

Lab Exercise 7

Integrating Maven with Jenkins

Name: Pulkit

Sap id: 500125835

Batch 3 DevOps

Objective: To install the Maven plugin in Jenkins for smooth integration and automation of Maven-based build processes within the Jenkins environment

Tools required: Git, GitHub, and Jenkins

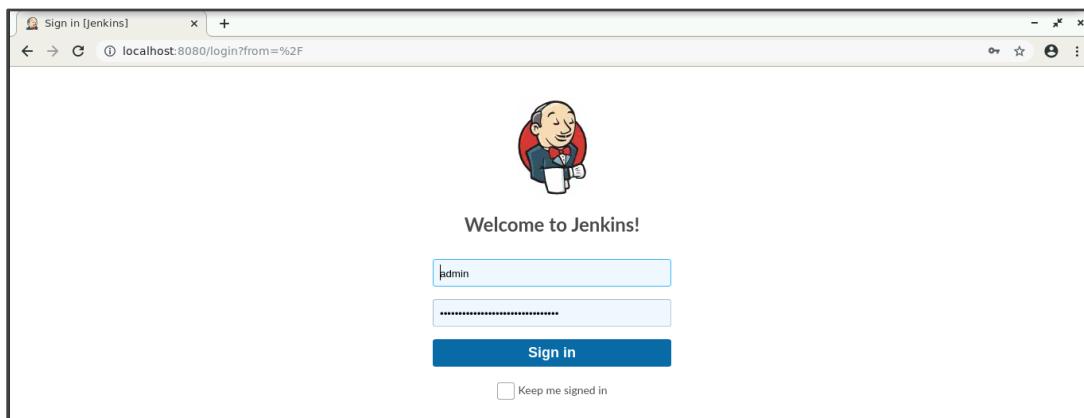
Prerequisites: None

Steps to be followed:

1. Install the Maven plugin
2. Set up Global Tool Configuration
3. Fork a sample repository
4. Integrate Maven with Jenkins

Step 1: Install the Maven plugin

1. Open the browser, go to the Jenkins Dashboard by typing **localhost:8080** in your browser, provide the credentials, and click the **Sign in** button



2. Click on the **Manage Jenkins** option as shown in the screenshot below:

The screenshot shows the Jenkins Manage Jenkins interface. At the top, there's a navigation bar with a back arrow, a forward arrow, and a search icon. The title "Manage Jenkins - Jenkins" is centered. Below the title is a breadcrumb trail: "Jenkins / Manage Jenkins". On the right side of the header are three buttons: "Set up agent", "Set up cloud", and "Dismiss". A search bar with the placeholder "Search settings" is also present.

The main content area is titled "Manage Jenkins". It features a prominent yellow banner at the top stating: "Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#)". Below this banner are several configuration sections:

- System Configuration**: Includes links for "System" (global settings), "Nodes" (node management), "Tools" (tool configuration), "Clouds" (cloud provisioning), "Plugins" (plugin management), and "Appearance" (UI customization).
- Security**: Includes links for "Security" (user access control) and "Users" (user management).
- Status Information**: Includes links for "System Information" (environmental info), "System Log" (log capture), and "Load Statistics" (resource utilization).

3. Click on the **Plugins** option as shown in the screenshot below:

This screenshot is identical to the one above, showing the Jenkins Manage Jenkins page. The "Plugins" section under the "System Configuration" heading is highlighted with a gray background, indicating it has been selected. The other sections (System, Nodes, Tools, Clouds, Appearance) are visible but not highlighted.

- Click on **Installed plugins** to verify whether the **Maven Integration plugin** has been installed

The screenshot shows the Jenkins 'Plugins' management interface. On the left, a sidebar lists 'Updates', 'Available plugins', 'Installed plugins' (which is selected and highlighted in blue), 'Advanced settings', and 'Download progress'. The main content area has a search bar at the top with the text 'maven'. Below the search bar, a table lists the 'Maven Integration plugin' version 3.27. The table includes columns for 'Name', 'Health' (green circle with '100'), 'Enabled' (blue switch with a checkmark), and a delete icon. A tooltip for the plugin describes its function: 'This plugin provides a deep integration between Jenkins and Maven. It adds support for automatic triggers between projects depending on SNAPSHOTs as well as the automated configuration of various Jenkins publishers such as Junit.' There are also links to 'Report an issue with this plugin' and 'View plugin page'. At the bottom right of the main area, it says 'REST API' and 'Jenkins 2.516.3'.

- Use the following command to check the Maven version:
mvn -version

```
[pulkit@Pulkits-MacBook-Air ~ % mvn -version
Apache Maven 3.9.11 (3e54c93a704957b63ee3494413a2b544fd3d825b)
[Maven home: /opt/homebrew/Cellar/maven/3.9.11/libexec
Java version: 25, vendor: Homebrew, runtime: /opt/homebrew/Cellar/openjdk/25/libexec/openjdk.jdk/Contents/Home
Default locale: en_IN, platform encoding: UTF-8
OS name: "mac os x", version: "15.6.1", arch: "aarch64", family: "mac"
pulkit@Pulkits-MacBook-Air ~ % ]
```

Step 2: Set up Global Tool Configuration

1. Go to the Jenkins Dashboard, click on Manage Jenkins, and then select Tools from the list of options

The screenshot shows the Jenkins Manage Jenkins interface. At the top, there is a banner with the message: "Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#)." Below the banner are several configuration sections:

- System**: Configure global settings and paths.
- Tools**: Configure tools, their locations and automatic installers. This section includes a link to Maven installations.
- Nodes**: Add, remove, control and monitor the various nodes Jenkins runs jobs on.
- Clouds**: Add, remove, and configure cloud instances to provision agents on-demand.
- Plugins**: Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- Appearance**: Configure the look and feel of Jenkins.

Below these sections is a **Security** section with links to Security, Credentials, and Credential Providers.

2. To configure Maven, click on the Maven installations button in the Maven section and enter a Name and MAVEN_HOME path

The screenshot shows the Jenkins Tools configuration page. Under the **Maven installations** section, there is a table with one entry:

Name	Version	Action
Maven-3.9.11	3.9.11	X

Below the table, there are buttons for **Add Maven** and **Install automatically**. A detailed view of the Maven row shows the **Install from Apache** section with a Version of 3.9.11 and an **Add Installer** dropdown.

