

Lesson 06 Demo 04

Publishing Test Cases Reports in Jenkins

Objective: To publish test case reports in Jenkins to enhance transparency, accountability, and quality in the software development process

Tools required: Jenkins

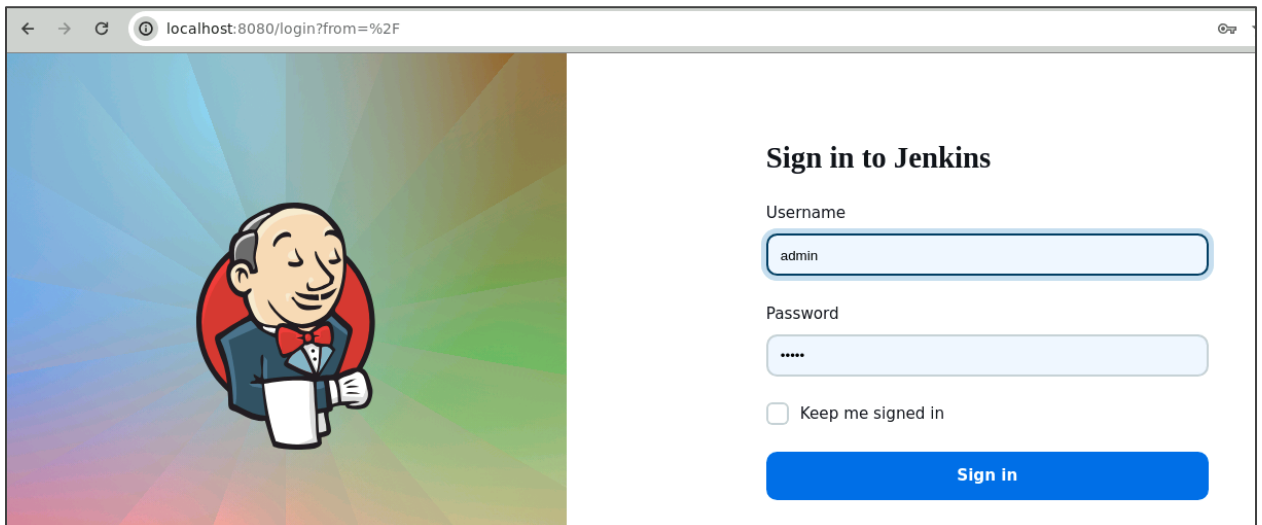
Prerequisites: You need to have a Jenkins up and running.

Steps to be followed:

1. Log in to the Jenkins CI tool and publish test case reports

Step 1: Log in to the Jenkins CI tool and publish test case reports

1.1 Log in to Jenkins using your credentials

A screenshot of a web browser showing the Jenkins login page. The browser's address bar displays 'localhost:8080/login?from=%2F'. The page features a large illustration of a man in a tuxedo on the left and a login form on the right. The form is titled 'Sign in to Jenkins' and includes fields for 'Username' (containing 'admin') and 'Password' (masked with dots). There is a checkbox for 'Keep me signed in' and a blue 'Sign In' button at the bottom.

← → ↻ 🔒 localhost:8080/login?from=%2F

Sign in to Jenkins

Username
admin

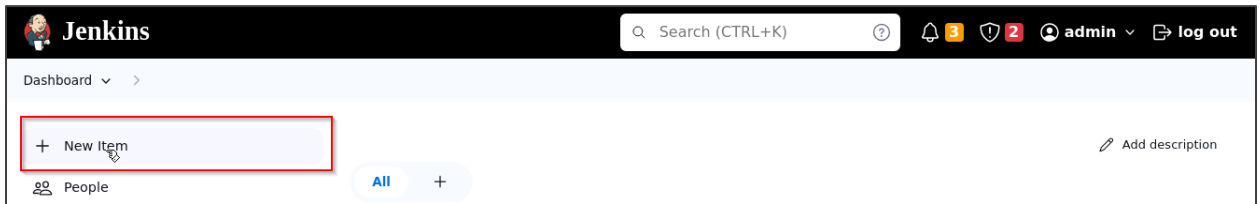
Password
.....

☐ Keep me signed in

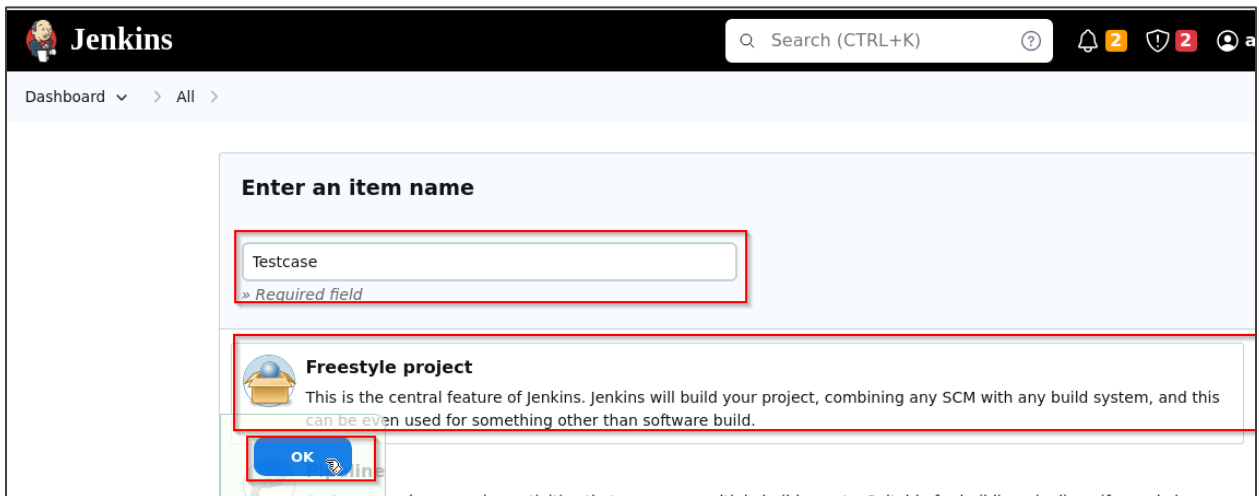
Sign In

Note: The credentials for accessing Jenkins in the lab are Username: **admin** and Password: **Root123\$**.

1.2 In the Jenkins dashboard, click on **New Item**



1.3 Select the **Freestyle project** while creating a Jenkins job, provide a custom job name, and click on **OK**

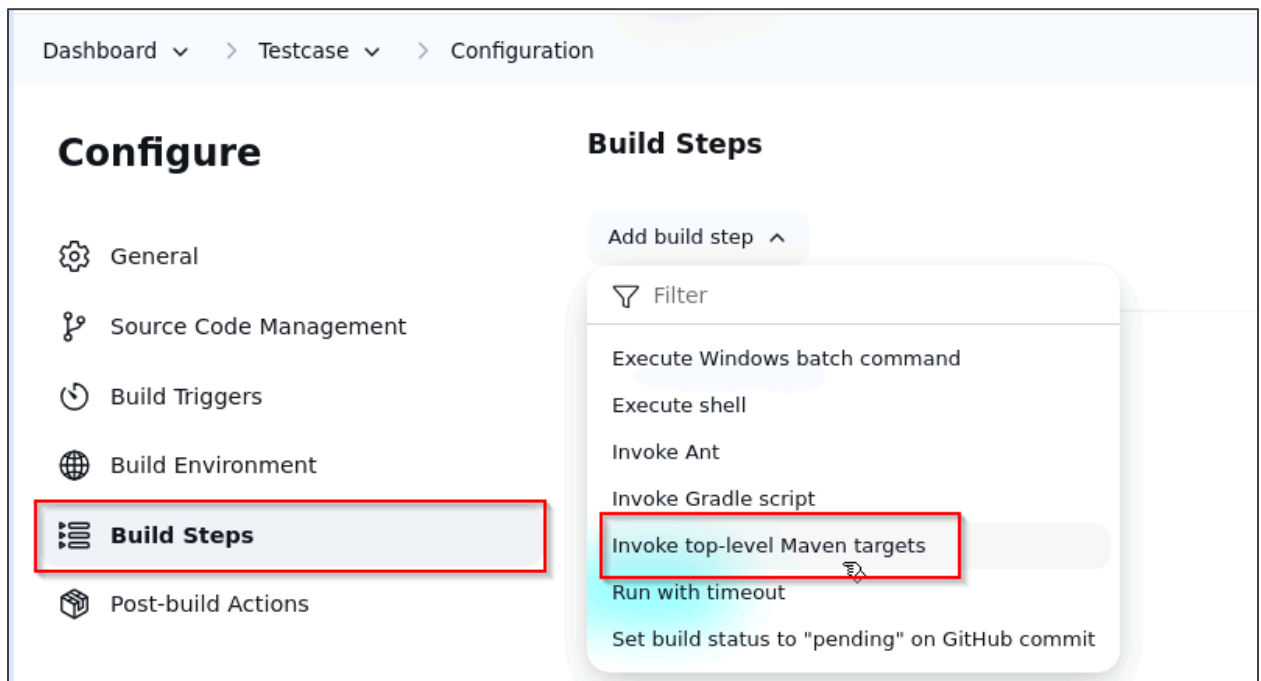


1.4 Now, in the Configure page, navigate to **Source Code Management** from the left navigation bar, select **Git**, and then provide the Git repository URL

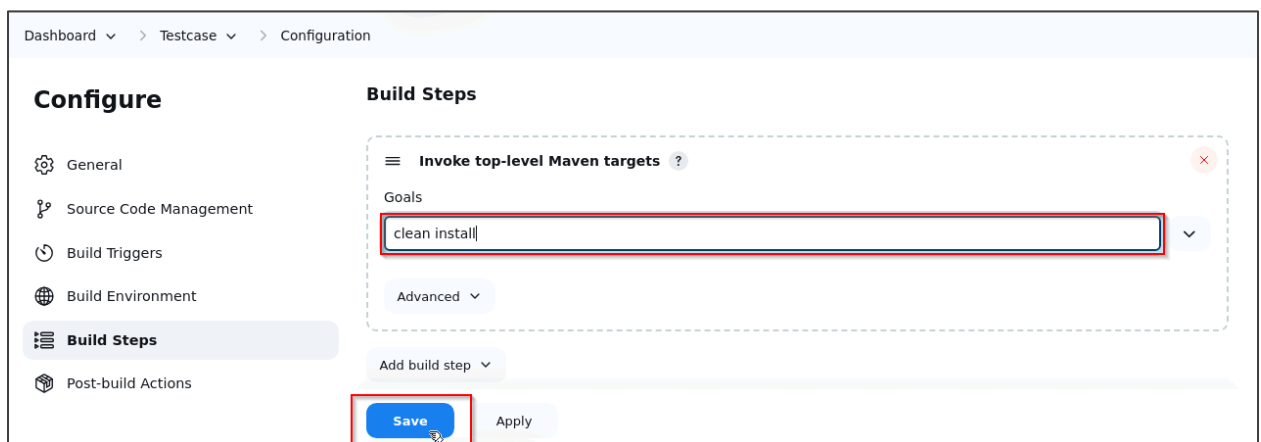


Note: Here, the repository URL is <https://github.com/github-simplilearn-net/MavenBuild.git>.

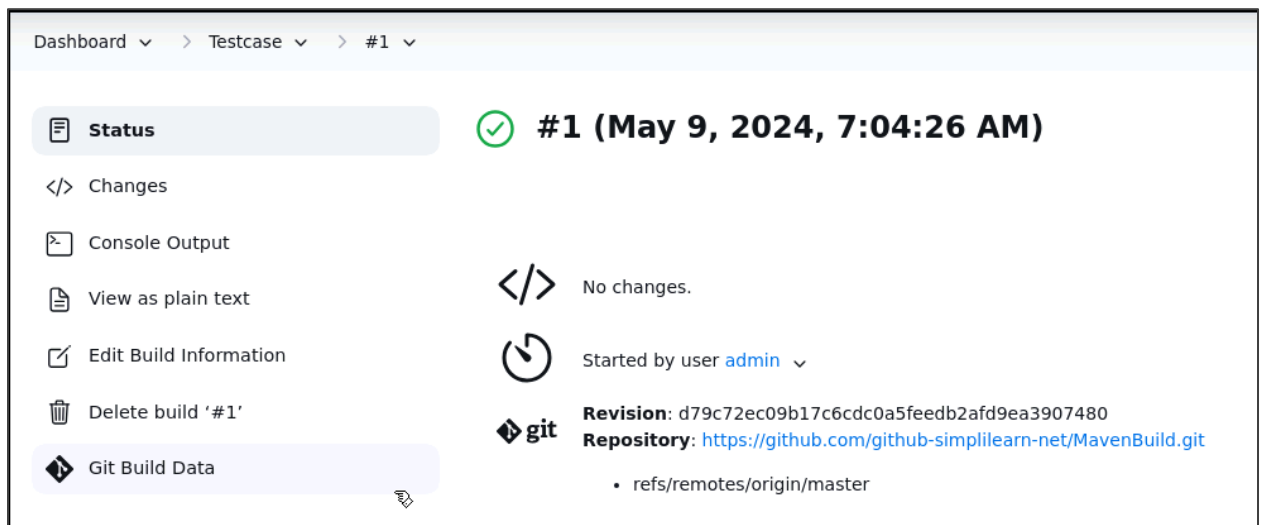
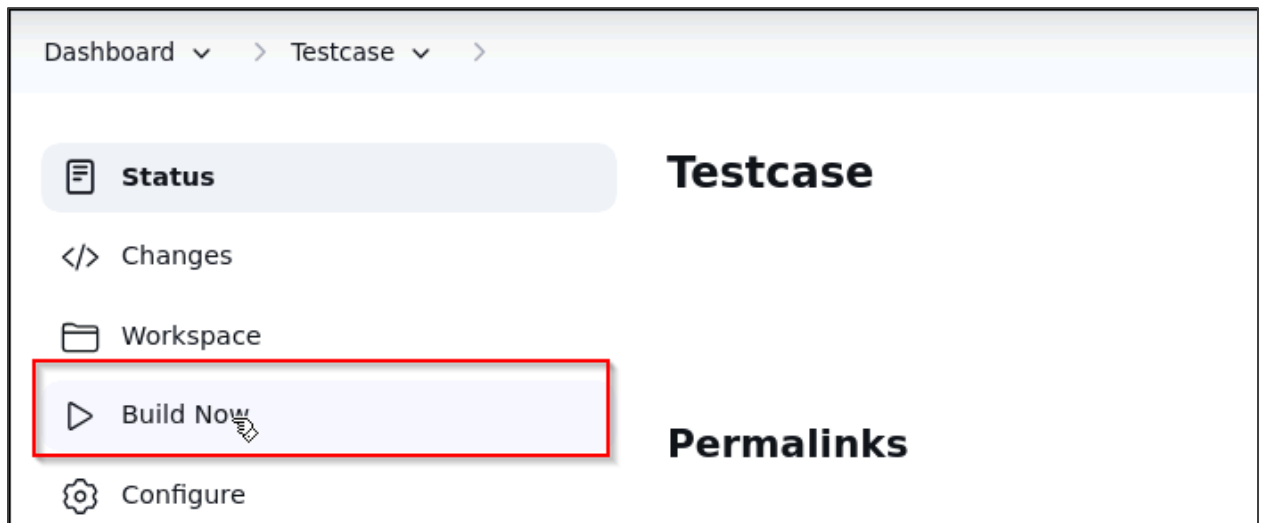
1.5 Now, navigate to **Build Steps**, click on **Add build step**, and then select **Invoke top-level Maven targets**



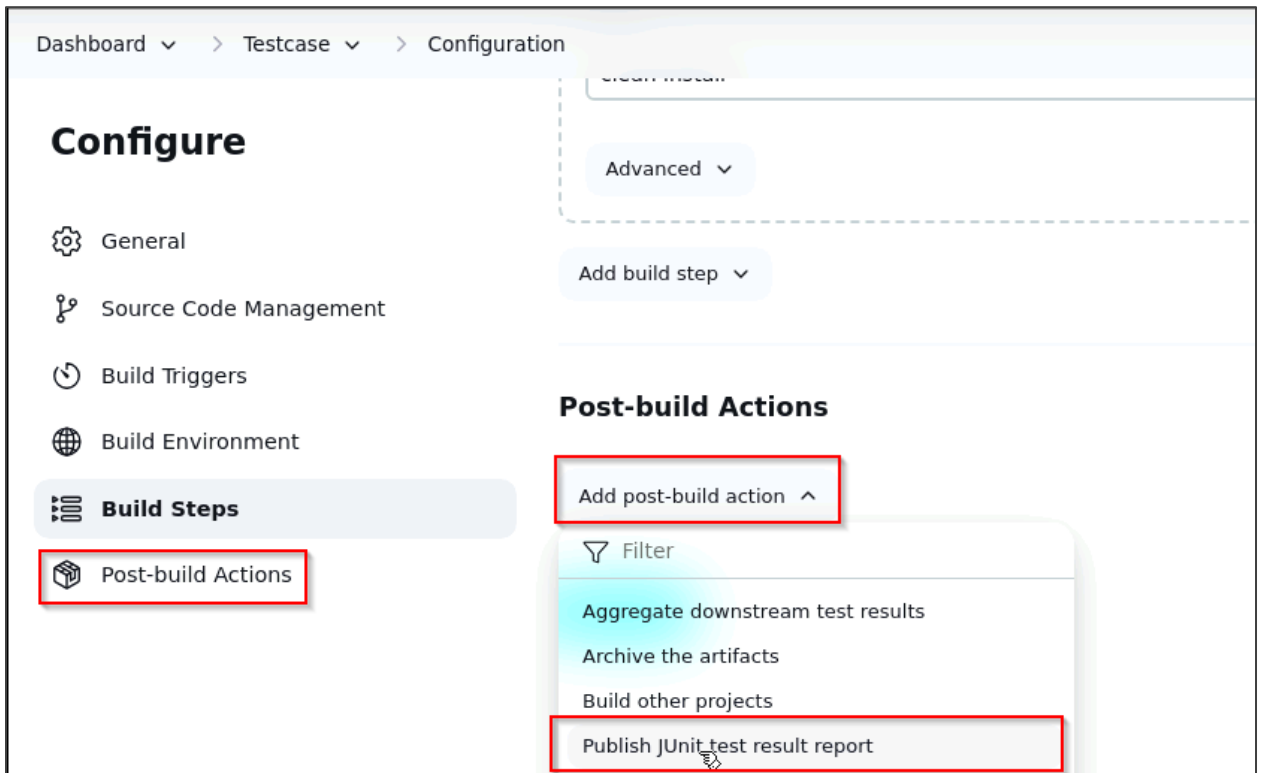
1.6 Now, provide **clean install** in the **Goals** section and click on **Save**



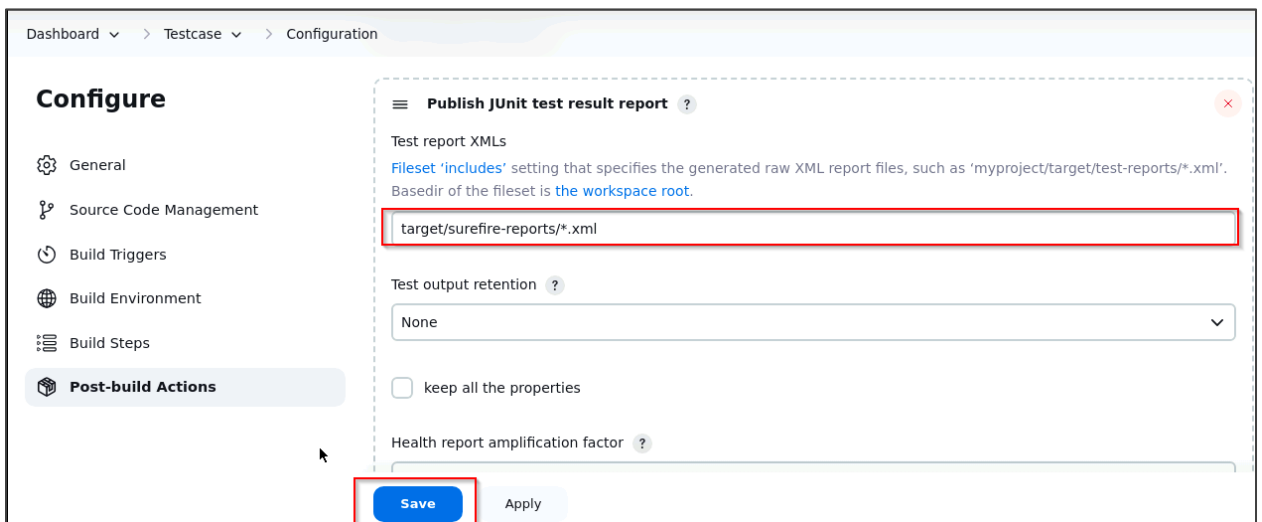
1.7 Then, click on **Build Now** to configure a build



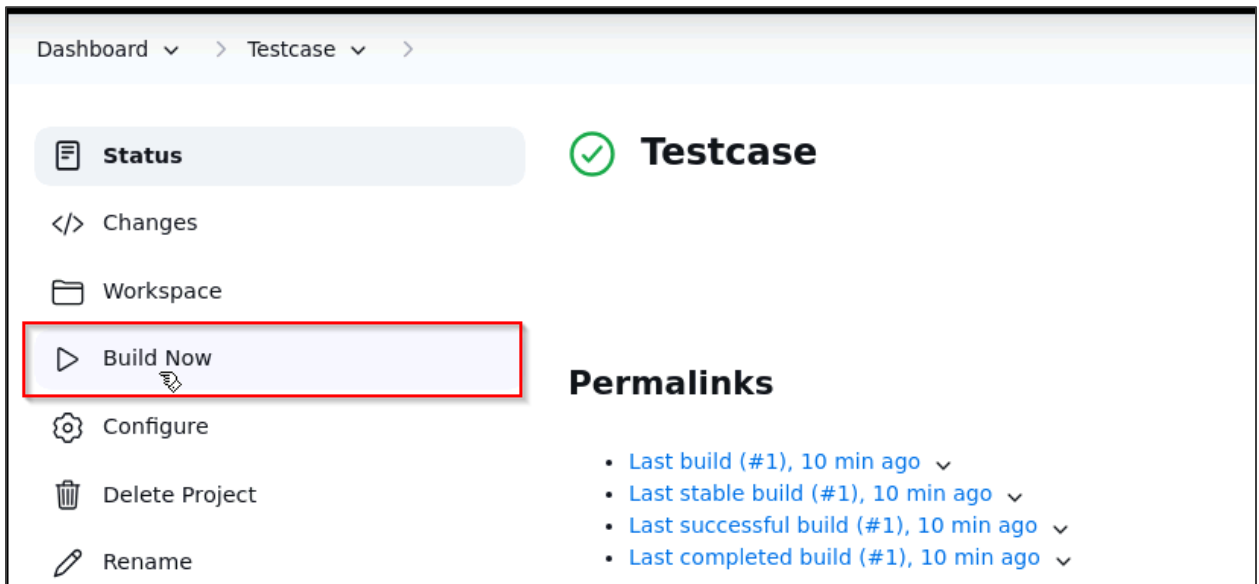
1.8 Now, navigate back to the **Testcase** job configuration page, click on **Post-build Actions**, then choose **Publish JUnit test result report** under the **Add post-build action**



1.9 Provide **target/surefire-reports/*.xml** in the **Test report XMLs** section and then click on **Save**



1.10 Now, click on **Build Now** to trigger new build, and build logs will show you build process logs



The screenshot shows the Jenkins 'Testcase' page. The breadcrumb navigation at the top is 'Dashboard > Testcase >'. On the left sidebar, the 'Build Now' button is highlighted with a red rectangular box. The main content area has a green checkmark icon and the title 'Testcase'. Below the sidebar, there are links for 'Changes', 'Workspace', 'Build Now', 'Configure', 'Delete Project', and 'Rename'. On the right, under the heading 'Permalinks', there is a list of build links: 'Last build (#1), 10 min ago', 'Last stable build (#1), 10 min ago', 'Last successful build (#1), 10 min ago', and 'Last completed build (#1), 10 min ago', each with a dropdown arrow.



The screenshot shows the Jenkins 'Console Output' for build #2. The breadcrumb navigation is 'Dashboard > Testcase > #2 > Console Output'. The output text is as follows:

```
-----
T E S T S
-----
Running com.geekcap.vmturbo.ThingTest
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.046 sec

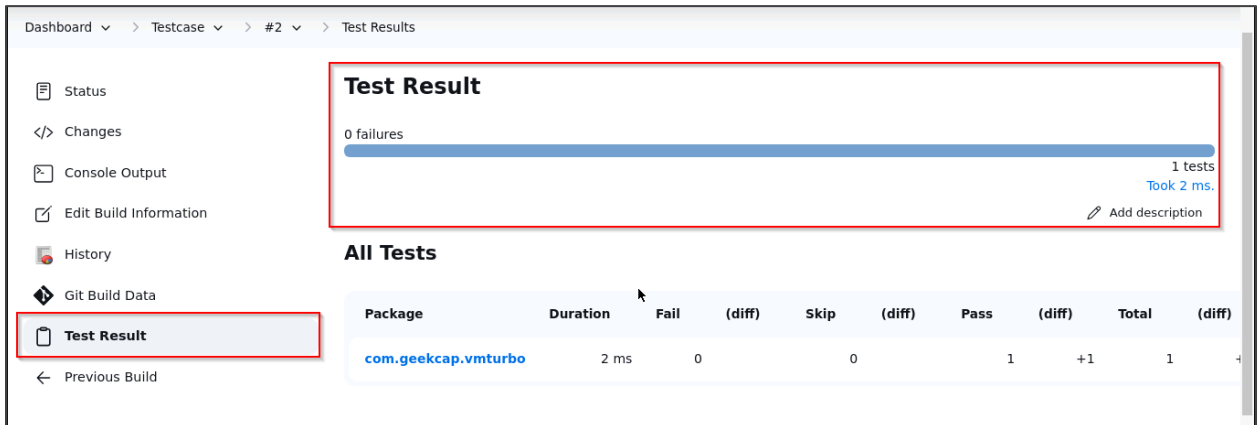
Results :

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

[[1;34mINFO[[m]
[[1;34mINFO[[m] [[1m--- [[0;32mmaven-war-plugin:3.3.2:war[[m [[1m(default-war)[[m @ [[36mjava-example[[0;1m ---[[m
[[1;34mINFO[[m] Packaging webapp
[[1;34mINFO[[m] Assembling webapp [java-example] in [/var/lib/jenkins/workspace/Testcase/target/java-example]
[[1;34mINFO[[m] Processing war project
[[1;34mINFO[[m] Copying webapp resources [/var/lib/jenkins/workspace/Testcase/src/main/webapp]
[[1;34mINFO[[m] Building war: /var/lib/jenkins/workspace/Testcase/target/java-example.war
[[1;34mINFO[[m]
[[1;34mINFO[[m] [[1m--- [[0;32mmaven-install-plugin:2.4:install[[m [[1m(default-install)[[m @ [[36mjava-
example[[0;1m ---[[m
[[1;34mINFO[[m] Installing /var/lib/jenkins/workspace/Testcase/target/java-example.war to
```

You can see that the build is configured successfully.

1.11 Now, click on **Test Result** to check the test reports



The screenshot shows the Jenkins 'Test Results' page for a specific build. The breadcrumb navigation at the top reads 'Dashboard > Testcase > #2 > Test Results'. On the left sidebar, the 'Test Result' option is highlighted with a red box. The main content area features a 'Test Result' section, also outlined in red, which displays '0 failures' and a blue progress bar. To the right of the bar, it indicates '1 tests' and 'Took 2 ms.', with an 'Add description' link. Below this is an 'All Tests' table. The table has columns for Package, Duration, Fail, (diff), Skip, (diff), Pass, (diff), Total, and (diff). A single row is visible for the package 'com.geekcap.vmturbo', showing a duration of 2 ms, 0 failures, 0 skips, 1 pass, and a total of 1 test.

Package	Duration	Fail	(diff)	Skip	(diff)	Pass	(diff)	Total	(diff)
com.geekcap.vmturbo	2 ms	0		0		1	+1	1	

You can see that the test cases have been successfully passed.

By following these steps, you have successfully published test case reports in Jenkins to ensure transparency, accountability, and quality in the software development process.