



JAVA 9 AND BEYOND: MODULES, THE REACTIVE STREAM API

OBJECT ORIENTED PROGRAMMING I

Sercan Külcü | Object Oriented Programming I | 10.01.2023

Contents

Introduction	2
Modules	3
Enhancements to the JVM	4
Reactive Streams API	5
Other Features	6
Conclusion	7

Introduction

Java 9 introduced several new features and improvements, including the introduction of modules, enhancements to the Java Virtual Machine (JVM), and support for reactive programming. In this chapter, we will explore these new features and how they can be used in Java applications.

Modules

Modules are a new feature in Java 9 that allow you to encapsulate code and define clear boundaries between different parts of your application. Modules make it easier to manage dependencies, improve performance, and reduce the risk of conflicts between different libraries and components.

With modules, you can define a set of packages that are accessible to other modules, and you can specify which modules your module depends on. This makes it easier to manage dependencies and ensure that your application has access to the required libraries and components.

Enhancements to the JVM

Java 9 introduced several enhancements to the JVM, including improved garbage collection and support for multi-release JARs. The new garbage collector, called the Garbage-First (G1) collector, improves performance and reduces pause times. Multi-release JARs allow you to package different versions of the same class for different versions of the JVM.

Reactive Streams API

The Reactive Streams API is a new API for reactive programming in Java. Reactive programming is a programming paradigm that focuses on asynchronous and event-driven programming. The Reactive Streams API provides a set of interfaces and classes that allow you to build reactive systems using Java.

The Reactive Streams API provides a set of standard interfaces for working with reactive streams, including Publisher, Subscriber, and Processor. These interfaces provide a common set of methods for producing, consuming, and processing streams of data in a reactive manner.

Other Features

Java 9 also introduced several other features, including:

JShell, an interactive tool for exploring and testing Java code

HTTP/2 support in the Java Standard Library

Support for private interface methods

Enhanced deprecation warnings

Conclusion

Java 9 introduced several new features and improvements, including modules, enhancements to the JVM, and support for reactive programming. These new features can help you build more modular, performant, and responsive applications. By mastering these new features, you can become a more productive and effective Java developer. However, it's important to remember that not all of these features may be appropriate for every project, and you should carefully consider the needs and requirements of your project when deciding whether to use these new features.