

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/UreshVel/Docker.git`
3. Navigate into the cloned directory: `cd Docker`

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 Docker`

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.

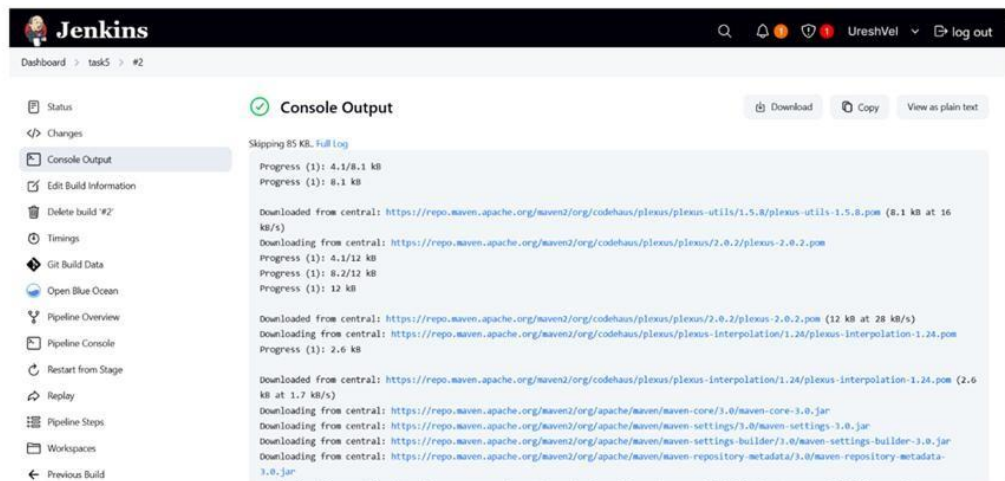
- Open Jenkins in your browser.
2. Click on New Item.
 3. Enter the item name (e.g., task-5).
 4. Select Pipeline and click OK.
 5. Open the newly created pipeline (task-5).

6. Click on Configure.
7. Under the Pipeline section, choose Pipeline script from SCM.
8. Select Git as the SCM.
9. Paste your Git repository URL (e.g., <https://github.com/UreshVel/Docker.git>).
10. Click Save.
11. 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:



```
[1;34mINFO[1;32mBUILD SUCCESS[1;34mINFO[1;34m-----[1;34mINFO[1;34m Total time: 01:59 min
[1;34mINFO[1;34m Finished at: 2025-03-21T09:20:17Z
[1;34mINFO[1;34m-----[1;34m
[0m[0m
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] cleanWs
[WS-CLEANUP] Deleting project workspace...
[WS-CLEANUP] Deferred wipeout is used...
[WS-CLEANUP] done
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```