

ASSIGNMENT-5

Setting Up and Running a Jenkins Pipeline with Git and Maven

Step 1: Clone the Repository

1. Open Command Prompt and navigate to a desired directory.
2. Run the command:
`git clone https://github.com/Parinithavarshini/Assignment_1.git`
3. Navigate into the cloned directory:
`cd Assignment_1`

Step 2: Access Jenkins Inside Docker

1. Run the following command to enter the Jenkins container:
`docker exec -it jenkins-blueocean /bin/bash`
2. Navigate to the cloned repository inside the container:
`cd Assignment_1`

Step 3: Install Maven

1. Inside the Jenkins container, install Maven by running:
`sudo apt-get install -y maven`
This will install Maven and set up dependencies.
2. Verify Maven installation:
`mvn -version`

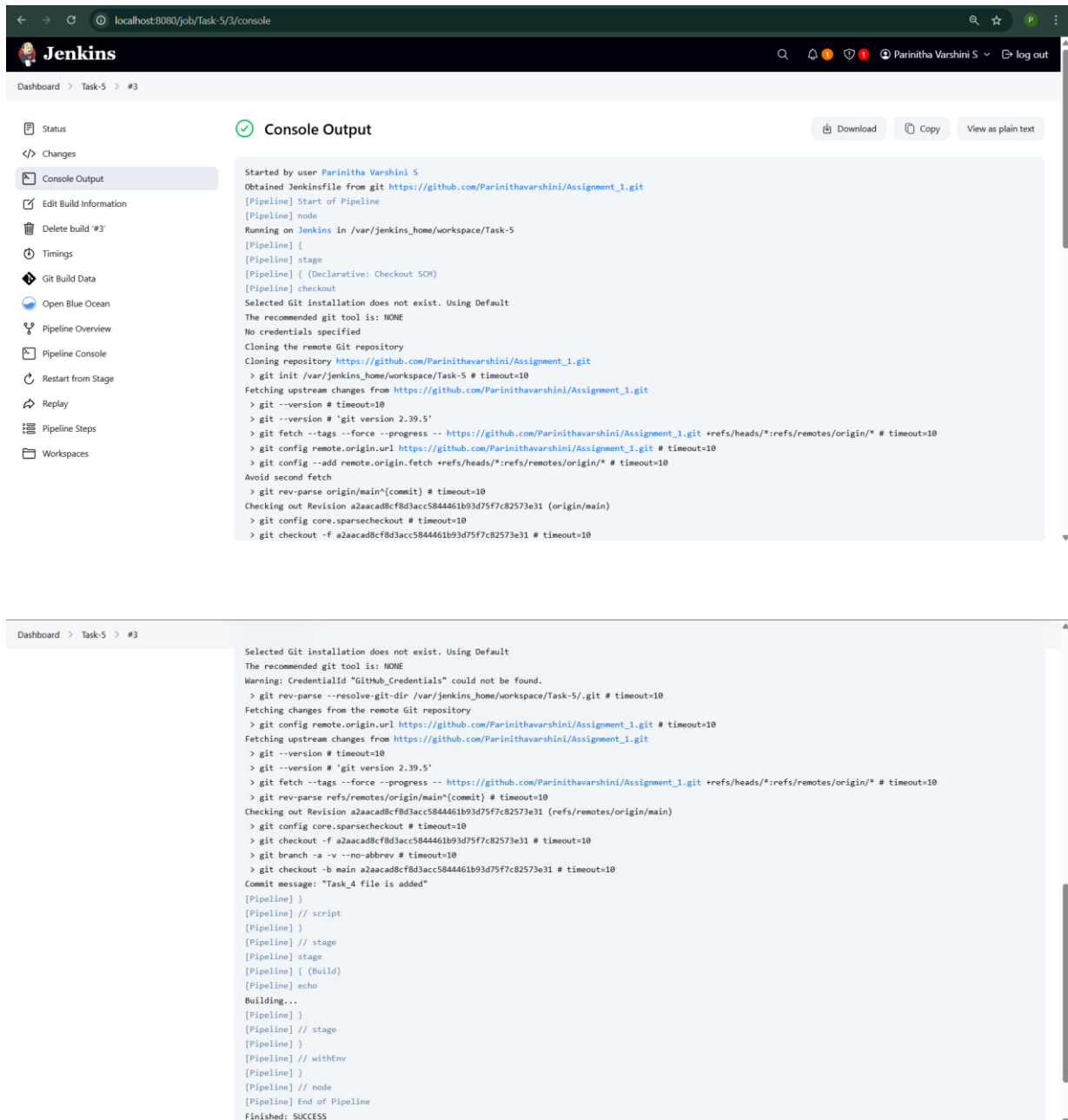
Step 4: Run the Jenkins Pipeline

1. Open Jenkins in your browser.
2. Click on New Item.
3. Enter the item name (e.g., Task-5).
4. Select Pipeline and click OK.
5. Open the newly created pipeline (Task-5).
6. Click on Configure.
7. Under the Pipeline section, choose Pipeline script from SCM.
8. Select Git as the SCM.
Paste your Git repository URL (e.g.,
`https://github.com/Parinithavarshini/Assignment_1.git`)
9. Click Save.
10. Click on Build Now to start the pipeline.

Step 5: Verify the Build Output

1. Monitor the build progress under Build History in Jenkins.
2. If the build is successful, verify the output.
3. If there are errors, check the logs in Console Output and fix any issues.

OUTPUT:



The screenshot displays the Jenkins web interface. The top navigation bar shows the Jenkins logo and the user 'Parinitha Varshini S'. The breadcrumb trail indicates the current view is 'Dashboard > Task-5 > #3'. On the left sidebar, the 'Console Output' option is selected. The main panel, titled 'Console Output' with a green checkmark icon, shows the build log. The log begins with 'Started by user Parinitha Varshini S' and 'Obtained Jenkinsfile from git https://github.com/Parinithavarshini/Assignment_1.git'. It then shows the pipeline stages: 'Start of Pipeline', 'node', and 'Jenkins in /var/jenkins_home/workspace/Task-5'. The 'Checkout' stage is detailed, showing the selection of the default Git installation, cloning the repository, and fetching upstream changes. The build concludes with 'Finished: SUCCESS'. The bottom portion of the image shows a continuation of the console output, including the 'script' stage, 'stage' declarations, and the 'Build' stage, which includes an 'echo' command and the final 'Finished: SUCCESS' status.

```
Started by user Parinitha Varshini S
Obtained Jenkinsfile from git https://github.com/Parinithavarshini/Assignment_1.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/jenkins_home/workspace/Task-5
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/Parinithavarshini/Assignment_1.git
> git init /var/jenkins_home/workspace/Task-5 # timeout=10
Fetching upstream changes from https://github.com/Parinithavarshini/Assignment_1.git
> git --version # timeout=10
> git --version # 'git version 2.39.5'
> git fetch --tags --force --progress -- https://github.com/Parinithavarshini/Assignment_1.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/Parinithavarshini/Assignment_1.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse origin/main^{commit} # timeout=10
Checking out Revision a2aacad8cf8d3acc5844461b93d75f7c82573e31 (origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f a2aacad8cf8d3acc5844461b93d75f7c82573e31 # timeout=10

Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
Warning: CredentialId "GitHub_Credentials" could not be found.
> git rev-parse --resolve-git-dir /var/jenkins_home/workspace/Task-5/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/Parinithavarshini/Assignment_1.git # timeout=10
Fetching upstream changes from https://github.com/Parinithavarshini/Assignment_1.git
> git --version # timeout=10
> git --version # 'git version 2.39.5'
> git fetch --tags --force --progress -- https://github.com/Parinithavarshini/Assignment_1.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision a2aacad8cf8d3acc5844461b93d75f7c82573e31 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f a2aacad8cf8d3acc5844461b93d75f7c82573e31 # timeout=10
> git branch -a -v --no-abbrev # timeout=10
> git checkout -b main a2aacad8cf8d3acc5844461b93d75f7c82573e31 # timeout=10
Commit message: "Task_4 file is added"
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] echo
Building...
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```