↖ Topic Index

Emacs Support for Java 🚧

Description	Keystroke	Function	Note	
Java Programming Language Support See also: • Σ **Customize** Last updated on:	Emacs has built-in support for Java. The java-mode is one of the cc-modes. Emacs 30 and later also provides the tree sitter java-ts-mode. • PEL additional support for Java is activated by **pel-use-java** customizable user-option. 🚧 This page is a placeholder. It will be filled with more information as I perform more testing with Java code and the various facilities provided by tree-sitter and LSP. 2025-10-09			
Open this PDF file. See also: Σ **Help/ Info**	**<f11> SPC J <f1>** **<f12> <f1>**	(**pel-help-pdf** &optional OPEN-WEB-PAGE)	Open the **PEL - Java** local PDF. If the prefix argument (like **C-u** or **M--**) is used, then it opens the remote GitHub hosted raw PDF instead. If the **pel-flip-help-pdf-arg** user-option is set it's the other way around.	
Σ **Customize** PEL Java support	**<f11> SPC J <f2>** **<f12> <f2>**	(**pel-customize-pel** &optional OTHER-WINDOW)	Customize PEL Java support: pel-use-java • If OTHER-WINDOW is non-nil (use **C-u**), display in another window.	
Σ **Customize** Emacs Java support	**<f11> SPC J <f3>** **<f12> <f3>**	(**pel-customize-library** &optional OTHER-WINDOW)	Customize Emacs Java support: java, c • If OTHER-WINDOW is non-nil (use **C-u**), display in another window.	

Java & Emacs Reference

Java Language	• Java programming language @ Wikipedia
	  • Java Versions @ Wikipedia
	  • Java Editions @ Wikipedia
	• JRE, Java Runtime Environment @ Wikipedia		

JDK: Java Development Kit	• OpenJDK : Open Java Development Kit
	  • OpenJDK @ Wikipedia
	  • OpenJDK Builds @ Wikipedia
	  • Eclipse Adoptium : Temurin @ Wikipedia
	  • Eclipse Adoptium Temurin @ home
		Long Term Support (LTS) versions of JDK:
		• Java 8 (LTS)
		• Java 11 (LTS)
		• Java 17 (LTS)
		• Java 21 (LTS)
		• Java 25 (LTS)	

• **Install JDK**	• **Installing on macOS:**
	  • **with Temurin**
	  • with homebrew
		• Eclipse Temurin provides a free and open source implementation of the OpenJDK, provides several versions (including 25) and is well trusted. It provides a macOS installer. Several versions can be installed.
		• On a Apple ARM architecture system, it installs Eclipse Temurin 25+36-LTS to `/Library/Java/JavaVirtualMachines/temurin-25.jdk`
		• Executing '`java --version`' prints:
		`openjdk 25 2025-09-16 LTS`
		`OpenJDK Runtime Environment Temurin-25+36 (build 25+36-LTS)`
		`OpenJDK 64-Bit Server VM Temurin-25+36 (build 25+36-LTS, mixed mode, sharing)`	

	• **Installing on Linux:**	• Eclipse Temurin also supports Linux.
		  • The installation depends on your Linux distribution. The site provides a tarsal file and instructions to use the various package managers.
		  • Depending on your distribution the package manager provided package may not support the latest JDK version.	

	• **Using multiple JDK versions**	It is possible to install several JDK inside a system. • Select the JDK to use by setting the **JAVA_HOME** environment variable to the absolute path of the /Contents/Home sub-directory of the selected JDK directory. • For example, on macOS, set JAVA_HOME to `/Library/Java/JavaVirtualMachines/temurin-21.jdk/Contents/Home` to select version 21 of Temurin if there are multiple versions installed. If there is only one version that version is used even if there is no JAVA_HOME environment variable set. If there are several versions installed and no JAVA_HOME, *it seems* that the latest is used.	
		It's also possible to use the **jenv** script to control the Java environment globally, in a directory or in the shell.	
Query Java support	Tool from **USRHOME**	• **info-java** prints information related to Java environment available to the shell. • **use-sdkman** activates sdkman in the current shell. You must install sdkman separately.	
Java Build Tools			

• **sbt**	• sbt - Simple Built Tool
	  • sbt @ Wikipedia
	  • sbt home
	  • sbt @ Github
	  • sbt 1.x Documentation
	  • learning sbt
	  • sbt by example
	• sbt 2.x Documentation (preliminary as of Oct 2025)
		sbt is a built tool that can build Java, Scala and Kotlin projects. It is the de-factor build tool for Scala and its mainly written in Scala with some Java.	Installing sbt
			• The best way to install sbt is to use the **sdkman** tool and run the **`sdk install sbt`** command.
			  • This requires **sdkman** properly installed and available in your shell.
			  • The USRHOME system provides a command you can use to inject support of **sdkman** to your shell without having to modify the startup scripts. USRHOME provides the **envfor-sdkman** source script that is invoked by the use-sdkman alias.
			  • Once this script is executed, the sdk command becomes available.