








Java 下载:

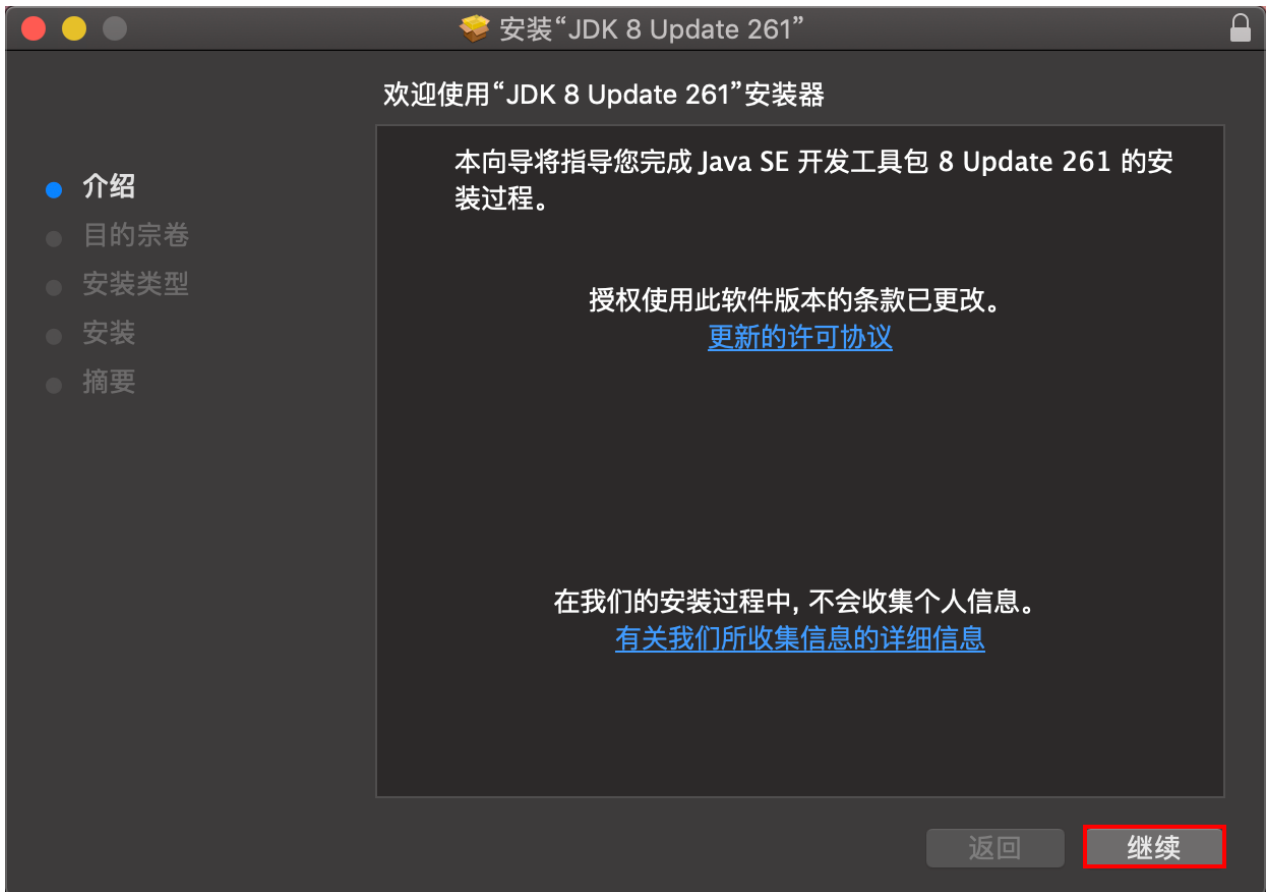
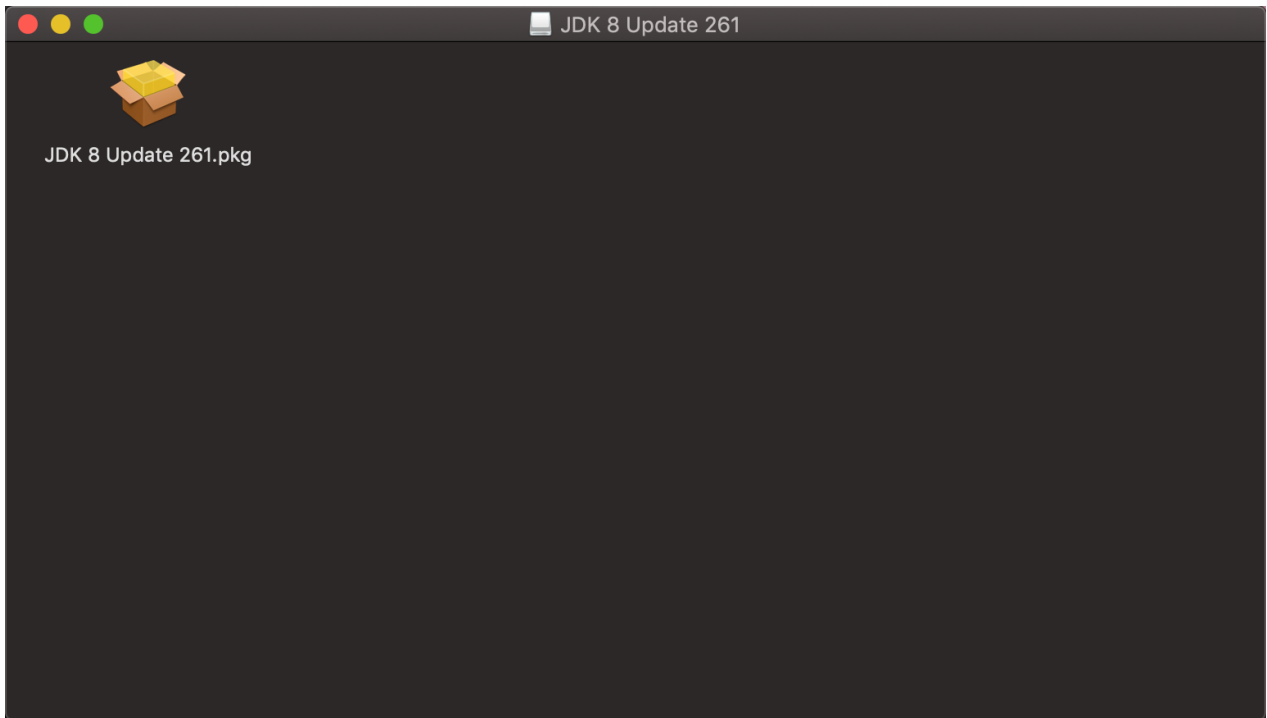
<https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>

Java SE Development Kit 8u261		
This software is licensed under the Oracle Technology Network License Agreement for Oracle Java SE		
Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	73.4 MB	 jdk-8u261-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	70.3 MB	 jdk-8u261-linux-arm64-vfp-hflt.tar.gz
Linux x86 RPM Package	121.92 MB	 jdk-8u261-linux-i586.rpm
Linux x86 Compressed Archive	136.81 MB	 jdk-8u261-linux-i586.tar.gz
Linux x64 RPM Package	121.53 MB	 jdk-8u261-linux-x64.rpm
Linux x64 Compressed Archive	136.48 MB	 jdk-8u261-linux-x64.tar.gz
macOS x64	203.94 MB	 jdk-8u261-macosx-x64.dmg

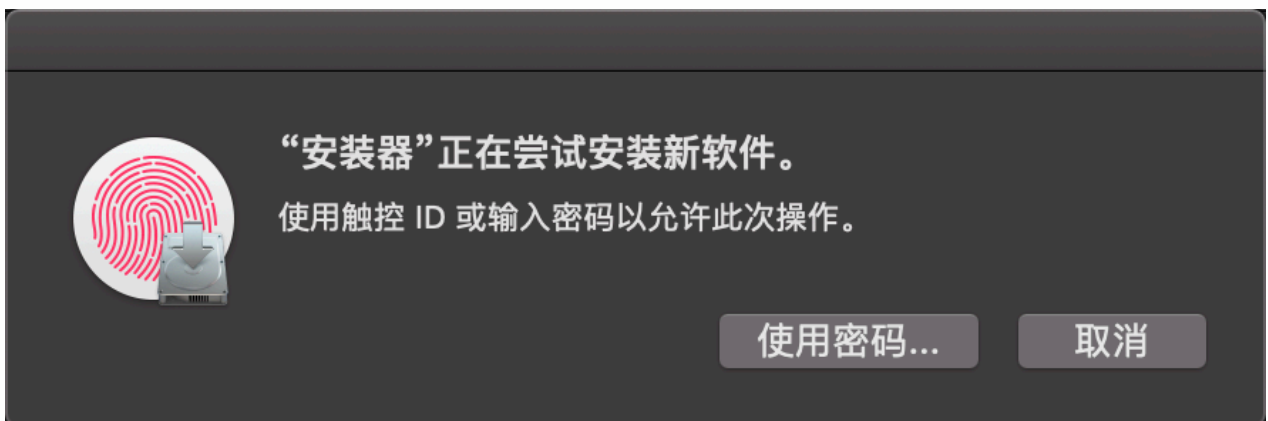
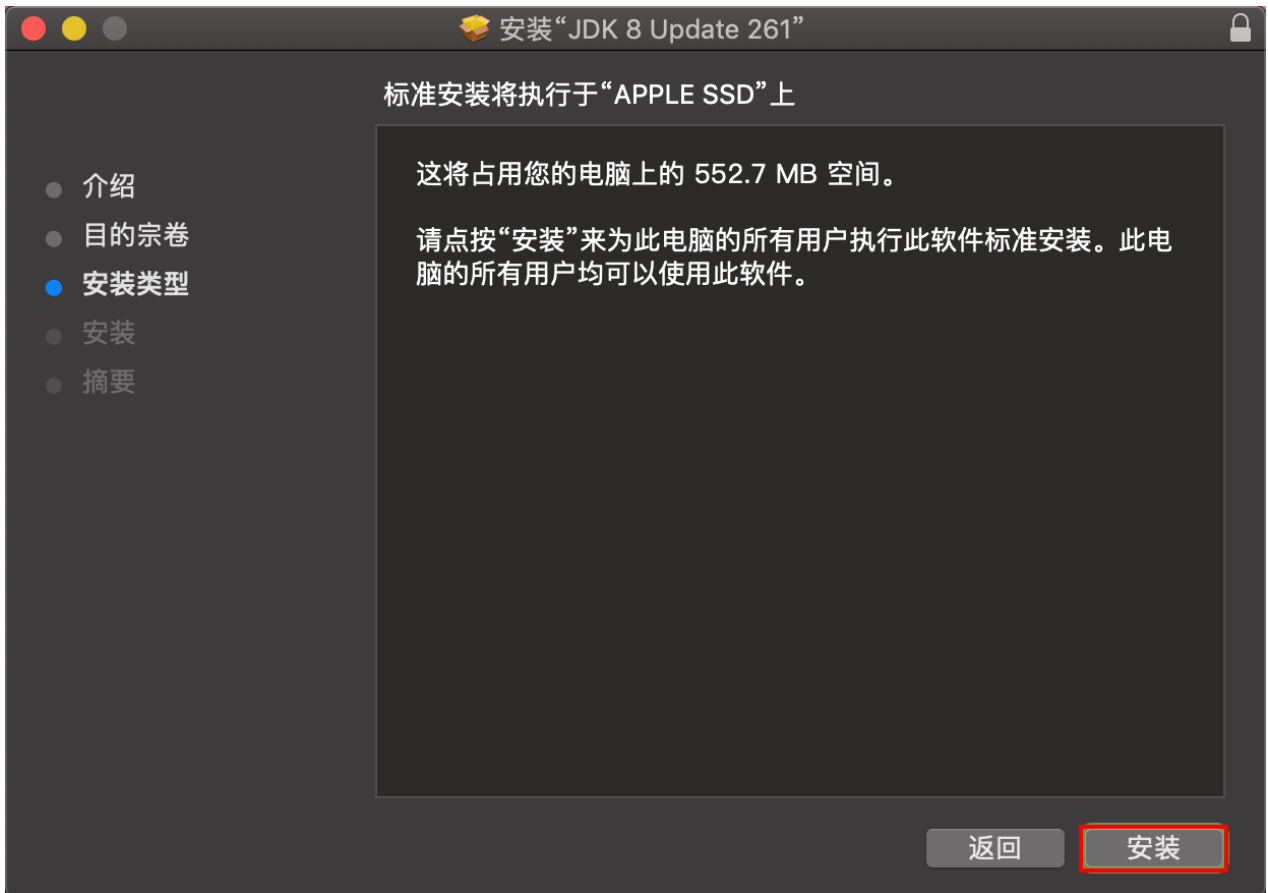
双击

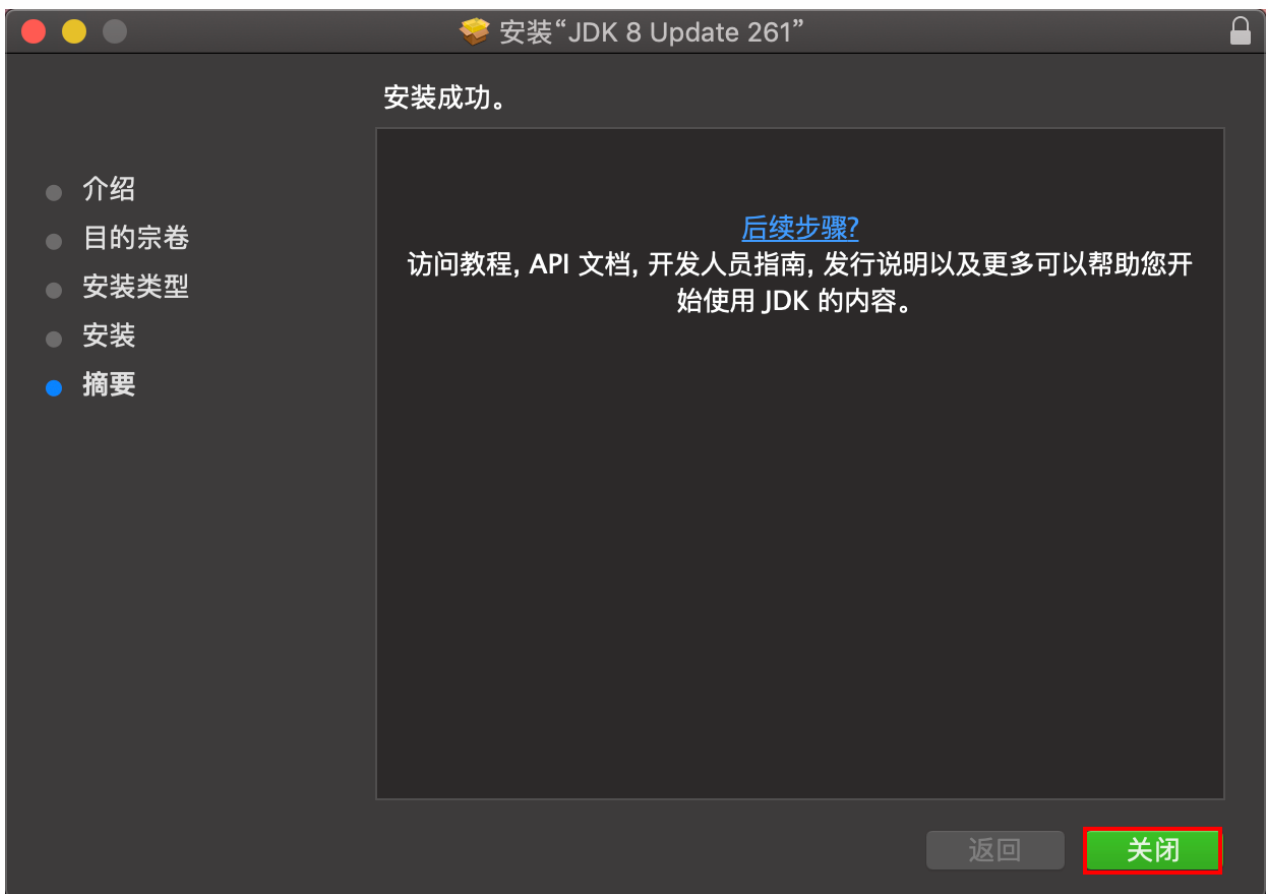
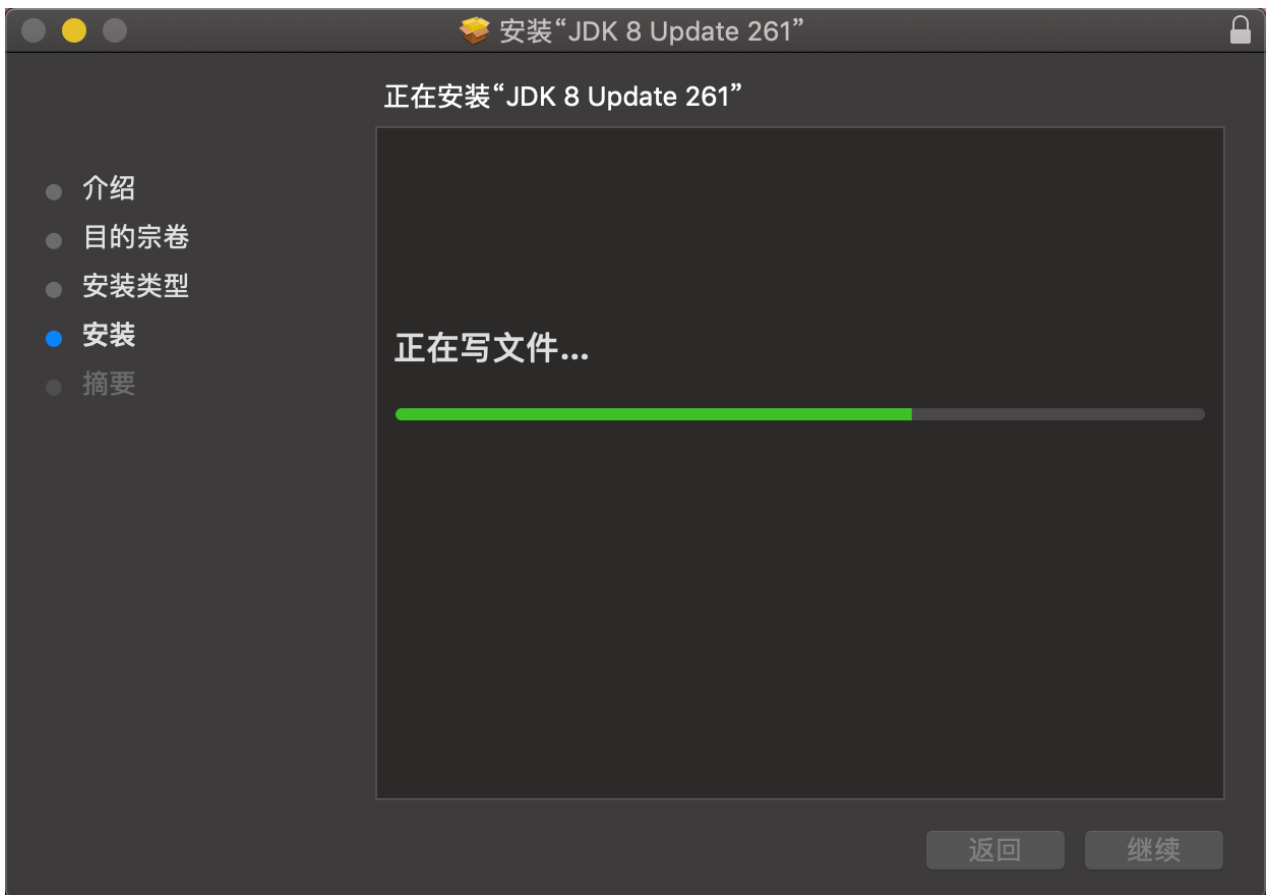


双击

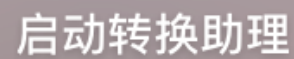


因为只有一块磁盘所以直接跳过了安装的安装的卷宗，直接到安装类型





配置环境变量:

[illegible]

```
hanlichao — vim ~/.bash_profile — 89x27
# Tell ls to be colourful
export CLICOLOR=1
export LSCOLORS=Exfxcdxbxegedabagacad

# Tell grep to highlight matches
export GREP_OPTIONS='--color=auto'
export TERM="xterm-color"
PS1='\033[012;32m\][\u@\t]\033[012;33m\]\w\[\033[012;36m\]\$\[\033[00m\] '
export PATH=/usr/local/bin:$PATH

# 添加 JDK 的环境变量
export JAVA_8_HOME=/Library/Java/JavaVirtualMachines/jdk1.8.0_261.jdk/Contents/Home

# 其中 jdk1.8.0_261.jdk 是安装好版本所对应的目录
# /Library/Java/JavaVirtualMachines/ 是 JDK 默认的安装路径

# 默认 JDK1.8
export JAVA_HOME=$JAVA_8_HOME

# alias 命令动态切换 JDK 版本
alias jdk8="export JAVA_HOME=$JAVA_8_HOME"
~
~
~
~
~
```

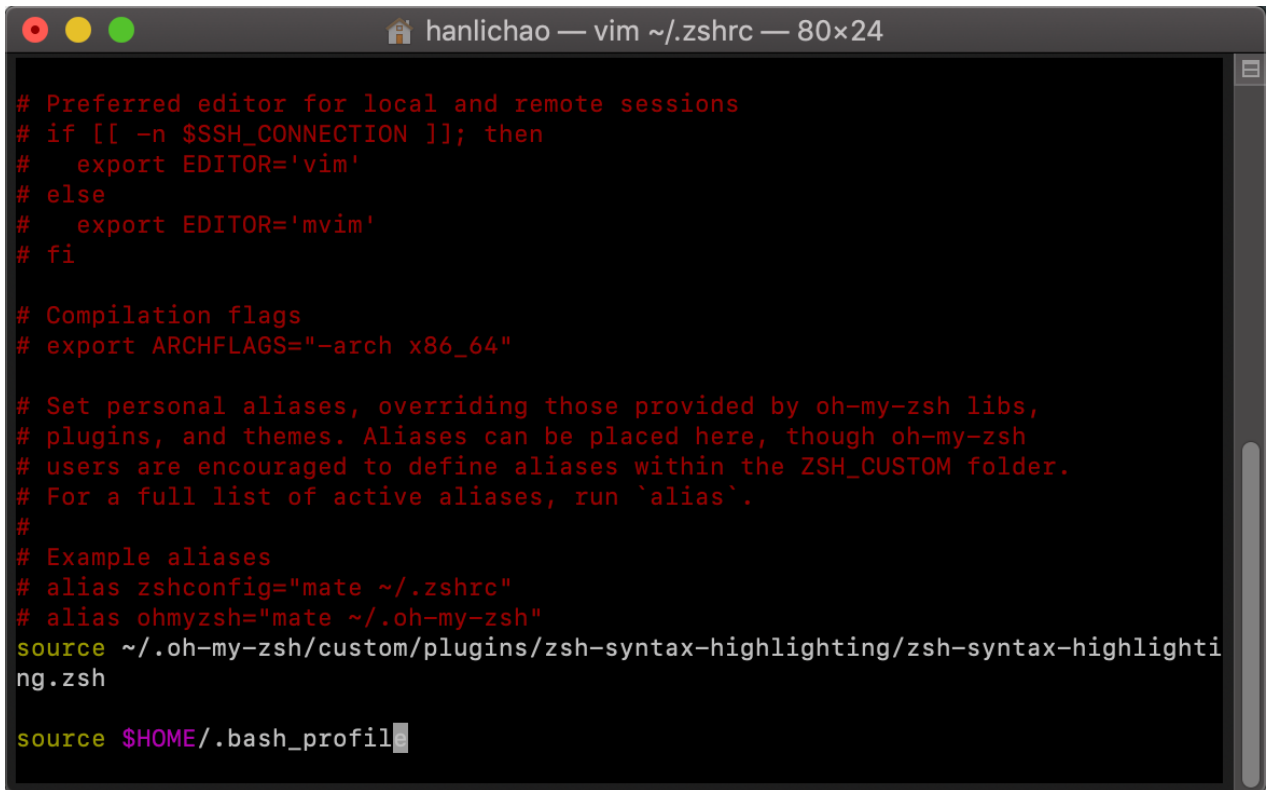
```
# 添加 JDK 的环境变量
export
JAVA_8_HOME=/Library/Java/JavaVirtualMachines/jdk1.8.0_261.jdk/Contents/Home

# 其中 jdk1.8.0_261.jdk 是安装好版本所对应的目录
# /Library/Java/JavaVirtualMachines/ 是 JDK 默认的安装路径

# 默认 JDK1.8
export JAVA_HOME=$JAVA_8_HOME

# alias 命令动态切换 JDK 版本
alias jdk8="export JAVA_HOME=$JAVA_8_HOME"
```

\$ vim ~/.zshrc



```
# Preferred editor for local and remote sessions
# if [[ -n $SSH_CONNECTION ]]; then
#   export EDITOR='vim'
# else
#   export EDITOR='mvim'
# fi

# Compilation flags
# export ARCHFLAGS="-arch x86_64"

# Set personal aliases, overriding those provided by oh-my-zsh libs,
# plugins, and themes. Aliases can be placed here, though oh-my-zsh
# users are encouraged to define aliases within the ZSH_CUSTOM folder.
# For a full list of active aliases, run `alias`.
#
# Example aliases
# alias zshconfig="mate ~/.zshrc"
# alias ohmyzsh="mate ~/.oh-my-zsh"
source ~/.oh-my-zsh/custom/plugins/zsh-syntax-highlighting/zsh-syntax-highlighting.zsh

source $HOME/.bash_profile
```

```
source $HOME/.bash_profile
```

检验是否安装成功:

```
hanlichao — -bash — 80x55
[[hanlichao@06:29:50]~$ java
用法: java [-options] class [args...]
      (执行类)
或 java [-options] -jar jarfile [args...]
      (执行 jar 文件)
其中选项包括:
-d32          使用 32 位数据模型 (如果可用)
-d64          使用 64 位数据模型 (如果可用)
-server       选择 "server" VM
              默认 VM 是 server,
              因为您是在服务器类计算机上运行。

-cp <目录和 zip/jar 文件的类搜索路径>
-classpath <目录和 zip/jar 文件的类搜索路径>
            用 : 分隔的目录, JAR 档案
            和 ZIP 档案列表, 用于搜索类文件。
-D<名称>=<值>
            设置系统属性
-verbose:[class|gc|jni]
            启用详细输出
-version      输出产品版本并退出
-version:<值>
            警告: 此功能已过时, 将在
            未来发行版中删除。
            需要指定的版本才能运行
-showversion 输出产品版本并继续
-jre-restrict-search | -no-jre-restrict-search
            警告: 此功能已过时, 将在
            未来发行版中删除。
            在版本搜索中包括/排除用户专用 JRE
-? -help     输出此帮助消息
-X           输出非标准选项的帮助
-ea[:<packagename>...|:<classname>]
-enableassertions[:<packagename>...|:<classname>]
            按指定的粒度启用断言
-da[:<packagename>...|:<classname>]
-disableassertions[:<packagename>...|:<classname>]
            禁用具有指定粒度的断言
-esa | -enablesystemassertions
            启用系统断言
-dsa | -disablesystemassertions
            禁用系统断言
-agentlib:<libname>[=<选项>]
            加载本机代理库 <libname>, 例如 -agentlib:hprof
            另请参阅 -agentlib:jdwp=help 和 -agentlib:hprof=help
-agentpath:<pathname>[=<选项>]
            按完整路径名加载本机代理库
-javaagent:<jarpath>[=<选项>]
            加载 Java 编程语言代理, 请参阅 java.lang.instrument
-splash:<imagepath>
            使用指定的图像显示启动屏幕
有关详细信息, 请参阅 http://www.oracle.com/technetwork/java/javase/documentation/index.html。
[[hanlichao@06:30:24]~$
```



```
hanlichao — -bash — 80x37
[hanlichao@06:30:24]~$ javac
用法: javac <options> <source files>
其中, 可能的选项包括:
-g 生成所有调试信息
-g:none 不生成任何调试信息
-g:{lines,vars,source} 只生成某些调试信息
-nowarn 不生成任何警告
-verbose 输出有关编译器正在执行的操作的消息
-deprecation 输出使用已过时的 API 的源位置
-classpath <路径> 指定查找用户类文件和注释处理程序的位置
-cp <路径> 指定查找用户类文件和注释处理程序的位置
-sourcepath <路径> 指定查找输入源文件的位置
-bootclasspath <路径> 覆盖引导类文件的位置
-extdirs <目录> 覆盖所安装扩展的位置
-endorseddirs <目录> 覆盖签名的标准路径的位置
-proc:{none,only} 控制是否执行注释处理和/或编译。
-processor <class1>[,<class2>,<class3>...] 要运行的注释处理程序的名称; 绕过默
认的搜索进程
-processorpath <路径> 指定查找注释处理程序的位置
-parameters 生成元数据以用于方法参数的反射
-d <目录> 指定放置生成的类文件的位置
-s <目录> 指定放置生成的源文件的位置
-h <目录> 指定放置生成的本机标头文件的位置
-implicit:{none,class} 指定是否为隐式引用文件生成类文件
-encoding <编码> 指定源文件使用的字符编码
-source <发行版> 提供与指定发行版的源兼容性
-target <发行版> 生成特定 VM 版本的类文件
-profile <配置文件> 请确保使用的 API 在指定的配置文件中可用
-version 版本信息
-help 输出标准选项的提要
-A关键字 [=值] 传递给注释处理程序的选项
-X 输出非标准选项的提要
-J<标记> 直接将 <标记> 传递给运行时系统
-Werror 出现警告时终止编译
@<文件名> 从文件读取选项和文件名

[hanlichao@06:31:30]~$
```

```
hanlichao — -bash — 80x10
[hanlichao@06:33:12]~$ java -version
java version "1.8.0_261"
Java(TM) SE Runtime Environment (build 1.8.0_261-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.261-b12, mixed mode)
[hanlichao@06:33:25]~$
```

Eclipse 下载:

<https://www.eclipse.org/downloads/download.php?file=/oomph/epp/2020-06/R/eclipse-inst-mac64.dmg>

All downloads are provided under the terms and conditions of the [Eclipse Foundation Software User Agreement](#) unless otherwise specified.

 Download

Download from: Taiwan - Computer Center, Shu-Te University (<http>)

File: [eclipse-inst-mac64.dmg](#) SHA-512

[>> Select Another Mirror](#)

OR Get It Faster from our Members