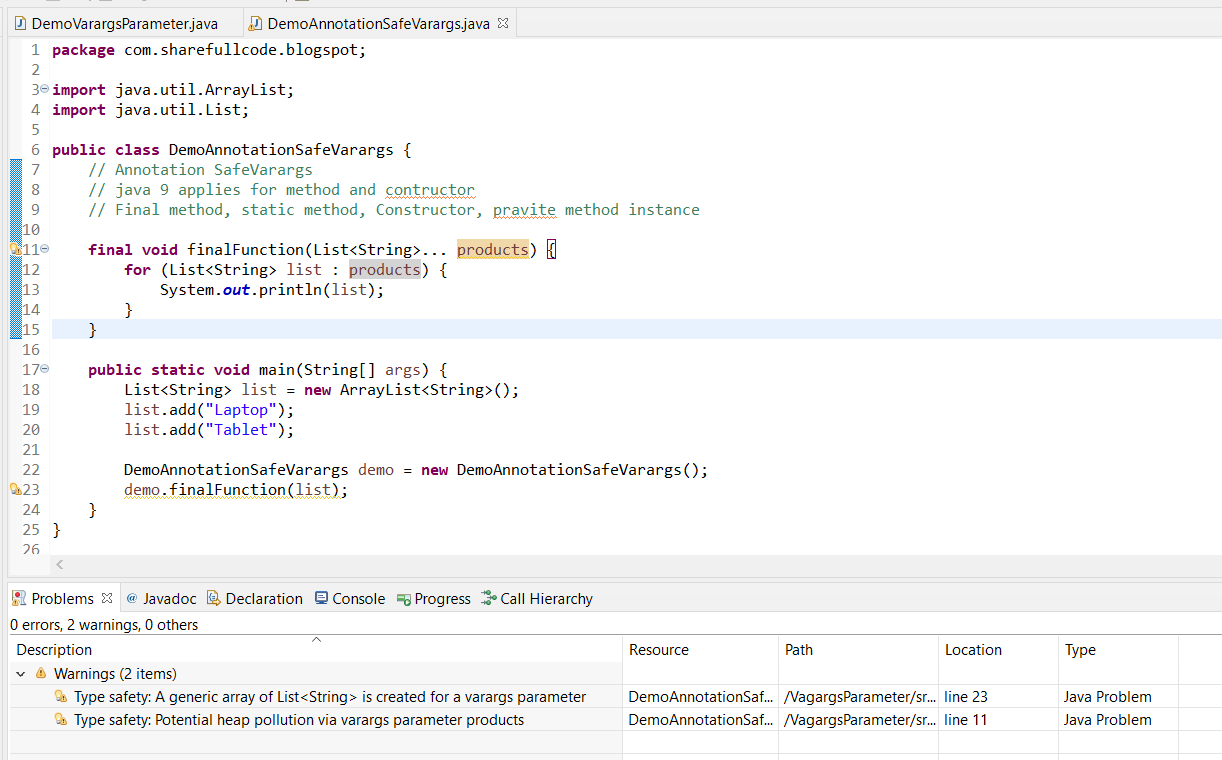
## Java annotation @SafeVarags And Variables Argument (Varargs)

### Annotation SafeVarargs

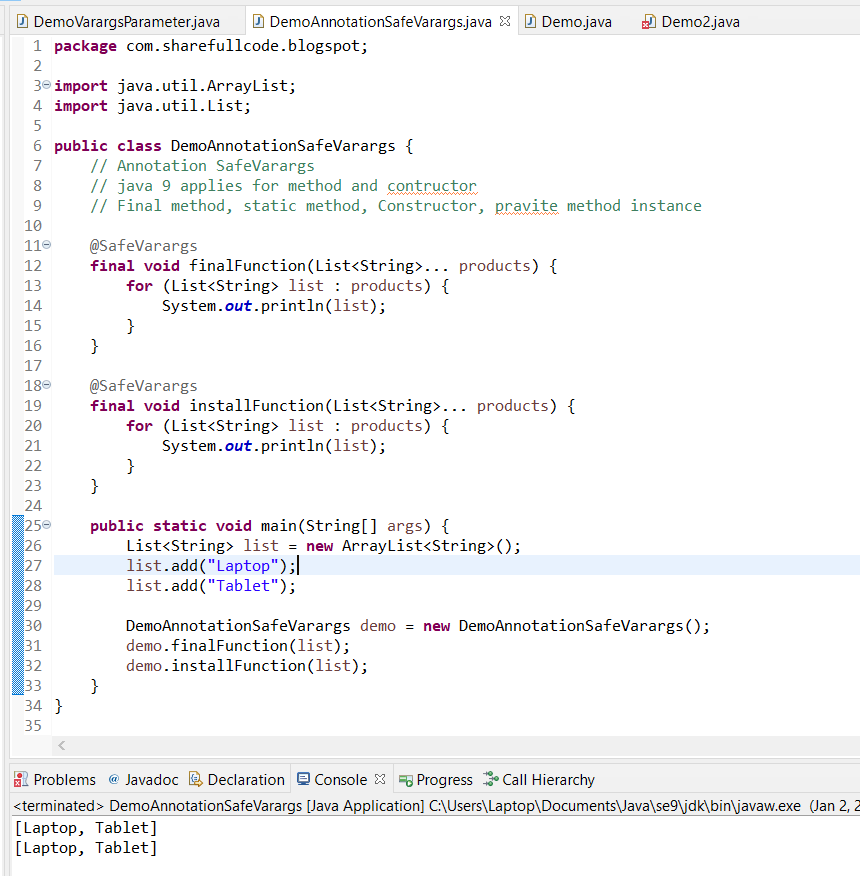
It is an annotation which applies on a method or constructor that takes varags parameters. It is used to ensure that the method does not perform unsafe operations on its varags parameters.

It was included in java 7 and can be only applied on : Final method, Static methods, Constructors.

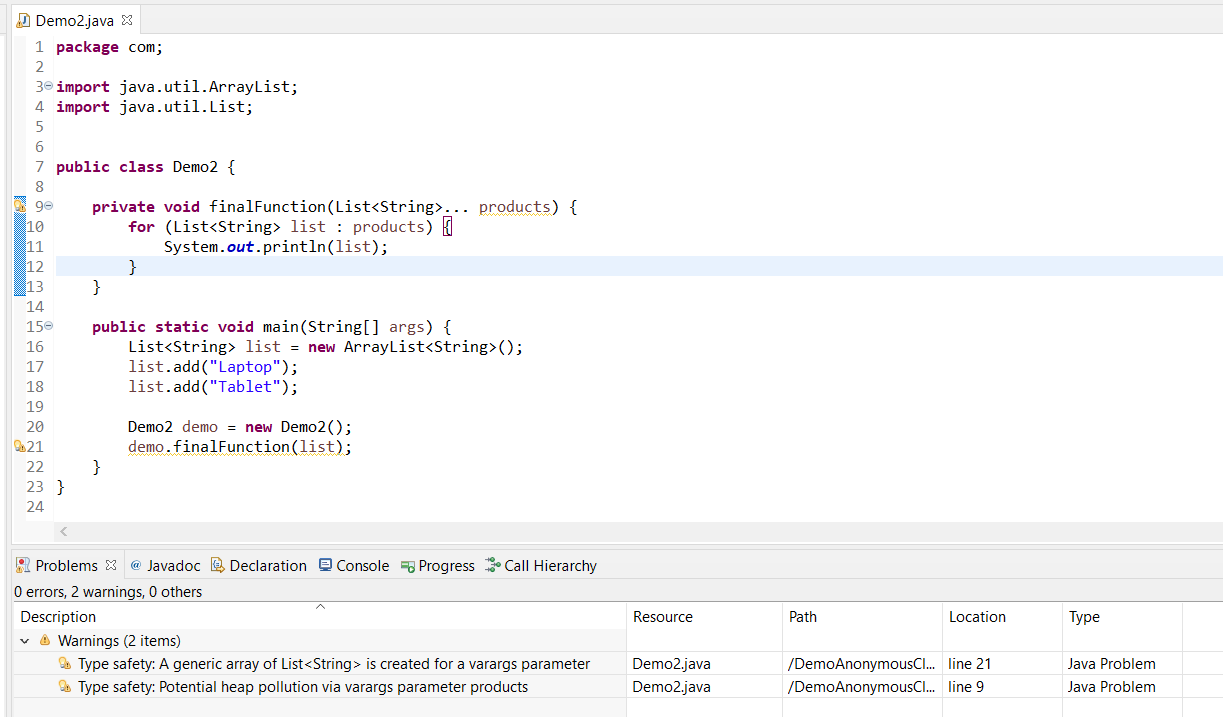
From java 9 it can also be used with private instance methods.

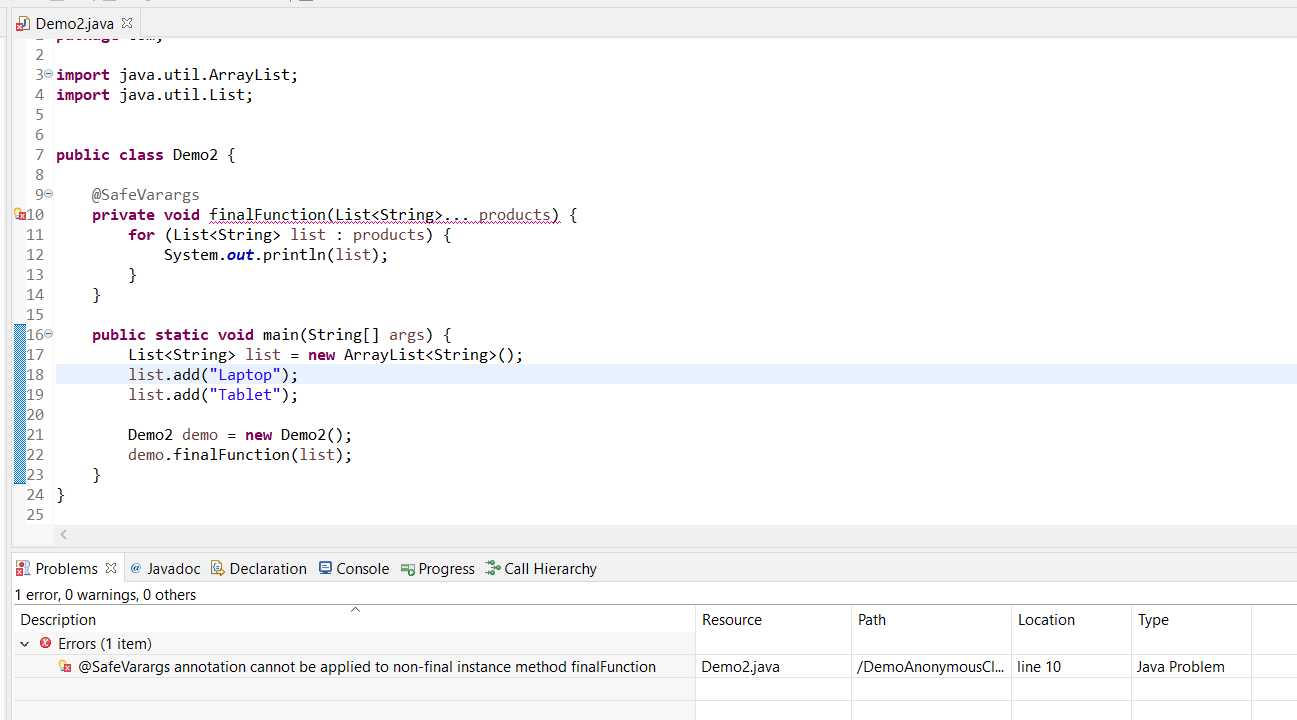


Add Annotation into class then run success



But run with JDK older compile error with private instance method, when not add annotation SafeVarargs that warning , but when add annotation run error because version JDK older se9 not support for private instance method





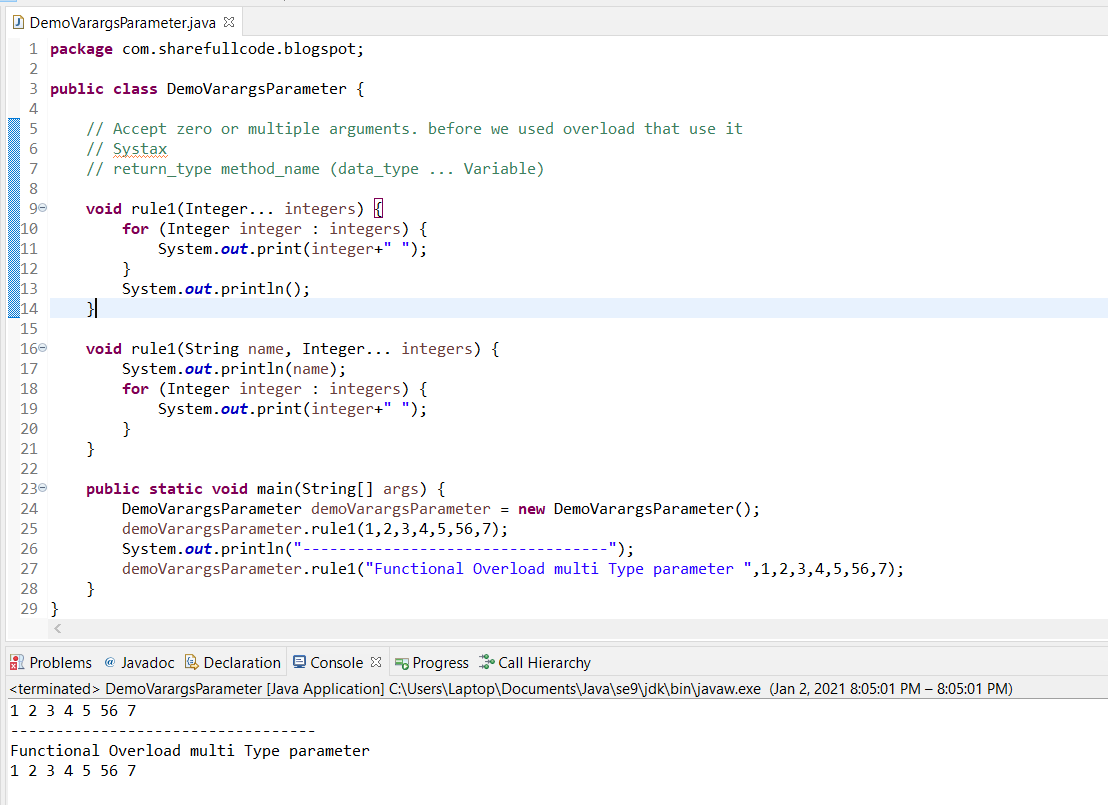
### Variables Argument (Varargs)

It replace Overload function when the parameter has same type.

Has Rules: There can only one Varargs in the method, Varargs must be the last argument if has more parameters.

// void method (Int … a, String … b) complie error

// void method (Int … a, String b) complie error



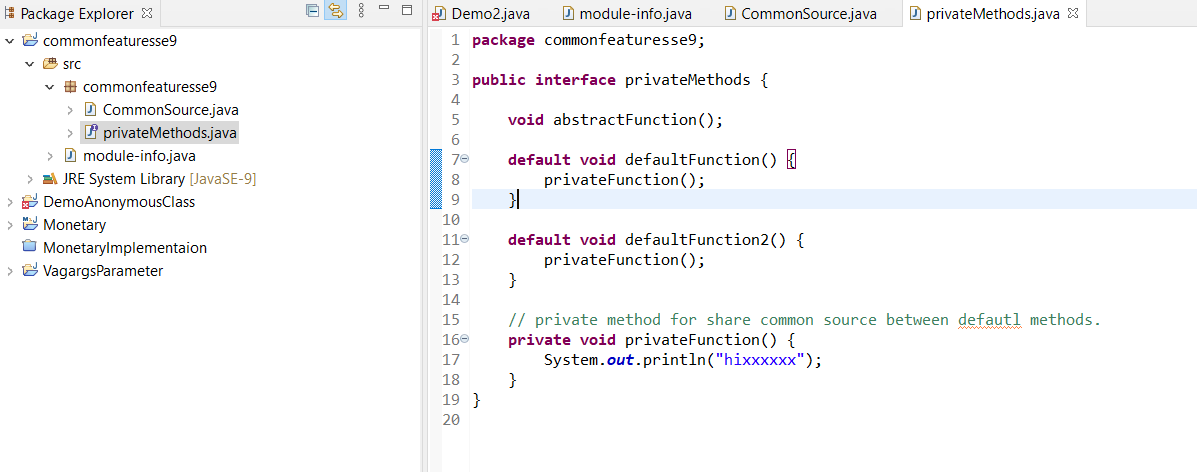
## Provided java linker.

Java linker is a tool that can be used to assemble set of modules into a runtime image. It also allow us to assemble module’s dependencies into custom runtime image.

Link time is a phase between the compile and runtime. Jlink works there for linking and assemble modules to runtime image.

## Interface Private Methods

In java 9 we can create private methods inside an interface. Interface allows us to declare private methods that help to share common code between non-abstract methods.



## HTTP 2 Client

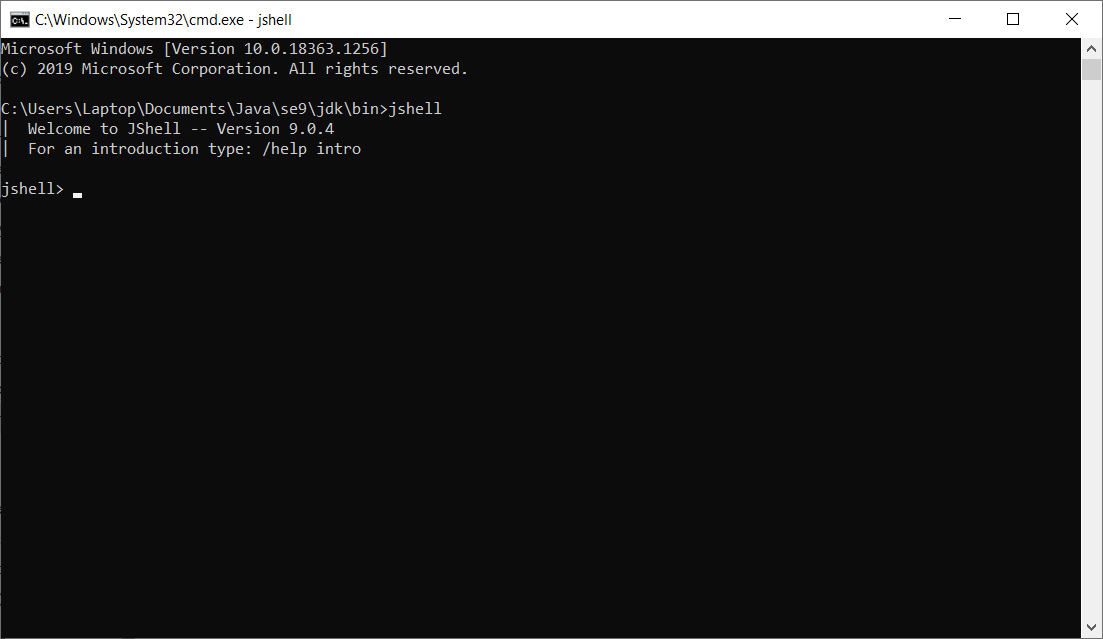
Http/1.1 client was released on 1997 a lot has changed since. So for Java 9 a new API been introduced that is cleaner and cleaner to use and which also support for http/2. New Api use major classes : HttpClient, HttpRequest, HttpResponse.

HttpClient support sych response by Future view detail demo:

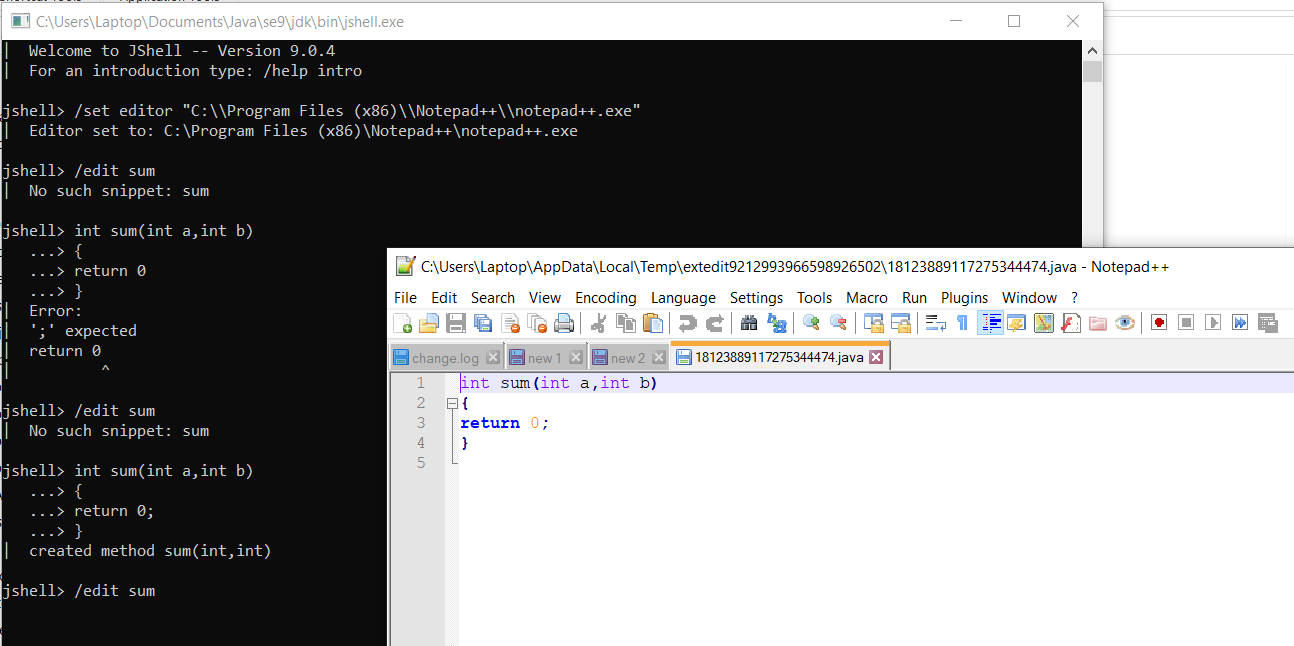


## JShell REPL Tool

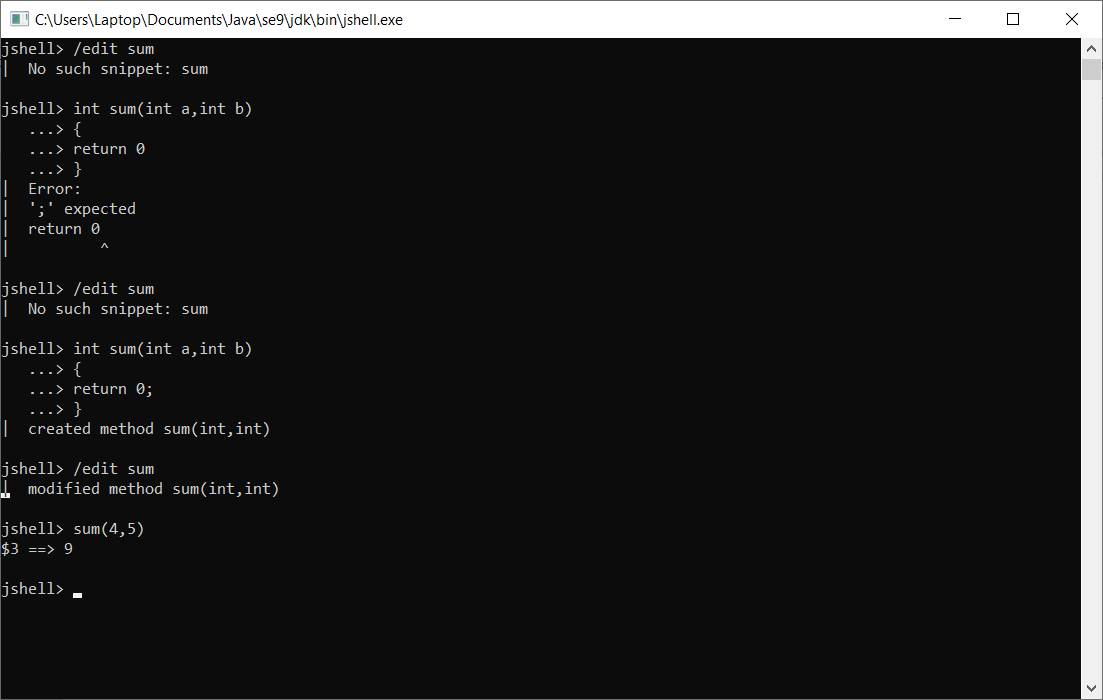
Jshell tool is a new command line interactive tool shipped with java 9 distribution to evaluate declearations, statements and expression written in java. JShell allows us to excute jav code snippet and get immediate results without having to create a solution or project.



You can set editor external when save it run in jShell



Edit -> save -> turn off editor extenal -> goi ham Sum



## Platform and JVM Logging

JDK has improved logging in platform classes JDK Class and JVM components, through new API.

The API is meant to be used by the classes in the JDK, not by Application.

For your application code, you will continue using other logging Apis as before. It automatic

Use java.lang.System.Logger, which provides the logging API you can implement it. Then create class LoggerFinder from extends java.lang.System.LoggerFinder to autocreate instance Logger. View detail here:

<https://sharecodefull.blogspot.com/2021/01/api-logging.html>

## Process API Updates

Prior to java 5, the only way to spawn a new process was to use the Runtime.getRuntime().exec(). Then in java 5, ProcessBuilder API was introduced which supported a cleaner way to of spawning new processes. Now java 9 is adding a new way of getting information about current and any spawned process.

To get information of any process, now you should use java.lang.ProcessHandle.Info interface.

Also use ProcessHandle.allProcesses() to get a stream of ProcessHandle of all processes available in system.

## Collection Factory Of Method

Since java 9, you can create immutable collections such as immutable list, immutable set and immutable map using new factory methods **OF**.

### Create immutable List

Static <E> List<E> of( E ... elements);

List immutable **cannot** Add, removed, replaced, sort. If call these methods exception UnsupportedOperationException. They do **not allow null** element. If given null appearance NullPointerException. They are **serializable** if all elements are serializable. The order of element in the list is the same as the **order of the provided** arguments, or of the elements in the provided array

### Create immutable Set

Static <E> Set<E> .of(E… elements);

Set do not allow duplicate elements as well. The interation order of set element is unspecified and is subject to change.

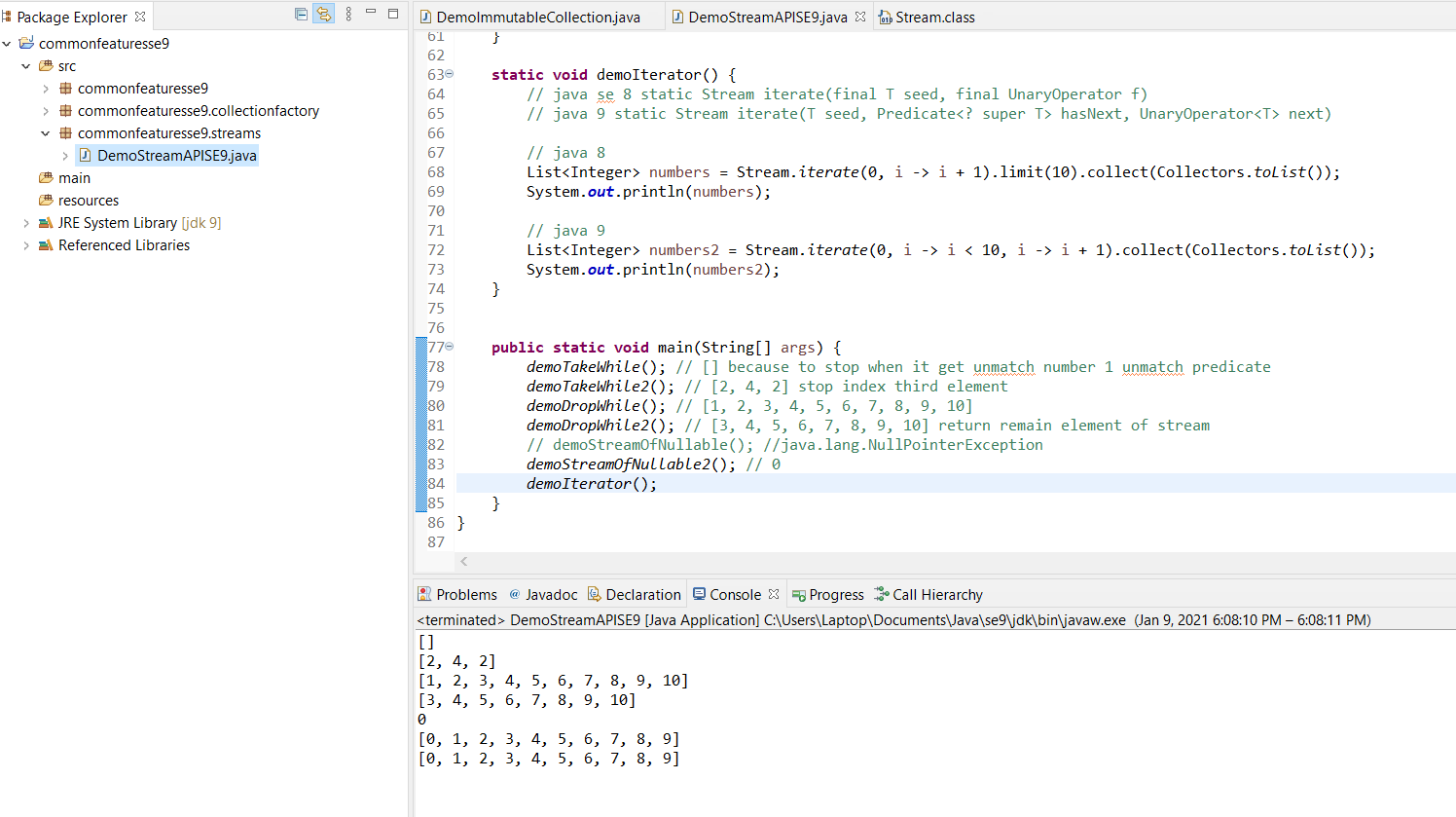
### Create immutable Map

Static <K,V> Map<K, V>.of (K1,V2, k2,v2 , …)

static <K, V> Map<K, V> ofEntries(Entry<? extends K, ? extends V>... entries)

## Stream API Improvements

In java 9, Stream API has improved and new methods added to the Stream interface. TakeWhile, dropWhile and ofNullable, and Overloaded iterate method are added to perform operations on stream elements



## Multi-Release JRA Files

This enhancement is related to how you package application classes in jar files. Perviously, you had to package all classes into a jar file and drop in the classpath of the another application , which wish to use it.

Using multi-release feature, now a jar can contains different versions of a class-compatiable to different JDK releases. The information regarding diffrent versions of a class, and in which jdk version which class shall be picked up by class loaded is stored in MANIFEST.MF file. In this case, MANIFEST.MF file includes the entry Multi-Release : True in its main section.

Ex

JAR content root

A.class

B.class

C.class

D.class

META-INF

MANIFEST.MF

versions

9

A.class

B.class

10

A.class

Let’s assume that in JDK 10, A.class isupdated to leverage some java 10 features, then this jar file can be updated like this above example.

## @Deprecated Tag Changes

From java 9, @Deprecated annotation will have to attributes: forRemoval and since.

forRemoval: indicates whether the annotated element is subject to removal in a future version.

Since: it returns the version which the annotated element became deprecated.

# Reference

## My Github source for java 9:

<https://github.com/nguyenthinhit996/sharefullcode/tree/java/Learn%20Java/NewFeatureSE9>

**This section is non commercial mainly sharing and advance knowlage for java.This tutorials has referenced document from the list below if you has complain for license, i will remove all from internet. Thank you all everything.**

<https://www.javatpoint.com/java-versions>

<https://howtodoinjava.com/java9/java9-new-features-enhancements>

<https://www.baeldung.com/java-money-and-currency>

<https://dzone.com/articles/looking-java-9-money-and>