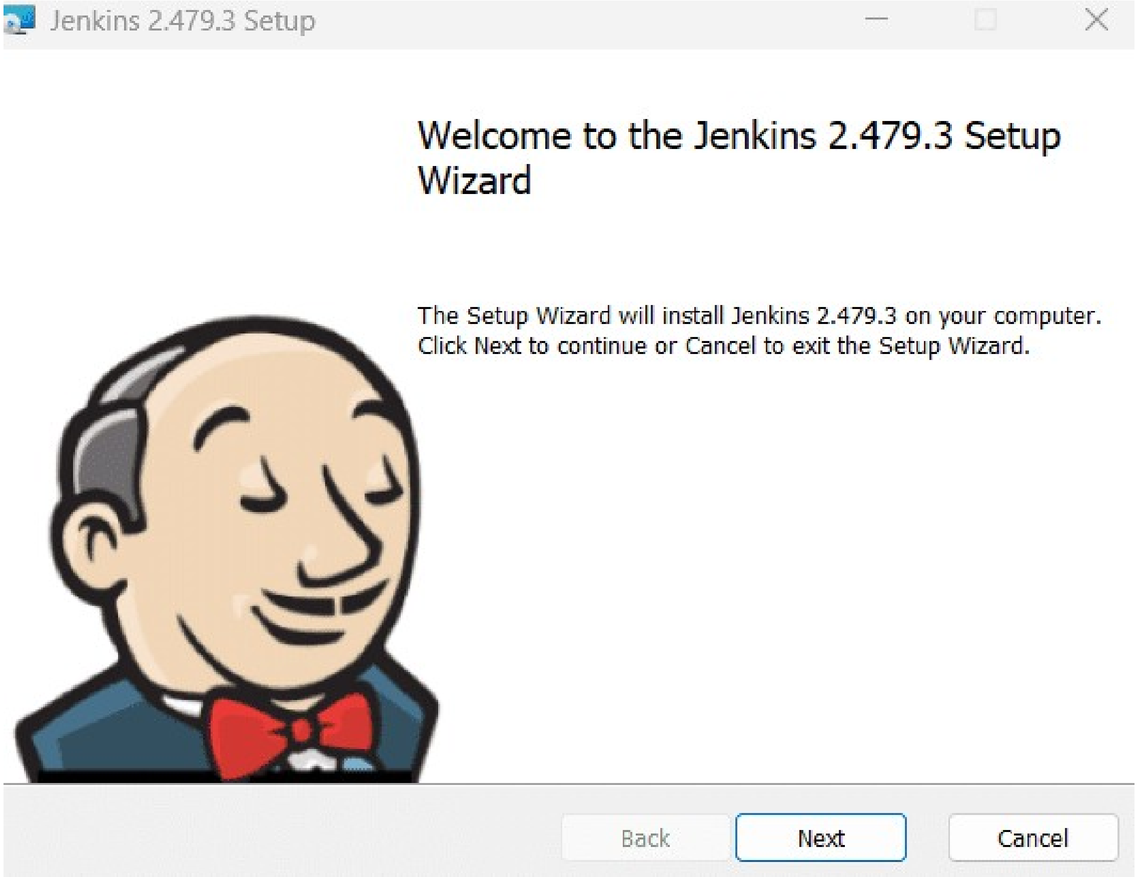
# Using Jenkins for CI/CD implementation

## 1. Download the Jenkins installer

To begin the Jenkins installation, visit the Jenkins Download page. There, you can choose between an LTS release and a weekly release for Windows.

After selecting the appropriate version, click the “Windows” option to download the installer package. Once the download is complete, run the installer to begin the installation process.



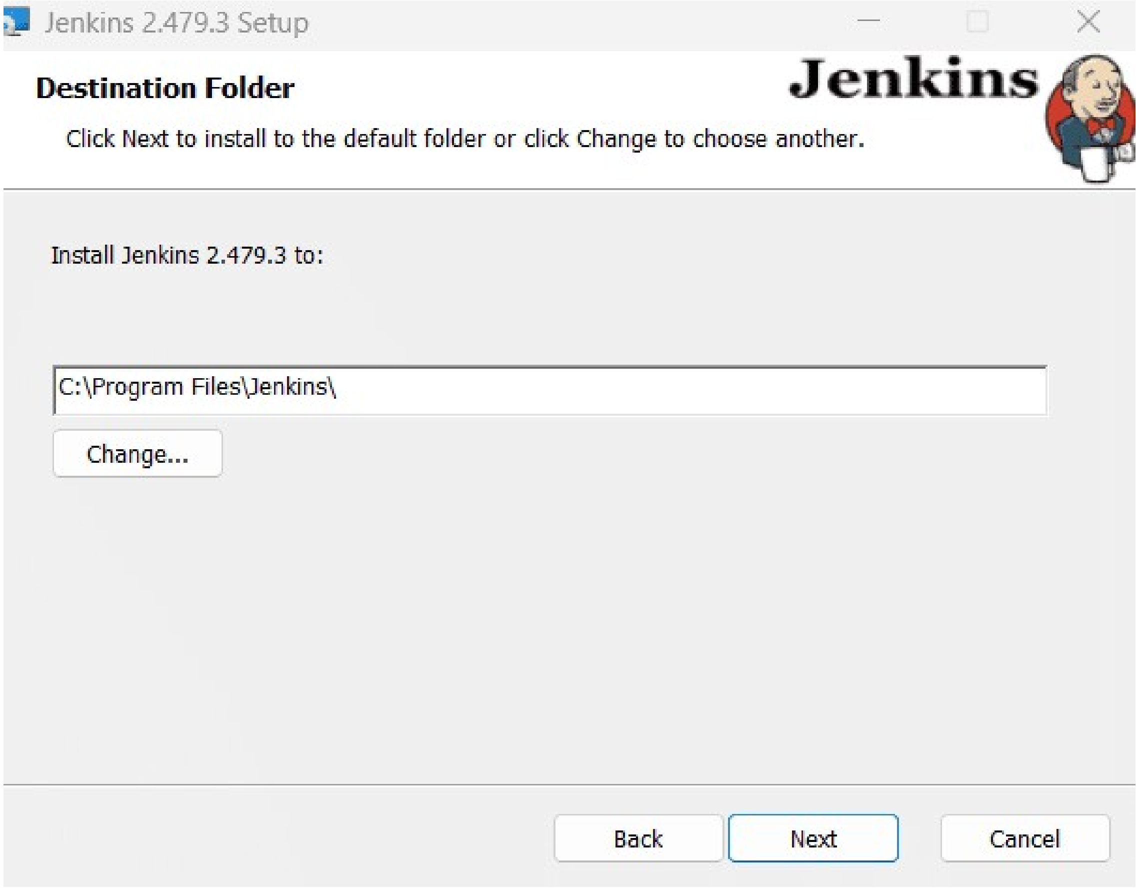
Jenkins 2.479.3 Setup

Welcome to the Jenkins 2.479.3 Setup Wizard

The Setup Wizard will install Jenkins 2.479.3 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

## 2. Initiate the setup wizard

Once you click *Next*, it directs you to specify the destination folder path.



Jenkins 2.479.3 Setup

**Destination Folder**

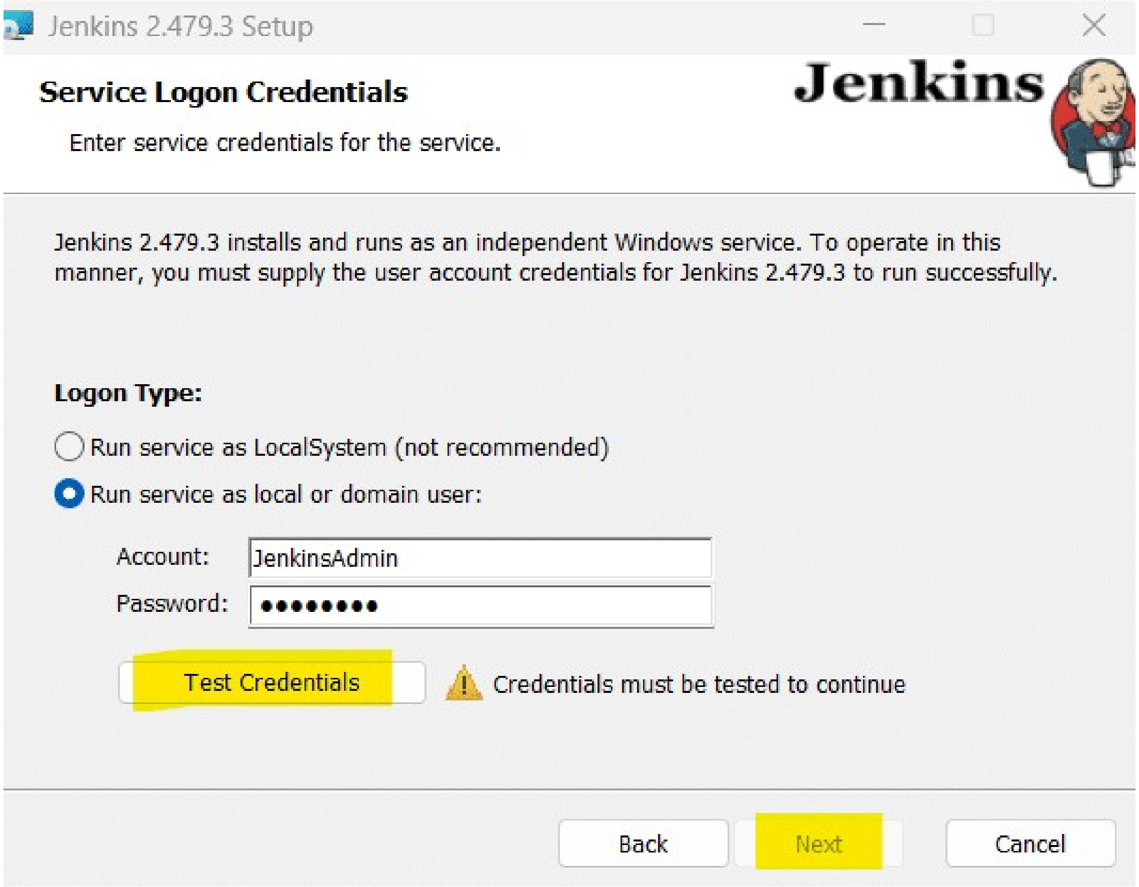
Click Next to install to the default folder or click Change to choose another.

Install Jenkins 2.479.3 to:

C:\Program Files\Jenkins\

**3. Configure installation directory and user settings**

Choose the directory where Jenkins should be installed and click *Next* to specify a local or domain user for running Jenkins. Click *Test Credentials* to validate permissions, and click *Next* after successful validation.



Jenkins 2.479.3 Setup

**Service Logon Credentials**

Enter service credentials for the service.

Jenkins 2.479.3 installs and runs as an independent Windows service. To operate in this manner, you must supply the user account credentials for Jenkins 2.479.3 to run successfully.

**Logon Type:**

○ Run service as LocalSystem (not recommended)

● Run service as local or domain user:

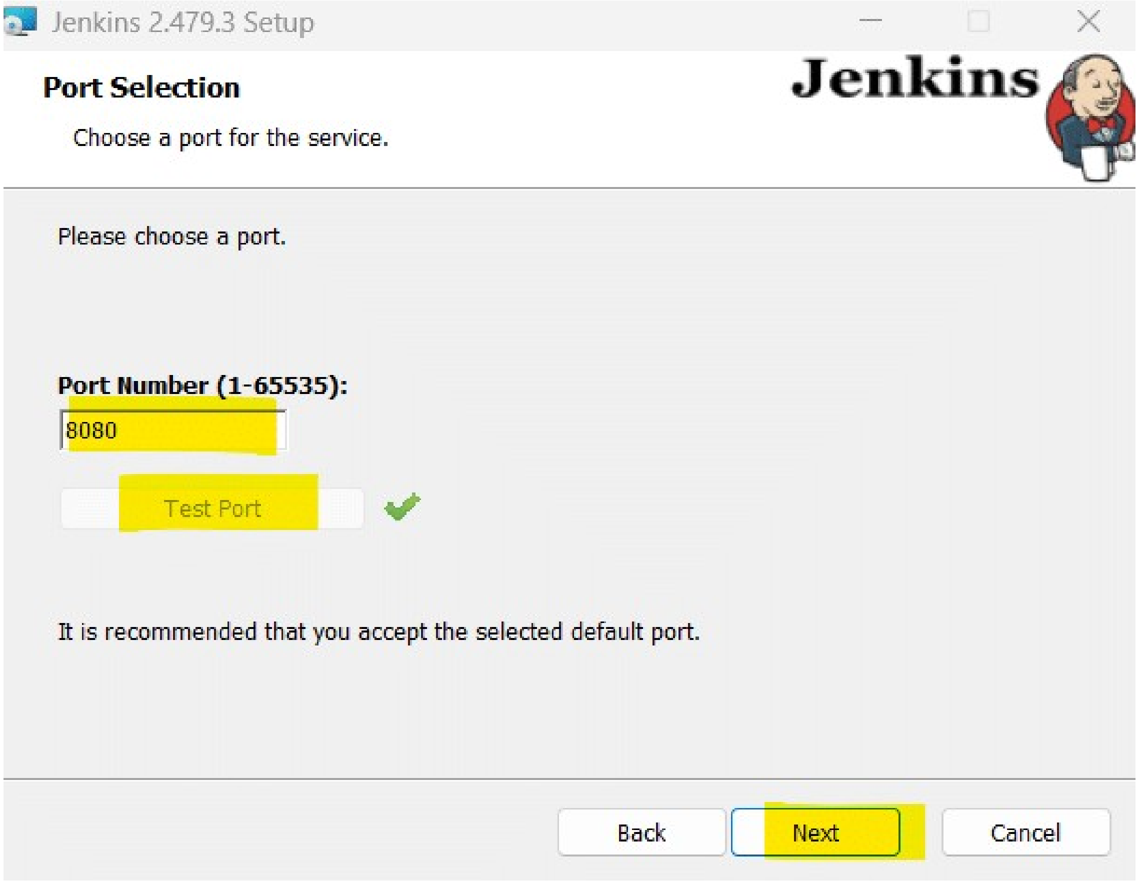
Account: JenkinsAdmin

Password: ●●●●●●●●●

⚠ Credentials must be tested to continue

**4. Port selection**

Choose the port Jenkins should listen on (default: 8080), click the *Test Port* option to ensure it is free, and choose *Next* to proceed.



Jenkins 2.479.3 Setup

Port Selection

Choose a port for the service.

Please choose a port.

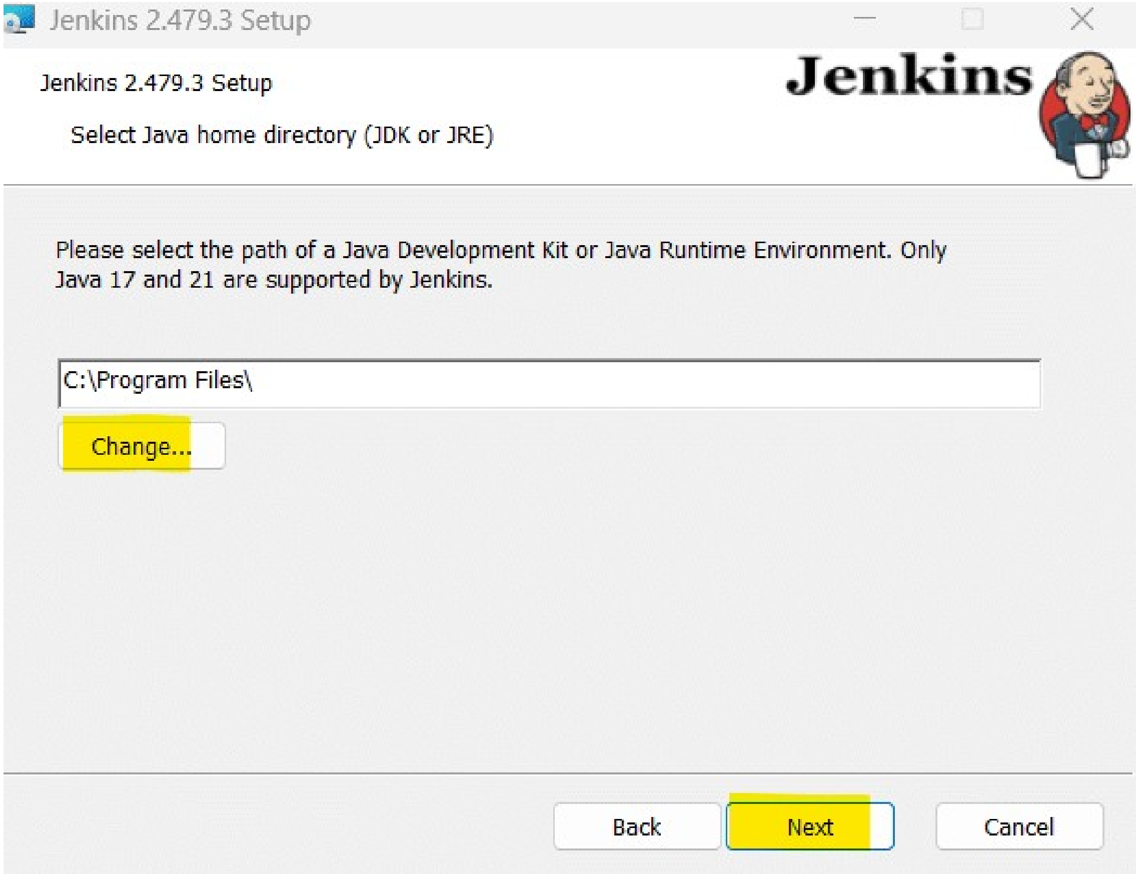
Port Number (1-65535):

8080

It is recommended that you accept the selected default port.

**5. Java home directory**

The installer detects your Java installation automatically. If Java is missing, install it and specify the Java home directory. Click *Next* to continue.



Jenkins 2.479.3 Setup

Jenkins 2.479.3 Setup

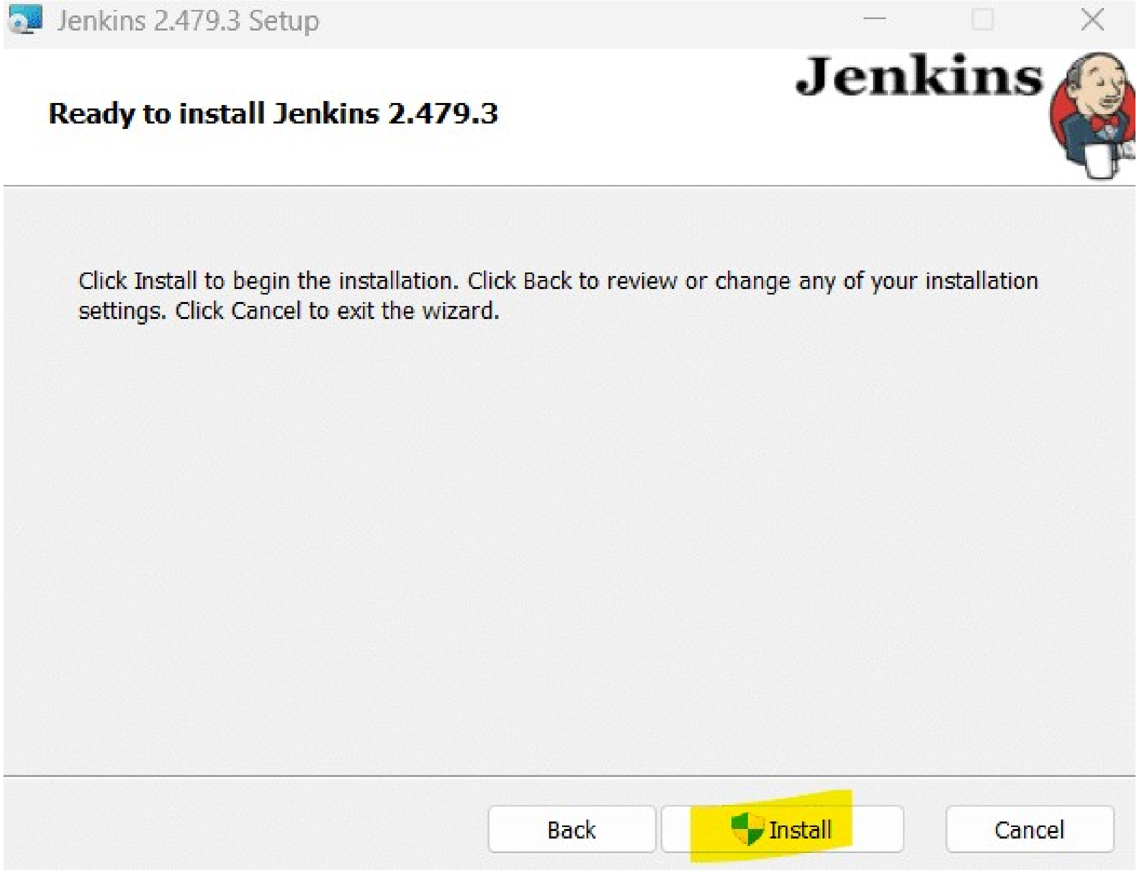
Select Java home directory (JDK or JRE)

Please select the path of a Java Development Kit or Java Runtime Environment. Only Java 17 and 21 are supported by Jenkins.

C:\Program Files\

### 6. Install Jenkins

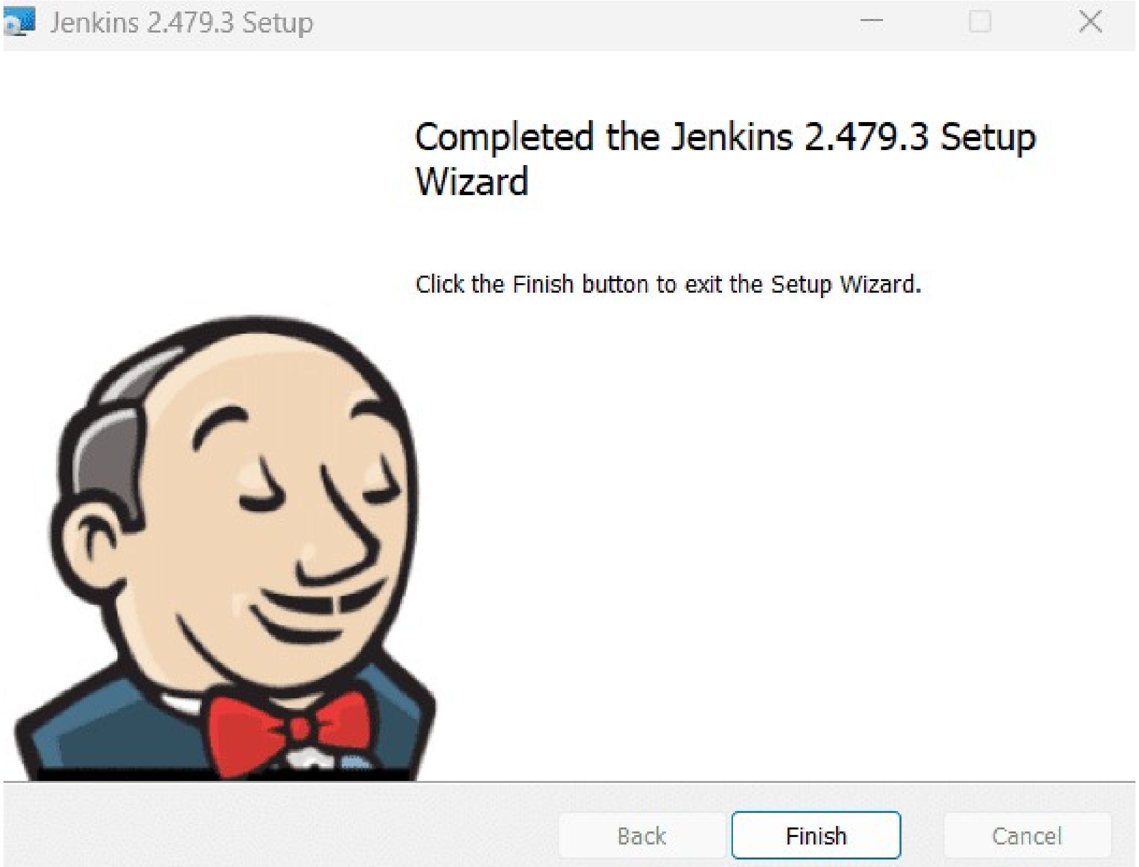
Click *Install* to start the installation and wait for the installation to complete.



**Ready to install Jenkins 2.479.3**

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

Click *Finish* when complete.

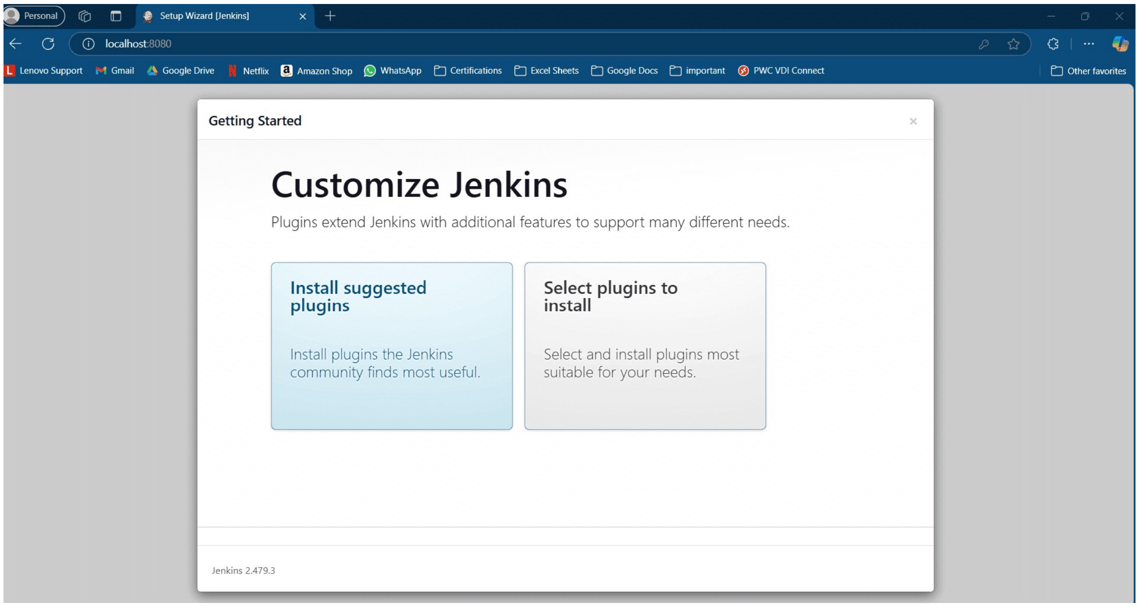


Completed the Jenkins 2.479.3 Setup Wizard

Click the Finish button to exit the Setup Wizard.

## 7. Post-installation validation

Open Windows Services and check if Jenkins is running to verify the installation. Otherwise, you can navigate to http://localhost:8080 in your browser for verification.



Getting Started

Customize Jenkins

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

Install suggested plugins

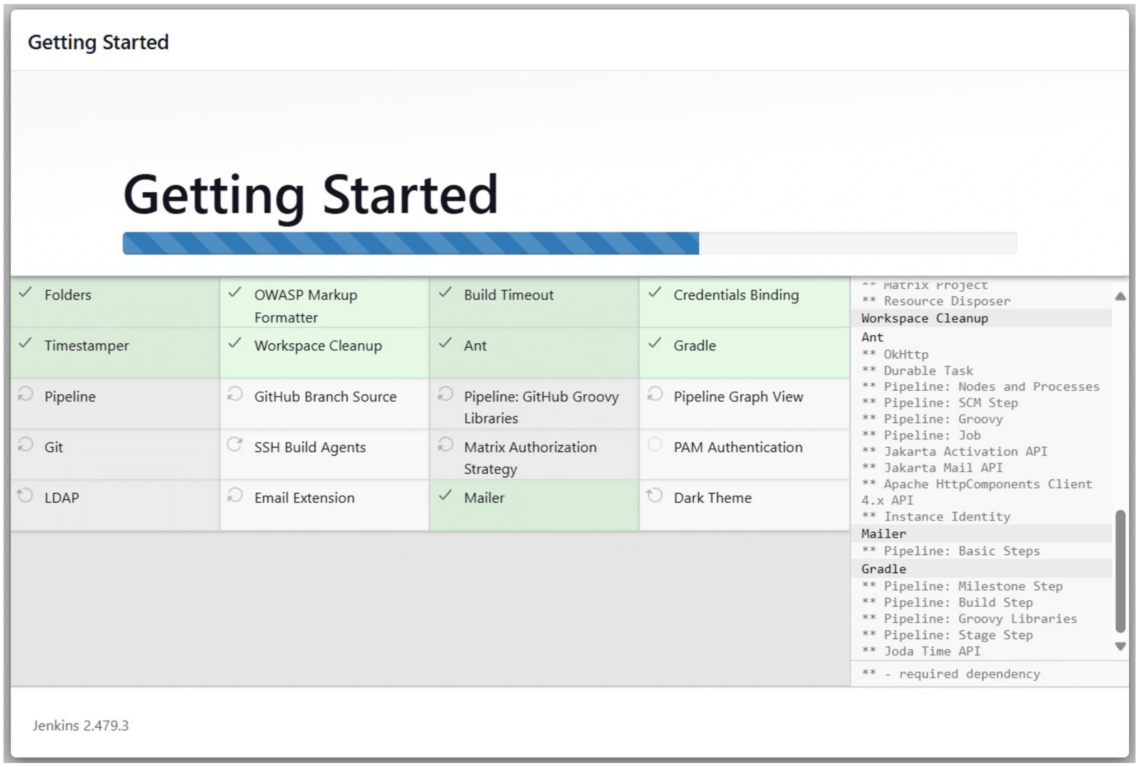
Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Once you retrieve the initialAdminPassword from C:/Program Files/Jenkinssecrets and enter it on the Unlock Jenkins page, you can customize Jenkins with Plugins.

You can click *Install Suggested Plugins* for a basic setup.



Getting Started

Getting Started

| | | | |
|---|---|---|---|
| ✓ Folders | ✓ OWASP Markup Formatter | ✓ Build Timeout | ✓ Credentials Binding |
| ✓ Timestamper | ✓ Workspace Cleanup | ✓ Ant | ✓ Gradle |
| Pipeline | GitHub Branch Source | Pipeline: GitHub Groovy Libraries | Pipeline Graph View |
| Git | SSH Build Agents | Matrix Authorization Strategy | PAM Authentication |
| LDAP | Email Extension | ✓ Mailer | Dark Theme |

```
** Matrix Project
** Resource Disposer
Workspace Cleanup
Ant
** OkHttp
** Durable Task
** Pipeline: Nodes and Processes
** Pipeline: SCM Step
** Pipeline: Groovy
** Pipeline: Job
** Jakarta Activation API
** Jakarta Mail API
** Apache HttpComponents Client
4.x API
** Instance Identity
Mailer
** Pipeline: Basic Steps
Gradle
** Pipeline: Milestone Step
** Pipeline: Build Step
** Pipeline: Groovy Libraries
** Pipeline: Stage Step
** Joda Time API
** - required dependency
```

Jenkins 2.479.3

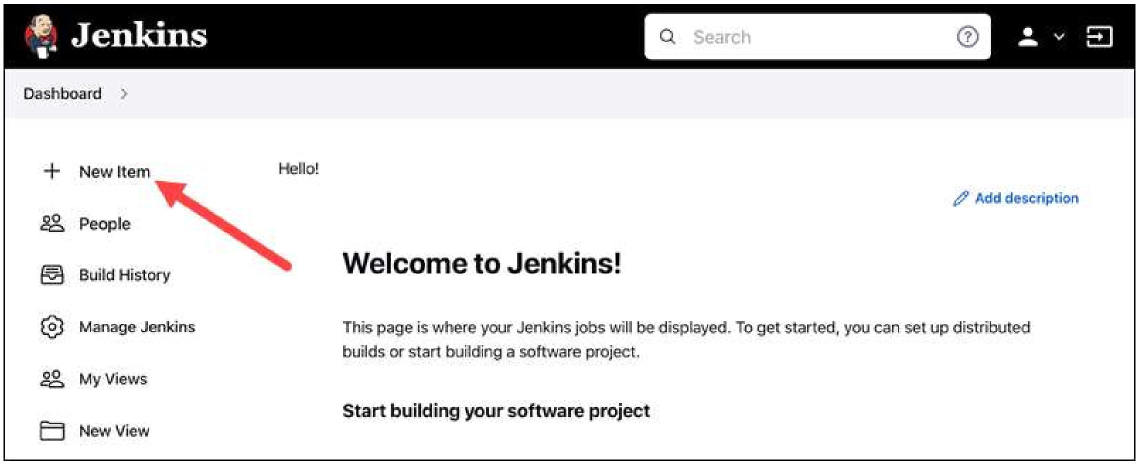
Now, create the First Administrator User by providing a username, password, and a valid email for the admin account. Click *Save and Continue*.

### How to Set up a Build Job in Jenkins

Follow the steps outlined below to set up and run a new Jenkins freestyle project.

### Step 1: Create a New Freestyle Project

1. Click the **New Item** link on the left-hand side of the Jenkins dashboard.



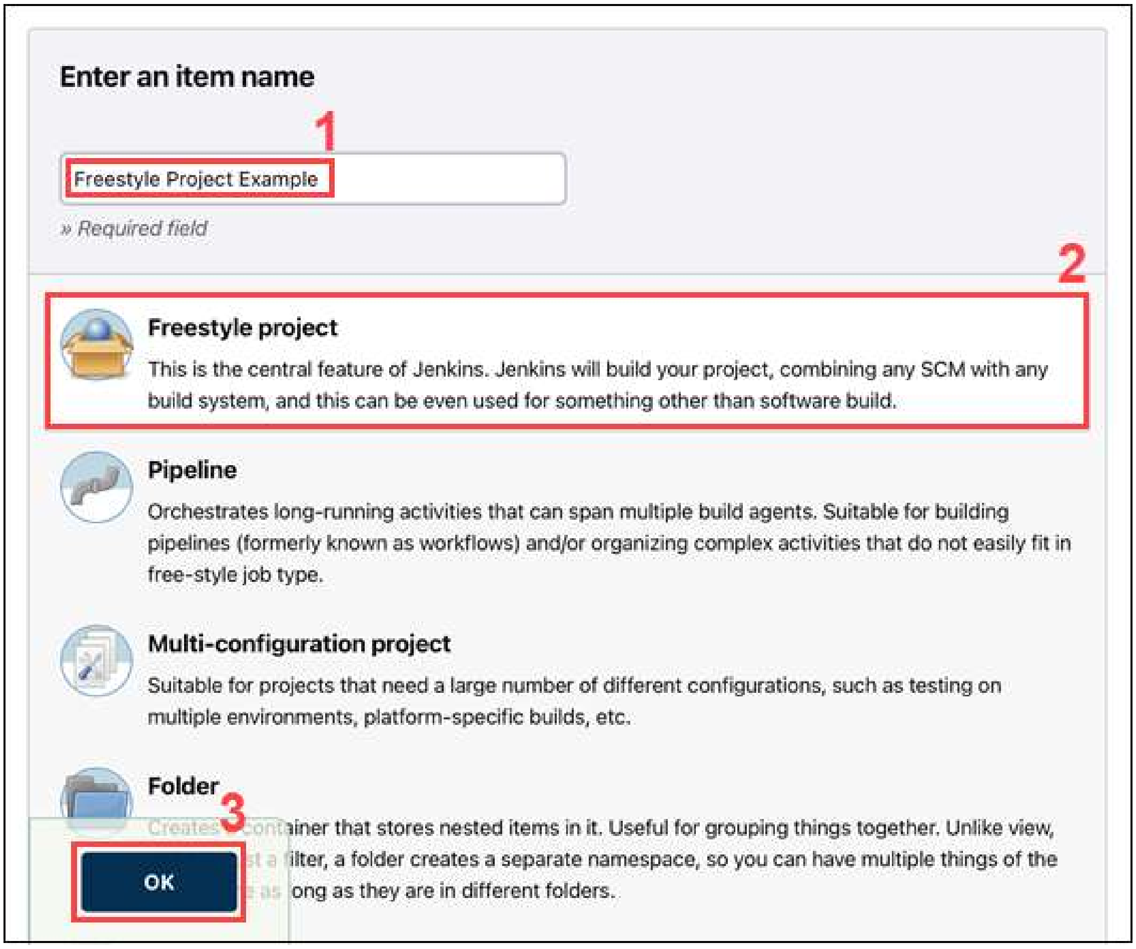
Hello!

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

2. Enter the new project's name in the **Enter an item name** field and select the **Freestyle project** type. Click **OK** to continue.



Enter an item name

Freestyle Project Example

» 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.

**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.

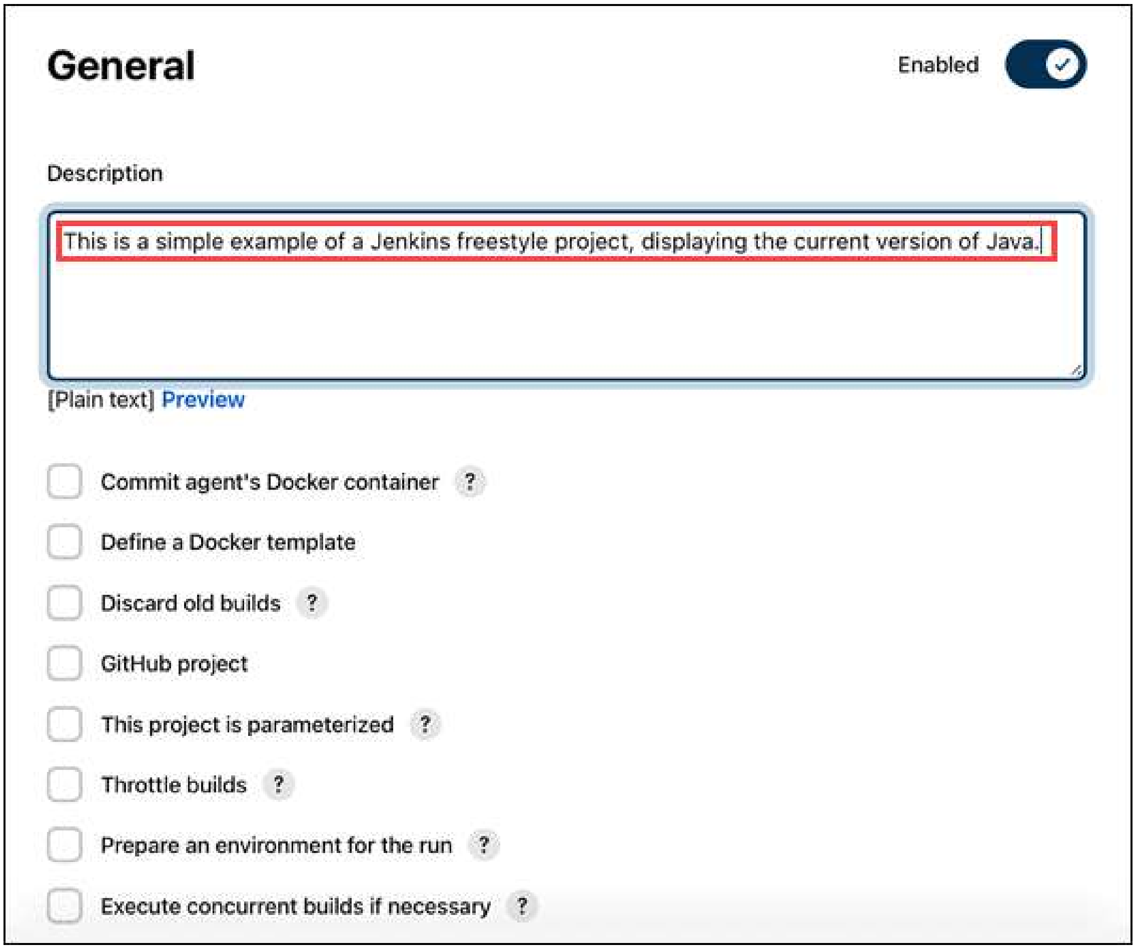
**Multi-configuration project**

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.

3. Under the *General* tab, add a project description in the **Description** field.



General

Enabled ✓

Description

This is a simple example of a Jenkins freestyle project, displaying the current version of Java.

[Plain text] Preview

□ Commit agent's Docker container ?

□ Define a Docker template

□ Discard old builds ?

□ GitHub project

□ This project is parameterized ?

□ Throttle builds ?

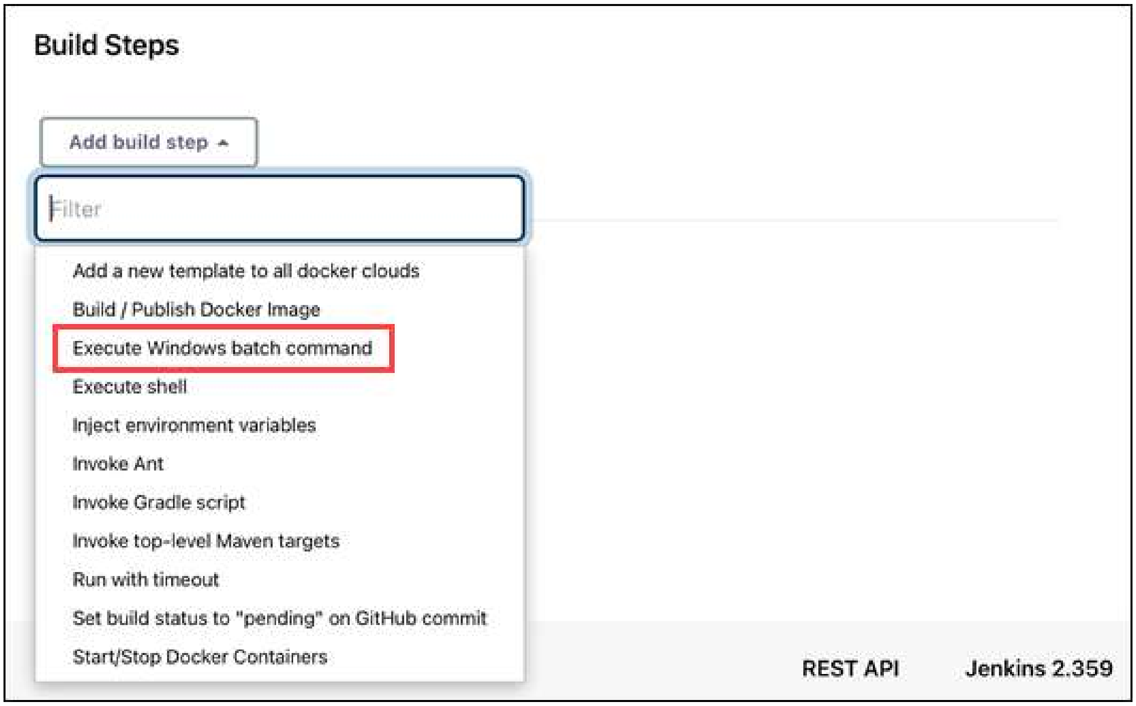
□ Prepare an environment for the run ?

□ Execute concurrent builds if necessary ?

**Step 2: Add a Build Step**

1. Scroll down to the *Build* section.

2. Open the **Add build step** drop-down menu and select **Execute Windows batch command**.

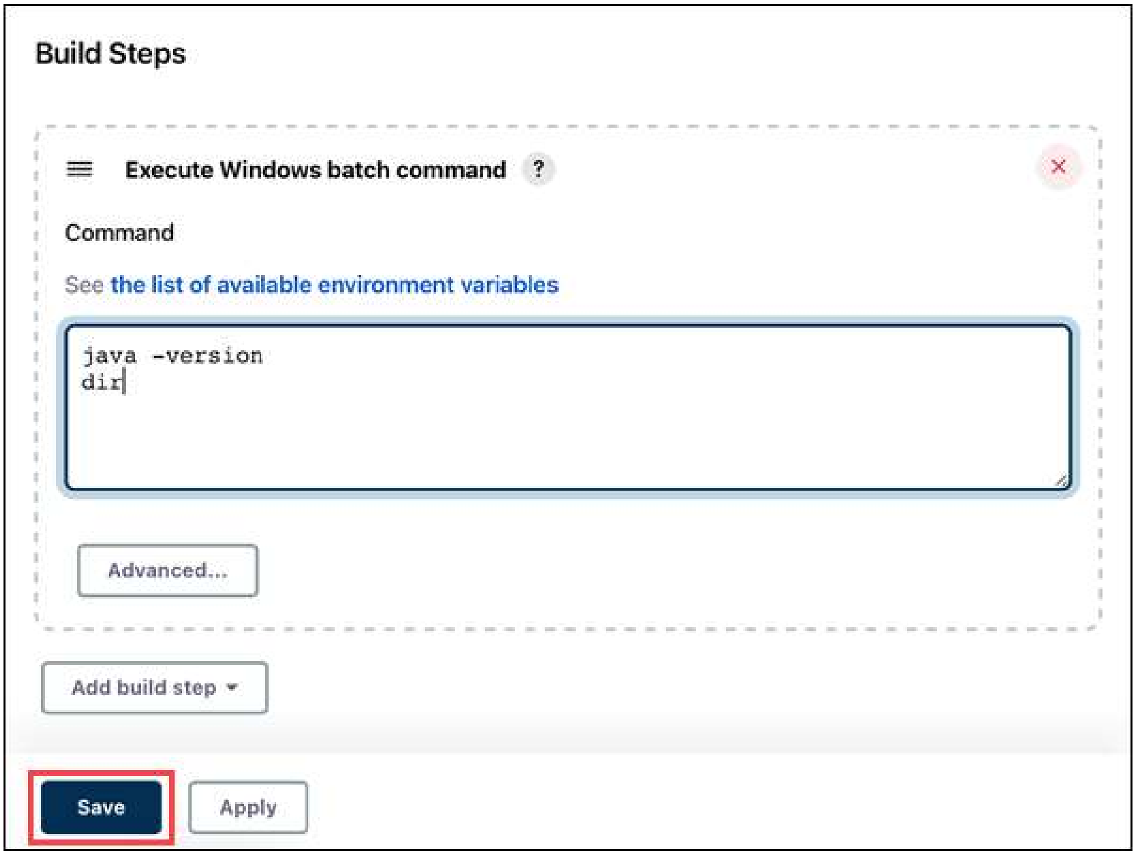

Build Steps
Add build step
Filter
Add a new template to all docker clouds
Build / Publish Docker Image
Execute Windows batch command
Execute shell
Inject environment variables
Invoke Ant
Invoke Gradle script
Invoke top-level Maven targets
Run with timeout
Set build status to "pending" on GitHub commit
Start/Stop Docker Containers
REST API
Jenkins 2.359

3. Enter the commands you want to execute in the **Command** field. For this tutorial, we are using a simple set of commands that display the current version of Java and Jenkins working directory:

java -version

dirCopy

4. Click the **Save** button to save changes to the project.



Build Steps

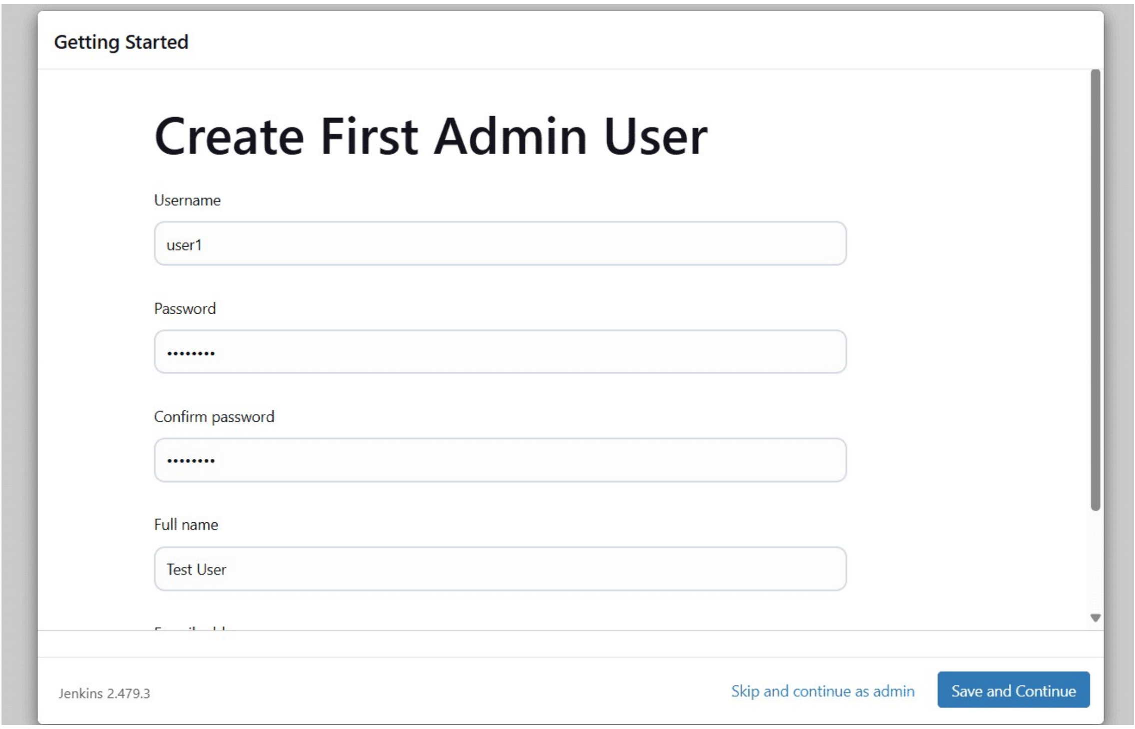
Execute Windows batch command

Command

See the list of available environment variables

```
java -version
dir
```

**Note**: Need a cheap sandboxing environment with



Getting Started

Create First Admin User

Username: user1

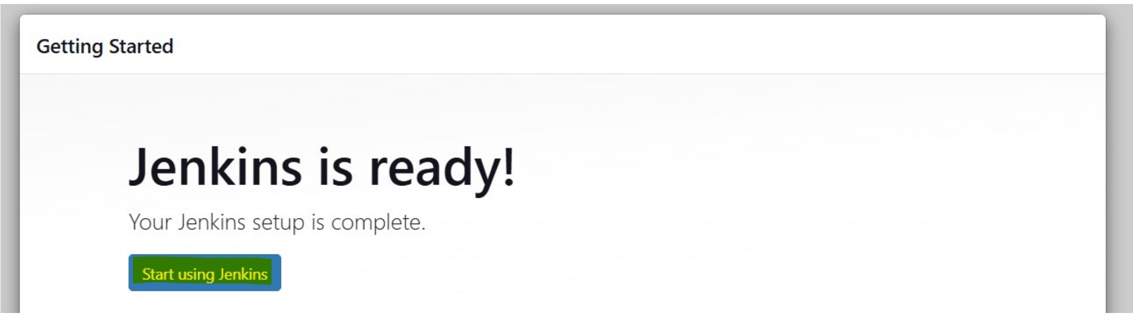
Password: ••••••••

Confirm password: ••••••••

Full name: Test User

Jenkins 2.479.3

Jenkins installation is complete.

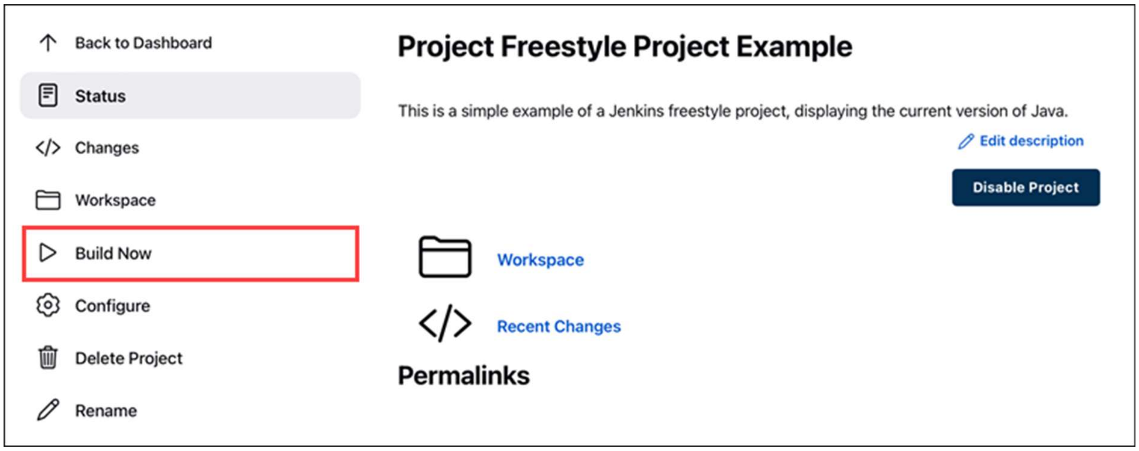


Jenkins is ready!

Your Jenkins setup is complete.

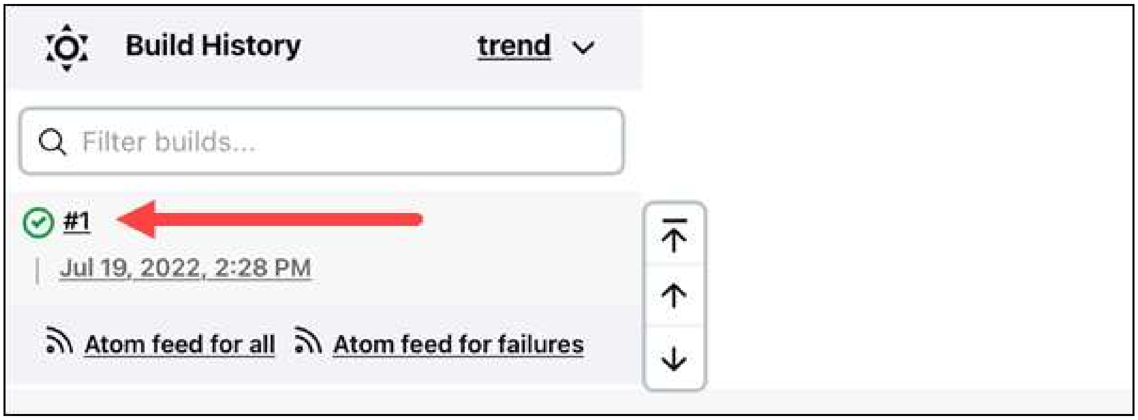
### Step 3: Build the Project

1. Click the **Build Now** link on the left-hand side of the new project page.

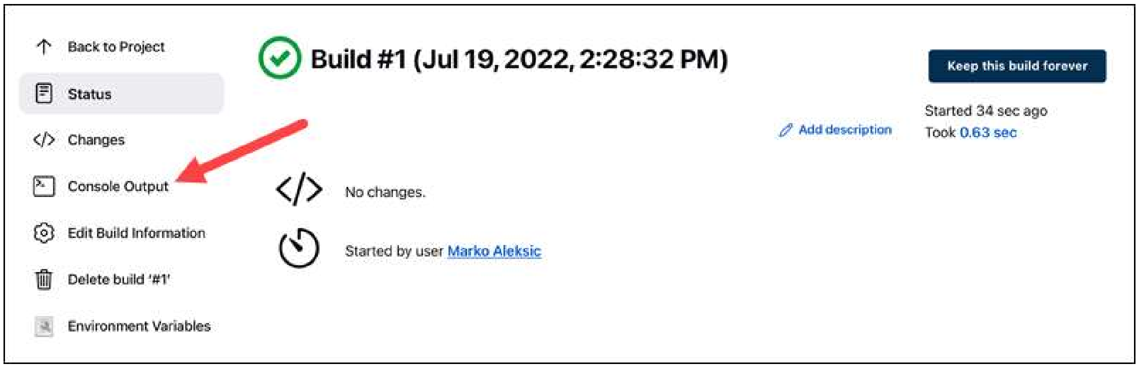


This is a simple example of a Jenkins freestyle project, displaying the current version of Java.

2. Click the link to the latest project build in the *Build History* section.



3. Click the **Console Output** link on the left-hand side to display the output for the commands you entered.



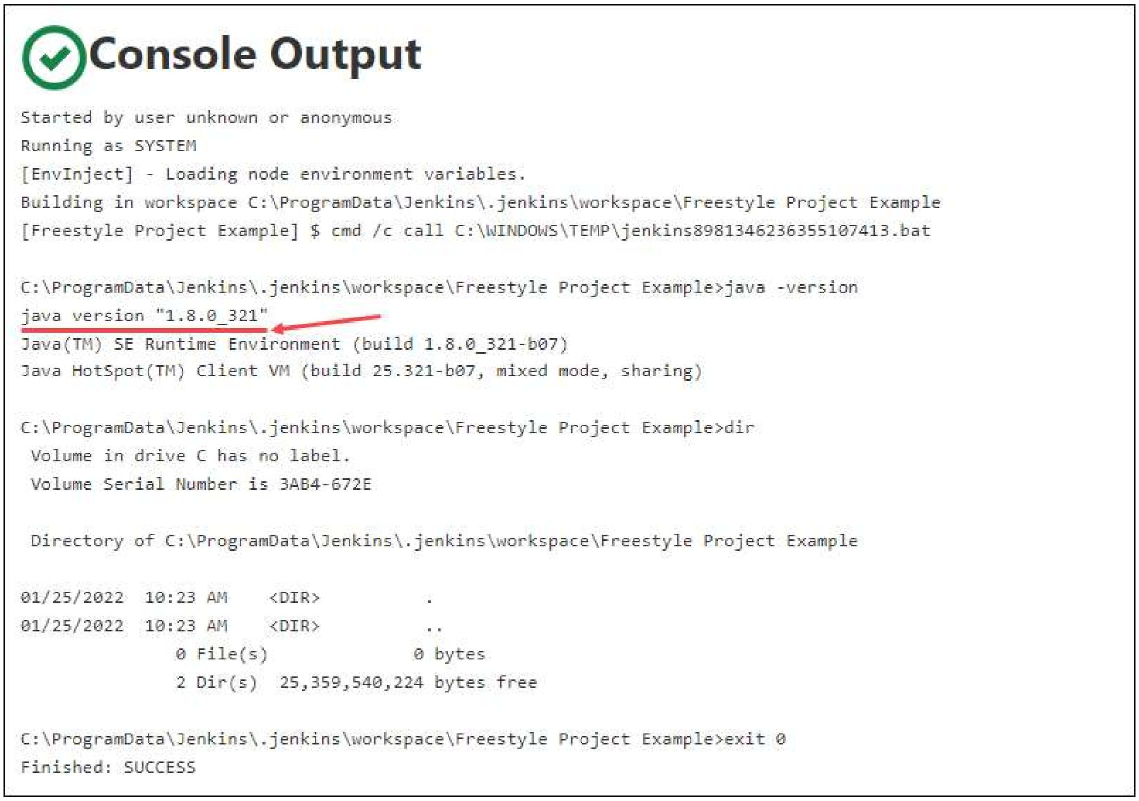
Build #1 (Jul 19, 2022, 2:28:32 PM)

Started 34 sec ago
Took 0.63 sec

No changes.

Started by user Marko Aleksic

4. The console output indicates that Jenkins is successfully executing the commands, displaying the current version of Java and Jenkins working directory.



Console Output

```
Started by user unknown or anonymous
Running as SYSTEM
[EnvInject] - Loading node environment variables.
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\Freestyle Project Example
[Freestyle Project Example] $ cmd /c call C:\WINDOWS\TEMP\jenkins8981346236355107413.bat

C:\ProgramData\Jenkins\.jenkins\workspace\Freestyle Project Example>java -version
java version "1.8.0_321"
Java(TM) SE Runtime Environment (build 1.8.0_321-b07)
Java HotSpot(TM) Client VM (build 25.321-b07, mixed mode, sharing)

C:\ProgramData\Jenkins\.jenkins\workspace\Freestyle Project Example>dir
 Volume in drive C has no label.
 Volume Serial Number is 3AB4-672E

 Directory of C:\ProgramData\Jenkins\.jenkins\workspace\Freestyle Project Example

01/25/2022  10:23 AM    <DIR>          .
01/25/2022  10:23 AM    <DIR>          ..
               0 File(s)              0 bytes
               2 Dir(s)  25,359,540,224 bytes free

C:\ProgramData\Jenkins\.jenkins\workspace\Freestyle Project Example>exit 0
Finished: SUCCESS
```