

**Name - Zuhair Ahmad**

**SapID: 500119134**

**Roll no:R2142230479**

## **Lab Exercise 9- Setting up Poll SCM Configuration in Jenkins**

**Objective:** To set up poll SCM configuration in Jenkins for automating build triggering and optimizing resource usage on Jenkins servers

**Tools required:** Jenkins

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

### **Step 1: Log in to the Jenkins CI tool and set up poll SCM configuration**

- 1.1 Log in to Jenkins using the credentials
- 1.2 In the Jenkins dashboard, click on **New Item**

DEVSECOPS-LAB-JULY-DEC-2025Dashboard - Jenkins

localhost:8080

Jenkins

+ New Item

Build History

Project Relationship

Check File Fingerprint

Build Queue

No builds in the queue.

Build Executor Status

0/2

All

+

S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☀	Job-for-Lab8	3 hr 15 min #1	N/A	42 sec
✓	☀	New-Second-Job	20 hr #13	N/A	4.8 sec

Icons: S M L

REST APIJenkins 2.516.1

16:5027-08-2025

29°C Light rain

ENG US

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

The screenshot shows the Jenkins 'Source Code Management' configuration page for a Git repository. At the top, there are two radio buttons: 'None' and 'Git'. The 'Git' option is selected and highlighted with a blue circle. Below this, the 'Repositories' section is expanded, showing a text input for 'Repository URL' containing 'https://github.com/hkshitesh/DEVSECOPS-MAVEN-REPO.git'. Underneath, the 'Credentials' section shows a dropdown menu with '- none -' selected, which is highlighted with a purple rectangular box. To the left of the dropdown is a '+ Add' button, and below it is an 'Advanced' dropdown menu. Below the 'Credentials' section is an 'Add Repository' button. The 'Branches to build' section is also expanded, showing a text input for 'Branch Specifier (blank for \'any\')' containing '\*/master'. Below this is an 'Add Branch' button. At the bottom of the configuration area, there is a 'Repository browser' section with a dropdown menu. At the very bottom of the page, there are two buttons: 'Save' (in blue) and 'Apply' (in grey).

1.5 Now, navigate to **Build Environment**, click on **Poll SCM**, provide **H/2\*\*\*\*** under the **Schedule** section, and then click on **Save (Triggers Section)**

DEVSECOPS-LAB-JULY-DEC-20

Job-PollSCM-lab9 Config - Jeni

Misha1207-code/DEVSECOPS-

localhost:3080/job/Job-PollSCM-lab9/configure

Jenkins

Job-PollSCM-lab9 / Configuration

Configure

General

Source Code Management

Triggers

Environment

Build Steps

Post-build Actions

Triggers

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

☐ Trigger builds remotely (e.g., from scripts) ?

☐ Build after other projects are built ?

☐ Build periodically ?

☐ GitHub hook trigger for GITScm polling ?

☒ Poll SCM ?

☐ Ignore post-commit hooks ?

Schedule ?

5

\* \* \* \* \*

Do you really mean "every minute" when you say "\* \* \* \* \*"? Perhaps you meant "H \* \* \* \* \*" to poll once per hour

Would last have run at Wednesday, 27 August, 2025 at 4:54:00 pm India Standard Time; would next run at Wednesday, 27 August, 2025 at 4:55:00 pm India Standard Time.

Save

Apply

29°C

Light rain

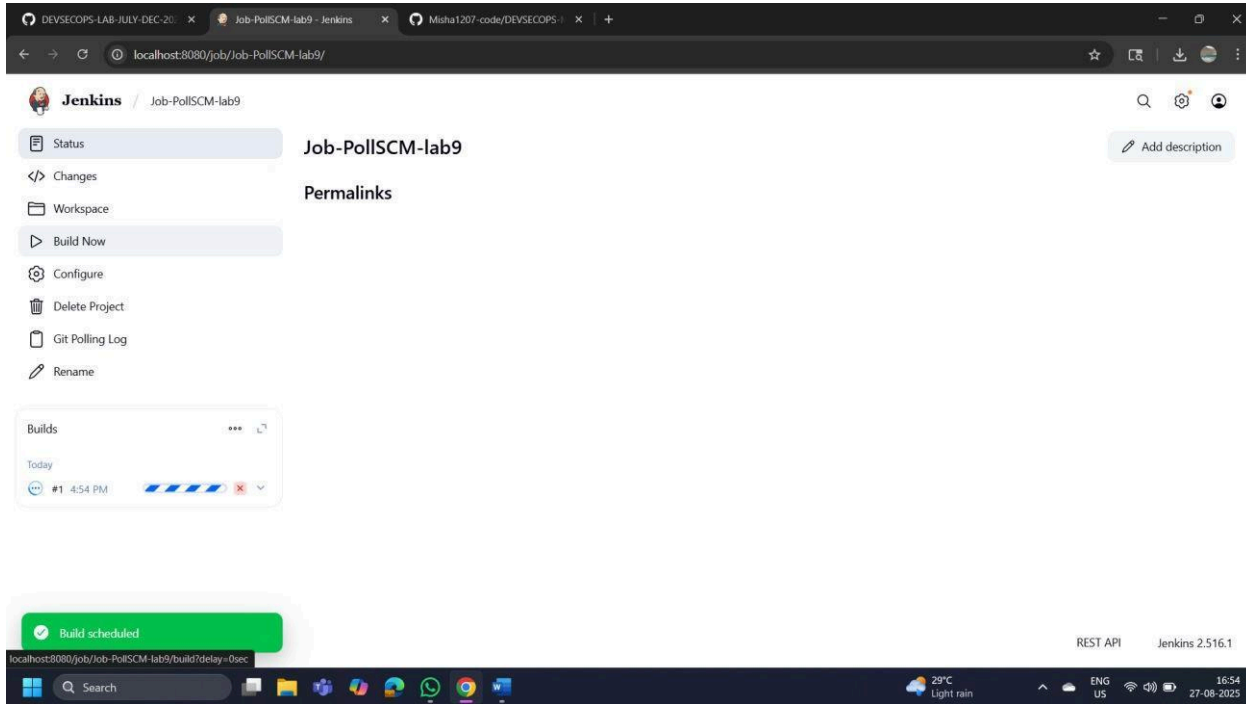
ENG

US

16:54

27-08-2025

1.6 Then, click on **Build Now** to configure a build, and you will be able to view the successful build in the **Console output** section.



```
Started by user admin
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\jenkins\workspace\Job-PollSCM-lab9
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/Misha1207-code/DEVSECOPS-MAVEN-REPO
> git.exe init C:\ProgramData\Jenkins\jenkins\workspace\Job-PollSCM-lab9 # timeout=10
Fetching upstream changes from https://github.com/Misha1207-code/DEVSECOPS-MAVEN-REPO
> git.exe --version # timeout=10
> git --version # 'git version 2.47.1.windows.2'
> git.exe fetch --tags --force --progress -- https://github.com/Misha1207-code/DEVSECOPS-MAVEN-REPO +refs/heads/*:refs/remotes/origin/* #
timeout=10
> git.exe config remote.origin.url https://github.com/Misha1207-code/DEVSECOPS-MAVEN-REPO # timeout=10
> git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision 04a6890beda18d9a0604c2a45835912c4cdb966b (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f 04a6890beda18d9a0604c2a45835912c4cdb966b # timeout=10
Commit message: "div() added"
First time build. Skipping changelog.
Finished: SUCCESS
```

You can see that the build is configured successfully.

By following these steps, you have successfully set up poll SCM configuration in Jenkins for automating build triggering and optimizing resource usage on Jenkins servers.