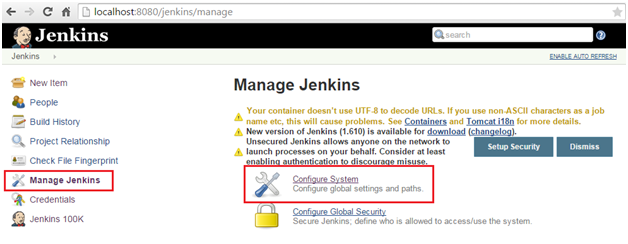
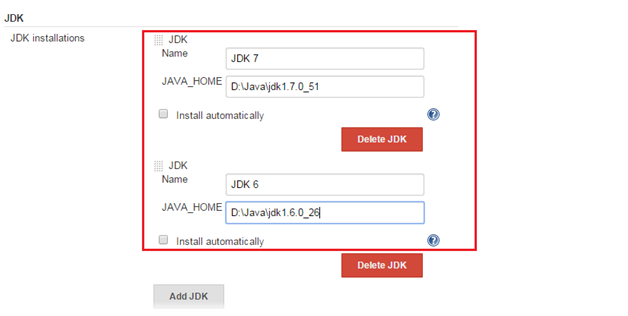
#### Configuring JDK

Below are the steps to be followed in order to configure the JDK in Jenkins :

1. Deploy the Jenkins.war into the tomcat server and start the server.
2. From the home page, click on the link **Manage Jenkins** and click on **Configure System** as shown below :

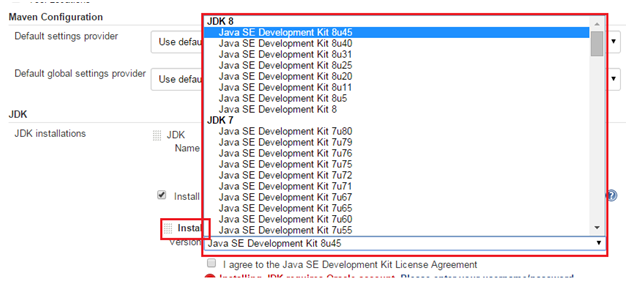


1. From the next page, click on "Add JDK". The easiest way to configure JDK is to provide a name and path of the java installation directory. (Consider the same path used for JAVA\_HOME). In the below screenshot JDK 6 and JDK 7 has been configured,



**NOTE :** The Jenkins will validate to check whether the installation directory exists for the specified JDK versions.

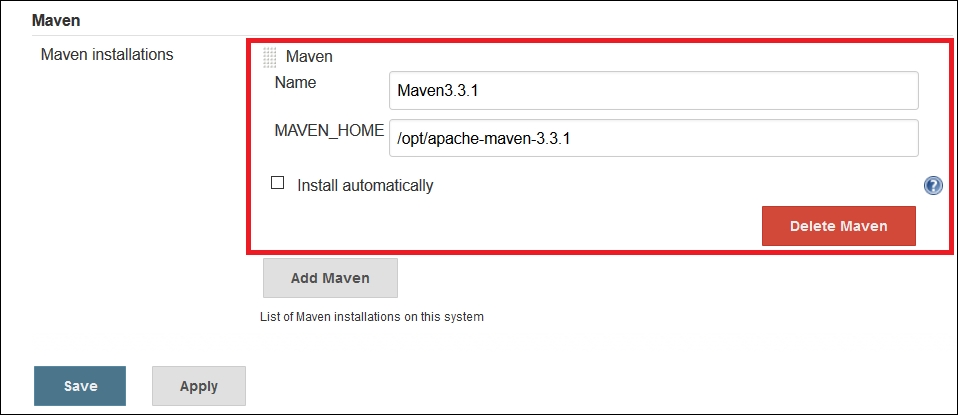
1. JDK can also be installed by intimating Jenkins to do the task. The check box **install automatically** needs to be checked in this case. Jenkins will download and installs the specified version of the JDK into the tools directory of the Jenkins home directory.



1. After all the settings made to configure the JDK, just click the "Save" at the bottom of the page to save the configuration.

### **Configuring Maven**

To configure Maven, download the Maven installer from <https://maven.apache.org/download.cgi> , and extract it to the directory on your Jenkins virtual machine. In the **Global Tool Configuration** section, provide the **Name** and **MAVEN\_HOME** path, or check **Install automatically** checkbox:



That's it! Our major configuration for running a simple build is done. Now, let's go to the home page of the Jenkins dashboard to create and configure a build job.

Open the Jenkins dashboard.

1. Go to **Manage Jenkins**.
2. Go to **Global Tool Configuration** to configure the tools, their locations, and automatic installers.
3. Go to the **Git** section.
4. Give the **Name** and click on **Install Automatically**, or provide a path to Git.
5. You can add multiple Gits based on the version or for the specific agent. You need to give a meaningful name so it can be identified easily while configuring the build job:

