

DEVOPS ENGINEER

HOW TO SET UP A BUILD JOB IN JENKINS

Introduction

Jenkins is an open-source server used for automating software development and deployment. It allows developers to create repeatable jobs that contain all the code necessary to build an application.

What Is a Build Job?

A Jenkins build job contains the configuration for automating a specific task or step in the application building process. These tasks include gathering dependencies, compiling, archiving, or transforming code, and testing and deploying code in different environments.

Jenkins supports several types of build jobs, such as freestyle projects, pipelines, multi-configuration projects, folders, multibranch pipelines, and organization folders.

What is a Jenkins Freestyle Project?

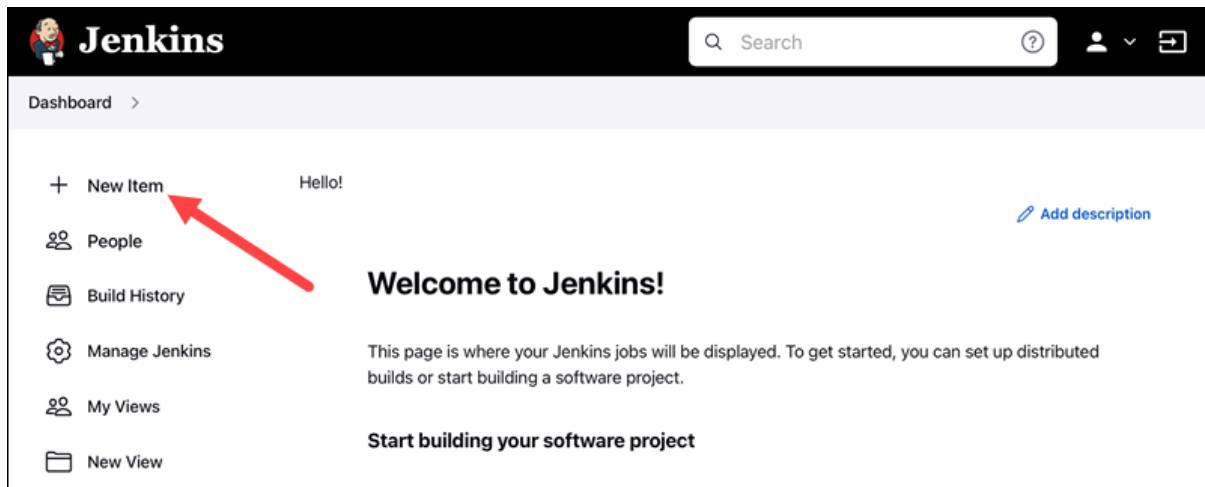
Jenkins freestyle projects allow users to automate simple jobs, such as running tests, creating and packaging applications, producing reports, or executing commands. Freestyle projects are repeatable and contain both build steps and post-build actions.

Even though freestyle jobs are highly flexible, they support a limited number of general build and post-build actions. Any specialized or non-typical action a user wants to add to a freestyle project requires additional plugins.

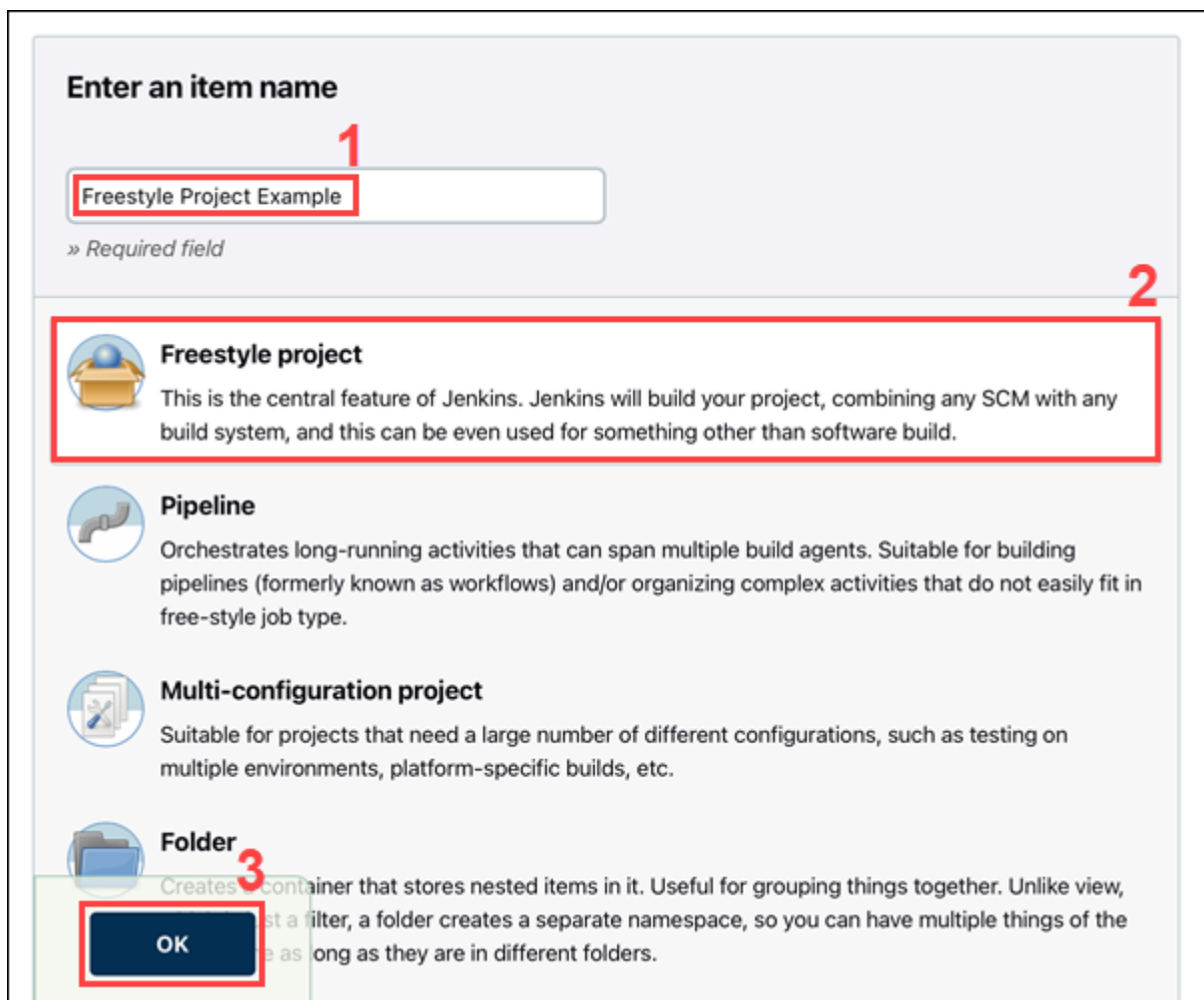
Follow the steps outlined below to set up and run a new Jenkins freestyle project.

Step 1: Create a New Freestyle Project

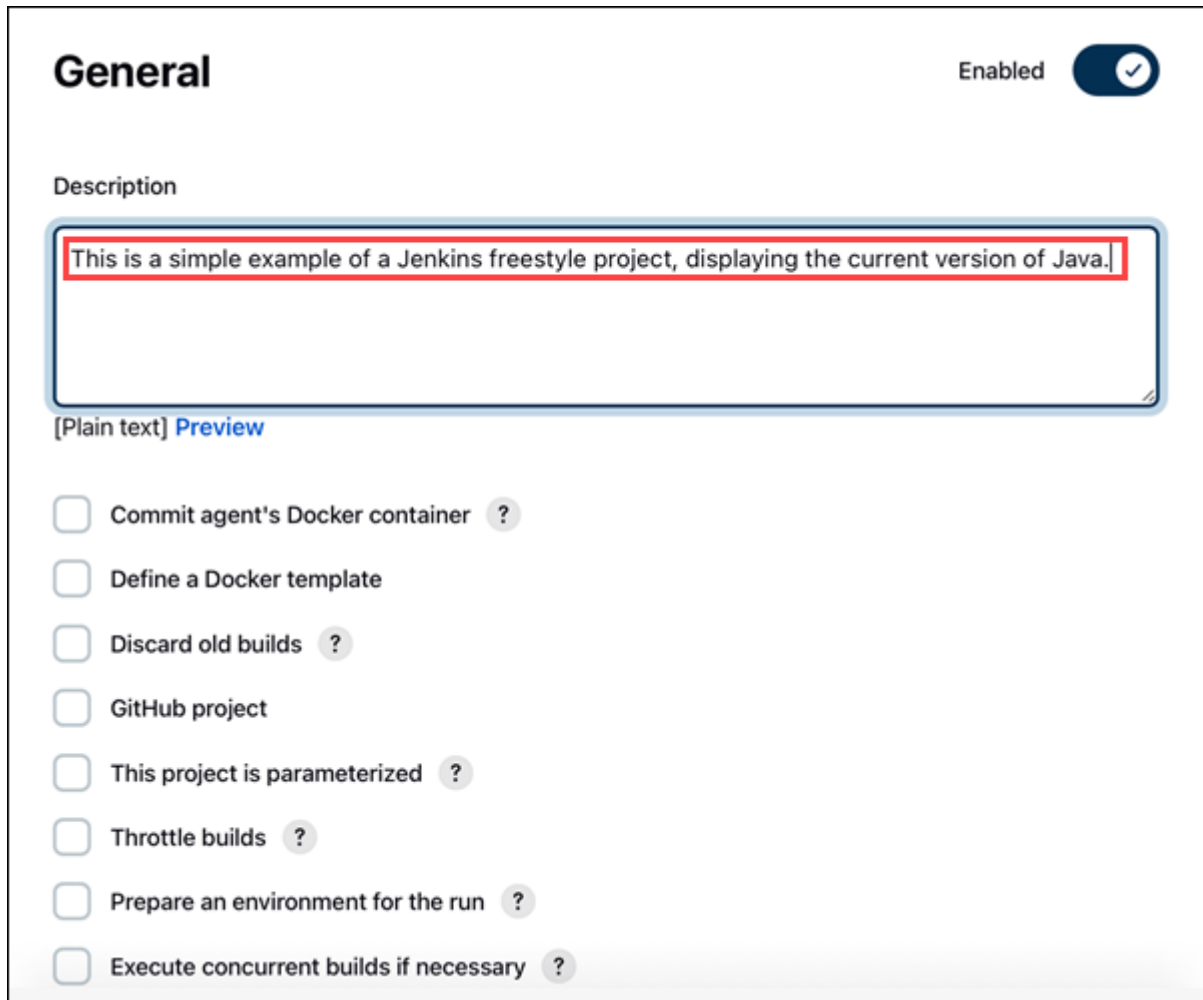
1. Click the **New Item** link on the left-hand side of the Jenkins dashboard.



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



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

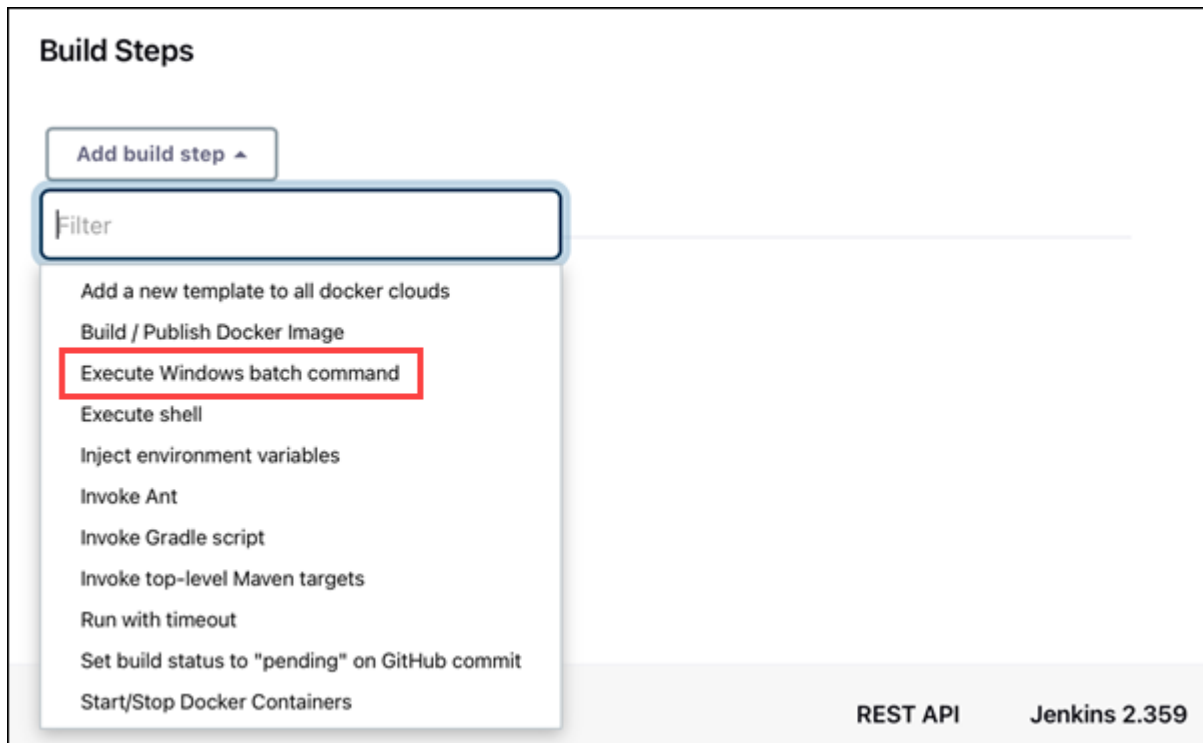


The screenshot shows the 'General' tab of a Jenkins project configuration. At the top right, there is a toggle switch labeled 'Enabled' which is turned on. Below this, the 'Description' section contains a text area with the text 'This is a simple example of a Jenkins freestyle project, displaying the current version of Java.' The text area is highlighted with a red border. Below the text area, there is a link '[Plain text] Preview'. At the bottom, there is a list of checkboxes with labels and help icons (question marks):

- ☐ 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**.



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
```

```
dir
```

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
```

Advanced...

Add build step ▾

Save Apply

Step 3: Build the Project

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

↑ Back to Dashboard

Project Freestyle Project Example

This is a simple example of a Jenkins freestyle project, displaying the current version of Java. [Edit description](#)

Build Now [Disable Project](#)

[Workspace](#)

[Recent Changes](#)

Permalinks

[Status](#)

[Changes](#)

[Workspace](#)


[Build Now](#)

[Configure](#)

[Delete Project](#)

[Rename](#)

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 History** trend ▾

✓ **#1**

Jul 19, 2022, 2:28 PM

Atom feed for all Atom feed for failures

⬆

⬆

⬇

⬆ Back to Project

Status

Changes

Console Output

Edit Build Information

Delete build '#1'

Environment Variables

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

No changes.

Started by user [Marko Aleksic](#)

[Add description](#)

Started 34 sec ago
Took 0.63 sec

Keep this build forever

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

Note: Learn how to configure [Jenkins Shared Libraries](#) to make your deployment and development more efficient.

Conclusion

After reading this tutorial, you should be able to create and run your first Jenkins freestyle project.

Check out our rundown of the [best CI/CD tools](#) for Jenkins alternatives or additions.