

The screenshot shows the Jenkins Dashboard at localhost:8080. The top navigation bar includes the Jenkins logo, a search bar, and user information (admin). The left sidebar contains links to New Item, People, Build History, Manage Jenkins, and My Views. The main content area displays a table of build history with columns for Status (S), Weather (W), Name, Last Success, Last Failure, and Last Duration. Two builds are listed: 'Artifactory' (failed) and 'Job1' (successful). Below the table, there are sections for 'Build Queue' (showing no builds) and 'Build Executor Status' (showing two idle executors).

S	W	Name	Last Success	Last Failure	Last Duration
❌	☁️	Artifactory	N/A	5 days 0 hr #1	3.6 sec
✅	☀️	Job1	6 days 19 hr #4	6 days 20 hr #1	29 sec

Download the code coverage plugin in Jenkins

Go to [Dashboard](#) > [Manage Jenkins](#) > [Plugins](#) > Available Plugins > search for JACOCO

The screenshot shows the Jenkins Plugins page. The left sidebar has links to Updates, Available plugins, Installed plugins, and Advanced settings. The main content area is titled 'Plugins' and shows a search bar with 'JACOCO' entered. Below the search bar, a table lists available plugins. The 'JaCoCo 3.3.5' plugin is highlighted, showing its name, version, and release date (12 days ago). A description states: 'This plugin integrates JaCoCo code coverage reports to Jenkins.'

Install	Name	Released
<input checked="" type="checkbox"/>	JaCoCo 3.3.5 Build Reports This plugin integrates JaCoCo code coverage reports to Jenkins.	12 days ago

Click on Install button

The screenshot shows the Jenkins Download progress page. The left sidebar has links to Updates, Available plugins, Installed plugins, Advanced settings, and Download progress. The main content area is titled 'Download progress' and shows the progress of downloading the JaCoCo plugin. The progress is shown as a list of steps: Preparation, JaCoCo, and Loading plugin extensions, all of which are marked as 'Success'. A link to 'Go back to the top page' is provided, along with a note that the user can start using the installed plugins right away. There is also a checkbox to 'Restart Jenkins when installation is complete and no jobs are running'.

Download progress

- Preparation
 - Checking internet connectivity
 - Checking update center connectivity
 - Success
- JaCoCo ✅ Success
- Loading plugin extensions ✅ Success

→ [Go back to the top page](#)
(you can start using the installed plugins right away)

→ ☐ Restart Jenkins when installation is complete and no jobs are running

Now go and create a Freestyle job

The screenshot shows the 'Configure' page for a Freestyle job in SonarQube, specifically the 'Source Code Management' section. The left sidebar contains a 'Configure' menu with options: General, Source Code Management (selected), Build Triggers, Build Environment, Build Steps, and Post-build Actions. The main content area is titled 'Source Code Management' and has two radio buttons: 'None' and 'Git' (selected). Below the 'Git' button is a 'Repositories' section with a 'Repository URL' field containing 'https://github.com/Sonal0409/calwebapp-sonar.git', a 'Credentials' dropdown menu set to 'none', and an 'Add' button. There is also an 'Advanced' dropdown and an 'Add Repository' button. At the bottom, there is a 'Branches to build' section.

The screenshot shows the 'Configure' page for a Freestyle job in SonarQube, specifically the 'Build Steps' section. The left sidebar contains a 'Configure' menu with options: General, Source Code Management, Build Triggers, Build Environment, Build Steps (selected), and Post-build Actions. The main content area is titled 'Build Steps' and has three checkboxes: 'Maven3-Artifactory Integration', 'Terminate a build if it's stuck', and 'With Ant' (selected). Below the checkboxes is a 'Build Steps' section with a 'Maven Version' dropdown menu set to 'mymaven' and a 'Goals' field containing 'package'. There is also an 'Advanced' dropdown and an 'Add build step' button.

In port build actions > Select Record JACOCO coverage reports

Dashboard > CodeCoverage > Configuration

Configure

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

Invoke top-level Maven targets ?

Maven Version

Filter

- Aggregate downstream test results
- Archive the artifacts
- Build other projects
- Publish JUnit test result report
- Publish Javadoc
- Publish TestNG Results
- Record JaCoCo coverage report
- Record fingerprints of files to track usage
- Git Publisher
- E-mail Notification
- Editable Email Notification
- Set GitHub commit status (universal)
- Set build status on GitHub commit [deprecated]
- Delete workspace when build is done

Add post-build action +

Save Apply

Execute the job

Jenkins

Search (CTRL+K)

admin log out

Dashboard > CodeCoverage >

Project CodeCoverage

Add description

Disable Project

Permalinks

- Last build (#1), 1 min 29 sec ago
- Last stable build (#1), 1 min 29 sec ago
- Last successful build (#1), 1 min 29 sec ago
- Last completed build (#1), 1 min 29 sec ago

Code Coverage Trend

enlarge

Build History trend

Filter builds...

#1

17-Oct-2023 11:23 AM

Status

Changes

Console Output

View as plain text

Edit Build Information

Delete build '#1'

Git Build Data

Coverage Report

JaCoCo Coverage Report

Download jacoco.exe binary coverage file



Overall Coverage Summary

name	instruction	branch	complexity	line	method	class
all classes	7% M: 56 C: 4	0% M: 4 C: 0	13% M: 7 C: 1	6% M: 17 C: 1	17% M: 5 C: 1	50% M: 1 C: 1

Coverage Breakdown by Package

name	instruction	branch	complexity	line	method	class
com.cruds.service	M: 7 C: 4 36%	M: 0 C: 0 100%	M: 2 C: 1 33%	M: 2 C: 1 33%	M: 2 C: 1 33%	M: 0 C: 1 100%
com.cruds.servlet	M: 49 C: 0 0%	M: 4 C: 0 0%	M: 5 C: 0 0%	M: 15 C: 0 0%	M: 3 C: 0 0%	M: 1 C: 0 0%