

RBG technologies

KNOWLEDGE CREATES UNLIMITED TECHNOLOGIES - LIKE PRIMARY COLORS

OUR ESTEEMED COURSES :

**JAVA
SELENIUM
API TESTING
MANUAL TESTING**



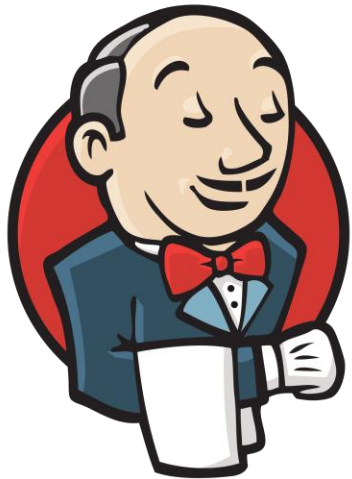
 CONTACT US :
+91 93986 94373



RBG technologies

Getting Started with Jenkins

A Beginner's Guide



Jenkins

www.rbgtechnologies.com

INDEX

<u>Content</u>	<u>Page Number</u>
• Welcome to Jenkins	1
• Installation of Jenkins	2-15
• Unlocking Jenkins and Account Creation	16-22
• Creating first Job, Local Maven Project	22-35
• Creating first job, Git Maven Project	36-41
• Tear Points (eclipse error)	42-44

WELCOME TO THE JENKINS

What is Jenkins?

- Jenkins is a self-contained, open-source automation server which can be used to automate all sorts of tasks related to building, testing, and delivering or deploying software.
- Jenkins can be installed through native system packages, Docker, or even run standalone by any machine with a Java Runtime Environment (JRE) installed.
- It's a core tool for continuous integration and continuous delivery (CI/CD) pipelines, allowing developers to automate the software delivery lifecycle, from code changes to deployment

INSTALLATION :

Prerequisites :

For this, you will require:

A machine with:

256 MB of RAM, although more than 2 GB is recommended
10 GB of drive space (for Jenkins and your Docker image)

The following software installed:

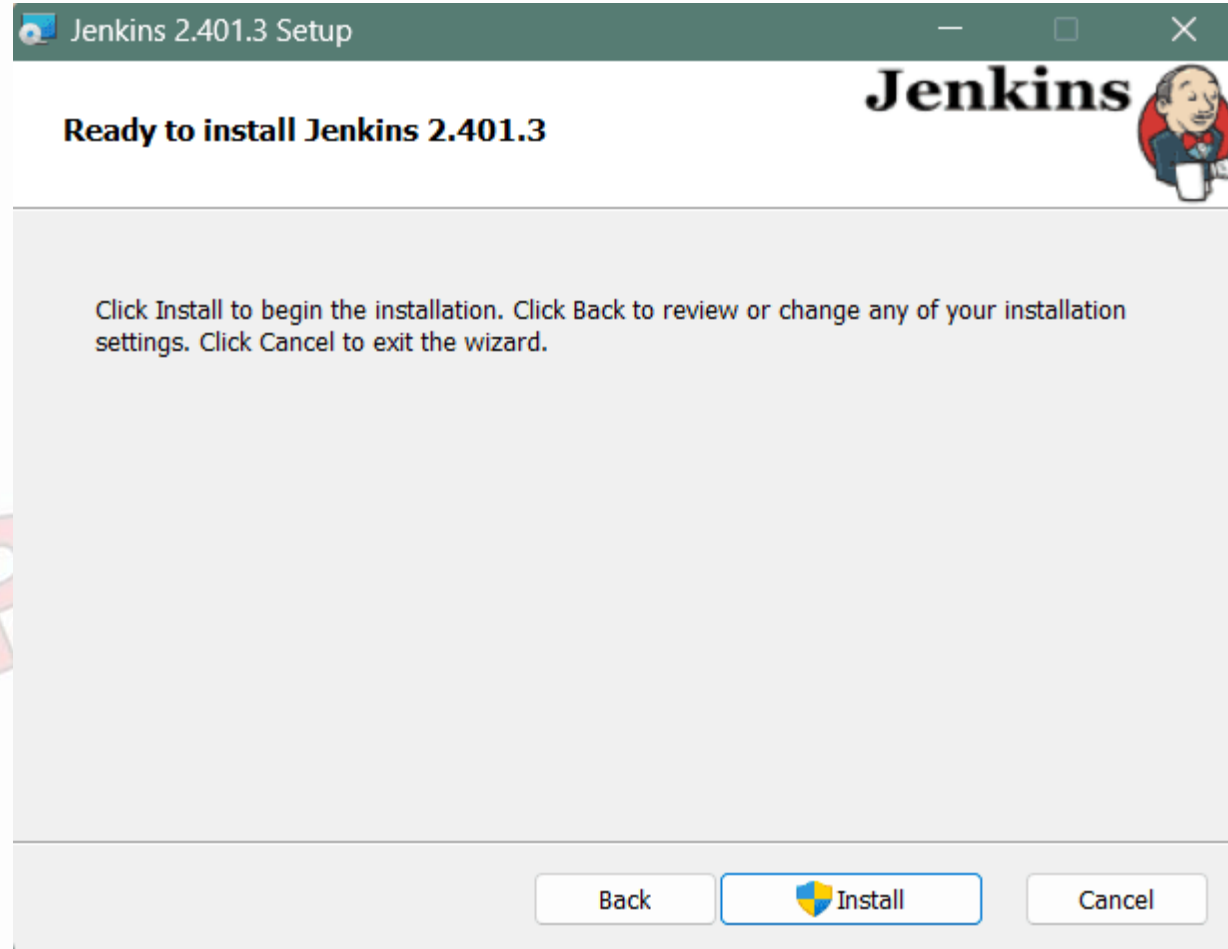
Java 17 or 21

Jenkins Download Link:

<https://www.jenkins.io/download/thank-you-downloading-windows-installer-stable/>

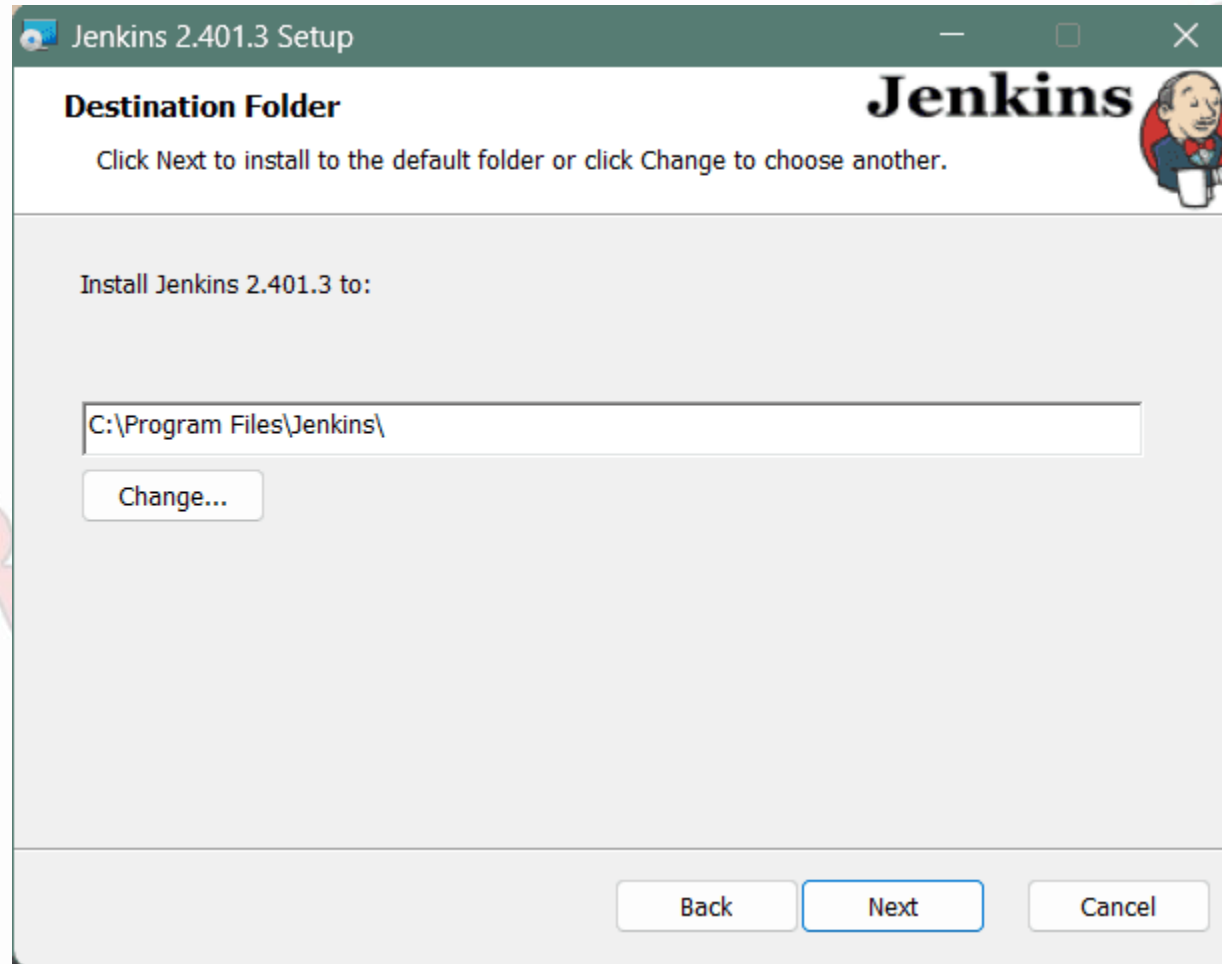
Step 1: Setup wizard

On opening the Windows Installer, an **Installation Setup Wizard** appears, Click **Next** on the Setup Wizard to start your installation.



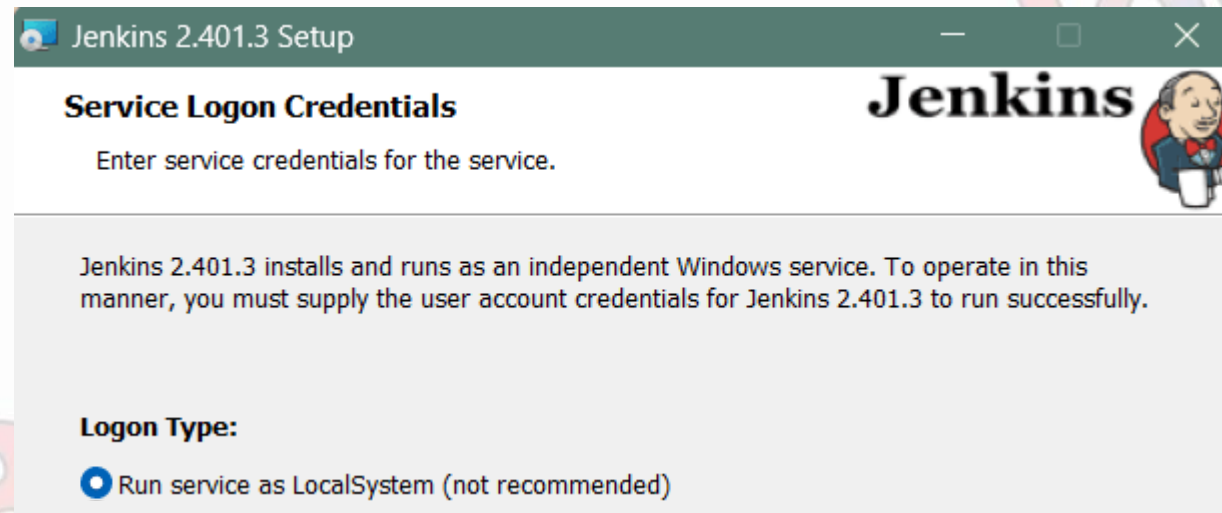
Step 2: Select destination folder

Select the destination folder to store your Jenkins Installation and click **Next** to continue.



Step 3: Service logon credentials

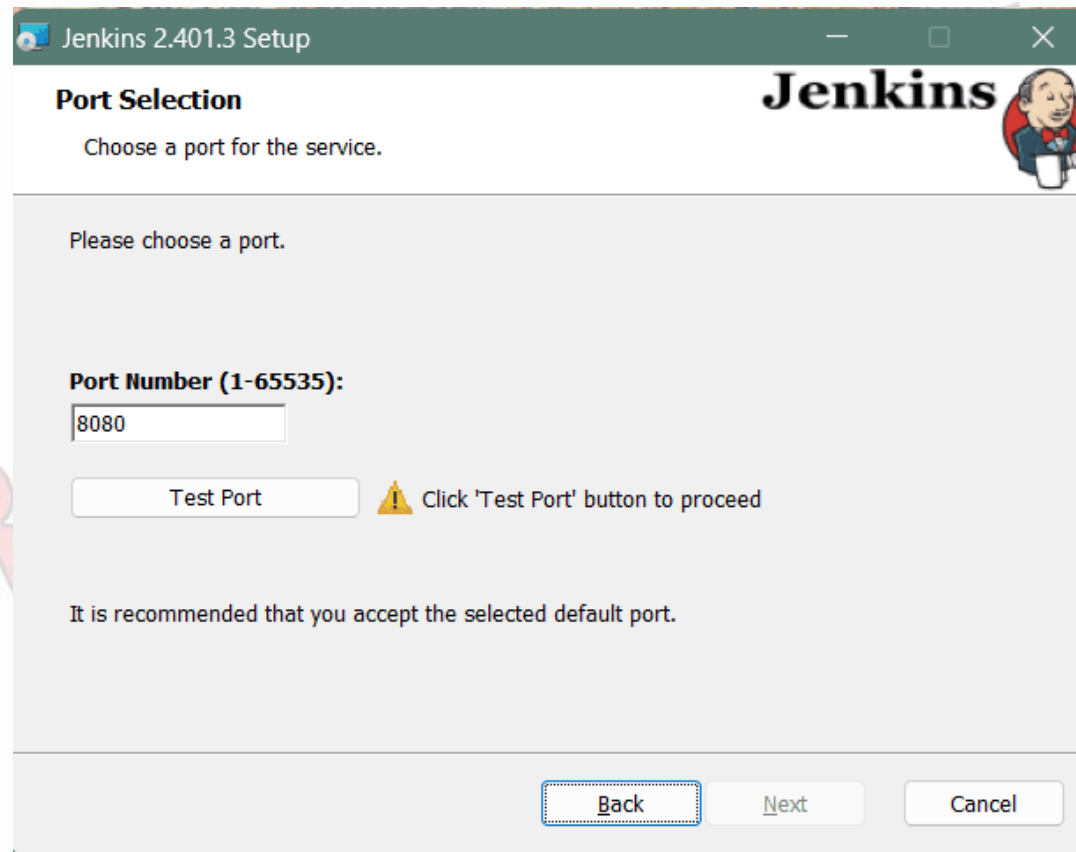
When Installing Jenkins, Click On **Run Service as Local System(Windows equivalent of root)** which will grant Jenkins full access to your machine and services.



After Selecting Local System, Click on Next

Step 4: Port selection

Specify the port on which Jenkins will be running, **Test Port** button to validate whether the specified port is free on your machine or not. Consequently, if the port is free, it will show a green tick mark as shown below, then click on **Next**.

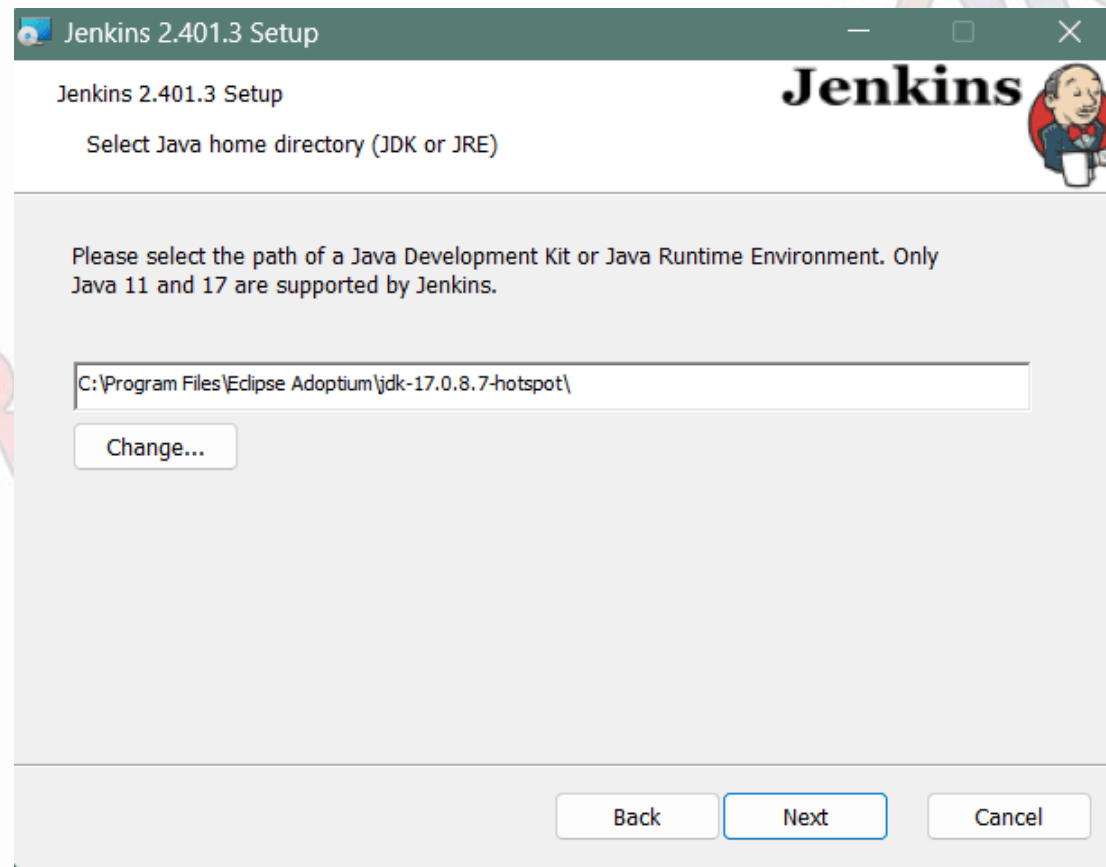


In Port Number we have to Use these ports example : 8001,8002,8081

Step 5: Select Java home directory

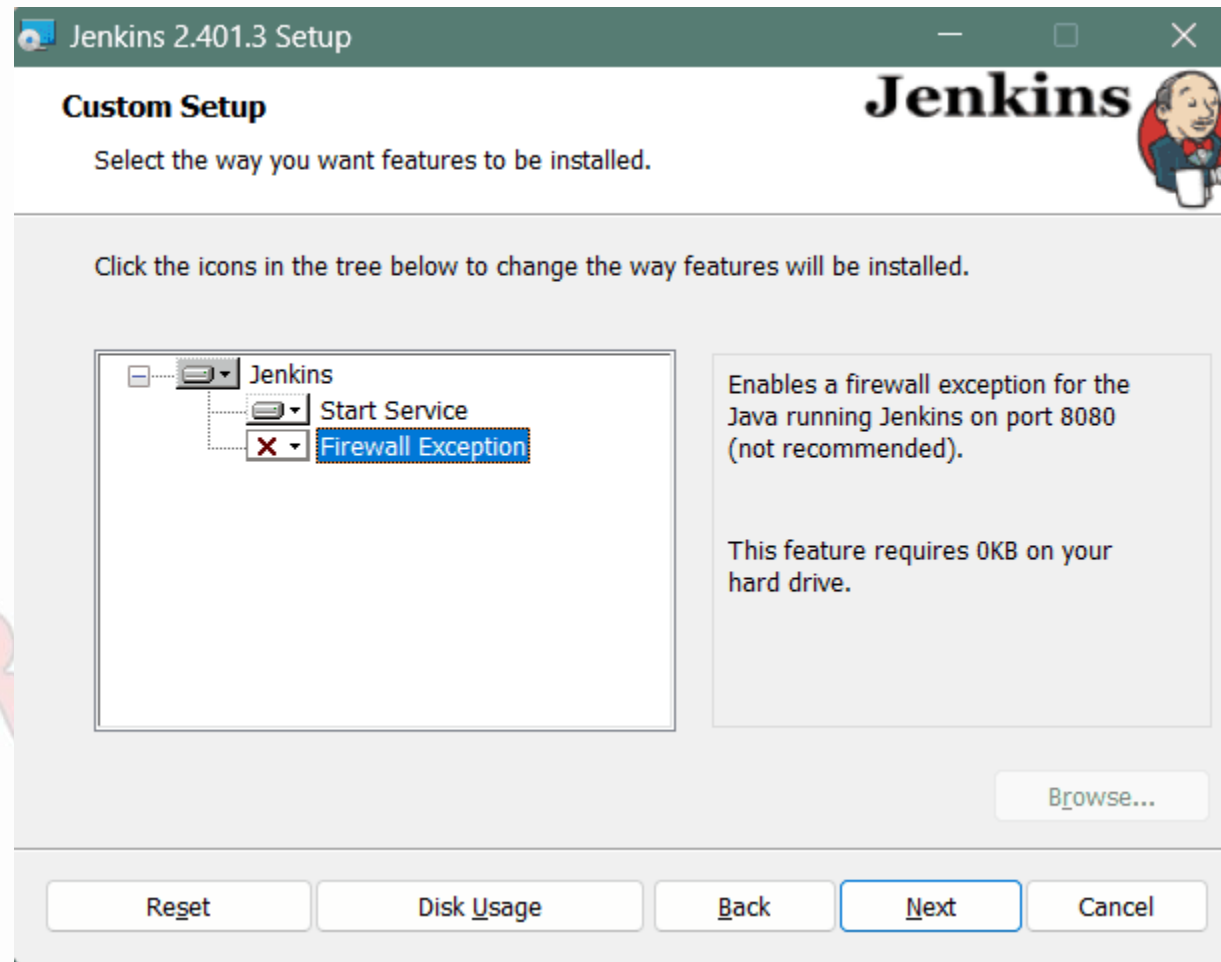
The installation process checks for Java on your machine and prefills the dialog with the Java home directory. If the needed Java version is not installed on your machine, you will be prompted to install it.

Once your Java home directory has been selected, click on **Next** to continue.



Step 6: Custom setup

Select other services that need to be installed with Jenkins and click on **Next**



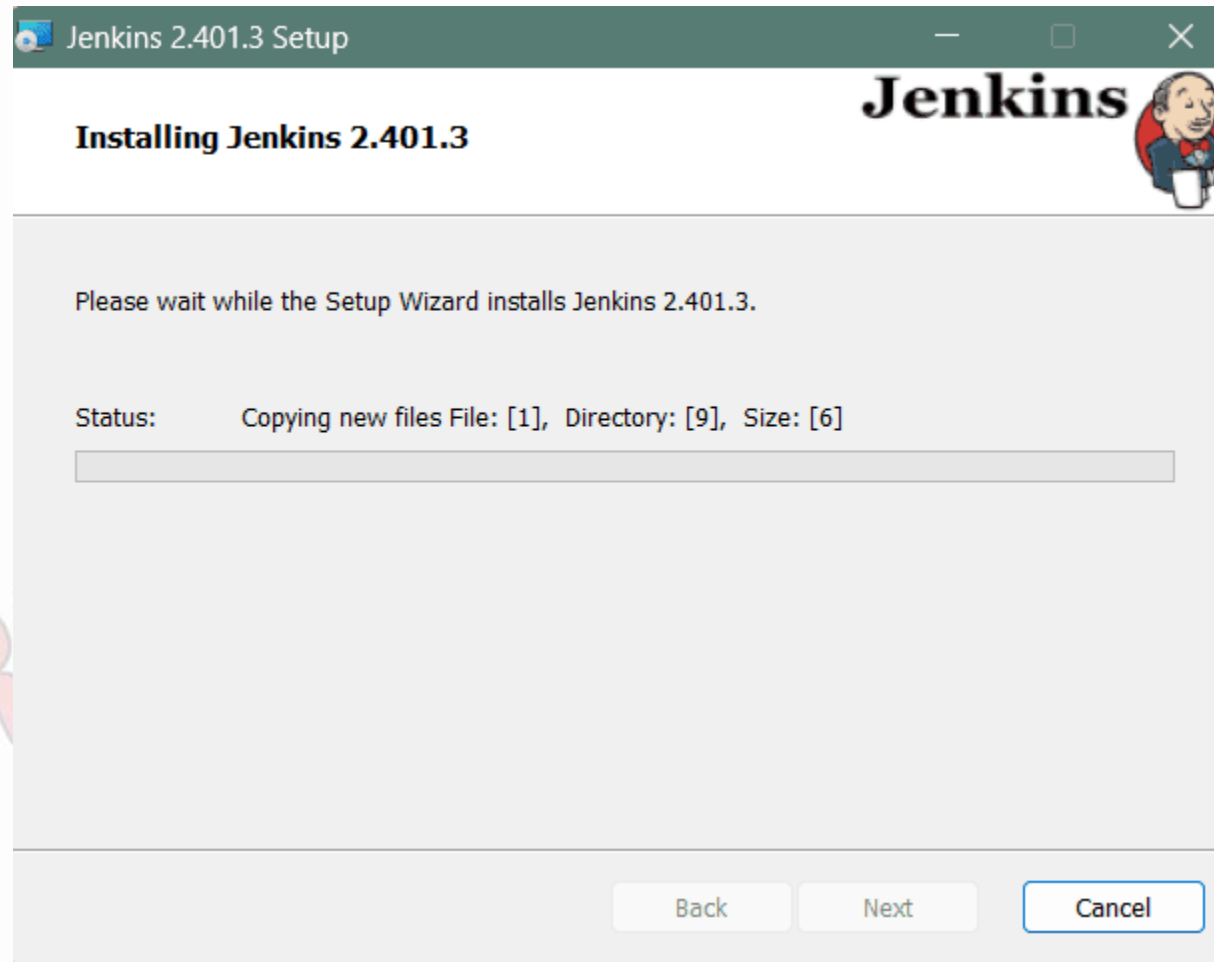
If we need any Firewall Exception we need to Enable.

Step 7: Install Jenkins

Click on the **Install** button to start the installation of Jenkins.



Additionally, clicking on the **Install** button will show the progress bar of installation, as shown below:



Step 8: Finish Jenkins installation

Once the installation completes, click on **Finish** to complete the installation.



Post-installation setup wizard

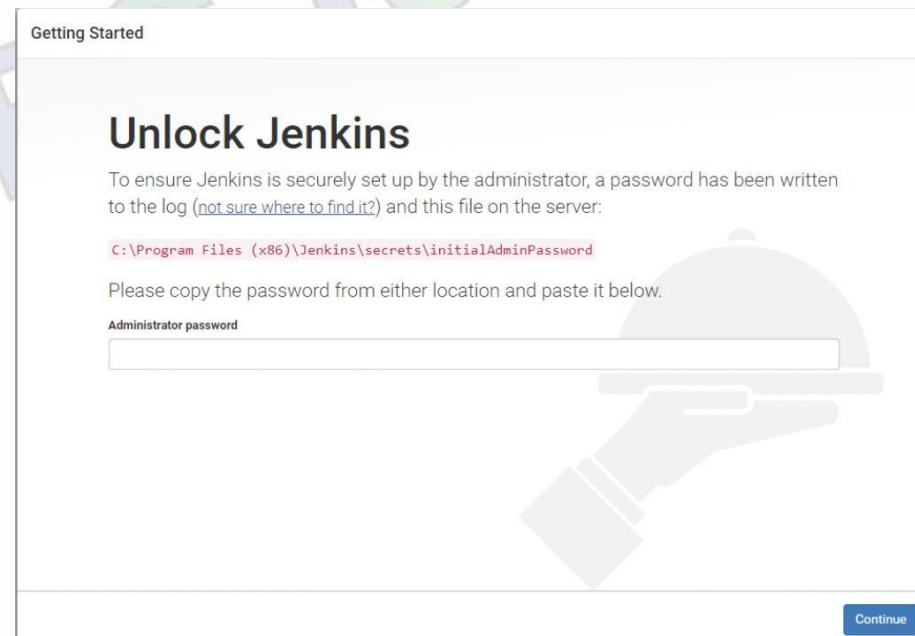
After downloading, installing and running Jenkins, the post-installation setup wizard begins.

Unlocking Jenkins

When you first access a new Jenkins controller, you are asked to unlock it using an automatically-generated password.

Step 1

Browse to `http://localhost:8080` (or whichever port you configured for Jenkins when installing it) and wait until the **Unlock Jenkins** page appears.



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:\Program Files (x86)\Jenkins\secrets\initialAdminPassword`

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

Administrator password

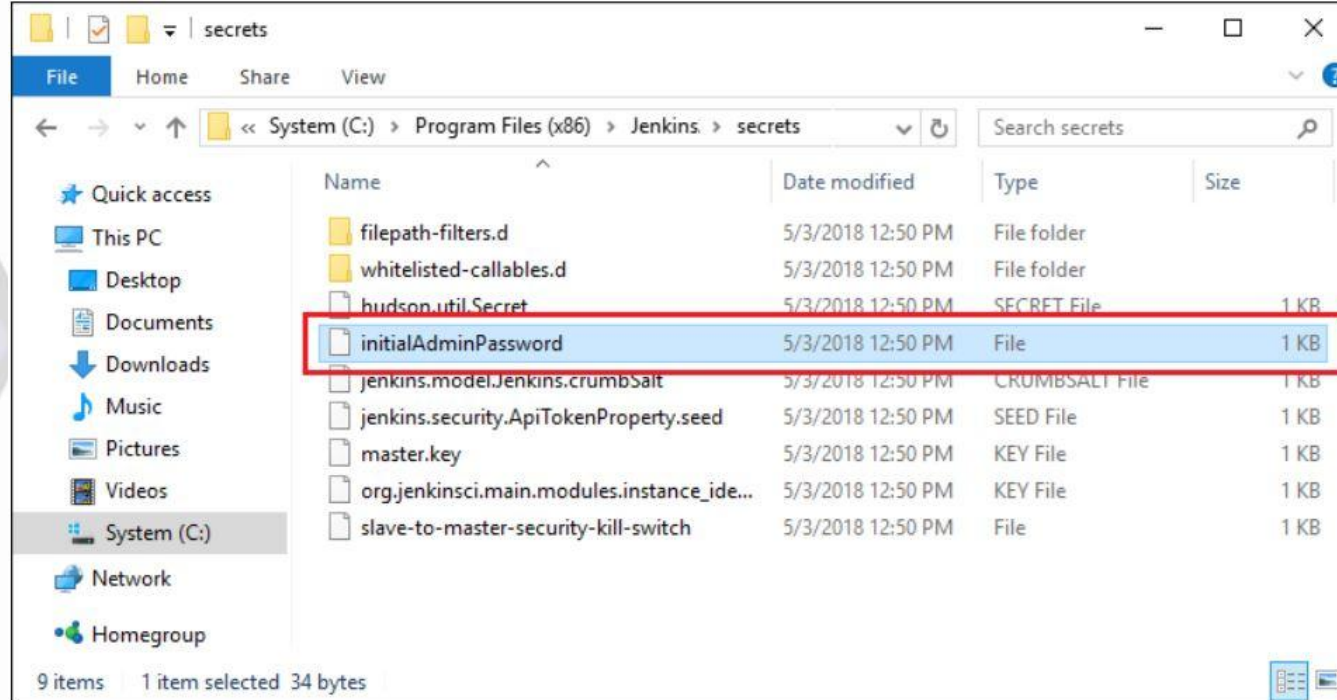
Continue

Step 2

The initial Administrator password should be found under the Jenkins installation path (set at Step 2 in Jenkins Installation).

For default installation location to C:\Program Files\Jenkins, a file called **initialAdminPassword** can be found under C:\Program Files\Jenkins\secrets.

However, If a custom path for Jenkins installation was selected, then you should check that location for **initialAdminPassword** file.



Step 3

Open the highlighted file and copy the content of the **initialAdminPassword** file.



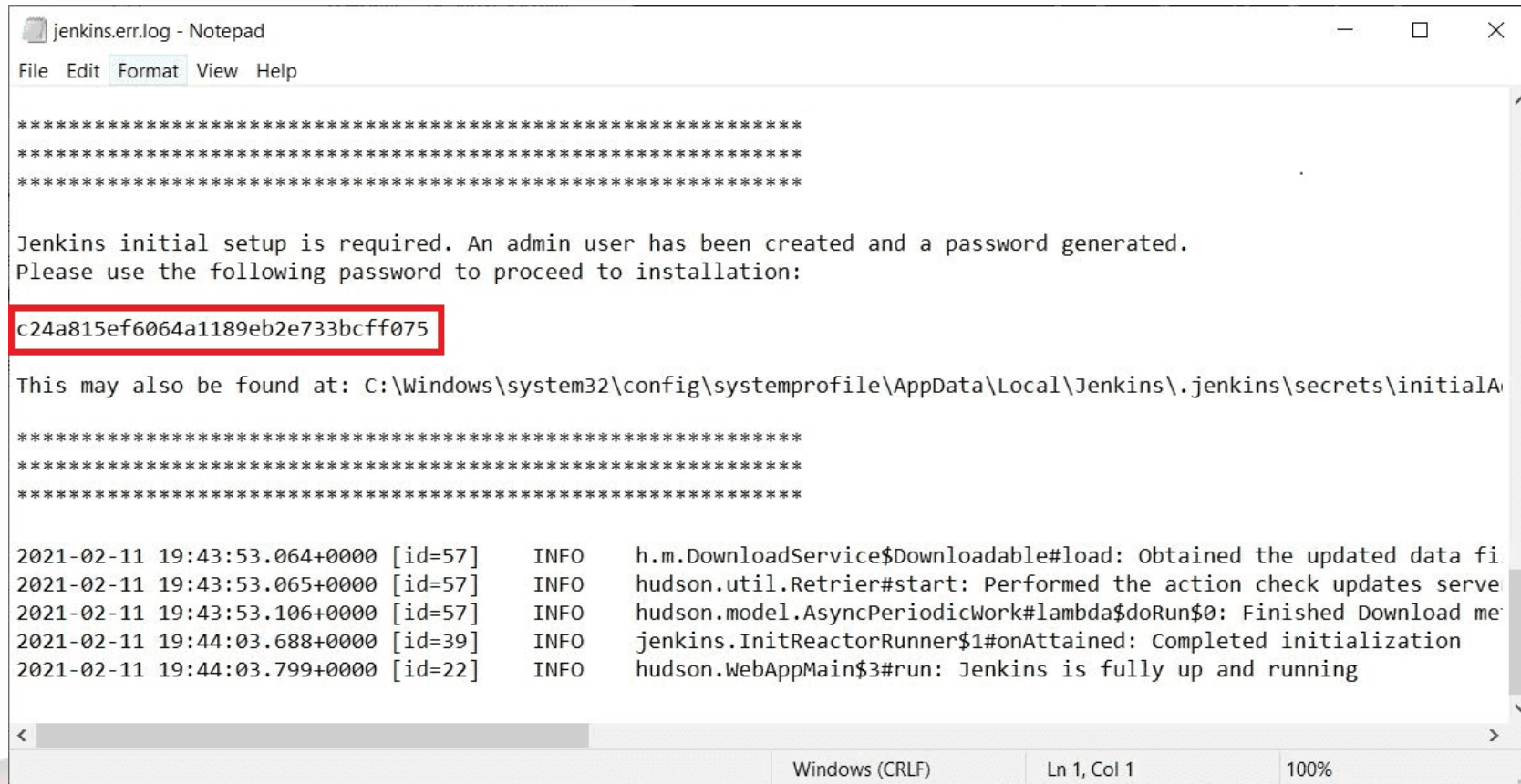
Step 4

On the **Unlock Jenkins** page, paste this password into the **Administrator password** field and click **Continue**.

Notes:

You can also access Jenkins logs in the **jenkins.err.log** file in your Jenkins directory specified during the installation.

The Jenkins log file is another location (in the Jenkins home directory) where the initial password can also be obtained.



```
jenkins.err.log - Notepad
File Edit Format View Help

*****
*****
*****

Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:

c24a815ef6064a1189eb2e733bcff075

This may also be found at: C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\secrets\initialAdminPassword

*****
*****
*****

2021-02-11 19:43:53.064+0000 [id=57] INFO h.m.DownloadService$Downloadable#load: Obtained the updated data fi
2021-02-11 19:43:53.065+0000 [id=57] INFO hudson.util.Retrier#start: Performed the action check updates serve
2021-02-11 19:43:53.106+0000 [id=57] INFO hudson.model.AsyncPeriodicWork#lambda$doRun$0: Finished Download me
2021-02-11 19:44:03.688+0000 [id=39] INFO jenkins.InitReactorRunner$1#onAttained: Completed initialization
2021-02-11 19:44:03.799+0000 [id=22] INFO hudson.WebAppMain$3#run: Jenkins is fully up and running

< Windows (CRLF) Ln 1, Col 1 100%
```

This password must be entered in the setup wizard on new Jenkins installations before you can access Jenkins's main UI. This password also serves as the default administrator account's password (with username "admin") if you happen to skip the subsequent user-creation step in the setup wizard.

Jenkins download and deployment x Sign in [Jenkins] x +

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:

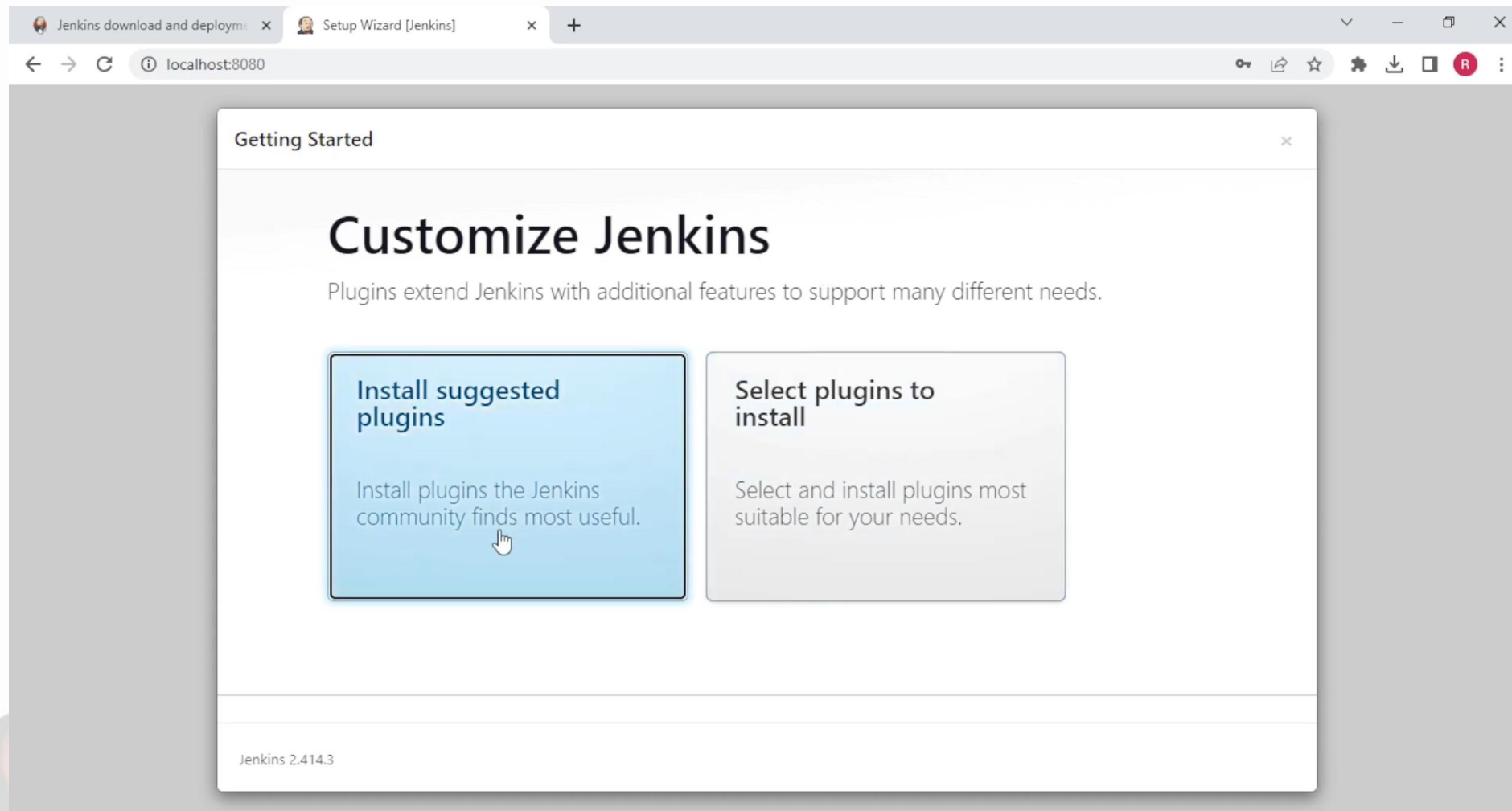
`C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword`

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

Administrator password

Continue

We need to enter Administrator Password, copied in local files, then click on continue.



After Successful install of Jenkins, we have to navigate to <http://localhost:8888/>
And you will see this Getting Started window, then after click on
Install Suggested Plugins

Getting Started

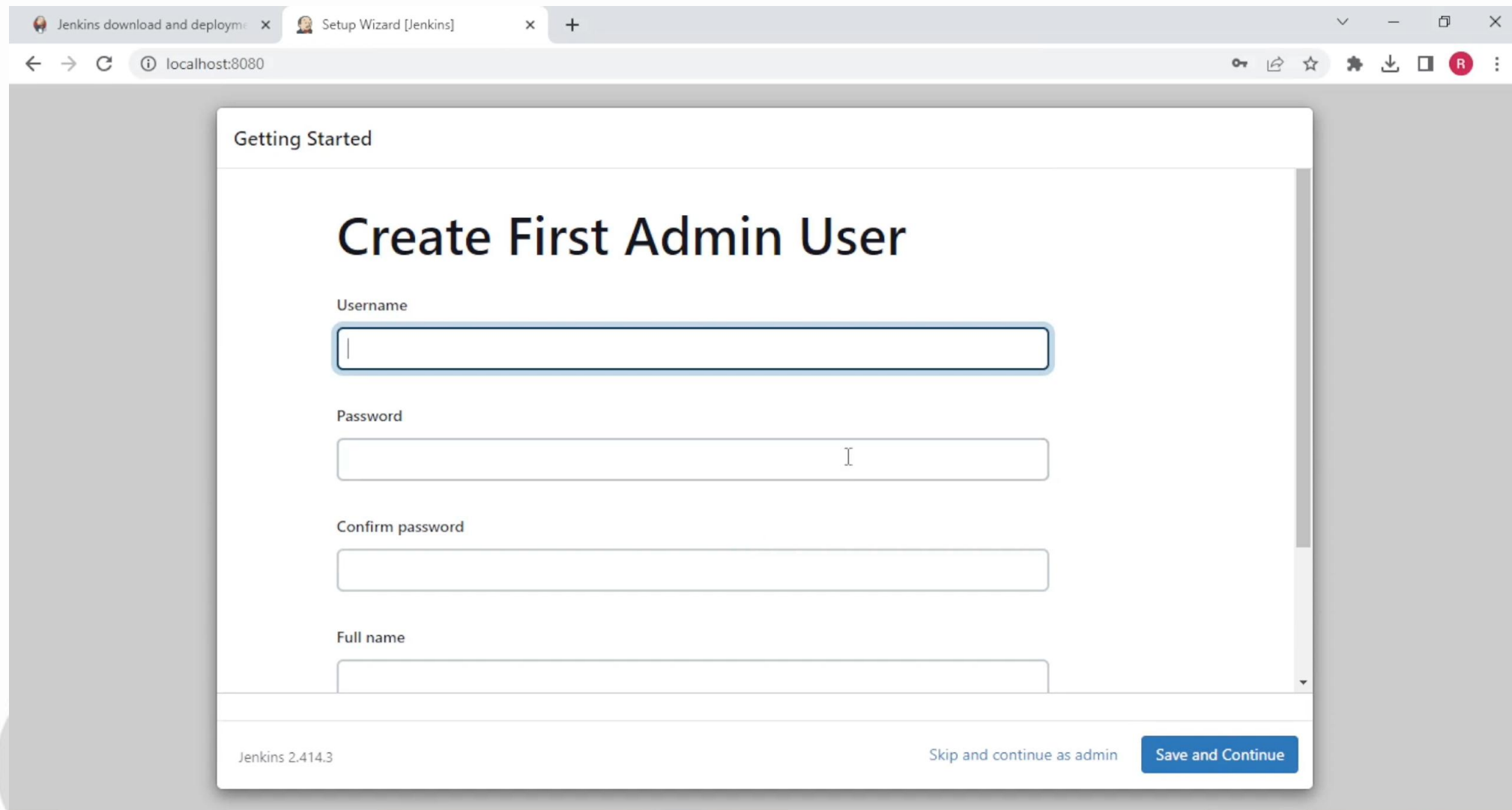
✓ Folders	✓ OWASP Markup Formatter	✓ Build Timeout	✓ Credentials Binding
✓ Timestampers	✓ Workspace Cleanup	✓ Ant	✓ Gradle
✓ Pipeline	✓ GitHub Branch Source	✓ Pipeline: GitHub Groovy Libraries	✓ Pipeline: Stage View
✓ Git	✓ SSH Build Agents	✓ Matrix Authorization Strategy	⌚ PAM Authentication
⌚ LDAP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

Dependency tree:

- ** Mina SSHD API :: Common
- ** Mina SSHD API :: Core
- ** Git client
- ** Pipeline: Input Step
- ** Pipeline: Declarative
- Pipeline
- ** Java JSON Web Token (JJWT)
- ** OkHttp
- ** GitHub API
- Git
- ** GitHub
- GitHub Branch Source
- Pipeline: GitHub Groovy Libraries
- ** Pipeline Graph Analysis
- ** Pipeline: REST API
- Pipeline: Stage View
- Git
- SSH Build Agents
- Matrix Authorization Strategy
- ** - required dependency

Jenkins 2.414.3

in this window all required Plugins for Jenkins will be downloaded here.



The screenshot shows a web browser window with two tabs: 'Jenkins download and deployment' and 'Setup Wizard [Jenkins]'. The address bar shows 'localhost:8080'. The main content area is titled 'Getting Started' and features a large heading 'Create First Admin User'. Below the heading are four input fields: 'Username', 'Password', 'Confirm password', and 'Full name'. The 'Username' field is currently active, indicated by a blue border. At the bottom of the form, there is a footer area with 'Jenkins 2.414.3' on the left, a link 'Skip and continue as admin' in the center, and a blue 'Save and Continue' button on the right.

Getting Started

Create First Admin User

Username

Password

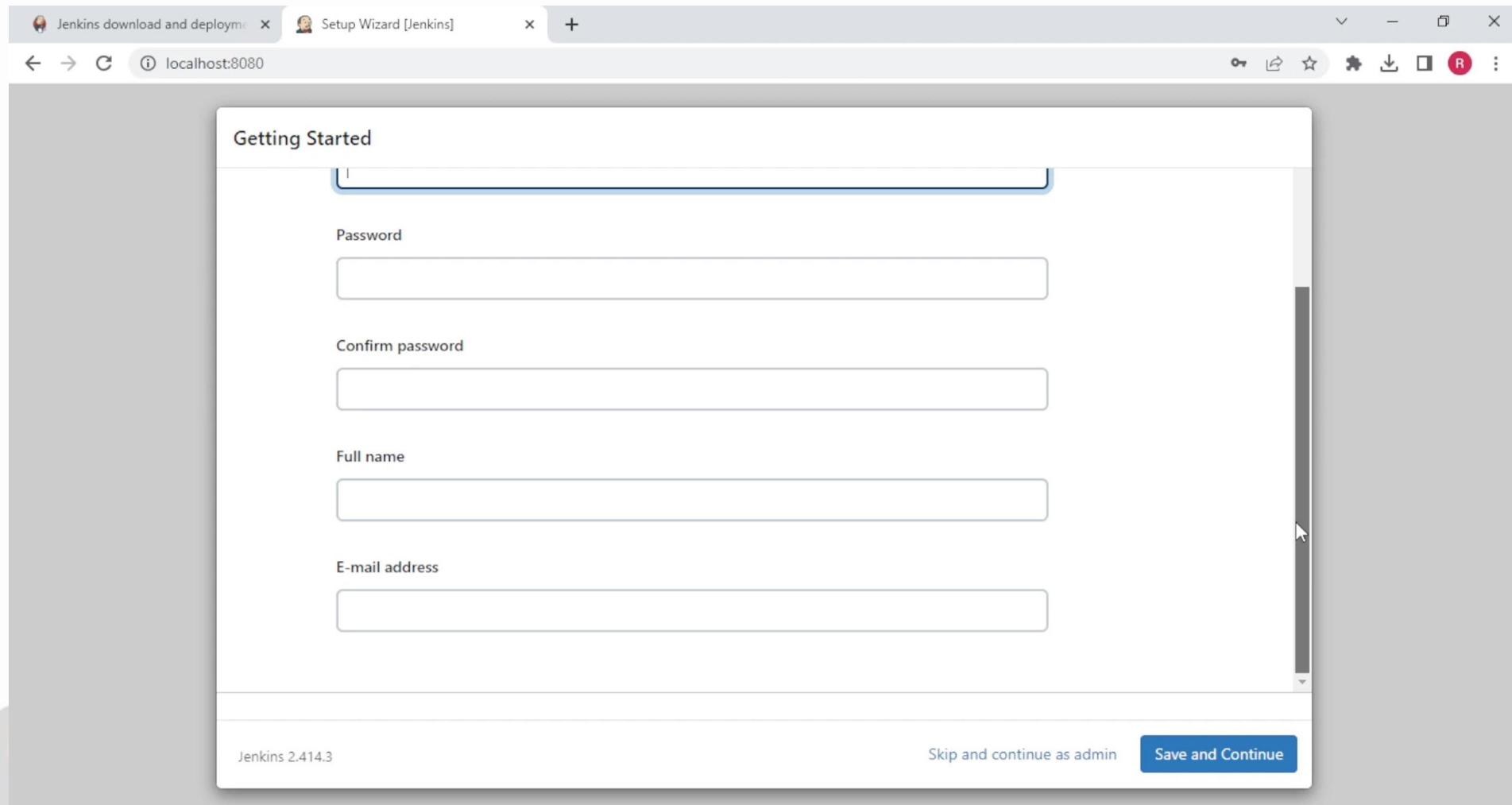
Confirm password

Full name

Jenkins 2.414.3

[Skip and continue as admin](#) [Save and Continue](#)

We need to enter Username, password to create Jenkins account.



The screenshot shows a web browser window with two tabs: 'Jenkins download and deployment' and 'Setup Wizard [Jenkins]'. The address bar shows 'localhost:8080'. The main content area displays the 'Getting Started' setup wizard for Jenkins 2.414.3. The form includes fields for Username, Password, Confirm password, Full name, and E-mail address. At the bottom, there are two buttons: 'Skip and continue as admin' and 'Save and Continue'.

Getting Started

Username

Password

Confirm password

Full name

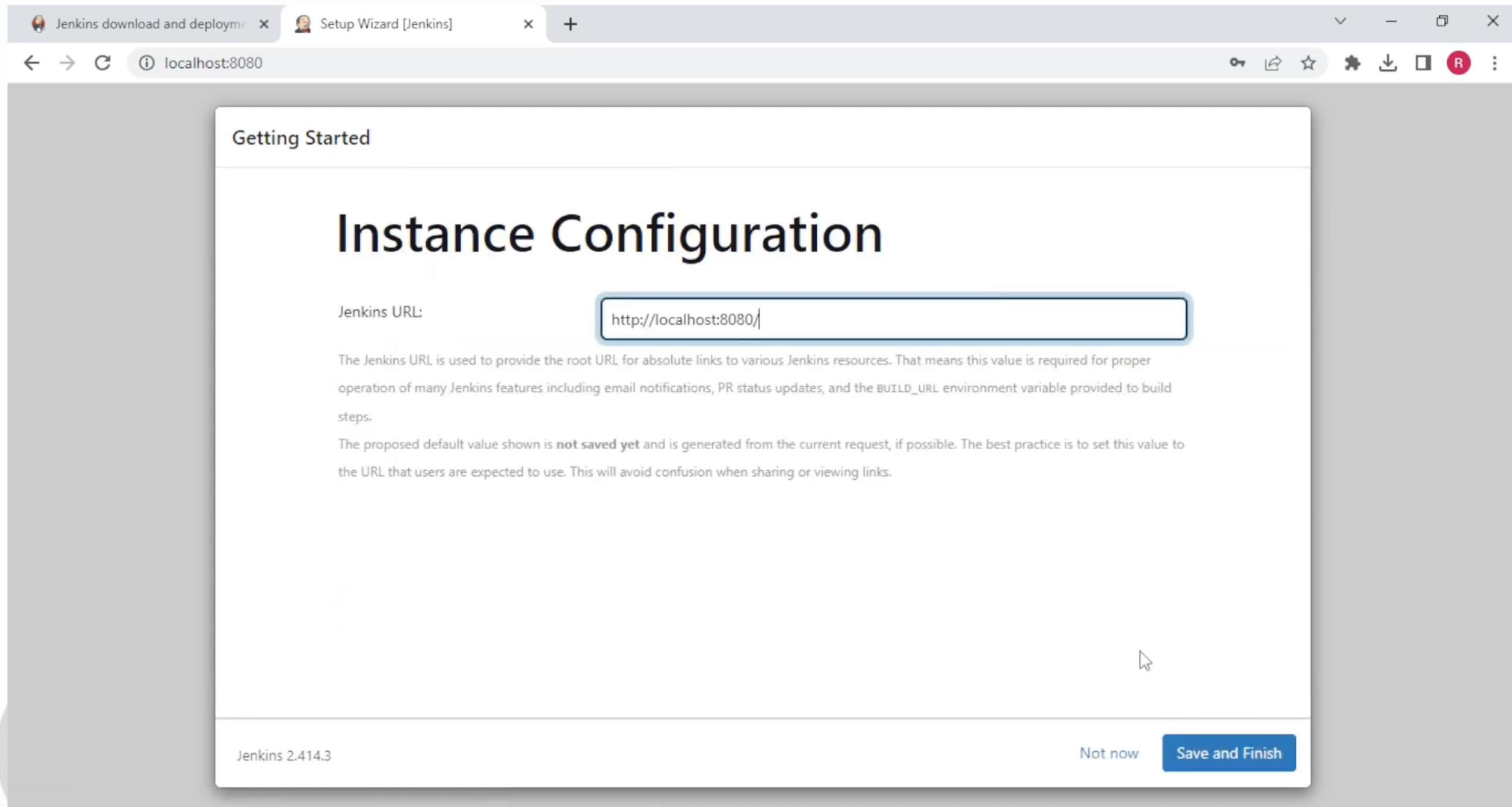
E-mail address

Jenkins 2.414.3

Skip and continue as admin

Save and Continue

We have to enter Username, Password, Full name, Email to Login to Jenkins Account. Then Click on Save and Continue. *remember Credentials.



The screenshot shows a web browser window with two tabs: 'Jenkins download and deployment' and 'Setup Wizard [Jenkins]'. The address bar shows 'localhost:8080'. The main content area is titled 'Getting Started' and 'Instance Configuration'. It features a text input field for 'Jenkins URL' containing 'http://localhost:8080/'. Below the input field, there is explanatory text about the Jenkins URL and its importance. At the bottom of the window, there are two buttons: 'Not now' and 'Save and Finish'.

Getting Started

Instance Configuration

Jenkins URL:

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.

Jenkins 2.414.3

Not now [Save and Finish](#)

This window will be displayed Instance Configuration, here we need to set our URL, then click on Save and Finish.

[+ New Item](#)[Build History](#)

Build Queue



No builds in the queue.

[Build Executor Status](#)

0/2

[Add description](#)

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

[Create a job](#)

Set up a distributed build

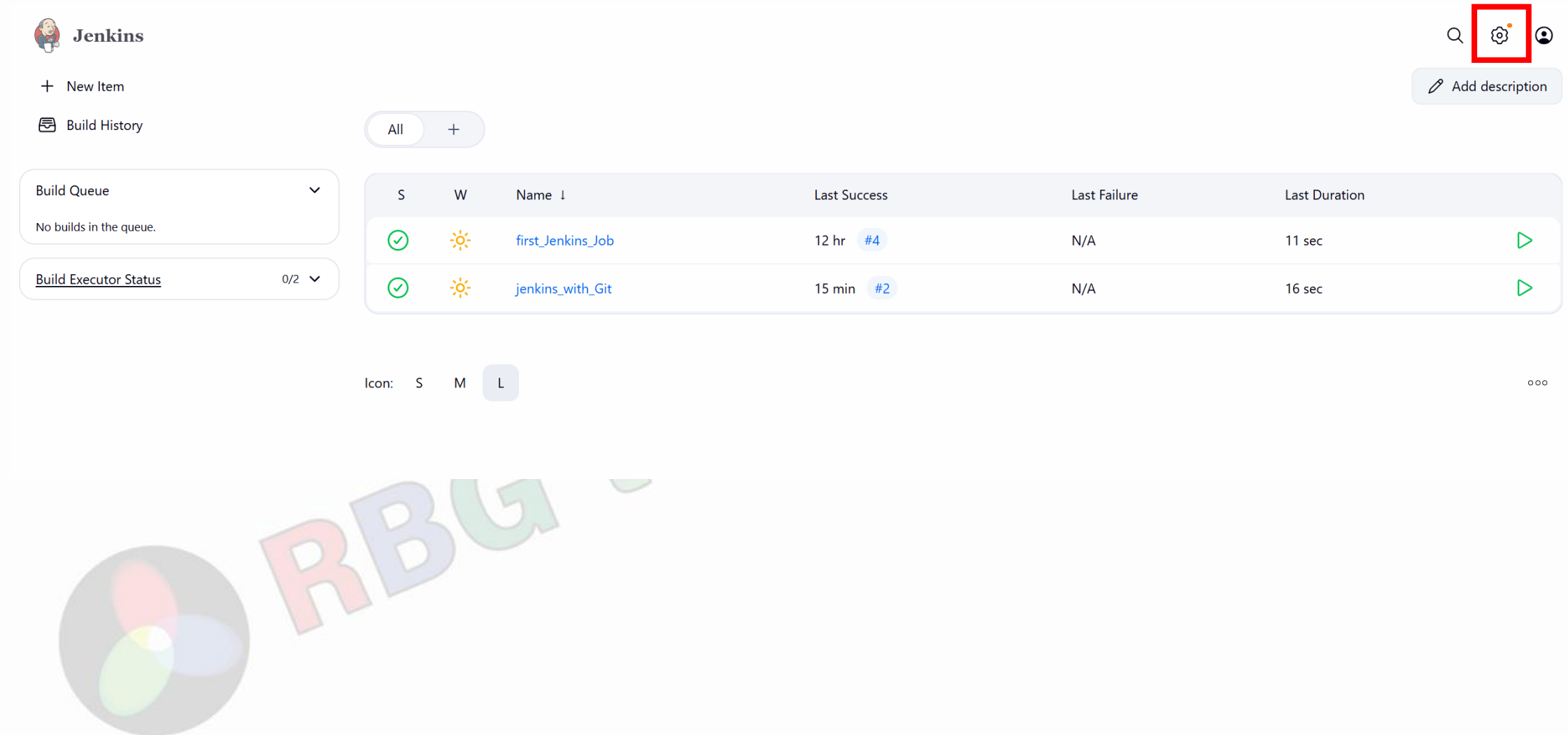
[Set up an agent](#)[Configure a cloud](#)[Learn more about distributed builds](#)[REST API](#)

Jenkins 2.528.1

After Account creation & Setup, we will be Re-Directed to Jenkins **Homepage**.

How to Create Jenkins Job

Using Local Maven Project



The screenshot shows the Jenkins dashboard. At the top left is the Jenkins logo and name. Below it are links for '+ New Item' and 'Build History'. On the right, there are icons for search, settings (highlighted with a red box), and a user profile. Below the settings icon is a button labeled 'Add description'. On the left side, there are two panels: 'Build Queue' showing 'No builds in the queue.' and 'Build Executor Status' showing '0/2'. The main area is a table with columns: S, W, Name ↓, Last Success, Last Failure, Last Duration, and a play button icon. The table contains two rows of build data.

S	W	Name ↓	Last Success	Last Failure	Last Duration	
✓	☀	first_Jenkins_Job	12 hr #4	N/A	11 sec	▶
✓	☀	jenkins_with_Git	15 min #2	N/A	16 sec	▶

Below the table, there is a section for 'Icon' with buttons for 'S', 'M', and 'L' (selected). To the right of these buttons is a three-dot menu icon.

Before going to create a job, Firstly we have to add Plugin for Maven for that we have to click on settings,



Jenkins

/ Manage Jenkins ▾ / Plugins



Plugins



Update



Updates

26



Available plugins



Installed plugins



Advanced settings



Name ↓

Released

Installed

Health

**Maven Integration plugin** 3.27

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.

[Report an issue with this plugin](#)


100

Maven Release Plug-in Plug-in 0.16.4

A plug-in that enables you to perform releases using the [maven-release-plugin](#) from Jenkins.

[Report an issue with this plugin](#)

Search for Maven Integration & select that and click on install

 **Jenkins**

[+ New Item](#)

[Build History](#)

Build Queue

No builds in the queue.

Build Executor Status

0/2

REST API

Jenkins 2.528.1

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job

Set up a distributed build

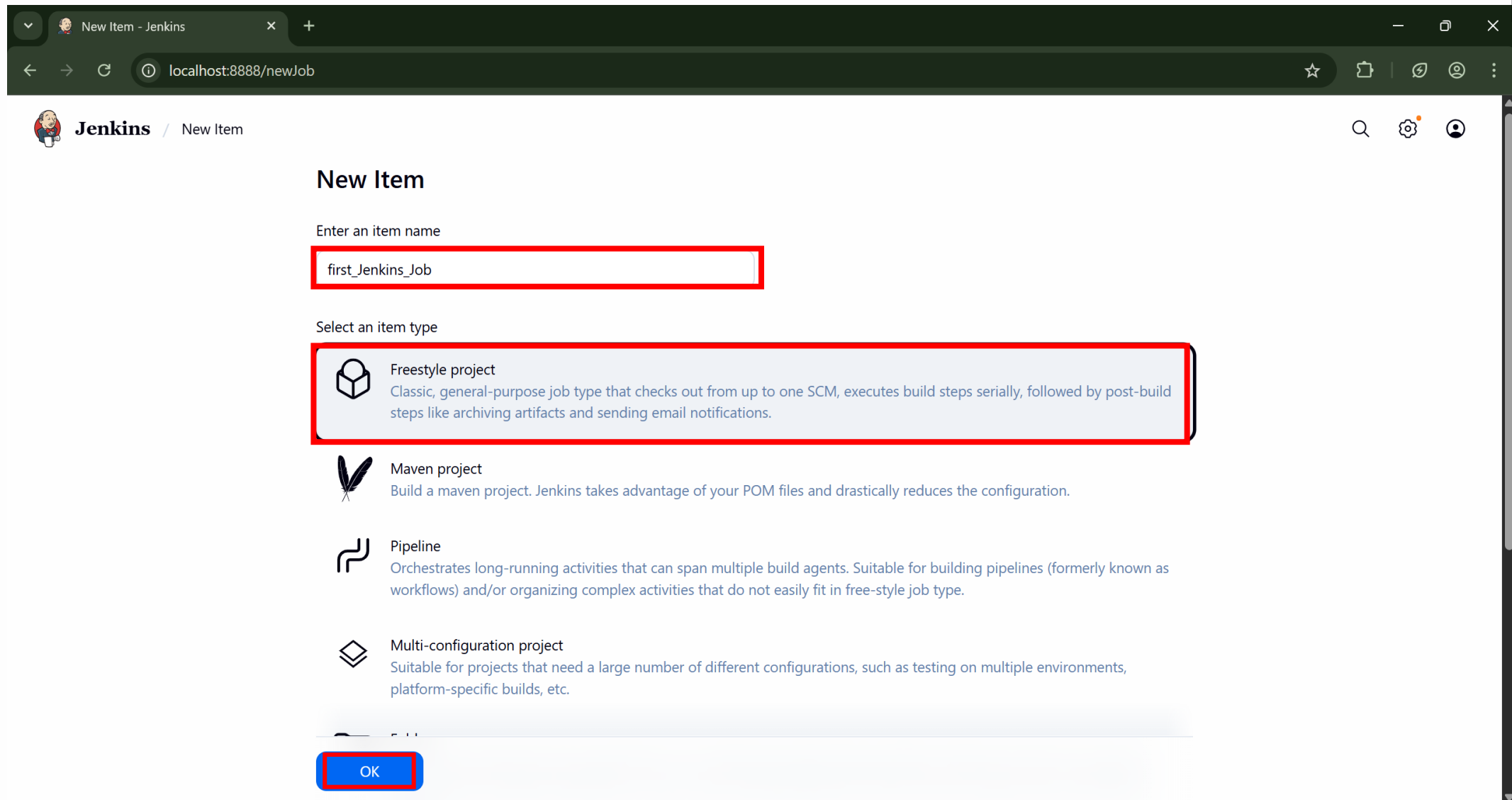
Set up an agent

Configure a cloud

Learn more about distributed builds

[Add description](#)

Click on Create a job.



New Item - Jenkins

localhost:8888/newJob





Jenkins / New Item

New Item

Enter an item name

first_Jenkins_Job

Select an item type

-  **Freestyle project**
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
-  **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.

OK

Firstly, we have to give a **name** for Jenkins Job.
Then select **Freestyle project**.
Click on **Ok**.

The screenshot shows the Jenkins web interface in a browser. The address bar indicates the URL is `localhost:8888/job/first_Jenkins_Job/configure`. The page title is "first_Jenkins_Job Config - Jenkins". The main content area is titled "Configure" and "General". On the left, there is a sidebar with navigation links: "General" (selected), "Source Code Management", "Triggers", "Environment", "Build Steps", and "Post-build Actions". The "General" section is active, showing a "Description" text area, a "Plain text" label, and a "Preview" link. Below these are several checkboxes: "Discard old builds", "GitHub project", "This project is parameterized", "Throttle builds", and "Execute concurrent builds if necessary". The "Advanced" dropdown menu is highlighted with a red box. At the bottom, there are "Save" and "Apply" buttons. The status "Enabled" with a toggle switch is visible in the top right corner.

Jenkins / first_Jenkins_Job / Configuration

Configure

General

Enabled

General

Source Code Management

Triggers

Environment

Build Steps

Post-build Actions

Description

Plain text [Preview](#)

☐ Discard old builds ?

☐ GitHub project

☐ This project is parameterized ?

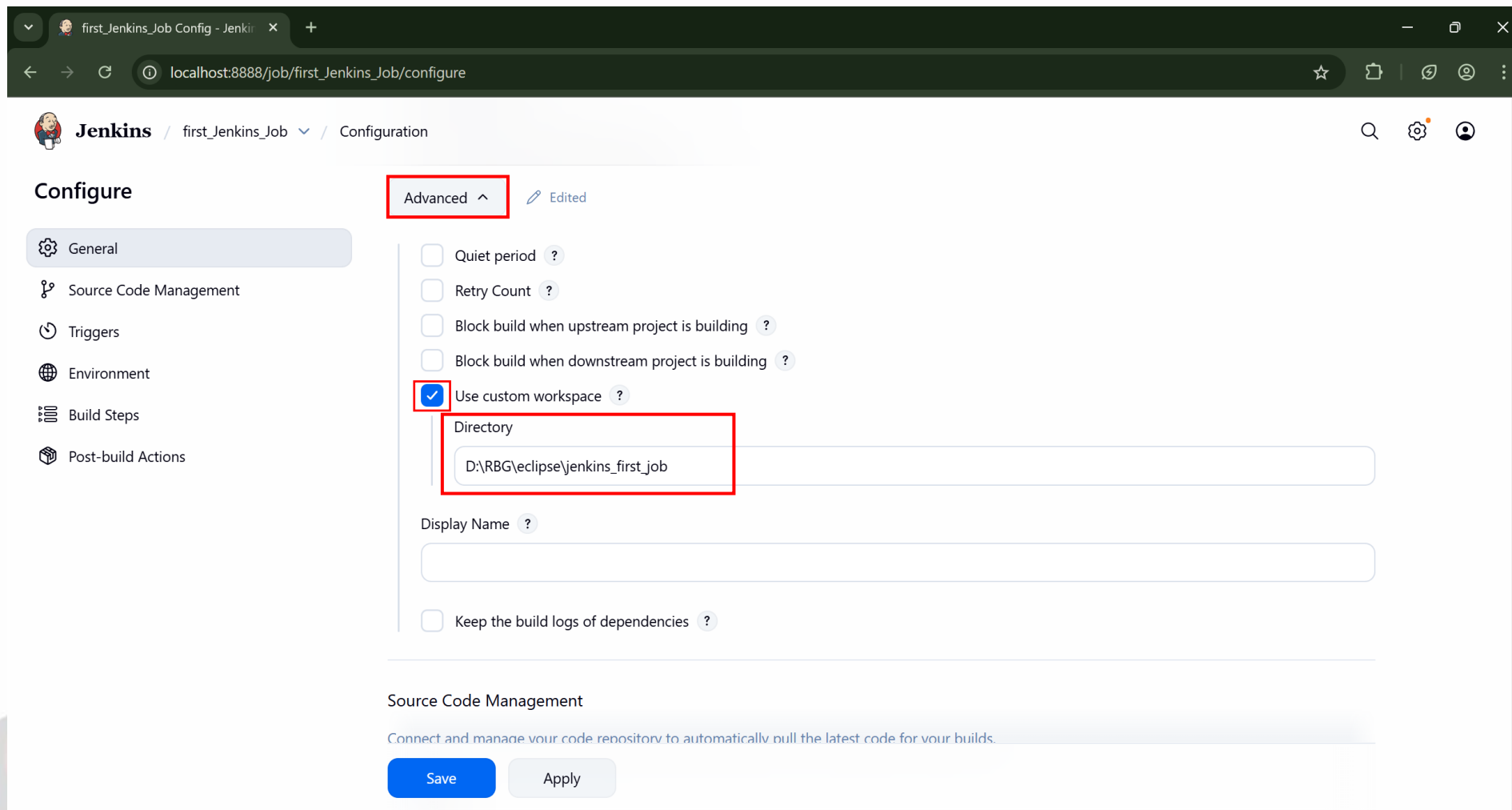
☐ Throttle builds ?

☐ Execute concurrent builds if necessary ?

Advanced

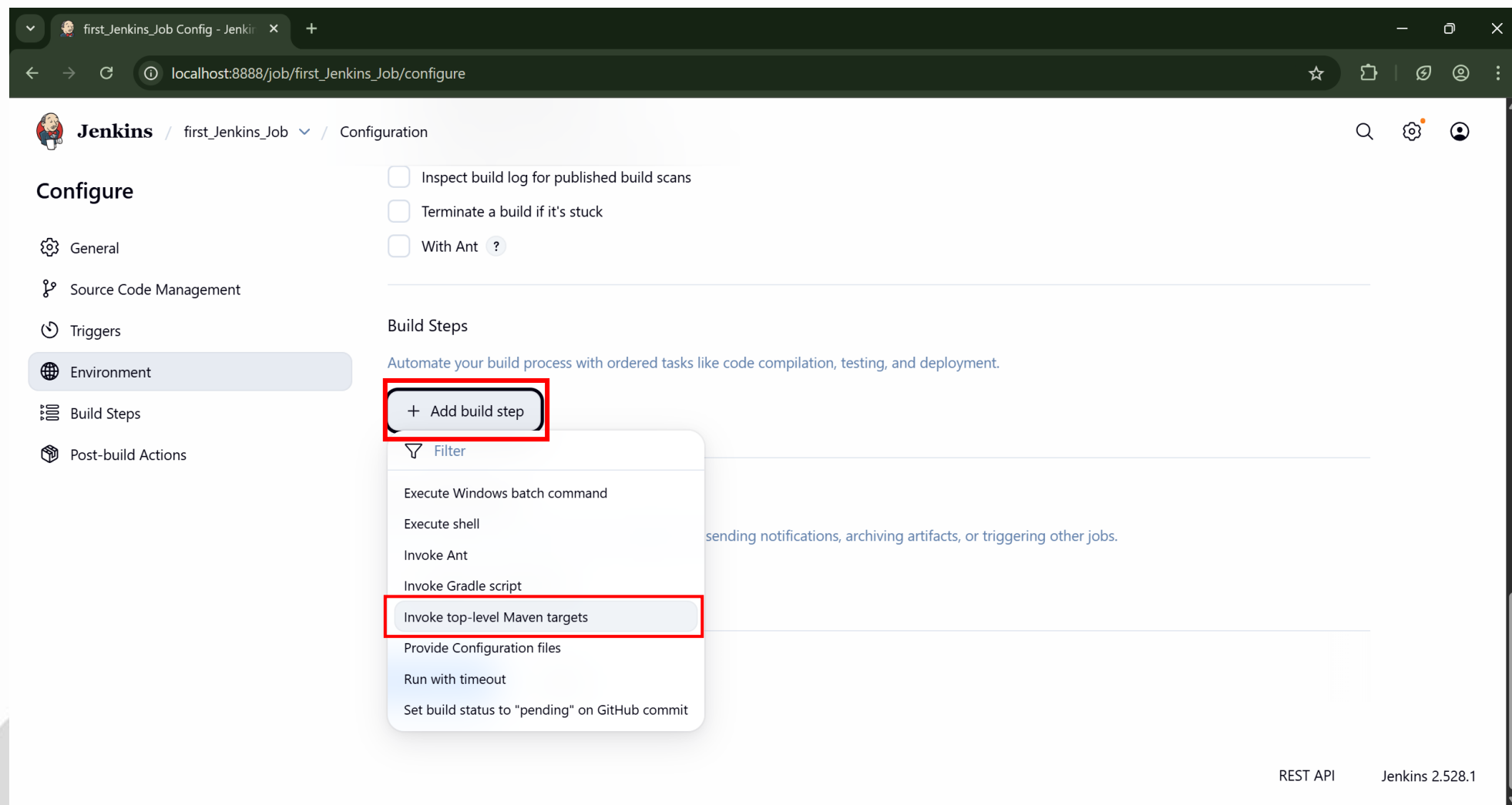
Save Apply

We will see this after clicking on Ok in before slide.



The screenshot shows the Jenkins web interface in a browser. The address bar indicates the URL is `localhost:8888/job/first_Jenkins_Job/configure`. The page title is "first_Jenkins_Job Config - Jenkins". The breadcrumb navigation shows "Jenkins / first_Jenkins_Job / Configuration". The left sidebar contains a "Configure" section with a list of tabs: "General", "Source Code Management", "Triggers", "Environment", "Build Steps", and "Post-build Actions". The "Advanced" tab is selected and highlighted with a red box. Below the "Advanced" tab, there is a list of checkboxes: "Quiet period", "Retry Count", "Block build when upstream project is building", "Block build when downstream project is building", "Use custom workspace", and "Keep the build logs of dependencies". The "Use custom workspace" checkbox is checked and highlighted with a red box. Below this checkbox, there is a "Directory" label and a text input field containing the path `D:\RBG\eclipse\jenkins_first_job`, which is also highlighted with a red box. Below the "Directory" field is a "Display Name" field. At the bottom of the "Advanced" section, there is a "Save" button and an "Apply" button. Below the "Advanced" section, there is a "Source Code Management" section with a description: "Connect and manage your code repository to automatically pull the latest code for your builds."

Select "use custom workspace" checkbox,
Then paste your maven project file location in directory. That should contain
pom.xml in the folder



The screenshot shows the Jenkins web interface for configuring a job named 'first_Jenkins_Job'. The left sidebar contains navigation links: 'Configure', 'General', 'Source Code Management', 'Triggers', 'Environment' (selected), 'Build Steps', and 'Post-build Actions'. The main content area is titled 'Configuration' and includes several checkboxes for build options: 'Inspect build log for published build scans', 'Terminate a build if it's stuck', and 'With Ant'. Below these is the 'Build Steps' section, which has a description: 'Automate your build process with ordered tasks like code compilation, testing, and deployment.' A red box highlights the '+ Add build step' button. A dropdown menu is open below this button, listing various build actions. The option 'Invoke top-level Maven targets' is highlighted with a red box. Other options in the menu include 'Filter', 'Execute Windows batch command', 'Execute shell', 'Invoke Ant', 'Invoke Gradle script', 'Provide Configuration files', 'Run with timeout', and 'Set build status to "pending" on GitHub commit'. The bottom right corner of the page shows 'REST API' and 'Jenkins 2.528.1'.

In Build step, we have to click on "+Add build Step", & Then select "Invoke top-level Maven targets".

Configure

- General
- Source Code Management
- Triggers
- Environment
- Build Steps**
- Post-build Actions

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Invoke top-level Maven targets ?

Maven Version

Maven_3.9.9

Goals

clean test

Advanced ^

✎ Edited

POM ?

pom.xml

Properties ?

JVM Options ?

☐ Inject build variables ?☐ Use private Maven repository


Save

Apply





Here in Goals, we have to enter “clean test” and then after mention POM file > pom.xml then click on Save.

The screenshot shows the Jenkins web interface in a browser. The address bar displays 'localhost:8888/job/first_Jenkins_Job/'. The page title is 'first_Jenkins_Job'. On the left sidebar, the 'Build Now' button is highlighted with a red rectangle. The main content area shows the job name 'first_Jenkins_Job' and a section titled 'Permalinks'. Below this, there is a 'Builds' section that currently displays 'No builds'. At the bottom right, the text 'REST API' and 'Jenkins 2.528.1' is visible.

After clicking on save, will redirect to this page, In this we have option **Build Now**, click Build now to run your First Jenkins Job.


 **Jenkins** / first_Jenkins_Job

- Status
- Changes
- Workspace
- Build Now**
- Configure
- Delete Project
- Rename
- Credentials




Builds ... 
Today
#4 10:35 AM   

first_Jenkins_Job

Permalinks

 **Jenkins** / first_Jenkins_Job

- Status
- Changes
- Workspace
- Build Now
- Configure
- Delete Project
- Rename
- Credentials

Builds ... 
Today
 #4 10:35 AM 

first_Jenkins_Job

Permalinks

After clicking on Build Now button, we will see the Builds down there, that's Running our Job.

After the job has run **Successfully**, that we can see in second image as Green tick mark

The screenshot displays the Jenkins web interface in a browser. The address bar shows the URL `localhost:8888/job/first_Jenkins_Job/4/`. The page header includes the Jenkins logo and the breadcrumb `first_Jenkins_Job / #4`. On the left sidebar, the `Console Output` link is highlighted with a red rectangle. The main content area shows build #4 as successful (green checkmark) with the timestamp `(Dec 10, 2025, 10:35:18 AM)`. It includes buttons for `Add description` and `Keep this build forever`. The build details show it was started by user `admin` and provides a breakdown of time: 5 ms waiting, 11 sec build duration, and 11 sec total from scheduled to completion. A `Timings` section indicates `No changes`. The footer shows `REST API` and `Jenkins 2.528.1`.

Here we can check status for the build, for output click on Console output.



```
[WARNING] source value 8 is obsolete and will be removed in a future release
[WARNING] target value 8 is obsolete and will be removed in a future release
[WARNING] To suppress warnings about obsolete options, use -Xlint:-options.
[INFO]
[INFO] --- surefire:3.2.5:test (default-test) @ TestCaseDemo ---
[INFO] Using auto detected provider org.apache.maven.surefire.testng.TestNGProvider
[INFO]
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running jenkins.TestCaseDemo.jenkinsTest
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.
Dec 10, 2025 10:35:27 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
WARNING: Unable to find CDP implementation matching 143
Dec 10, 2025 10:35:27 AM org.openqa.selenium.chromium.ChromiumDriver lambda$new$5
WARNING: Unable to find version of CDP to use for 143.0.7499.41. You may need to include a dependency on a specific version of the CDP using something
similar to `org.seleniumhq.selenium:selenium-devtools-v86:4.13.0` where the version ("v86") matches the version of the chromium-based browser you're
using and the version number of the artifact is the same as Selenium's.
Browser launched successfully ←
Facebook login page opened successfully ←
Browser closed successfully ←
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 4.817 s -- in jenkins.TestCaseDemo.jenkinsTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 9.172 s
[INFO] Finished at: 2025-12-10T10:35:29+05:30
[INFO] -----
Finished: SUCCESS ←
```

www.slf4j.org/codes.html#StaticLoggerBinder

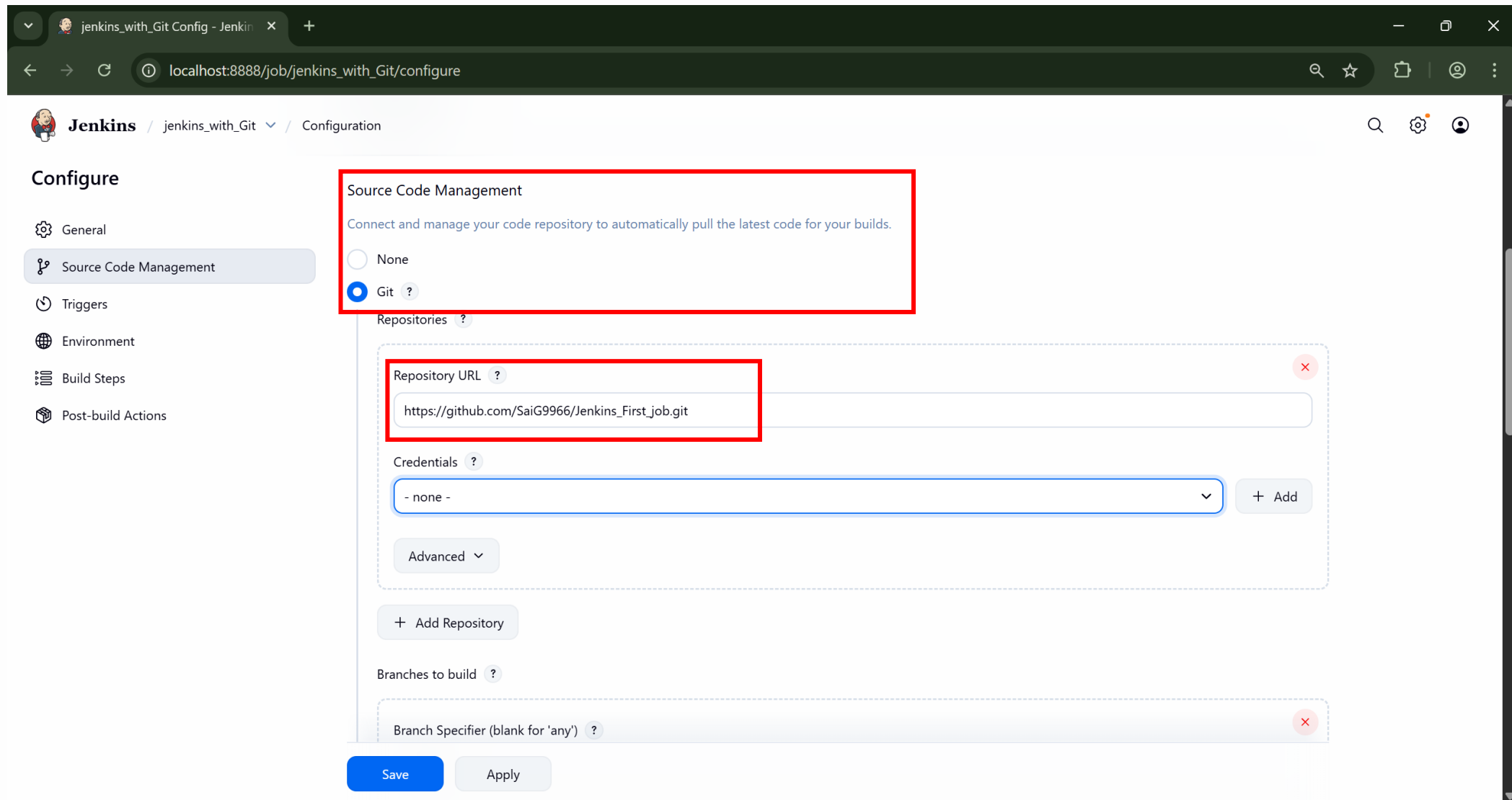
Here we can see the Output for Jenkins Job. This is the process to create and run job in Jenkins using local maven project.

How to Create Jenkins Job

Using Git Maven Project

The screenshot shows the Jenkins web interface in a browser. The address bar indicates the URL is `localhost:8888/job/first_Jenkins_Job/configure`. The page title is "first_Jenkins_Job Config - Jenkins". The main header shows the Jenkins logo and the breadcrumb "first_Jenkins_Job / Configuration". On the right, there are search, settings, and user icons. The left sidebar is titled "Configure" and lists several options: General (selected), Source Code Management, Triggers, Environment, Build Steps, and Post-build Actions. The main content area is titled "General" and has an "Enabled" toggle switch. It contains a "Description" text area, a "Plain text" label with a "Preview" link, and five unchecked checkboxes: "Discard old builds", "GitHub project", "This project is parameterized", "Throttle builds", and "Execute concurrent builds if necessary". Each checkbox has a help icon. Below these is an "Advanced" dropdown menu. At the bottom, there are "Save" and "Apply" buttons.

Here we can see the Output for Jenkins Job. This is the process to create and run job in Jenkins using git maven project. Scroll down to Source code management



The screenshot shows the Jenkins web interface for configuring a job named 'jenkins_with_Git'. The 'Source Code Management' tab is selected in the left sidebar. The 'Source Code Management' section is highlighted with a red box, showing the 'Git' option selected. Below this, the 'Repositories' section is also highlighted with a red box, showing a 'Repository URL' field with the value 'https://github.com/SaiG9966/Jenkins_First_job.git'. The 'Credentials' dropdown is set to '- none -'. At the bottom, there are 'Save' and 'Apply' buttons.

jenkins_with_Git Config - Jenkin x +

localhost:8888/job/jenkins_with_Git/configure

Jenkins / jenkins_with_Git Configuration

Configure

- General
- Source Code Management**
- Triggers
- Environment
- Build Steps
- Post-build Actions

Source Code Management

Connect and manage your code repository to automatically pull the latest code for your builds.

☐ None

☒ Git ?

Repositories ?

Repository URL ?

https://github.com/SaiG9966/Jenkins_First_job.git

Credentials ?

- none - + Add

Advanced ▾

+ Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

Save Apply

In Source Code Management, select Git to run Job, after in Repository we have to enter our Git maven project URL,

The screenshot shows the Jenkins web interface for configuring a job named 'jenkins_with_Git'. The 'Build Steps' section is selected in the left sidebar. In the 'Build Steps' area, the '+ Add build step' button is highlighted with a red rectangle. A dropdown menu is open below this button, listing various build steps. The option 'Invoke top-level Maven targets' is highlighted with a red rectangle. Other options in the dropdown include 'Execute Windows batch command', 'Execute shell', 'Invoke Ant', 'Invoke Gradle script', 'Provide Configuration files', 'Run with timeout', and 'Set build status to "pending" on GitHub commit'. The top of the page shows the browser address bar with 'localhost:8888/job/jenkins_with_Git/configure'. The bottom right corner displays 'REST API' and 'Jenkins 2.528.1'.

Here in Build Steps, we have to select Invoke top-level Maven Targets

The screenshot shows the Jenkins configuration interface for a job named 'jenkins_with_Git'. The left sidebar contains the following menu items: General, Source Code Management, Triggers, Environment, Build Steps (highlighted), and Post-build Actions. The main content area is titled 'Configure' and includes options to 'Terminate a build if it's stuck' and 'With Ant'. The 'Build Steps' section is active, showing a step named 'Invoke top-level Maven targets' with 'Maven Version' set to 'Maven_3.9.9' and 'Goals' set to 'clean test'. The 'Save' button at the bottom is highlighted with a red box.

Here we have to click on Build steps, and in Goals we have enter a command “clean test” and then click on Save.

The screenshot illustrates the Jenkins workflow for running a job and viewing its console output. On the left, the Jenkins dashboard for 'jenkins_with_Git' is shown. The 'Build Now' button is highlighted with a red box. Below it, the 'Builds' section shows a list of builds, with build '#2' at 10:20 PM highlighted by a red box. A red arrow points from this build to the right-hand screenshot. The right-hand screenshot shows the Jenkins interface for build '#2'. The 'Console Output' button is highlighted with a red box. To the right of this, the console output is displayed, showing the execution of 'jenkins.TestCaseDemo.jenkinsTest'. The output includes various log messages, warnings, and a final 'BUILD SUCCESS' status.

Builds

Build	Status	Time
#2	Success	10:20 PM
#1	Success	10:19 PM

Console Output

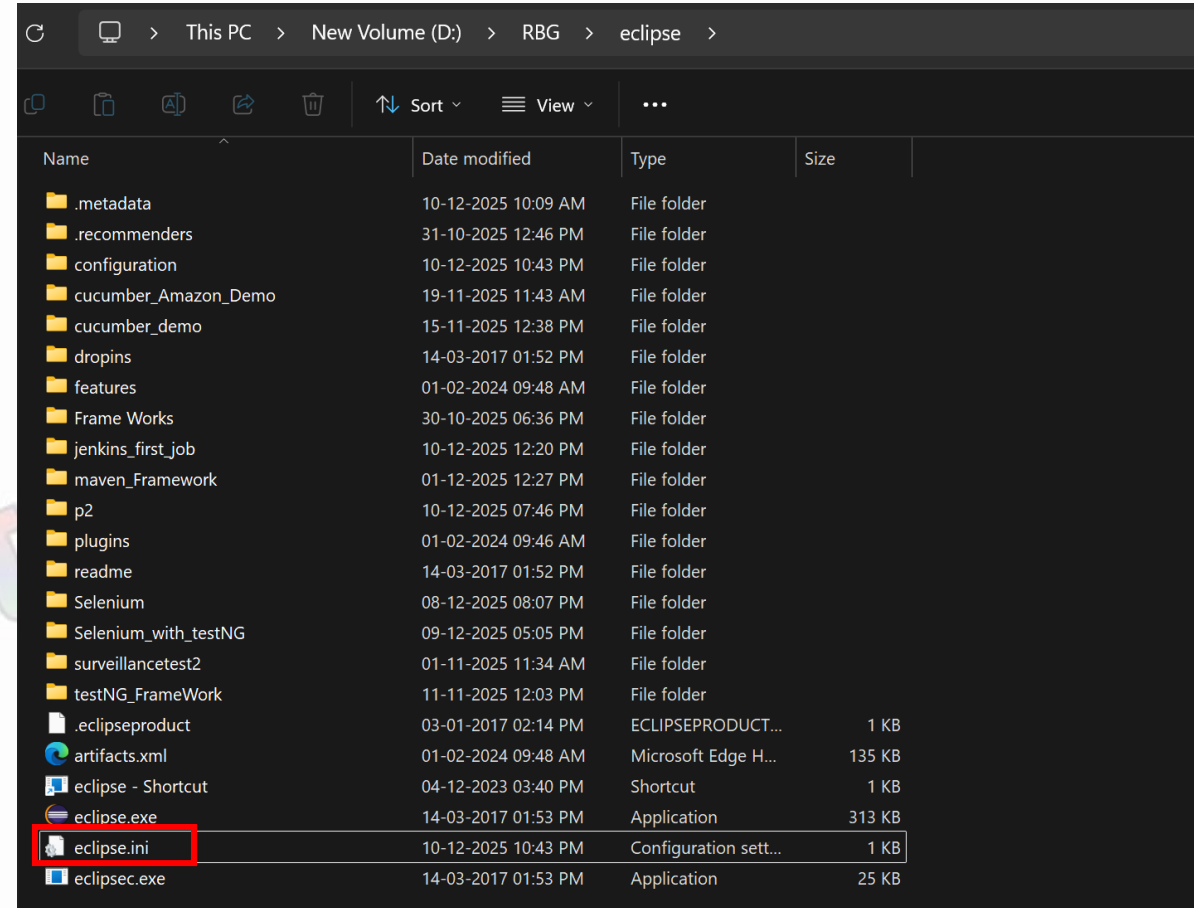
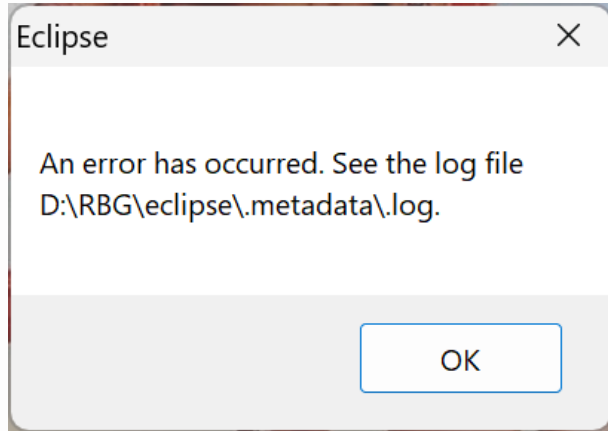
```
[INFO] T E S T S
[INFO] -----
[INFO] Running jenkins.TestCaseDemo.jenkinsTest
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Dec 10, 2025 10:20:54 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
WARNING: Unable to find CDP implementation matching 143
Dec 10, 2025 10:20:54 PM org.openqa.selenium.chromium.ChromiumDriver lambda$new$5
WARNING: Unable to find version of CDP to use for 143.0.7499.41. You may need to include a dependency on a specific version of the CDP using something
similar to 'org.seleniumhq.selenium:selenium-devtools-v86:4.13.0' where the version ("v86") matches the version of the chromium-based browser you're
using and the version number of the artifact is the same as Selenium's.
Browser launched successfully
Facebook login page opened successfully
Browser closed successfully
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 6.044 s -- in jenkins.TestCaseDemo.jenkinsTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 11.594 s
[INFO] Finished at: 2025-12-10T22:20:56+05:30
[INFO] -----
Finished: SUCCESS
```

After clicking on Save, we have to click on Build Now, the Job will run, and u can see the number of jobs run in Builds, click on the job and we can see console output click on that we going to see the Console output.

Tear Points

In case of Eclipse Error

After installing Java 17 or 21 we get an issue with eclipse, if you get this type of issue then follow these steps to follow



Navigate to your eclipse directory, then search for eclipse.ini file, open with note pad & type "-vm C:\Program Files\Java\jdk-1.8\jre\bin\server"

We have to mention JVM path here in the note pad above -vm args.

```
-startup
plugins/org.eclipse.equinox.launcher_1.3.201.v20161025-1711.jar
--launcher.library
plugins/org.eclipse.equinox.launcher.win32.win32.x86_64_1.1.401.v20161122-1740
-product
org.eclipse.epp.package.java.product
--launcher.defaultAction
openFile
-showsplash
org.eclipse.platform
--launcher.defaultAction
openFile
--launcher.appendVmargs
-vm
C:\Program Files\Java\jdk-1.8\jre\bin\server
-vmargs
-Dosgi.requiredJavaVersion=1.8
-XX:+UseG1GC
-XX:+UseStringDeduplication
-Dosgi.requiredJavaVersion=1.8
-Xms256m
-Xmx1024m
```

Above -vmargs, we have to paste the JVM path then save the notepad file, then your Eclipse will be Works Fine.

***locate JVM path from your local directory**

Thank You...

REGISTRATIONS NOW OPEN

THE CAREER COURSE FOR YOUR FUTURE FOR YOUR SUCCESS

AUTOMATION & MANUAL

CORE JAVA WITH SELENIUM



**Plot no: 623, Vivekananda Nagar,
Kukatpally, Hyderabad-500072**

LAND MARK NEAR UDUPI GRAND HOTEL.



Contact Us

+91 93986 94373