Lab3

what are different method to trigger pipeline in jenkins ?

1. Trigger builds remotely .
2. Build periodically .
3. GitHub webhook trigger .
4. Poll SCM
5. Build after other piplines are built .
6. Build manually

what is the benefit of using master-slave architecture rather than building on master only ?

- Produce backups — Obviously, the major utility of master-slave database architecture is its ability to provide reliable backups through their chain of slave databases. The slave database can be turned off without affecting the master database's activities. This is because live data snapshots will be replicated to the slave database, and data resources will remain intact even if the master database fails.
- Scale out the application — When your number of users increases and your app's usage skyrockets, it's critical to scale out your app so that your audience has a consistent experience. By splitting your data load across numerous databases, the master-slave database architecture can help you scale out your application.
- Split your work without causing a mess — You can use your slave nodes to collect the necessary data and generate a complete report, while the master node pushes new updates and features to your application server.
- Top-notch performance — It is extremely fast and fetches data without any issues, imposing no restrictions regarding performance and time to load the data.

what is different between authorization and authentication?

- Authentication verifies the identity of a user or service

- Authorization determines their access rights

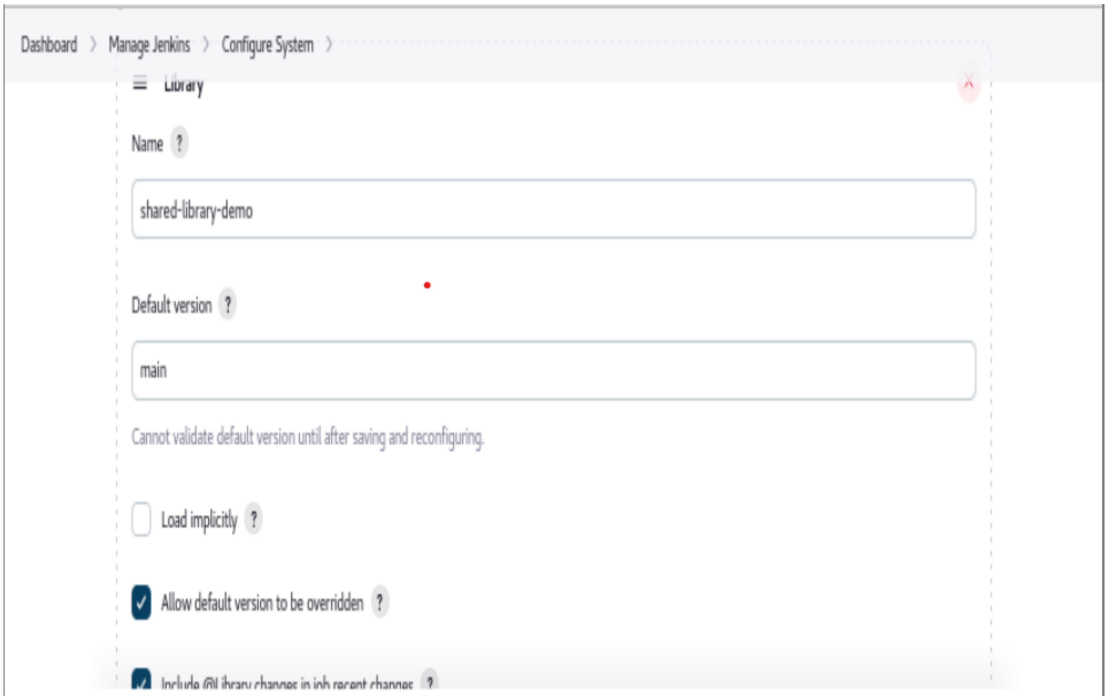
what is the benefit of making organization job in jenkins?

- Organization Folders enable Jenkins to monitor an entire GitHub Organization, Bitbucket

Team/Project, GitLab organization, or Gitea organization

- automatically create new Multibranch Pipelines for repositories which contain branches

and pull requests containing a Jenkinsfile make jenkins-shared-library and make your jenkinsfile which you used in lab2 to point to this library



Dashboard › Manage Jenkins › Configure System ›

≡ Library

Name ?

shared-library-demo

Default version ?

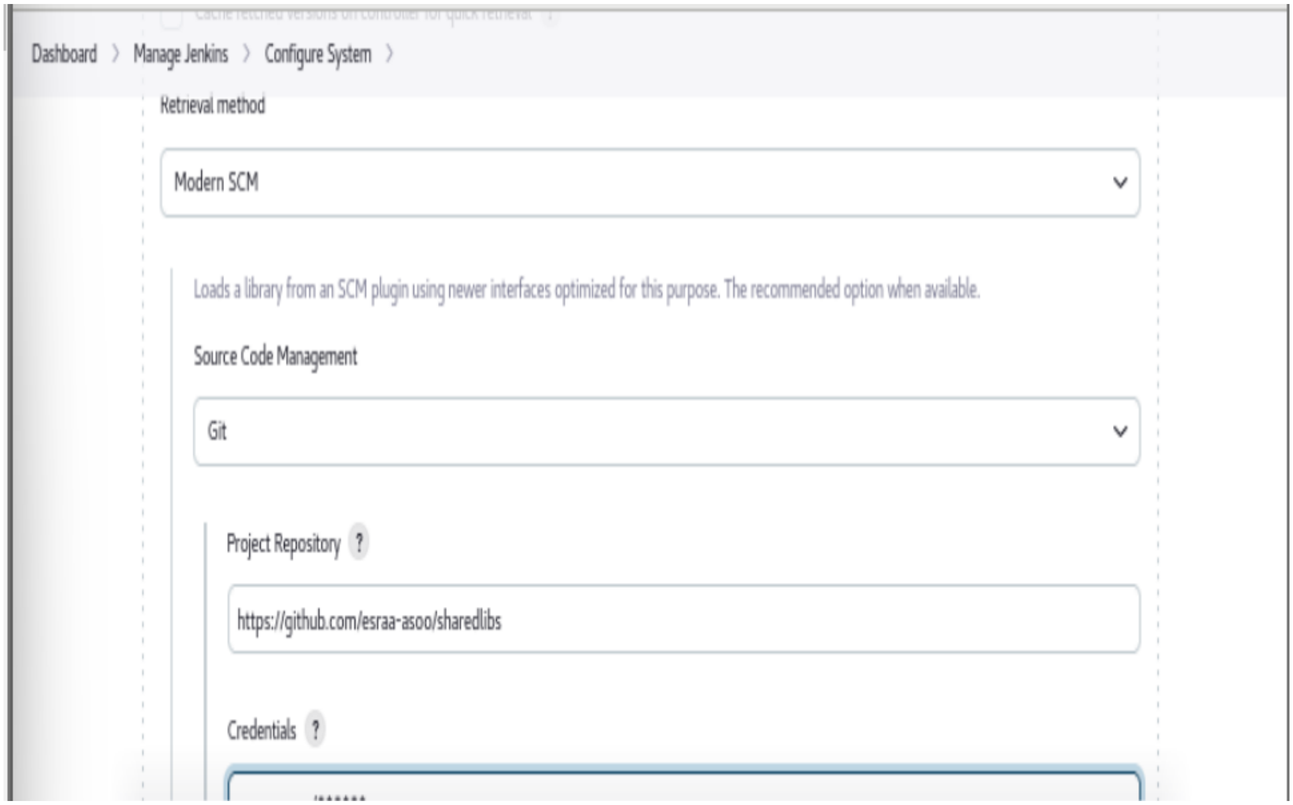
main

Cannot validate default version until after saving and reconfiguring.

□ Load implicitly ?

☑ Allow default version to be overridden ?

☑ Include @Library changes in job recent changes ?



□ Cache fetched versions on controller for quick retrieval ?

Dashboard › Manage Jenkins › Configure System ›

Retrieval method

Modern SCM

Loads a library from an SCM plugin using newer interfaces optimized for this purpose. The recommended option when available.

Source Code Management

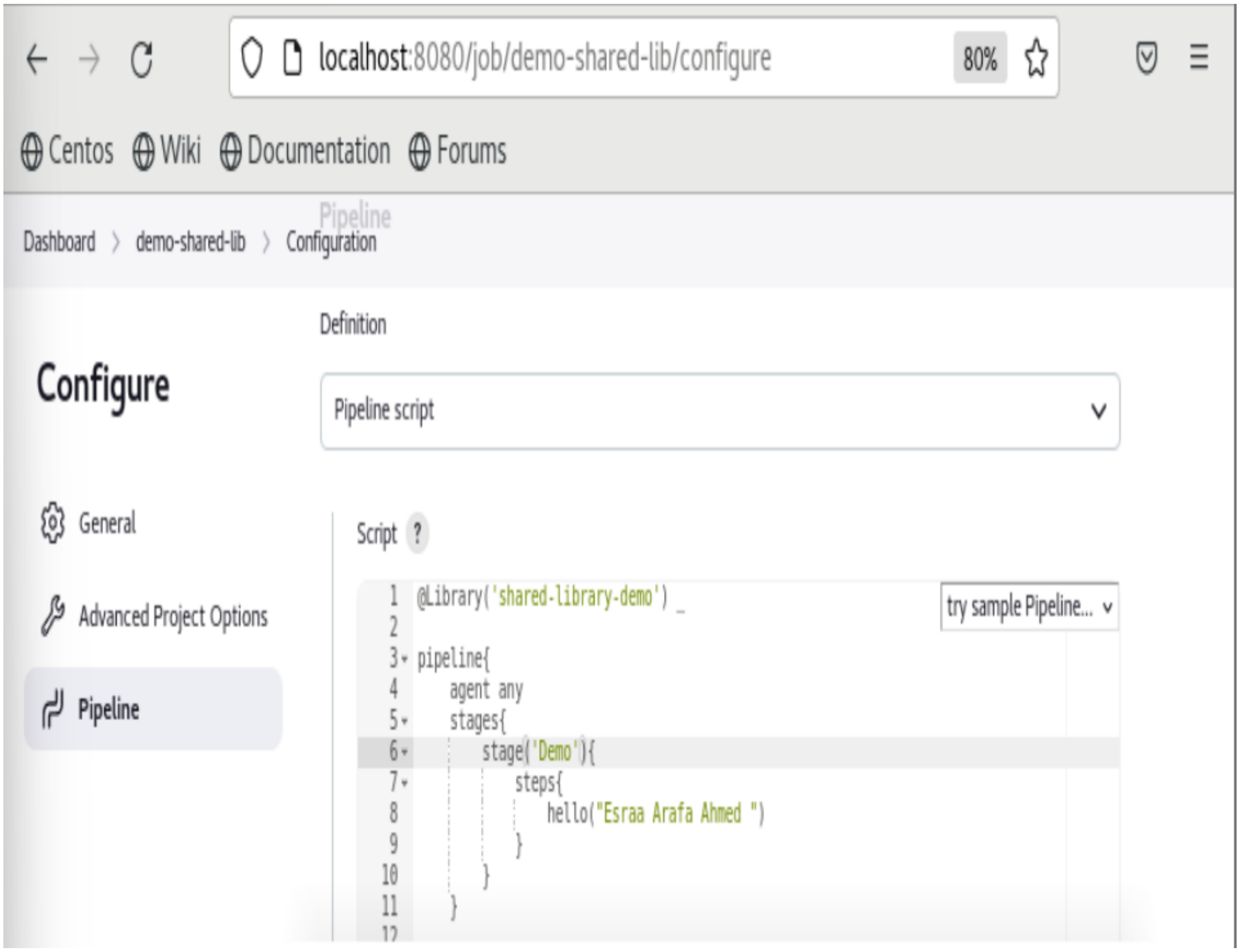
Git

Project Repository ?

https://github.com/esraa-asoo/sharedlibs

Credentials ?

******

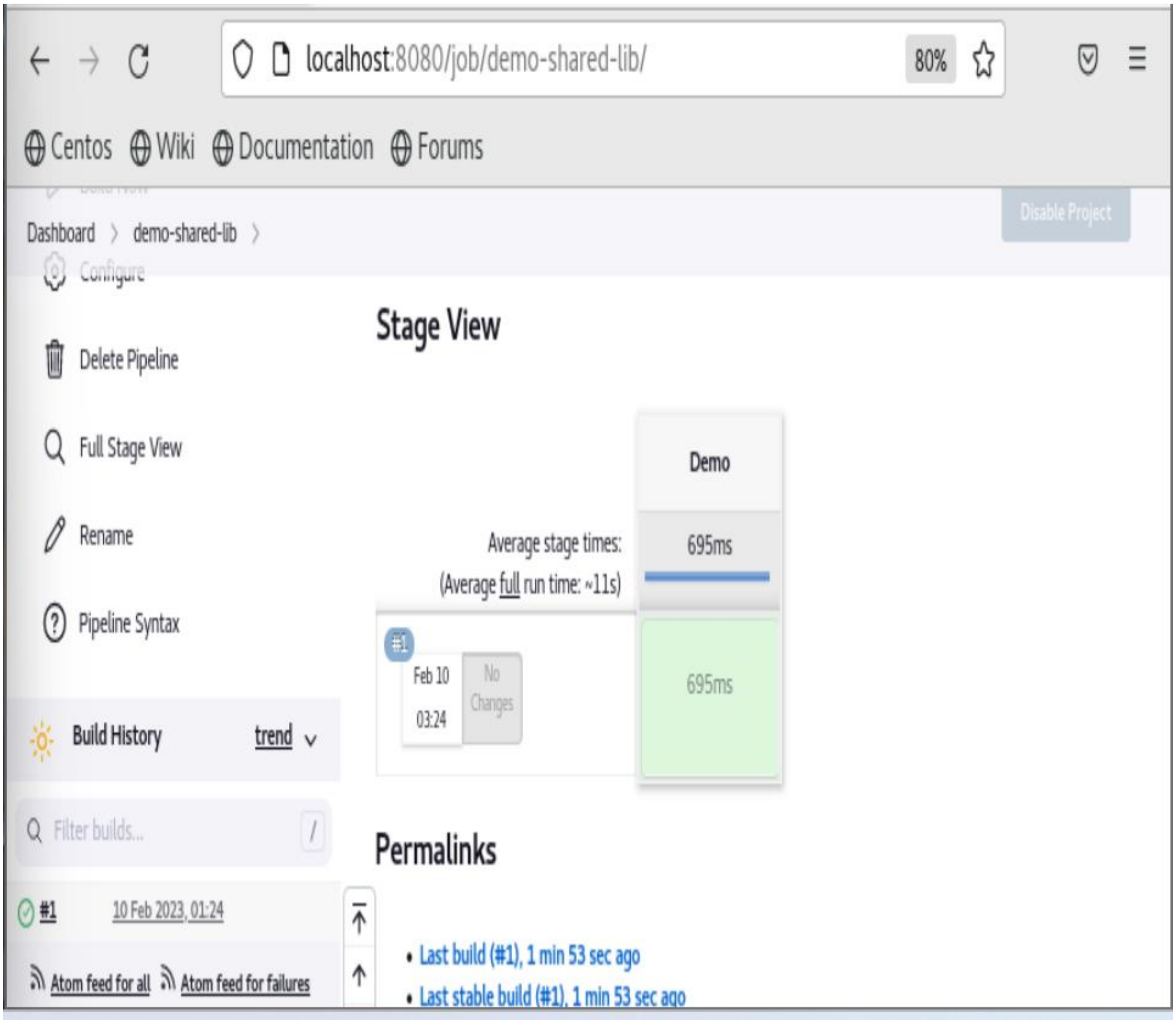


Definition

Pipeline script

```
@Library('shared-library-demo') _

pipeline{
    agent any
    stages{
        stage('Demo'){
            steps{
                hello("Esraa Arafa Ahmed ")
            }
        }
    }
```



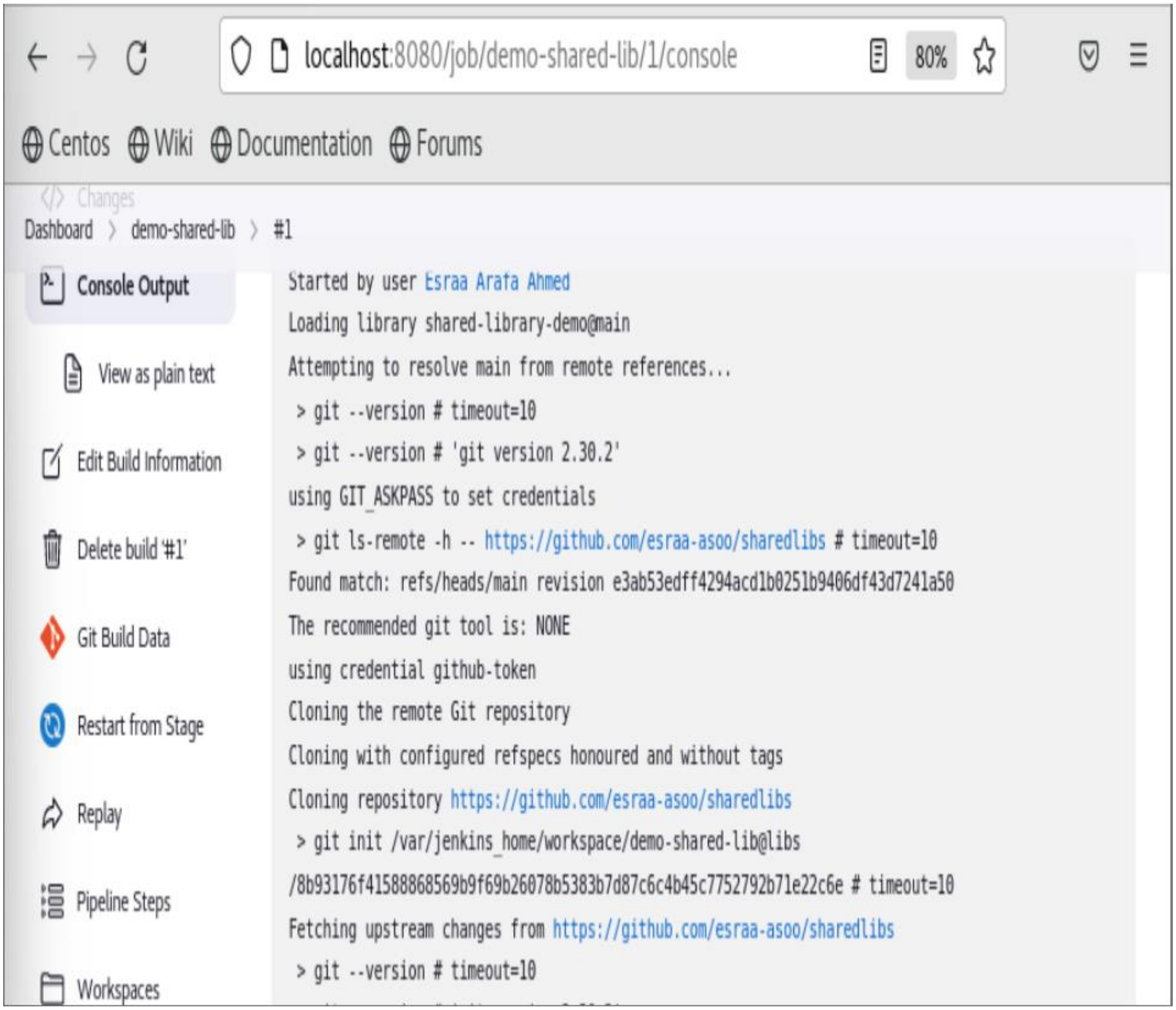
Stage View

	Demo
Average stage times: (Average full run time: ~11s)	695ms
#1 Feb 10 03:24 No Changes	695ms

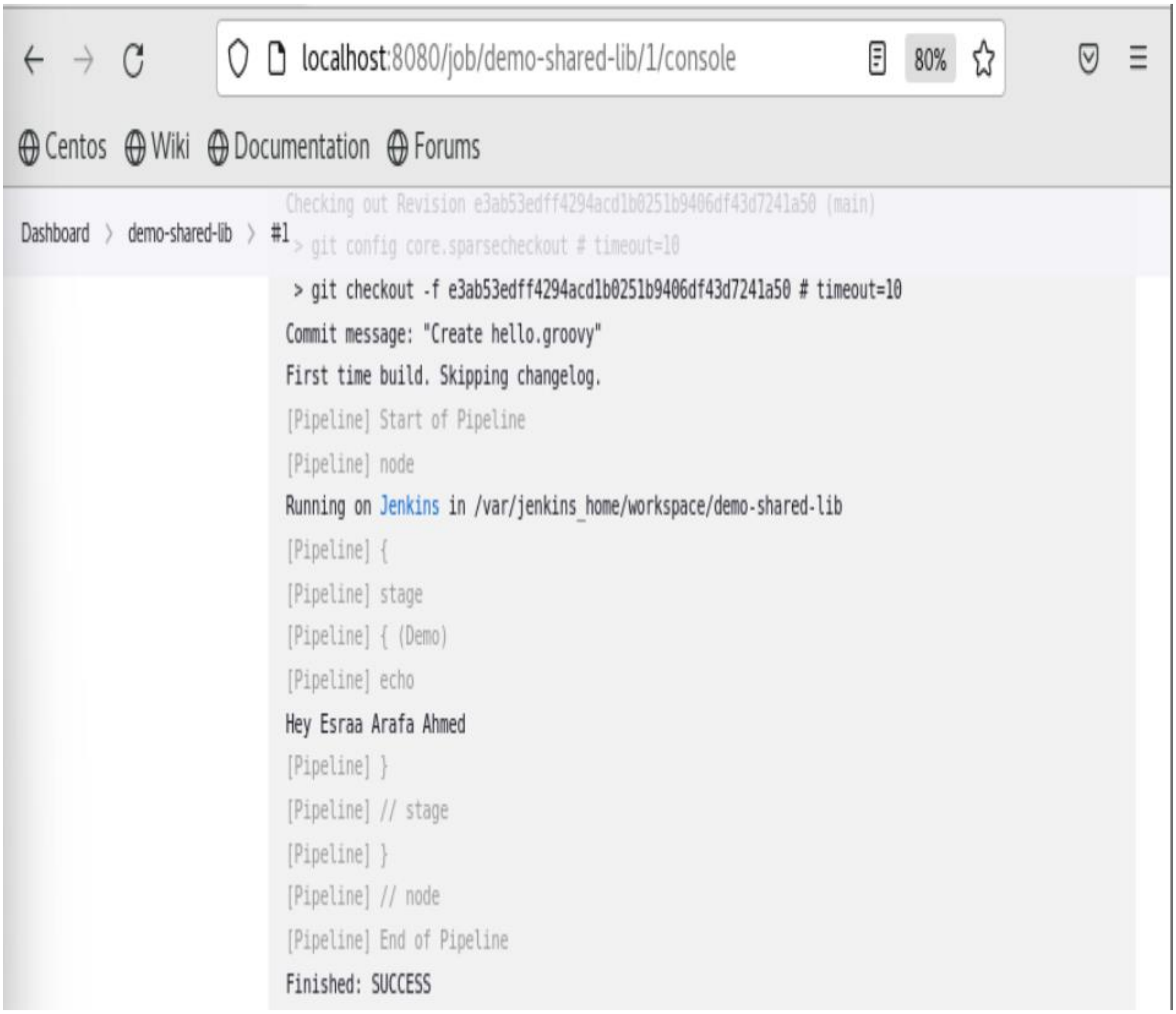
#1 10 Feb 2023, 01:24

Permalinks

- Last build (#1), 1 min 53 sec ago
- Last stable build (#1), 1 min 53 sec ago



```
Started by user Esraa Arafa Ahmed
Loading library shared-library-demo@main
Attempting to resolve main from remote references...
 > git --version # timeout=10
 > git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials
 > git ls-remote -h -- https://github.com/esraa-asoo/sharedlibs # timeout=10
Found match: refs/heads/main revision e3ab53edff4294acd1b0251b9406df43d7241a50
The recommended git tool is: NONE
using credential github-token
Cloning the remote Git repository
Cloning with configured refspecs honoured and without tags
Cloning repository https://github.com/esraa-asoo/sharedlibs
 > git init /var/jenkins_home/workspace/demo-shared-lib@libs
/8b93176f41588868569b9f69b26078b5383b7d87c6c4b45c7752792b71e22c6e # timeout=10
Fetching upstream changes from https://github.com/esraa-asoo/sharedlibs
 > git --version # timeout=10
```



```
Checking out Revision e3ab53edff4294acd1b0251b9406df43d7241a50 (main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f e3ab53edff4294acd1b0251b9406df43d7241a50 # timeout=10
Commit message: "Create hello.groovy"
First time build. Skipping changelog.
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/jenkins_home/workspace/demo-shared-lib
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Demo)
[Pipeline] echo
Hey Esraa Arafa Ahmed
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```