

# Grouper Book - Installing the quickstart - Linux

[Wiki Home](#)[Download Grouper](#)[Grouper Guides](#)[Community Contributions](#)[Developer Resources](#)[Grouper Website](#)

Note: As of January 2012, the Quickstart has been replaced by [an easier installation method](#)

## Section contents

- [Installing the quickstart - Linux](#)
- [Installing a JDK](#)
- [Installing the RPM version of the JDK](#)
- [Installing the self-extracting version of the JDK](#)
- [Installing Tomcat](#)
- [Installing Apache Ant](#)
- [Setting up the quick start](#)
- [Starting the quickstart](#)

## Installing the quickstart - Linux

Before you start on the quickstart, you'll need to have a java standard edition development kit (Java SE JDK) installed. A JRE alone is not enough as the quickstart runs inside Tomcat, which requires a JDK. At this stage I'd recommend a Sun/Oracle JDK rather than another flavour (such as IDM or OpenJDK). The reason for this is that all the Grouper binary releases are built using the Sun/Oracle JDK. The quickstart also requires that Apache Ant be installed.

### Installing a JDK

If your distribution has a packaged version of the Sun/Oracle JDK then you can install it. For example, on Ubuntu I can install it with

```
sudo apt-get install sun-java6-jdk
```

If your distribution does not have a packaged version, or you prefer to download it yourself, get a recent version of the JDK (Java 6 is required for Grouper) from <http://www.oracle.com/technetwork/java/javase/downloads/index.html> - make sure you download the JDK. This is available in 64 bit and 32 bit versions (choose whichever suits your operating system, Grouper runs on both), and packaged as RPM or self-extracting. For distributions which use RPMs - including RedHat, SUSE and Centos - you can use either the RPM or the self-extracting. If your distribution doesn't support RPM - including Debian and Ubuntu - download the self-extracting file.

### Installing the RPM version of the JDK

1. Open a command shell and cd into the directory into which you saved the downloaded file
2. su to become root (if you are not already root)
3. Make the file executable by running

```
chmod a+x jdk-6u<version>-linux-<architecture>.rpm.bin
```

4. Execute the file by running

```
./jdk-6u<version>-linux-<architecture>.rpm.bin
```

this runs a script that displays a license and extracts the RPM file after you've accepted the license terms. The RPM is then installed and standard java executables are added to the system path. You can find out where the JDK had been installed by looking at where `/usr/java/latest` links to.

5. Check that Java is installed and in your path by entering the following command

```
java -version
```

and you should see an output giving you details of the version of java you have installed. If you get an error, then the java executable is not in your path and you will need to consult the documentation for the java package you installed.

### Installing the self-extracting version of the JDK

1. Open a command shell and cd into the directory into which you saved the downloaded file
2. Make the file executable by running

```
chmod a+x jdk-6u<version>-linux-<architecture>.bin
```

3. Execute the file by running

```
./jdk-6u<version>-linux-<architecture>.bin
```

this runs a script that displays a license and extracts the JDK file after you've accepted the license terms.

4. You will have a new directory in the same location as the file you just executed called `jdk1.6.0_<version>` which contains all of java. The java executables are in the `jdk1.6.0_<version>/bin` folder and you may want to add this to your path by editing `~/.profile` (or similar). I usually move the extracted java to `/usr/local/java` with

```
sudo mv jdk1.6.0_<version> /usr/local/java
```

and add the bin directory to my path by adding the following to the end of my `.profile` file in my home directory

```
PATH="$PATH:/usr/local/java/jdk1.6.0<version>/bin/ "
```

. If you do this you'll need to log out and back in to load the new path. Alternatively, you can temporarily add it to your path in the current console with

```
export PATH=$PATH:/usr/local/java/jdk1.6.0<version>/bin/
```

I'll assume the java executables are in your system path from now on, so if it isn't you'll need to use the full path to the whichever executable you are using (which may involve editing some scripts)

5. Check that Java is installed and in your path by entering the following command

```
java -version
```

and you should see an output giving you details of the version of java you have installed. If you get an error, then the java executable is not in your path - go back to step 5 to check you've used one of the methods for adding it to your path.

## Installing Tomcat

1. Download a copy of Tomcat stable from <http://tomcat.apache.org/download-60.cgi> (this link is to Tomcat 6, the current stable version). Choose the `.tar.gz` download (`apache-tomcat-6.0.29.tar.gz` at the time of writing) and save it to a directory
2. `cd` to the directory in which you saved the download and extract it with

```
tar -xzf apache-tomcat-<version>.tar.gz
```

3. You will have a new directory called `apache-tomcat-<version>` within which there is a `bin` directory containing executable scripts for starting tomcat. You can move the whole `apache-tomcat-<version>` directory to another location (I usually put it in `/usr/local/`). From now on I'll call the root directory where Tomcat is located `TOMCAT_HOME`
4. To start Tomcat you need to define a variable `JAVA_HOME`, or have java in your system path. If you've followed the steps so far you should have java in your path. Alternatively you can define `JAVA_HOME` as a variable with

```
export JAVA_HOME=<path to root of JDK>
```

5. `cd` into the `apache-tomcat-<version>/bin` directory and start Tomcat with

```
./startup.sh
```

6. Test that Tomcat is running by opening a browser and visiting <http://localhost:8080> you should see a Tomcat welcome page. If you don't see this, but something else instead, then another service is listening on port 8080, you'll need either find out what it is and stop it, or change the port that Tomcat listens on by editing the `TOMCAT_HOME/conf/server.xml` file

## Installing Apache Ant

You will also need to install Apache Ant. This is packaged for most distributions and you should be able to install using the package manager. On Ubuntu I can install it with `sudo apt-get install ant1.8`. You can also download it from <http://ant.apache.org/bindownload.cgi> (choose the `.tar.gz` file, from the Binary Distributions, Core list) and save it.

1. Open a command shell and `cd` into the directory into which you saved the downloaded file
2. Extract the contents of the downloaded file with

```
tar xzvf apache-ant-<version>-bin.tar.gz
```

3. You will have a new directory called apache-ant-<version> within which there is a bin directory containing an executable for called ant
4. Add ant to your path (either permanently using the same method as we used to add the bin directory in the JDK to our path), or temporarily in the current console with

```
export PATH=$PATH:<path to ant executable>
```

## Setting up the quick start

Once we have java and ant set up and in our path, very little effort is needed to install the quickstart. You simply need to extract it and run a script.

1. Open a command line and CD into the directory into which you downloaded the grouper-quickstart-1.6.1.tar.gz file
2. Extract the contents of the file with

```
tar xzvf grouper.quickstart-1.6.1.tar.gz
```

You will now have a new directory in the same directory as where you were when you extracted the file. It will be called grouper.quickstart-1.6.1

The first task is to build the web application and install it into Tomcat

1. In the directory grouper.quickstart-1.6.1 run the following command

```
ant ui
```

2. You will then be presented with a list of options, type default and press enter,. The UI application will now be built
3. Before installing it into Tomcat, you need to copy a file to you Tomcat installation (or merge it if you have made other changes to the file for other reasons). You need to copy the tomcat-user.xml file from grouper.quickstart-1.6.1/grouper-ui to TOMCAT\_HOME/conf where it will overwrite another file

```
cp grouper.quickstart-1.6.1/grouper-ui/tomcat-users.xml <TOMCAT_HOME>/conf
```

4. Install it into Tomcat by copying the grouper.quickstart-1.6.1/grouper-ui/build/grouper directory into TOMCAT\_HOME/webapps

```
cp -r grouper.quickstart-1.6.1/grouper-ui/build/grouper <TOMCAT_HOME>/webapps
```

## Starting the quickstart

1. Start the HSQL database that comes with Grouper. To do this
  - a. cd into the grouper1.6.1/grouper.quickstart-1.6.1 directory
  - b. enter the command

```
ant db
```

2. Restart tomcat in another command window with:

```
cd apache-tomcat-<version>/bin  
./shutdown.sh  
./startup.sh
```

Now open a browser and go to <http://localhost:8080/grouper> you will see the login screen and can log in using the username GrouperSystem and the password 123 (these are contained in the tomcat-users.xml file). You should now have a working copy of Grouper.