

# How To Install Java with Apt-Get on Ubuntu 16.04

Posted April 23, 2016

653k

JAVA

UBUNTU

UBUNTU 16.04

## Introduction

Java and the JVM (Java's virtual machine) are widely used and required for many kinds of software. This article will guide you through the process of installing and managing different versions of Java using `apt-get`.

## Prerequisites

To follow this tutorial, you will need:

- One Ubuntu 16.04 server.
- A sudo non-root user, which you can set up by following [the Ubuntu 16.04 initial server setup guide](#).

## Installing the Default JRE/JDK

The easiest option for installing Java is using the version packaged with Ubuntu. Specifically, this will install OpenJDK 8, the latest and recommended version.

First, update the package index.

- `sudo apt-get update`

Next, install Java. Specifically, this command will install the Java Runtime Environment (JRE).

- `sudo apt-get install default-jre`

There is another default Java installation called the JDK (Java Development Kit). The JDK is usually only needed if you are going to compile Java programs or if the software that will use Java specifically requires it.

The JDK does contain the JRE, so there are no disadvantages if you install the JDK instead of the JRE, except for the larger file size.

You can install the JDK with the following command:

- `sudo apt-get install default-jdk`

## Installing the Oracle JDK

If you want to install the Oracle JDK, which is the official version distributed by Oracle, you will need to follow a few more steps. If you need Java 6 or 7, which are not available in the default Ubuntu 16.04 repositories (not recommended), this installation method is also available.

First, add Oracle's PPA, then update your package repository.

- `sudo add-apt-repository ppa:webupd8team/java`
- `sudo apt-get update`

Then, depending on the version you want to install, execute one of the following commands:

### Oracle JDK 6 or 7

These are very old versions of Java which reached end of life in February 2013 and April 2015 respectively. It's not recommended to use them, but they might still be required for some programs.

To install JDK 6, use the following command:

- `sudo apt-get install oracle-java6-installer`

To install JDK 7, use the following command:

- `sudo apt-get install oracle-java7-installer`

## Oracle JDK 8

This is the latest stable version of Java at time of writing, and the recommended version to install. You can do so using the following command:

- `sudo apt-get install oracle-java8-installer`

## Oracle JDK 9

This is a developer preview and the general release is scheduled for March 2017. It's not recommended that you use this version because there may still be security issues and bugs. There is more information about Java 9 on the [official JDK 9 website](#).

To install JDK 9, use the following command:

- `sudo apt-get install oracle-java9-installer`

## Managing Java

There can be multiple Java installations on one server. You can configure which version is the default for use in the command line by using `update-alternatives`, which manages which symbolic links are used for different commands.

- `sudo update-alternatives --config java`

The output will look something like the following. In this case, this is what the output will look like with all Java versions mentioned above installed.

Output

There are 5 choices for the alternative java (providing /usr/bin/java).

Selection	Path	Priority	Status
* 0	/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java	1081	auto mode
1	/usr/lib/jvm/java-6-oracle/jre/bin/java	1	manual mode
2	/usr/lib/jvm/java-7-oracle/jre/bin/java	2	manual mode
3	/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java	1081	manual mode

```
4      /usr/lib/jvm/java-8-oracle/jre/bin/java      3      manual mode
5      /usr/lib/jvm/java-9-oracle/bin/java        4      manual mode
```

Press <enter> to keep the current choice[\*], or type selection number:

You can now choose the number to use as a default. This can also be done for other Java commands, such as the compiler ( `javac` ), the documentation generator ( `javadoc` ), the JAR signing tool ( `jarsigner` ), and more. You can use the following command, filling in the command you want to customize.

- `sudo update-alternatives --config command`

## Setting the JAVA\_HOME Environment Variable

Many programs, such as Java servers, use the `JAVA_HOME` environment variable to determine the Java installation location. To set this environment variable, we will first need to find out where Java is installed. You can do this by executing the same command as in the previous section:

- `sudo update-alternatives --config java`

Copy the path from your preferred installation and then open `/etc/environment` using `nano` or your favorite text editor.

- `sudo nano /etc/environment`

At the end of this file, add the following line, making sure to replace the highlighted path with your own copied path.

```
/etc/environment
```

```
JAVA_HOME="/usr/lib/jvm/java-8-oracle"
```

Save and exit the file, and reload it.

- `source /etc/environment`

You can now test whether the environment variable has been set by executing the following command:

- `echo $JAVA_HOME`

This will return the path you just set.

## Conclusion

You have now installed Java and know how to manage different versions of it. You can now install software which runs on Java, such as Tomcat, Jetty, Glassfish, Cassandra, or Jenkins.