



jenkins

tutorialspoint

SIMPLY EASY LEARNING

www.tutorialspoint.com



<https://www.facebook.com/tutorialspointindia>



<https://twitter.com/tutorialspoint>

About the Tutorial

Jenkins is a powerful application that allows continuous integration and continuous delivery of projects, regardless of the platform you are working on. It is a free source that can handle any kind of build or continuous integration. You can integrate Jenkins with a number of testing and deployment technologies. In this tutorial, we would explain how you can use Jenkins to build and test your software projects continuously.

Audience

This tutorial is going to help all those software testers who would like to learn how to build and test their projects continuously in order to help the developers to integrate the changes to the project as quickly as possible and obtain fresh builds.

Prerequisites

Jenkins is a popular tool for performing continuous integration of software projects. This is a preliminary tutorial that covers the most fundamental concepts of Jenkins. Any software professional having a good understanding of Software Development Life Cycle should benefit from this tutorial.

Disclaimer & Copyright

© Copyright 2018 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute, or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness, or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

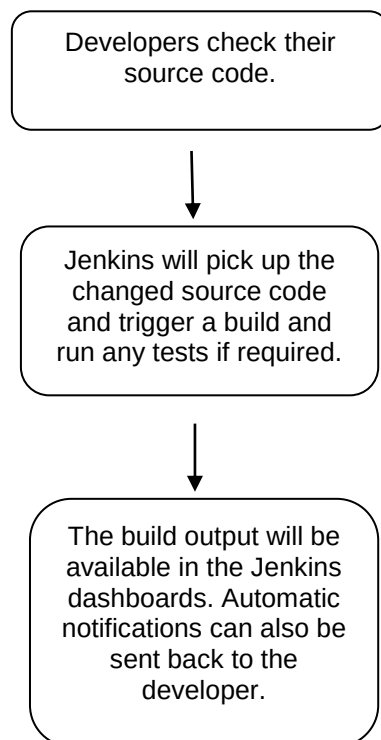
About the Tutorial	i
Audience.....	i
Prerequisites.....	i
Disclaimer & Copyright	i
Table of Contents.....	ii
 1. JENKINS – OVERVIEW.....	 1
Why Jenkins?	1
What is Continuous Integration?	1
System Requirements	2
 2. JENKINS – INSTALLATION	 3
Download Jenkins	3
Starting Jenkins.....	4
Accessing Jenkins	5
 3. JENKINS – TOMCAT SETUP	 6
 4. JENKINS – GIT SETUP.....	 10
 5. JENKINS – MAVEN SETUP	 16
 6. JENKINS – CONFIGURATION.....	 23
 7. JENKINS – MANAGEMENT.....	 27
Configure System	28
Reload Configuration from Disk	28
Manage Plugins	29
System Information	29
 8. JENKINS – SETUP BUILD JOBS.....	 32

9. JENKINS – UNIT TESTING	43
Example of a Junit Test in Jenkins	44
10. JENKINS – AUTOMATED TESTING.....	53
11. JENKINS – NOTIFICATION	60
12. JENKINS – REPORTING	63
13. JENKINS – CODE ANALYSIS	64
14. JENKINS – DISTRIBUTED BUILDS.....	65
15. JENKINS – AUTOMATED DEPLOYMENT	70
16. JENKINS – METRICS AND TRENDS	73
17. JENKINS – SERVER MAINTENANCE	87
URL Options	87
Backup Jenkins Home	87
18. JENKINS – CONTINUOUS DEPLOYMENT	89
19. JENKINS – MANAGING PLUGINS.....	102
Uninstalling Plugins	102
Installing another Version of a Plugin	103
20. JENKINS – SECURITY.....	104
21. JENKINS – BACKUP PLUGIN	109
22. JENKINS – REMOTE TESTING.....	117

1. Jenkins – Overview

Why Jenkins?

Jenkins is a software that allows **continuous integration**. Jenkins will be installed on a server where the central build will take place. The following flowchart demonstrates a very simple workflow of how Jenkins works.



Along with Jenkins, sometimes, one might also see the association of **Hudson**. Hudson is a very popular open-source Java-based continuous integration tool developed by Sun Microsystems which was later acquired by Oracle. After the acquisition of Sun by Oracle, a fork was created from the Hudson source code, which brought about the introduction of Jenkins.

What is Continuous Integration?

Continuous Integration is a development practice that requires developers to integrate code into a shared repository at regular intervals. This concept was meant to remove the problem of finding later occurrence of issues in the build lifecycle. Continuous integration requires the developers to have frequent builds. The common practice is that whenever a code commit occurs, a build should be triggered.

System Requirements

JDK	JDK 1.5 or above
Memory	2 GB RAM (recommended)
Disk Space	No minimum requirement. Note that since all builds will be stored on the Jenkins machines, it has to be ensured that sufficient disk space is available for build storage.
Operating System Version	Jenkins can be installed on Windows, Ubuntu/Debian, Red Hat/Fedora/CentOS, Mac OS X, openSUSE, FreeBSD, OpenBSD, Gentoo.
Java Container	The WAR file can be run in any container that supports Servlet 2.4/JSP 2.0 or later.(An example is Tomcat 5).

2. Jenkins – Installation

Download Jenkins

The official website for Jenkins is <https://jenkins-ci.org/>. If you click the given link, you can get the home page of the Jenkins official website as shown below.



By default, the latest release and the Long-Term support release will be available for download. The past releases are also available for download. Click the Long-Term Support Release tab in the download section.



Click the link "Older but stable version" to download the Jenkins war file.

Starting Jenkins

Open the command prompt. From the command prompt, browse to the directory where the jenkins.war file is present. Run the following command

```
D:\>Java -jar Jenkins.war
```

After the command is run, various tasks will run, one of which is the extraction of the war file which is done by an embedded webserver called winstone.

```
D:\>Java -jar Jenkins.war
Running from: D:\jenkins.war
Webroot: $user.home/ .jenkins
Sep 29, 2015 4:10:46 PM winstone.Logger logInternal
INFO: Beginning extraction from war file
```

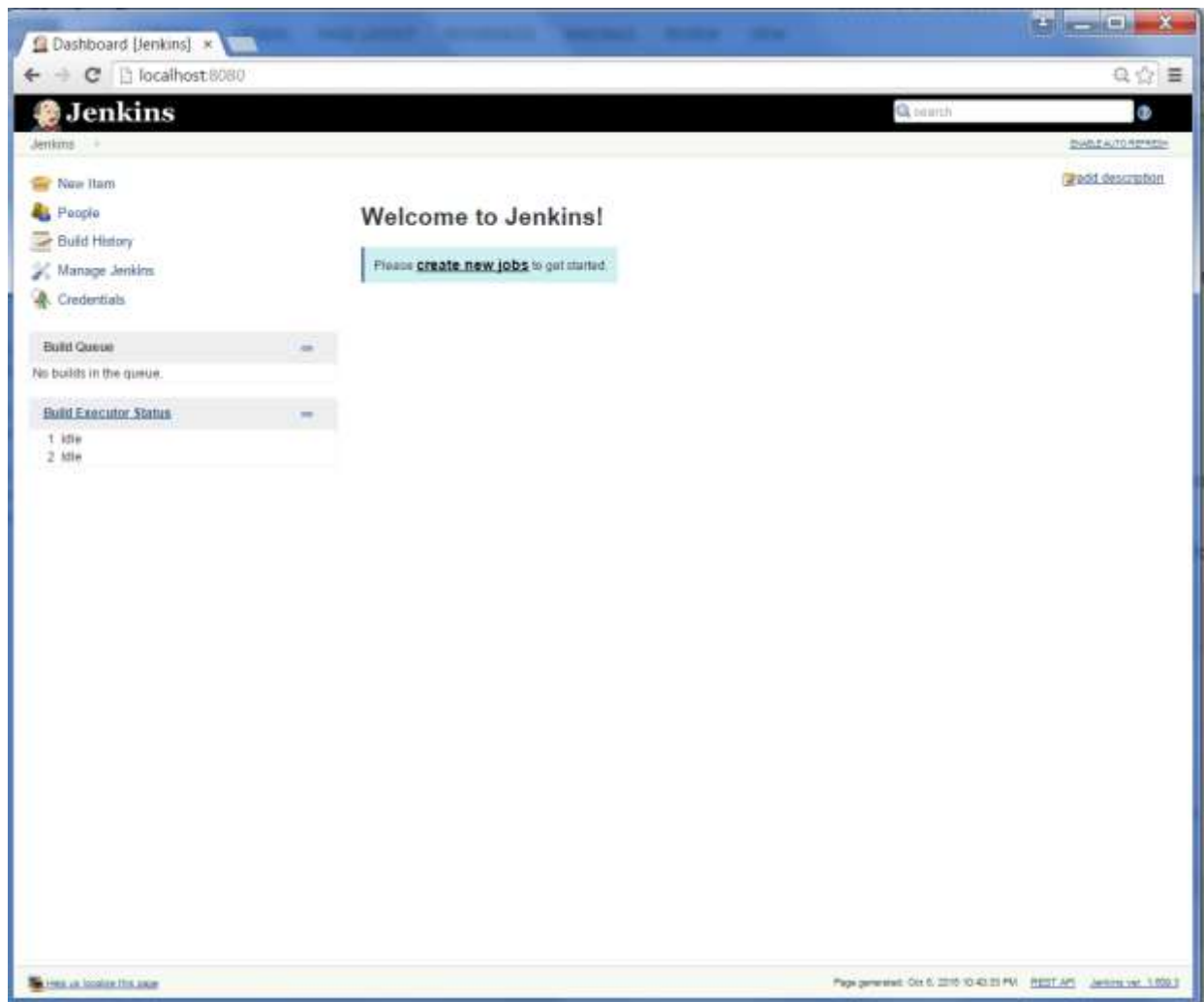
Once the processing is complete without major errors, the following line will come in the output of the command prompt.

INFO: Jenkins is fully up and running

Accessing Jenkins

Once Jenkins is up and running, one can access Jenkins from the link – <http://localhost:8080>

This link will bring up the Jenkins dashboard.



3. Jenkins – Tomcat Setup

The following prerequisites must be met for Jenkins Tomcat setup.

Step 1: Verifying Java Installation

To verify Java installation, open the console and execute the following java command.

OS	Task	Command
Windows	Open command console	\>java -version
Linux	Open command terminal	\$java -version

If Java has been installed properly on your system, then you should get one of the following outputs, depending on the platform you are working on.

OS	Output
Windows	Java version "1.7.0_60" Java (TM) SE Run Time Environment (build 1.7.0_60-b19) Java Hotspot (TM) 64-bit Server VM (build 24.60-b09, mixed mode)
Linux	java version "1.7.0_25" Open JDK Runtime Environment (rhel-2.3.10.4.el6_4-x86_64) Open JDK 64-Bit Server VM (build 23.7-b01, mixed mode)

We assume the readers of this tutorial have Java 1.7.0_60 installed on their system before proceeding for this tutorial.

In case you do not have Java JDK, you can download it from the link <http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>

Step 2: Verifying Java Installation

Set the JAVA_HOME environment variable to point to the base directory location where Java is installed on your machine. For example,

OS	Output
Windows	Set Environmental variable JAVA_HOME to C:\ProgramFiles\java\jdk1.7.0_60

Linux	export JAVA_HOME=/usr/local/java-current
-------	--

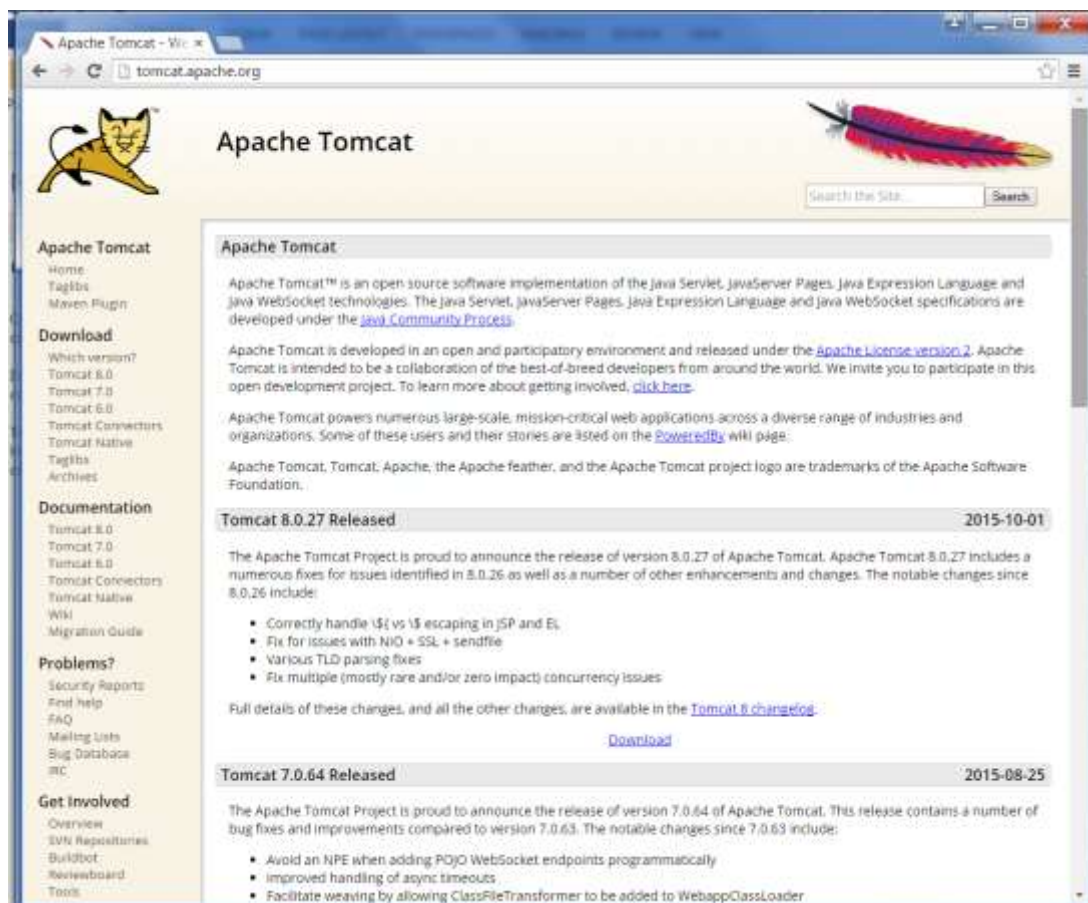
Append the full path of the Java compiler location to the System Path.

OS	Output
Windows	Append the String; C:\Program Files\Java\jdk1.7.0_60\bin to the end of the system variable PATH.
Linux	export PATH=\$PATH:\$JAVA_HOME/bin/

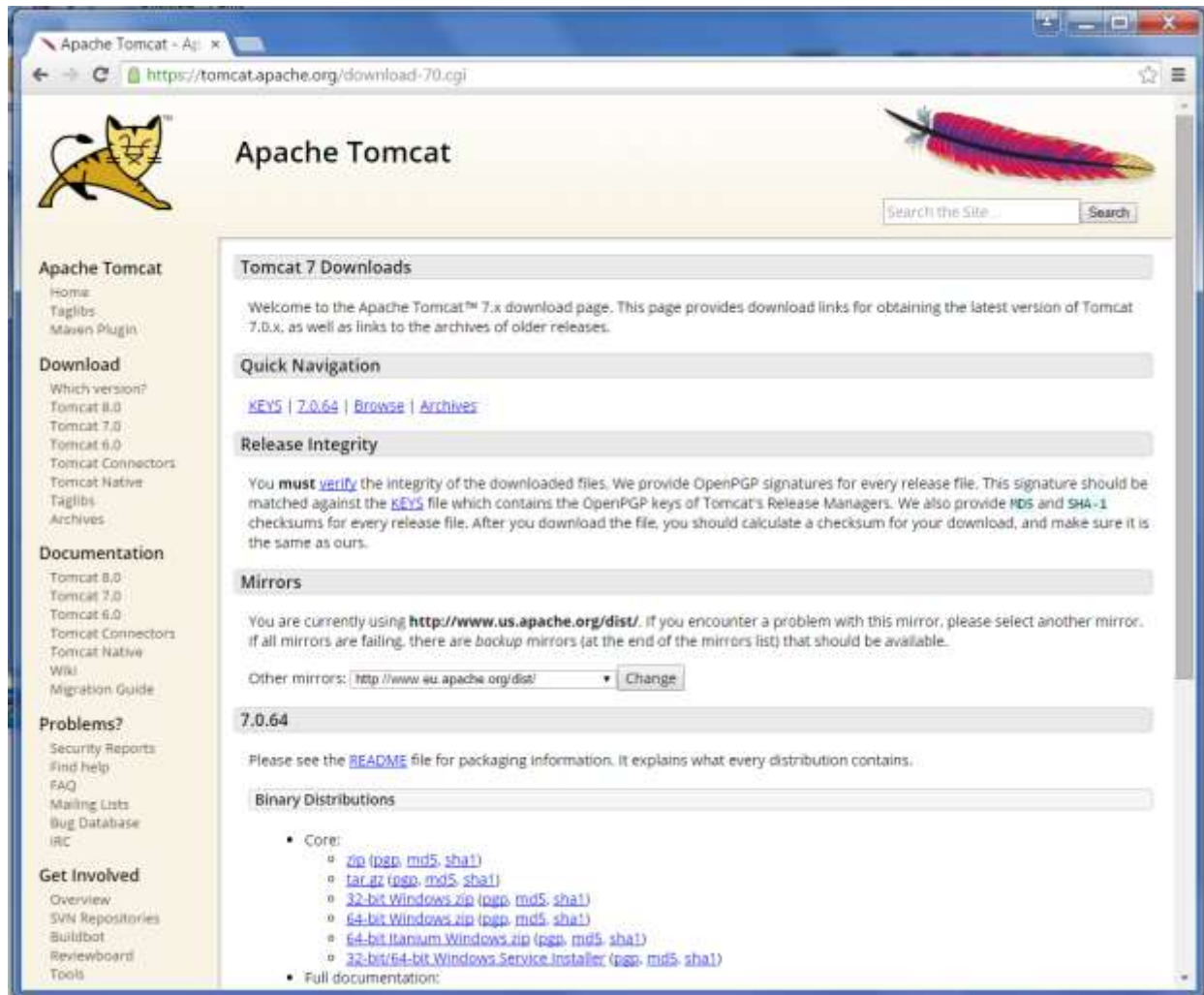
Verify the command java-version from command prompt as explained above.

Step 3: Download Tomcat

The official website for tomcat is <http://tomcat.apache.org/>. If you click the given link, you can get the home page of the tomcat official website as shown below.



Browse to the link <https://tomcat.apache.org/download-70.cgi> to get the download for tomcat.



Go to the 'Binary Distributions' section. Download the 32-bit Windows zip file.

Then unzip the contents of the downloaded zip file.

Step 4: Jenkins and Tomcat Setup

Copy the Jenkins.war file which was downloaded from the previous section and copy it to the webapps folder in the tomcat folder.

Now open the command prompt. From the command prompt, browse to the directory where the tomcat7 folder is location. Browse to the bin directory in this folder and run the start.bat file

```
E:\Apps\tomcat7\bin>startup.bat
```

Once the processing is complete without major errors, the following line will come in the output of the command prompt.

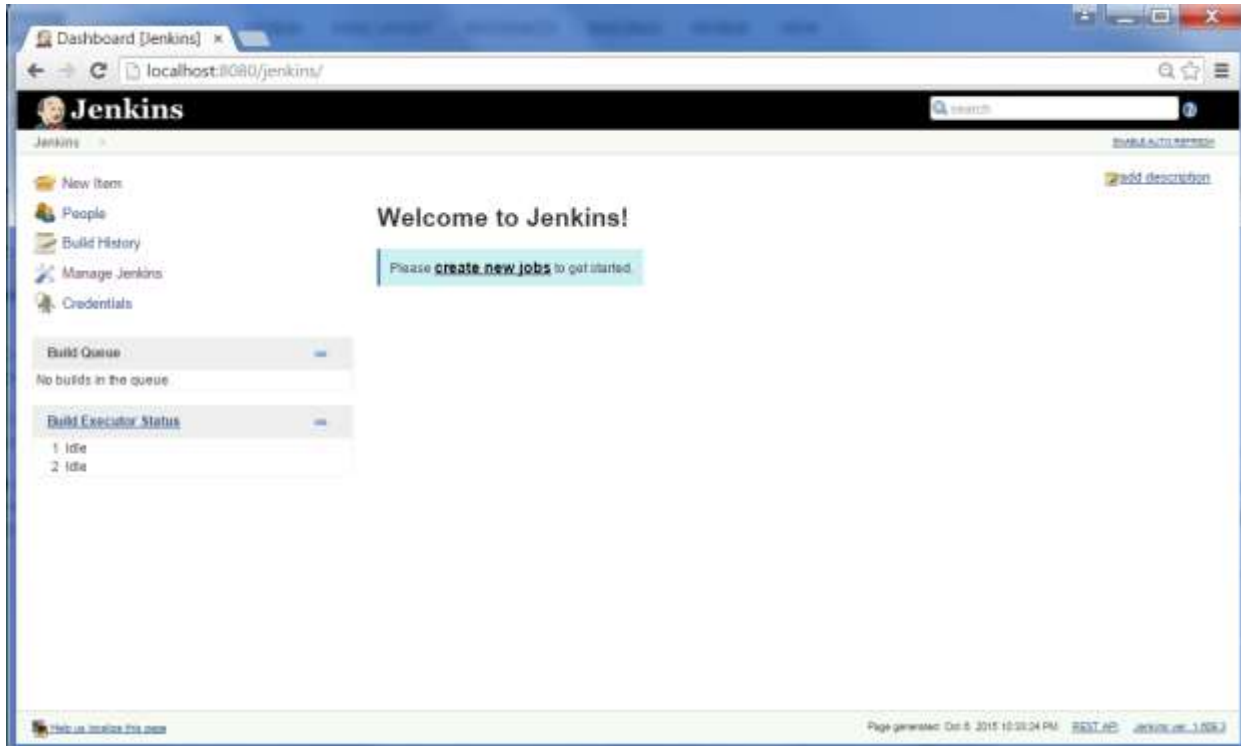
```
INFO: Server startup in 1302 ms
```

Open the browser and go to the link – <http://localhost/jenkins/>. Jenkins will be up and running on tomcat.

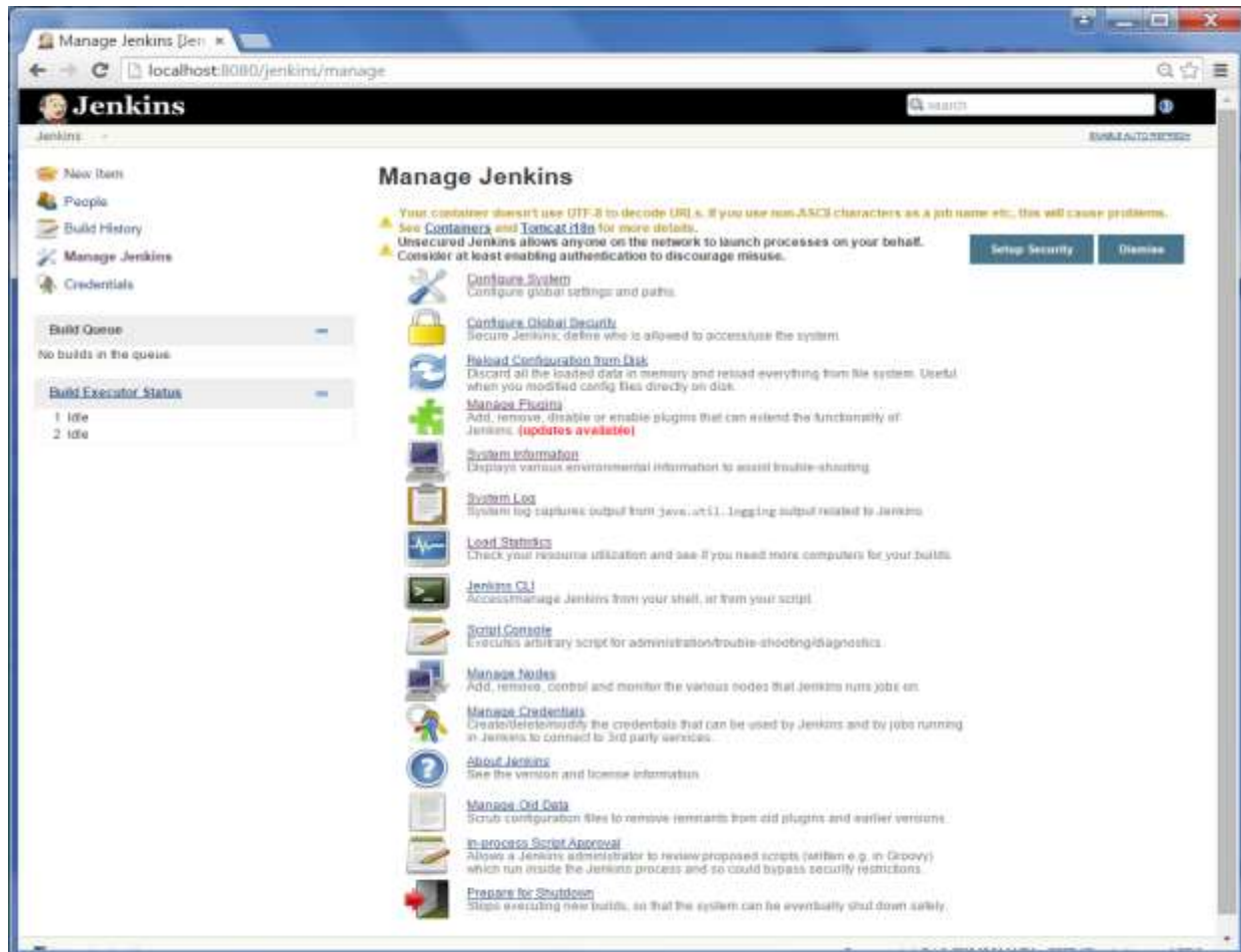


4. Jenkins – Git Setup

For this exercise, you have to ensure that Internet connectivity is present from the machine on which Jenkins is installed. In your Jenkins Dashboard (Home screen), click the Manage Jenkins option on the left hand side.



In the next screen, click the 'Manage Plugins' option.



End of ebook preview
If you liked what you saw...
Buy it from our store @ **<https://store.tutorialspoint.com>**