SDKMAN! JVM SDK manager

CONCEPTS

SDKMAN! keeps parallel versions of JVM SDKs, called candidates (java, maven, gradle, kotlin, ...). sdk is a shell function loaded by sdkman-init.sh.

Candidate java, maven, ...
↓ install
Installed version candidates/<c>/<v>
↓ default
current link new shells
↓ use / env
This shell PATH and *_HOME

sdk use affects only this shell; sdk default moves the current symlink that new shells pick up. JAVA_HOME follows the active java.

FIND AND INSTALL

`sdk list`
Browse all candidates

`sdk list <sdk>`
List versions of one candidate

`sdk install <sdk>`
Install the default version

`sdk install <sdk> <version>`
Install a specific version

`sdk uninstall <sdk> <version>`
Remove an installed version

`sdk update`
Refresh the candidate list

SWITCH VERSIONS

`sdk use <sdk> <version>`
Use a version in this shell

`sdk default <sdk> <version>`
Set default for new shells

`sdk current`
Show versions in use

`sdk current <sdk>`
Show the version of one candidate

SWITCH VERSIONS (cont.)

`sdk home <sdk> <version>`
Print the install path

`sdk upgrade`
List outdated candidates

JAVA VENDORS

`sdk list java`
Vendor table with identifiers

`sdk install java <version>-tem`
Install an Eclipse Temurin JDK

Identifier suffixes

Suffix	Distribution
`-tem`	Eclipse Temurin
`-amzn`	Amazon Corretto
`-zulu`	Azul Zulu
`-librca`	BellSoft Liberica
`-graalce`	GraalVM CE
`-graal`	Oracle GraalVM
`-oracle`	Oracle JDK
`-ms`	Microsoft OpenJDK

PROJECT ENV (.SDKMANRC)

`sdk env init`
Create .sdkmanrc with current java

`sdk env`
Switch to versions in .sdkmanrc

`sdk env install`
Install missing versions and switch

MAINTENANCE

`sdk version`
Print script and native versions

`sdk help [<subcommand>]`
Show help, or help for one command

`sdk selfupdate`
Update SDKMAN! itself

CONFIGURATION

`sdk config`
Edit ~/.sdkman/etc/config