



Install Java JDK 21 or OpenJDK 21 on Debian 12/11/10

By [Josphat Mutai](#) - December 1, 2023

Java is an open source, and high level programming language known to be reliable, robust and portable. It was originally created by Sun Microsystems in min-1990s but is now being owned and maintained by Oracle Corporation. Java is one of the widely adopted programming languages in banking, governments, schools and powering all types of applications worldwide.



Java Development Kit (JDK) can be described as a set of tools and libraries created by Oracle and other community contributors to enable you quickly develop Java applications. JDK includes Java Runtime Environment (JRE), a core engine for running Java applications, and extras such as debugger, compiler, and utilities that will increase your efficiency when programming.

Every 3 years, Oracle will release an LTS version of Java programming language. The Long-Term Support (LTS) versions of Java get extended updates, bug fixes, security updates, and support from Oracle as opposed to the short term non-LTS releases. When using Java to develop mission critical applications in an enterprise and business setup, an LTS is very crucial for SLA and general stability.



For those new to **OpenJDK**, it is a community driven and open-source implementation of Java Platform, Standard Edition (Java SE) specification. Being released under the open-source license means developers can modify and easily

contribute to its source code. Most of the contributors are individuals and other private companies.



Installing JDK 21 or OpenJDK 21 on Debian 12/11/10

In this installation, we will consider use of OpenJDK and the official Oracle JDK. When choosing between the two, factors like support, licensing are important. OpenJDK is a common choice since it's open-source and ticks all the boxes in Java SE specification.

1. Install Oracle JDK 21 on Debian

The Oracle JDK is not fully open source implementation of Java SE specification. It is packaged with additional tools and features that are not available in the standard Java APIs. If your application needs support from Oracle on updates and security patches in a timely manner, then this is a better option for you.



Oracle JDK 21 can be downloaded from [Oracle Downloads](#) pages. If you're a CLI user, this can be downloaded using `wget` or `curl`.



```
wget https://download.oracle.com/java/21/latest/jdk-21_linux-x64_bin.deb
```

Once the package is downloaded, perform install with the `apt` or `dpkg` package management tool available for Debian users.

```
$ sudo dpkg -i jdk-21_linux-x64_bin.deb
Selecting previously unselected package jdk-21.
(Reading database ... 34139 files and directories currently
installed.)
Preparing to unpack jdk-21_linux-x64_bin.deb ...
Unpacking jdk-21 (21.0.1-ga) ...
Setting up jdk-21 (21.0.1-ga) ...
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jar to
provide /usr/bin/jar (jar) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-
x64/bin/jarsigner to provide /usr/bin/jarsigner (jarsigner) in auto
mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/java to
provide /usr/bin/java (java) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/javac
to provide /usr/bin/javac (javac) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/javadoc
to provide /usr/bin/javadoc (javadoc) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/javap
to provide /usr/bin/javap (javap) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jcmm
to provide /usr/bin/jcmm (jcmm) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-
x64/bin/jconsole to provide /usr/bin/jconsole (jconsole) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jdb to
provide /usr/bin/jdb (jdb) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-
x64/bin/jdeprscan to provide /usr/bin/jdeprscan (jdeprscan) in auto
mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jdeps
to provide /usr/bin/jdeps (jdeps) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jfr to
```





provide /usr/bin/jfr (jfr) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jhsdb
to provide /usr/bin/jhsdb (jhsdb) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jimage
to provide /usr/bin/jimage (jimage) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jinfo
to provide /usr/bin/jinfo (jinfo) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jlink
to provide /usr/bin/jlink (jlink) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jmap to
provide /usr/bin/jmap (jmap) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jmod to
provide /usr/bin/jmod (jmod) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-
x64/bin/jpackage to provide /usr/bin/jpackage (jpackage) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jps to
provide /usr/bin/jps (jps) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-
x64/bin/jrunscript to provide /usr/bin/jrunscript (jrunscript) in
auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jshell
to provide /usr/bin/jshell (jshell) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jstack
to provide /usr/bin/jstack (jstack) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jstat
to provide /usr/bin/jstat (jstat) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/jstatd
to provide /usr/bin/jstatd (jstatd) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-
x64/bin/jwebserver to provide /usr/bin/jwebserver (jwebserver) in
auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/bin/keytool
to provide /usr/bin/keytool (keytool) in auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-
x64/bin/rmiregistry to provide /usr/bin/rmiregistry (rmiregistry) in
auto mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-
x64/bin/serialver to provide /usr/bin/serialver (serialver) in auto
mode
update-alternatives: using /usr/lib/jvm/jdk-21-oracle-x64/lib/jexec
to provide /usr/bin/jexec (jexec) in auto mode

Your JAVA HOME should be ***/usr/lib/jvm/jdk-21-oracle-x64/***



Check current default version of Java. If no old version it should show the one we just installed.

```
$ java --version
java 21.0.1 2023-10-17 LTS
Java(TM) SE Runtime Environment (build 21.0.1+12-LTS-29)
Java HotSpot(TM) 64-Bit Server VM (build 21.0.1+12-LTS-29, mixed
mode, sharing)
```

The `update-alternatives` command is used to update the default version of Java in systems with multiple versions.

To list all installed versions of Java run:





```
$ sudo update-alternatives --list java
/usr/lib/jvm/java-17-openjdk-amd64/bin/java
/usr/lib/jvm/jdk-21-oracle-x64/bin/java
```

Example on setting to Java 21:

```
sudo update-java-alternatives --set /usr/lib/jvm/jdk-21-oracle-x64
```

2. Install OpenJDK 21 on Debian

This is a manual process which involves archive download, extracting and copying the files to a system's directory.

Go to [OpenJDK 21 release](#) website and download a version suitable for your CPU architecture.



```
wget
https://download.java.net/java/GA/jdk21.0.2/f2283984656d49d69e91c5584
76027ac/13/GPL/openjdk-21.0.2_linux-x64_bin.tar.gz
```

We can extract the file downloaded to get all files and directories containing OpenJDK.

```
⚙️ tar xvf openjdk-21.0.2_linux-x64_bin.tar.gz
```

Move created folder into `/usr/local/` directory.

```
sudo mv jdk-21.0.2/ /usr/local/jdk-21
```



To set Java environment, create a new source script.



```
sudo tee -a /etc/profile.d/jdk21.sh<<EOF
export JAVA_HOME=/usr/local/jdk-21
export PATH=\$PATH:\$JAVA_HOME/bin
EOF
```

Source to effect the change in your current shell session.

```
source /etc/profile.d/jdk21.sh
```

Check Java version to verify it works.

```
$ java --version
openjdk 21.0.2 2024-01-16
OpenJDK Runtime Environment (build 21.0.2+13-58)
OpenJDK 64-Bit Server VM (build 21.0.2+13-58, mixed mode, sharing)
```

Choosing between OpenJDK and Oracle JDK has a high dependency on your application needs and long term company plans. In an ideal general use case, OpenJDK is an easy choice being 100% open source. But with Oracle JDK you can still use the free parts on small projects without the need for licensing.





Until our paths intertwine again in content distribution channel, take care!.

YOU CAN SUPPORT OUR WORK WITH A CUP OF COFFEE

As we continue to grow, we would wish to reach and impact more people who visit and take advantage of the guides we have on our blog. This is a big task for us and we are so far extremely grateful for the kind people who have shown amazing support for our work over the time we have been online.

Thank You for your support as we work to give you the best of guides and articles. Click below to buy us a coffee.



Josphat Mutai

<https://computingforgeeks.com/>

Founder of Computingforgeeks. Expertise in Virtualization, Cloud, Linux/UNIX Administration, Automation, Storage Systems, Containers, Server Clustering e.t.c.

in

