



链滴

# Java SDK 多版本管理

作者: [bugless](#)

原文链接: <https://ld246.com/article/1617958089228>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

根据 JRebel 的调查报告显示，58% 的开发人员依旧使用 Java 8。虽然，近半同行依旧坚守这 Java 8 但是，Java 8 以后，6 个月更新一次，目前最新的 Java 12 已经发布，Java 11 已成为 LTS 版本。以，紧跟时代潮流，不断学习，是开发者基本的素质（我学不懂啊），要体验新特性，就需要安装不同的 JDK，此时，电脑就有多套环境了，那学习管理 JDK 环境的正确姿势是很有必要的。



很多中间件/框架（比如 Dubbo）需要兼容多个 JDK 版本，开发、测试时需要不断切换版本，通过 `JAVA_HOME` 手动修改让人痛苦不堪。

之前了解到 Node.js 有很多版本管理工具，例如 nvm，通过 nvm 切换 Node.js 版本只需要一个命令，非常方便高效，幸运的是 Java 也有类似工具或方式来管理多个 JDK 版本，一行命令快速切换环境无需重复配置环境变量。

本文主要介绍了以下四种工具方式：

1. SDKMan，目前在用
2. jEnv
3. Jabba
4. 自定义脚本

Mac BigSur 11.1

<https://stackoverflow.com/questions/19039752/removing-java-8-jdk-from-mac>

```
#移除JDK
sudo rm -rf /Library/Java/JavaVirtualMachines/adoptopenjdk-8.jdk
#移除其他插件
sudo rm -rf /Library/PreferencePanes/JavaControlPanel.prefPane
sudo rm -rf /Library/Internet\ Plug-Ins/JavaAppletPlugin.plugin
sudo rm -rf /Library/LaunchAgents/com.oracle.java.Java-Updater.plist
sudo rm -rf /Library/PrivilegedHelperTools/com.oracle.java.JavaUpdateHelper
sudo rm -rf /Library/LaunchDaemons/com.oracle.java.Helper-Tool.plist
sudo rm -rf /Library/Preferences/com.oracle.java.Helper-Tool.plist
#验证
java --version
#出现以下弹框，说明清理干净
```



您需要安装JDK才能使用“java”命令行工具。

点按“更多信息...”来访问Java开发工具包下载网站。

更多信息...

好

<http://bugless.site>

□



<http://bugless.site>

SDKMAN! 是用于在大多数基于Unix的系统上管理多个软件开发套件的并行版本的工具。它提供了便的命令行界面 (CLI) 和API, 用于安装, 切换, 删除和列出候选人。它以前被称为Groovy enViroment Manager GVM, 它受到了非常有用的RVM和rbenv工具的启发, 而Ruby社区广泛使用了RV和rbenv工具。

```
#默认命令行安装, 安装位置在:$HOME/.sdkman  
curl -s "https://get.sdkman.io" | bash  
source "$HOME/.sdkman/bin/sdkman-init.sh"
```

```
#自定义安装  
export SDKMAN_DIR="/usr/local/sdkman" && curl -s "https://get.sdkman.io" | bash
```

原文链接: [Java SDK 多版本管理](#)

```
#验证
sdk version
```

```
> sdk version
SDKMAN 5.11.0+644
```

所有命令请参见: <https://sdkman.io/usage>

```
sdk list //查看支持的软件
```

支持的软件库: Ant、AsciidoctorJ、Bpipe、BTrace、Ceylon、Concurmas、CRash、CUBA、CXF、Gaiden、Gradle profiler、Grails、Groovy、http4k、Infrastructor、java、JBake、J'Bang、Kara、Kotlin、Kscipt、Lazybones、Leiningen、Maven、Micronaut、sbt、Scala、Spark、Spring Bot、Sshoogr、Vert.x、VisualVM

```
#查看 java 相关的版本
```

```
sdk list java
```

```
#安装软件语法
```

```
# sdk install 软件 你想安装的版本 (sdk default java IDENTIFIER)
```

```
sdk install java 8.0.265.hs-adp # 安装 jdk8
```

```
sdk install java 12.0.2.hs-adpt #安装 jdk12
```

```
# 如果你是通过 brew 安装了 JDK, 则只需纳入 SDK 管理即可
```

```
sdk install java JDK8 /Library/Java/JavaVirtualMachines/adoptopenjdk-8.jdk/Contents/Home
```

```
sdk install java JDK12 /Library/Java/JavaVirtualMachines/adoptopenjdk-12.jdk/Contents/Hom
```

```
sdk install java JDK14 /Library/Java/JavaVirtualMachines/adoptopenjdk-14.jdk/Contents/Hom
```

```
#-----
```

```
#卸载软件语法
```

```
# sdk uninstall 软件 软件版本
```

```
sdk uninstall java 8.0.265.hs-adp
```

```
#查看软件现有版本
```

```
sdk current java
```

```
#查看所有本地包的当前版本
```

```
sdk current
```

```
#升级
```

```
sdk upgrade java
```

```
sdk upgrade
```

```
#自我升级
```

```
skd selfupdate
```

```
#强制重新安装
```

```
sdk selfupdate fore
```

□

Java版本设置为默认版本 `sdk default java JDK8`

▾arrow\_down\_small仅将Java版本用于当前终端会话

`sdk use java 14.0.2-oracle`





4).

```
curl -sL https://github.com/shyiko/jabba/raw/master/install.sh | bash && . ~/.jabba/jabba.sh
```

```
# list available JDK's
jabba ls-remote
# you can use any valid semver range to narrow down the list
jabba ls-remote zulu@~1.8.60
jabba ls-remote "*@>=1.6.45 <1.9" --latest=minor

# install Oracle JDK
jabba install 1.15.0
# install Oracle Server JRE
jabba install sjre@1.8
# install Adopt OpenJDK (Hotspot)
jabba install adopt@1.8-0
# install Adopt OpenJDK (Eclipse OpenJ9)
jabba install adopt-openj9@1.9-0
# install Zulu OpenJDK
jabba install zulu@1.8
jabba install zulu@~1.8.144 # same as "zulu@>=1.8.144 <1.9"
# install IBM SDK, Java Technology Edition
jabba install ibm@1.8
# install GraalVM CE
jabba install graalvm@1.0-0
# install OpenJDK
jabba install openjdk@1.10-0
# install OpenJDK with Shenandoah GC
jabba install openjdk-shenandoah@1.10-0

# install from custom URL
# (supported qualifiers: zip (since 0.3.0), tgz, tgx (since 0.10.0), dmg, bin, exe)
jabba install 1.8.0-custom=tgz+http://example.com/distribution.tar.gz
jabba install 1.8.0-custom=tgx+http://example.com/distribution.tar.xz
jabba install 1.8.0-custom=zip+file:///opt/distribution.zip

# uninstall JDK
jabba uninstall zulu@1.6.77

# link system JDK
jabba link system@1.8.72 /Library/Java/JavaVirtualMachines/jdk1.8.0_72.jdk

# list all installed JDK's
jabba ls

# switch to a different version of JDK (it must be already `install`ed)
jabba use adopt@1.8
jabba use zulu@~1.6.97
```

```
echo "1.8" > .jabbarc
# switch to the JDK specified in .jabbarc (since 0.5.0)
jabba use

# set default java version on shell (since 0.2.0)
# this version will automatically be "jabba use"d every time you open up a new terminal
jabba alias default 1.8
```



```
# 1. 安装所需的jdk版本
brew cask install homebrew/cask-versions/java8
brew update
brew cask install adoptopenjdk8
brew cask install adoptopenjdk9
brew cask install adoptopenjdk10
brew cask install adoptopenjdk11
brew cask install adoptopenjdk12
#安装路径在: /Library/Java/JavaVirtualMachines
```

```
#展示所有的jdk版本
/usr/libexec/java_home -V
```

```
#2. 编写脚本
```

```
//最简单但繁琐的方式
export JAVA_HOME=/Library/Java/JavaVirtualMachines/1.6.0.jdk/Contents/Home
export JAVA_HOME=/Library/Java/JavaVirtualMachines/1.8.0.jdk/Contents/Home
```

```
//最佳实践
```

```
//-----.bash_profile编辑开始-----
vi ~/.bash_profile
# JDK 6、JDK 8、JDK 12 的 export 命令
export JDK6_HOME="/Library/Java/JavaVirtualMachines/1.6.0.jdk/Contents/Home"
export JDK8_HOME="/Library/Java/JavaVirtualMachines/adoptopenjdk-8.jdk/Contents/Home"
export JDK12_HOME="/Library/Java/JavaVirtualMachines/openjdk-12.0.1.jdk/Contents/Home"
```

```
//或者按以下方式
```

```
export JAVA_HOME=$(/usr/libexec/java_home -v14)

export JAVA_8_HOME=$(/usr/libexec/java_home -v1.8)
export JAVA_11_HOME=$(/usr/libexec/java_home -v11)
export JAVA_14_HOME=$(/usr/libexec/java_home -v14)
```

```
# alias 命令链接到 export 命令
alias jdk6="export JAVA_HOME=$JDK6_HOME"
alias jdk8="export JAVA_HOME=$JDK8_HOME"
alias jdk12="export JAVA_HOME=$JDK12_HOME"
```



```
# 或者以下方式
alias java8='export JAVA_HOME=$JAVA_8_HOME'
alias java11='export JAVA_HOME=$JAVA_11_HOME'
alias java14='export JAVA_HOME=$JAVA_14_HOME'
```

```
#设置默认的JDK
export jdk8
#或者以下方式
export JAVA_HOME=$(/usr/libexec/java_home -v14)
//-----.bash_profile编辑结束-----
```

```
:wq!
source ~/.bash_profile
```

//使用，在bash中输入 jdk6/jdk8/jdk12 回车，切换java环境变量  
jdk6

标 题：Java SDK 多版本管理

作 者：

出 处：<http://bugless.site>

创建时间：2021-04-09 16:46:19

关于博主：坐标广州，喜欢折腾各种东西的工程师，如有问题探讨可以直接下方留言。

声援博主：如果您觉得文章对您有帮助，可以评论、订阅、收藏。您的鼓励是博主的最大动力！