Jenkins Setup on Windows using war file

Step1: Download Jenkins (.war file)

https://get.jenkins.io/war-stable/2.289.1/jenkins.war

jenkins.war

Step2: Run below command in the windows command prompt (*Don't close command prompt until your work is done*)


```
Select C:\Windows\System32\cmd.exe - java -jar jenkins.war

Microsoft Windows [Version 10.0.19041.985]
(c) Microsoft Corporation. All rights reserved.

C:\Jenkins>java -jar jenkins.war
```

java -jar jenkins.war

Step3: Open below URL on browser and follow below steps as shown in screenshots.

http://localhost:8080/

localhost:8080/login?from=%2F

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (**not sure where to find it?**) and this file on the server:

`C:\Users\admin\.jenkins\secrets\initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Continue

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (**not sure where to find it?**) and this file on the server:

`C:\Users\admin\.jenkins\secrets\initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Continue

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Getting Started

Folders	OWASP Markup Formatter	Build Timeout	Credentials Binding	** SSH server Folders
Timestamper	Workspace Cleanup	Ant	Gradle	
Pipeline	GitHub Branch Source	Pipeline: GitHub Groovy Libraries	Pipeline: Stage View	
Git	SSH Build Agents	Matrix Authorization Strategy	PAM Authentication	
LDAP	Email Extension	Mailer		

** - required dependency

Create First Admin User

Username: admin

Password: •••••

Confirm password: •••••

Full name: admin

E-mail address: admin@gmail.com

Skip and continue as admin   Save and Continue

Instance Configuration

Jenkins URL:
`http://localhost:8080/`

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BUILD_URL` environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

Not now   **Save and Finish**

Getting Started

Jenkins is ready!

Your Jenkins setup is complete.

Start using Jenkins
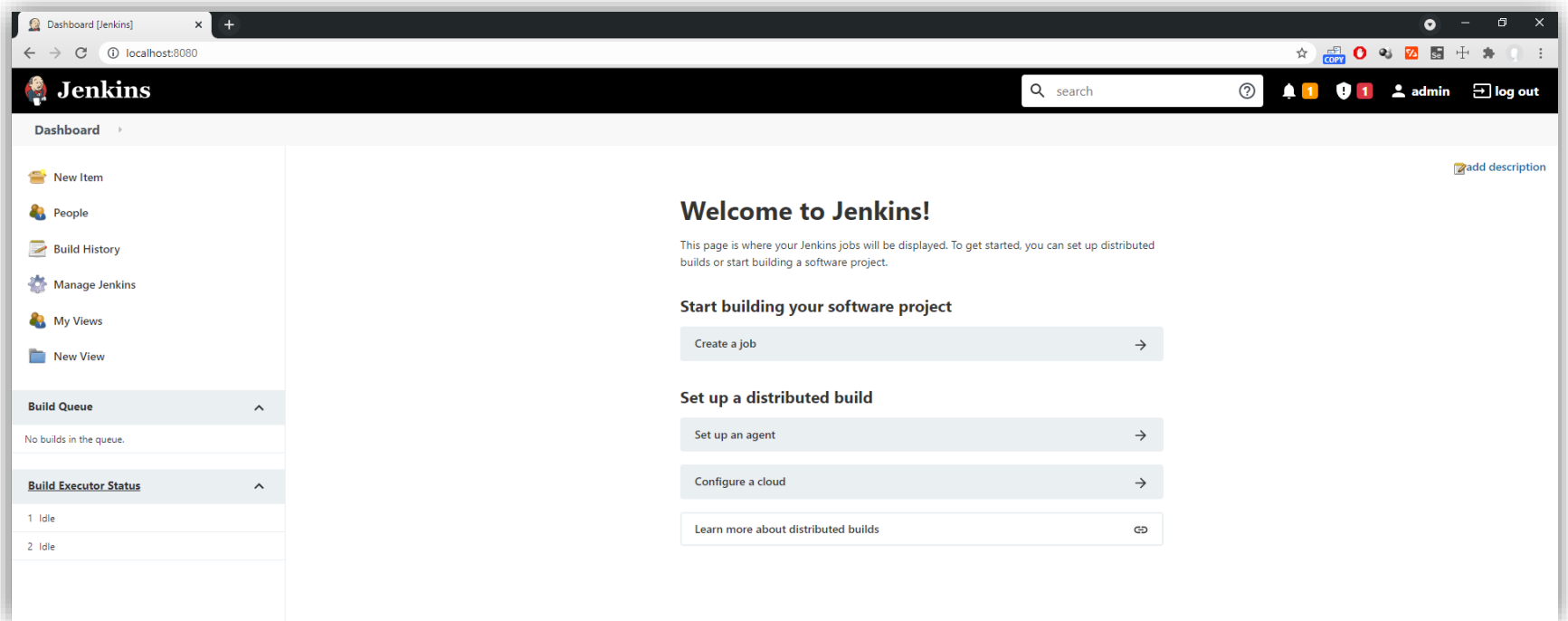
Here you can see Jenkins dashboard.

Install Maven Plugin

Manage Jenkins→ System Configuration→ Click on Manage Plugins
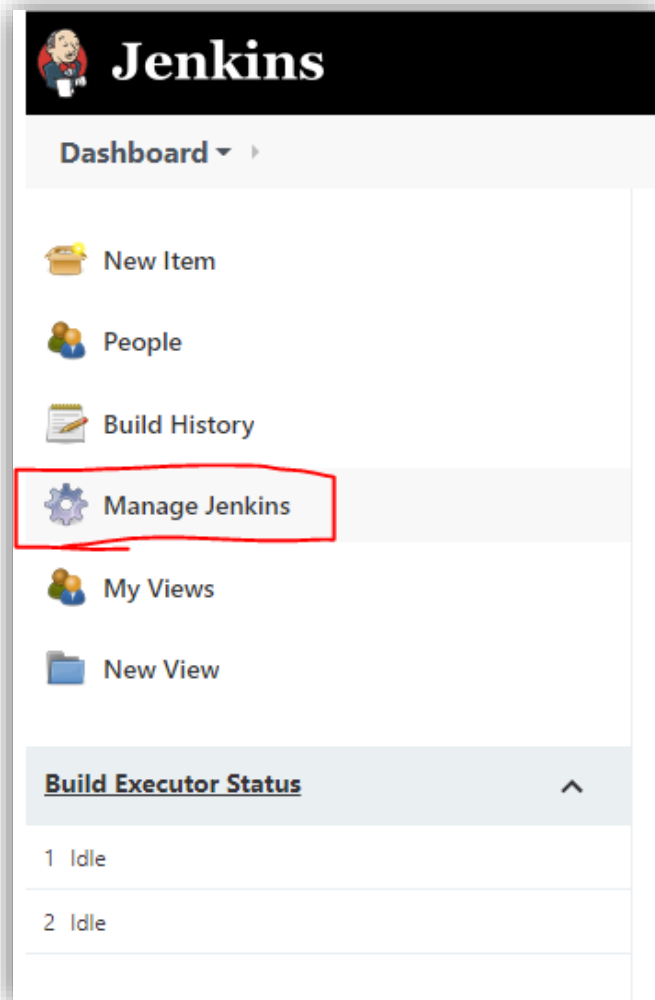
System Configuration

Configure System
Configure global settings and paths.

Global Tool Configuration
Configure tools, their locations and automatic installers.

Manage Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

Here select all Maven related plugins from the list.

Q maven

Updates	Available	Installed	Advanced

Install ↑	Name
☑	**Maven Integration** [Build Tools] This plug-in provides, for better and for worse, a deep integration of Jenkins and Maven: Aut Jenkins publishers (Junit, ...).

Install without restart

After installation restart Jenkins & Login

Global Tools Configurations (Java. Maven, Git configurations in Jenkins)

Dashboard→Manage Jenkins→ Global tool configuration
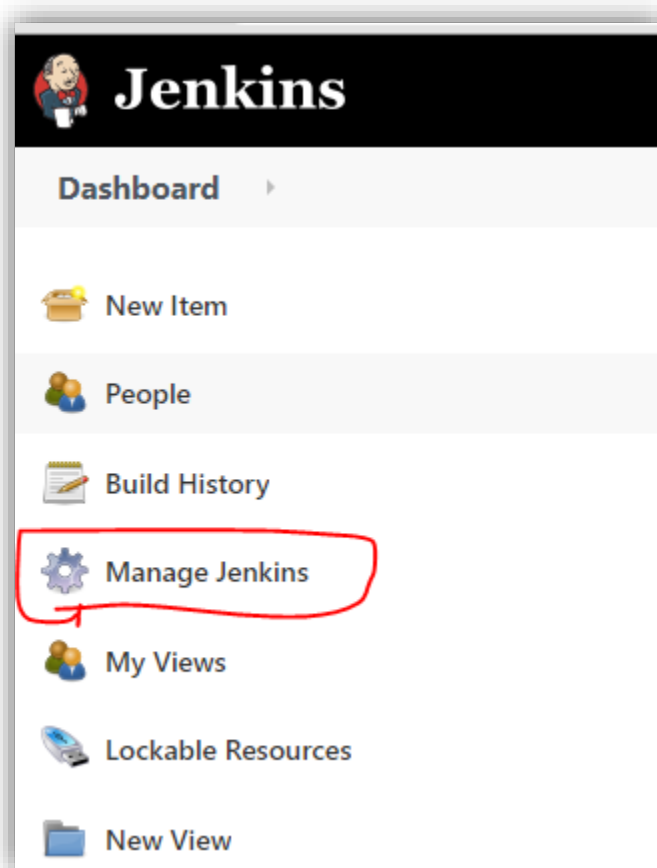
Manage Jenkins

Building on the controller node can be a security issue. You should set up distributed builds. See **the documentation**.

System Configuration

Configure System
Configure global settings and paths.

Global Tool Configuration
Configure tools, their locations and automatic installers.

Manage Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

Install as Windows Service
Installs Jenkins as a Windows service to this system, so that Jenkins starts automatically when the machine boots.

Maven

Maven installations

Add Maven

Maven

Name

mymaven

MAVEN_HOME

C:\apache-maven-3.8.1

☐ Install automatically

JDK

JDK installations

Add JDK

⣿ JDK

Name

myjava

JAVA_HOME

C:\Program Files\Java\jdk1.8.0_281

☐ Install automatically

Git

Git installations

Git

Name

Default

Path to Git executable ?

C:\Program Files\Git\bin\git.exe

☐ Install automatically ?

Save **Apply**

How to Run github Maven project in Jenkins

Dashboard → Create a new Item

Enter an item name

opencart
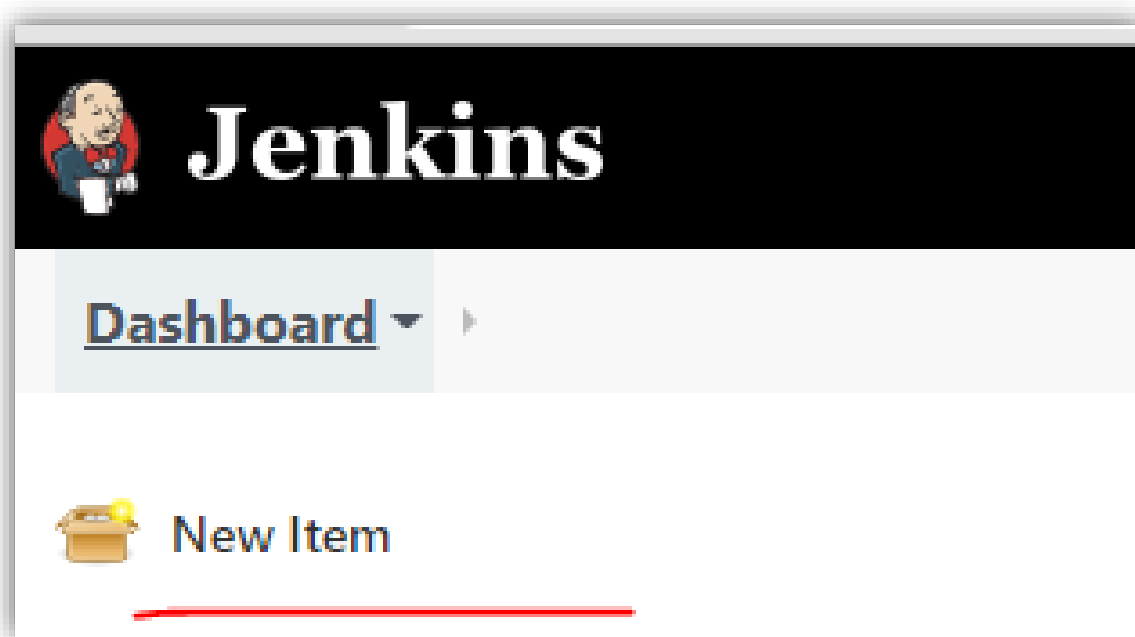
» *Required field*

Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

Maven project

Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder

Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

GitHub Organization

Scans a GitHub organization (or user account) for all repositories matching some defined markers.

Multibranch Pipeline

Creates a set of Pipeline projects according to detected branches in one SCM repository.

OK

Source Code Management

○ None

◉ Git

Repositories

Repository URL

https://github.com/pavanoltraining/opencart.git

Credentials

- none - ⌄   🔑Add ⌄

Build

Root POM

pom.xml

Goals and options

clean test

Jenkins

- ⬆ Back to Dashboard
- 🔍 Status
- 📝 Changes
- 📁 Workspace
- ▶ **Build Now**
- ⚙ Configure
- 🚫 Delete Maven project
- 📄 Modules
- 📝 Rename

Maven project

📁 Workspace

📝 Recent Changes

Permalinks

Build History   trend ∧

find	x

🔗 Atom feed for all 🔗 Atom feed for failures

 Back to Project

 Status

 Changes

 Console Output

 View as plain text

 Edit Build Information

 Git Build Data

 Build Artifacts As Maven Repository

💬 Console Output

```
Started by user admin
Running as SYSTEM
Building in workspace C:\Users\admin\.jenkins\workspace\opencart
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/pavanoltraining/opencart.git
 > C:\Program Files\Git\bin\git.exe init C:\Users\admin\.jenkins\workspace\opencart # timeout=10
Fetching upstream changes from https://github.com/pavanoltraining/opencart.git
 > C:\Program Files\Git\bin\git.exe --version # timeout=10
 > git --version # 'git version 2.31.1.windows.1'
 > C:\Program Files\Git\bin\git.exe fetch --tags --force --progress -- https://github.com/pavanoltra
 > C:\Program Files\Git\bin\git.exe config remote.origin.url https://github.com/pavanoltraining/open
 > C:\Program Files\Git\bin\git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/orig
Avoid second fetch
 > C:\Program Files\Git\bin\git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision 2672ee99052b417861c1011b4f43bc498c5453c2 (refs/remotes/origin/master)
 > C:\Program Files\Git\bin\git.exe config core.sparsecheckout # timeout=10
 > C:\Program Files\Git\bin\git.exe checkout -f 2672ee99052b417861c1011b4f43bc498c5453c2 # timeout=1
Commit message: "changes done in dummy2.txt"
First time build. Skipping changelog.
```

```
[INFO] ------------------------------------------------------------
[INFO]  T E S T S
[INFO] ------------------------------------------------------------
[INFO] Running TestSuite
[TestNGContentHandler] [WARN] It is strongly recommended to add "<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd" >" at the top of your file, otherwise TestNG may fail or not work as expected.
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Starting ChromeDriver 91.0.4472.19 (1bf021f248676a0b2ab3ee0561d83a59e424c23e-refs/branch-heads/4472@{#288}) on port 27106
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Jun 06, 2021 4:00:46 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
Jun 06, 2021 4:00:47 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 91
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 24.98 s - in TestSuite
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[JENKINS] Recording test results
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  01:12 min
[INFO] Finished at: 2021-06-06T16:01:01+05:30
[INFO] ------------------------------------------------------------------------
Waiting for Jenkins to finish collecting data
[JENKINS] Archiving C:\Users\admin\.jenkins\workspace\opencart\pom.xml to org/opencart/0.0.1-SNAPSHOT/opencart-0.0.1-SNAPSHOT.pom
channel stopped
Finished: SUCCESS
```