1) 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.

---

2) 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.

---

3) what is the difference between authorization and authentication?

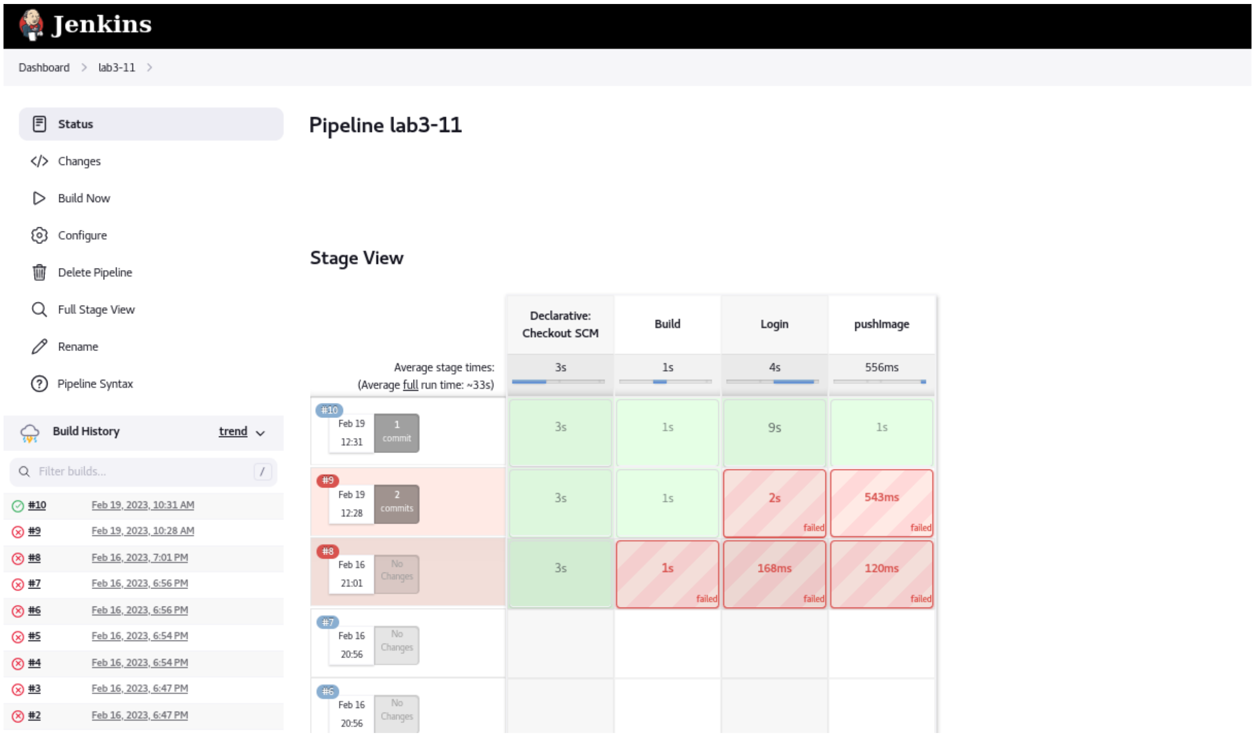
- Authentication verifies the identity of a user or service
- Authorization determines their access rights

---

4) 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

---



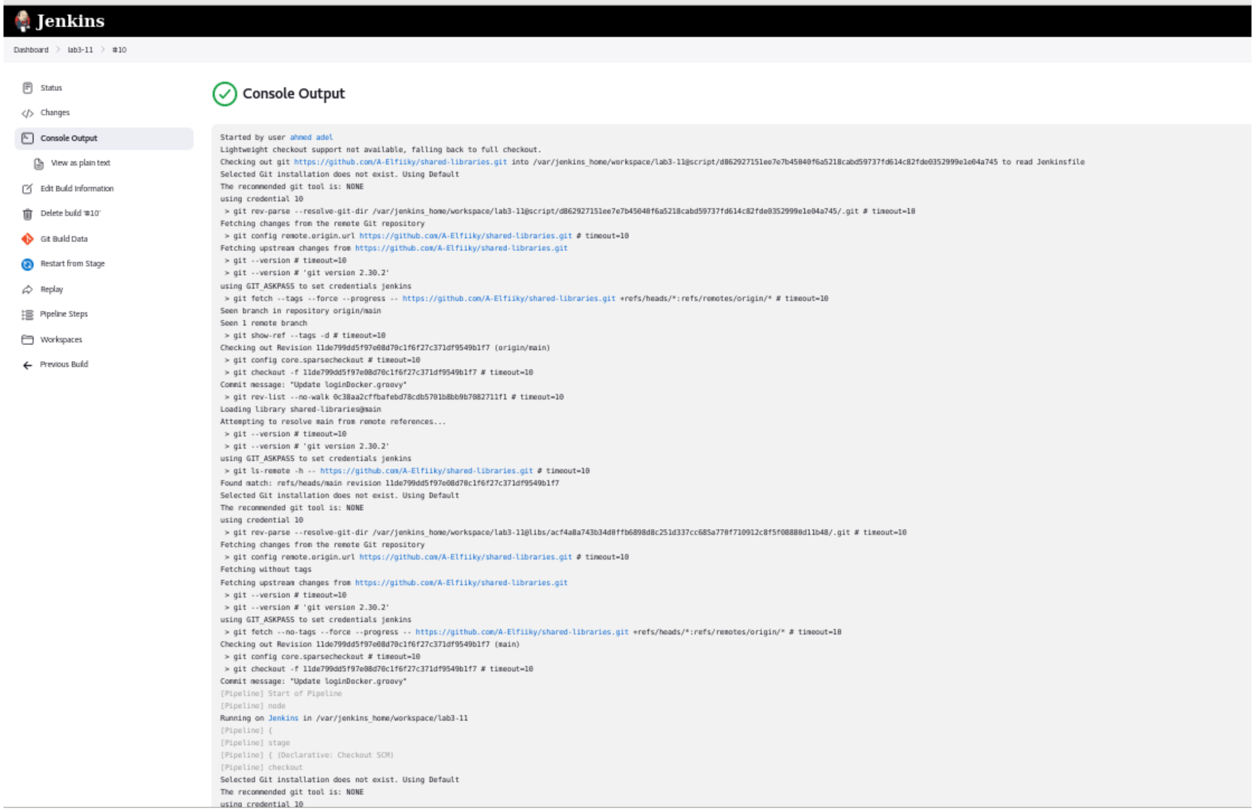
Pipeline lab3-11

Stage View

	Declarative: Checkout SCM	Build	Login	pushImage
Average stage times: (Average full run time: ~33s)	3s	1s	4s	556ms
#10 Feb 19 12:31 1 commit	3s	1s	9s	1s
#9 Feb 19 12:28 2 commits	3s	1s	2s failed	543ms failed
#8 Feb 16 21:01 No Changes	3s	1s failed	168ms failed	120ms failed
#7 Feb 16 20:56 No Changes				
#6 Feb 16 20:56 No Changes				

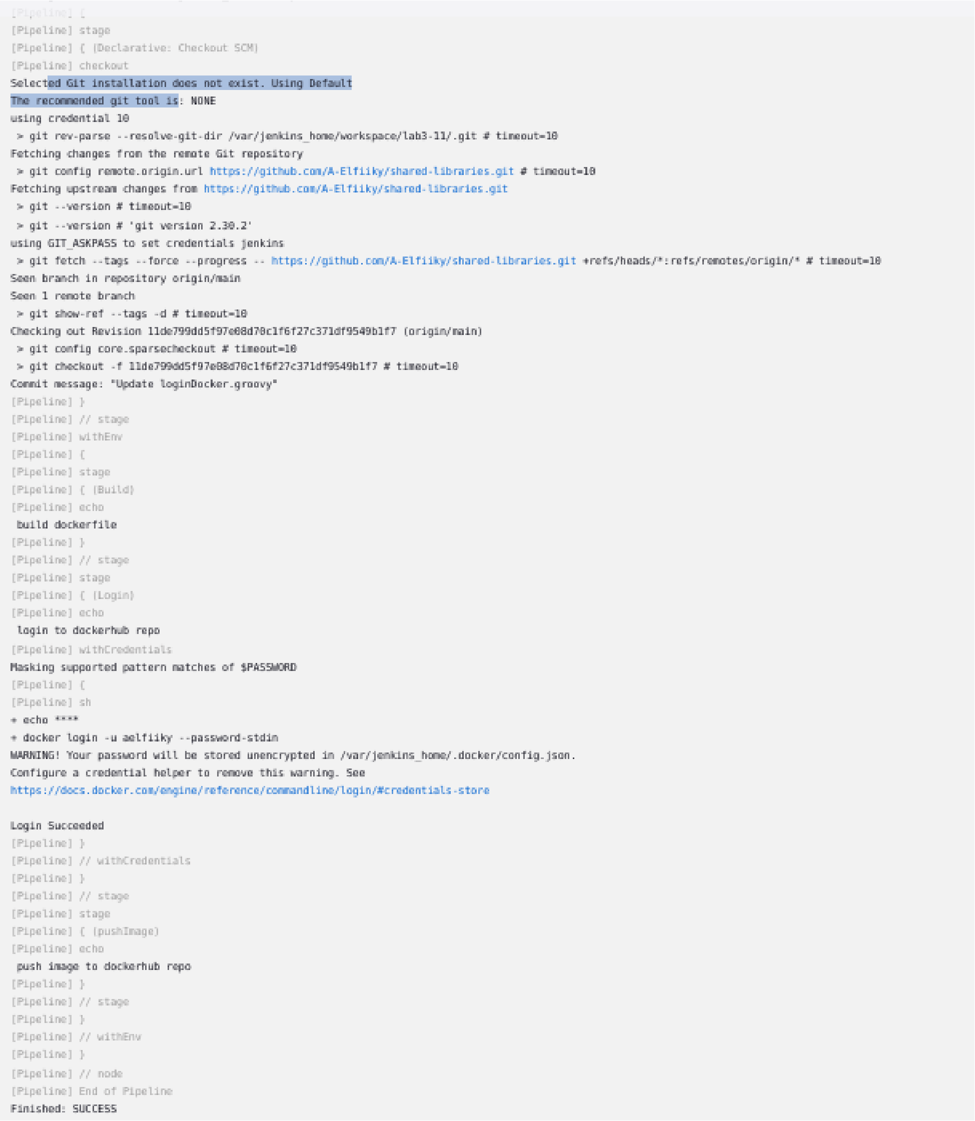
✓ #10	Feb 19, 2023, 10:31 AM
⊗ #9	Feb 19, 2023, 10:28 AM
⊗ #8	Feb 16, 2023, 7:01 PM
⊗ #7	Feb 16, 2023, 6:56 PM
⊗ #6	Feb 16, 2023, 6:56 PM
⊗ #5	Feb 16, 2023, 6:54 PM
⊗ #4	Feb 16, 2023, 6:54 PM
⊗ #3	Feb 16, 2023, 6:47 PM
⊗ #2	Feb 16, 2023, 6:47 PM

o/P



Console Output

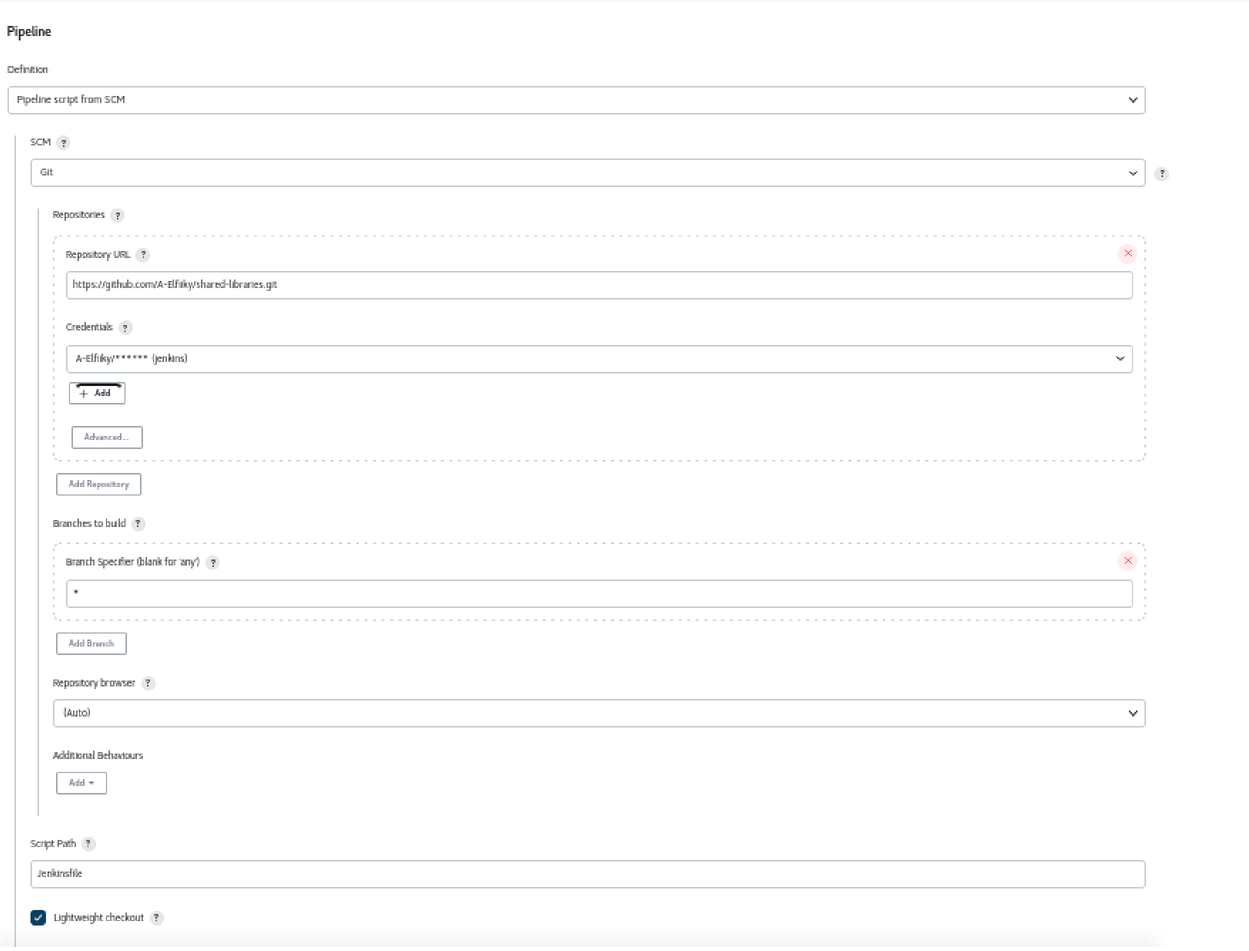
```
Started by user ahmed adel
Lightweight checkout support not available, falling back to full checkout.
Checking out git https://github.com/A-Elfiiky/shared-libraries.git into /var/jenkins_home/workspace/lab3-11@script/d862927151ee7e7b45040f6a5218cabd59737fd614c82fde0352999e1e04a745 to read Jenkinsfile
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential 10
 > git rev-parse --resolve-git-dir /var/jenkins_home/workspace/lab3-11@script/d862927151ee7e7b45040f6a5218cabd59737fd614c82fde0352999e1e04a745/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/A-Elfiiky/shared-libraries.git # timeout=10
Fetching upstream changes from https://github.com/A-Elfiiky/shared-libraries.git
 > git --version # timeout=10
 > git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials jenkins
 > git fetch --tags --force --progress -- https://github.com/A-Elfiiky/shared-libraries.git +refs/heads/*:refs/remotes/origin/* # timeout=10
Seen branch in repository origin/main
Seen 1 remote branch
 > git show-ref --tags -d # timeout=10
Checking out Revision 11de799dd5f97e08d70c1f6f27c371df9549b1f7 (origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 11de799dd5f97e08d70c1f6f27c371df9549b1f7 # timeout=10
Commit message: "Update loginDocker.groovy"
 > git rev-list --no-walk 0c38aa2cffbafebd78cdb5701b8bb9b7082711f1 # timeout=10
Loading library shared-libraries@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 jenkins
 > git ls-remote -h -- https://github.com/A-Elfiiky/shared-libraries.git # timeout=10
Found match: refs/heads/main revision 11de799dd5f97e08d70c1f6f27c371df9549b1f7
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential 10
 > git rev-parse --resolve-git-dir /var/jenkins_home/workspace/lab3-11@libs/acf4a8a743b34d0ffb6898d8c251d337cc685a778f710912c8f5f08888d11b48/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/A-Elfiiky/shared-libraries.git # timeout=10
Fetching without tags
Fetching upstream changes from https://github.com/A-Elfiiky/shared-libraries.git
 > git --version # timeout=10
 > git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials jenkins
 > git fetch --no-tags --force --progress -- https://github.com/A-Elfiiky/shared-libraries.git +refs/heads/*:refs/remotes/origin/* # timeout=10
Checking out Revision 11de799dd5f97e08d70c1f6f27c371df9549b1f7 (main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 11de799dd5f97e08d70c1f6f27c371df9549b1f7 # timeout=10
Commit message: "Update loginDocker.groovy"
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/jenkins_home/workspace/lab3-11
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential 10
```



```
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential 10
 > git rev-parse --resolve-git-dir /var/jenkins_home/workspace/lab3-11/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/A-Elfiiky/shared-libraries.git # timeout=10
Fetching upstream changes from https://github.com/A-Elfiiky/shared-libraries.git
 > git --version # timeout=10
 > git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials jenkins
 > git fetch --tags --force --progress -- https://github.com/A-Elfiiky/shared-libraries.git +refs/heads/*:refs/remotes/origin/* # timeout=10
Seen branch in repository origin/main
Seen 1 remote branch
 > git show-ref --tags -d # timeout=10
Checking out Revision 11de799dd5f97e88d70c1f6f27c371df9549b1f7 (origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 11de799dd5f97e88d70c1f6f27c371df9549b1f7 # timeout=10
Commit message: "Update loginDocker.groovy"
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] echo
 build dockerfile
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Login)
[Pipeline] echo
 login to dockerhub repo
[Pipeline] withCredentials
Masking supported pattern matches of $PASSWORD
[Pipeline] {
[Pipeline] sh
+ echo ****
+ docker login -u aelfiiky --password-stdin
WARNING! Your password will be stored unencrypted in /var/jenkins_home/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (pushImage)
[Pipeline] echo
 push image to dockerhub repo
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Pipeline setting



Pipeline

Definition

Pipeline script from SCM

SCM

Git

Repositories

Repository URL

https://github.com/A-Elfiky/shared-libraries.git

Credentials

A-Elfiky/****** (jenkins)

Branches to build

Branch Specifier (blank for 'any')

*

Repository browser

(Auto)

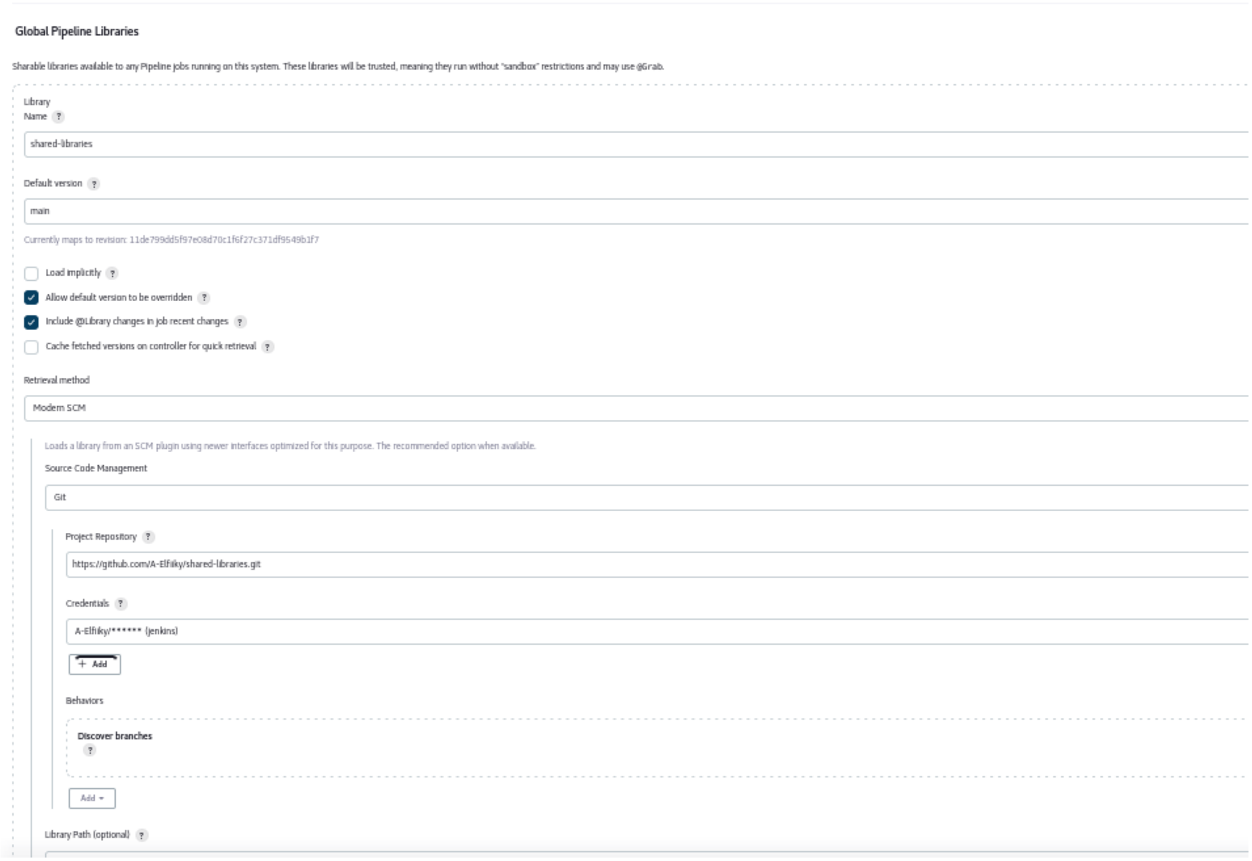
Additional Behaviours

Script Path

Jenkinsfile

☑ Lightweight checkout

Shared library setting >> jenkins manage >> global pipeline configuration



Global Pipeline Libraries

Sharable libraries available to any Pipeline jobs running on this system. These libraries will be trusted, meaning they run without "sandbox" restrictions and may use @Grab.

Library
Name ?

shared-libraries

Default version ?

main

Currently maps to revision: 11de799dd5f97e08d70c1f6f27c371df9549b1f7

□ Load implicitly ?

☑ Allow default version to be overridden ?

☑ Include @Library changes in job recent changes ?

□ Cache fetched versions on controller for quick retrieval ?

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/A-Elfiky/shared-libraries.git

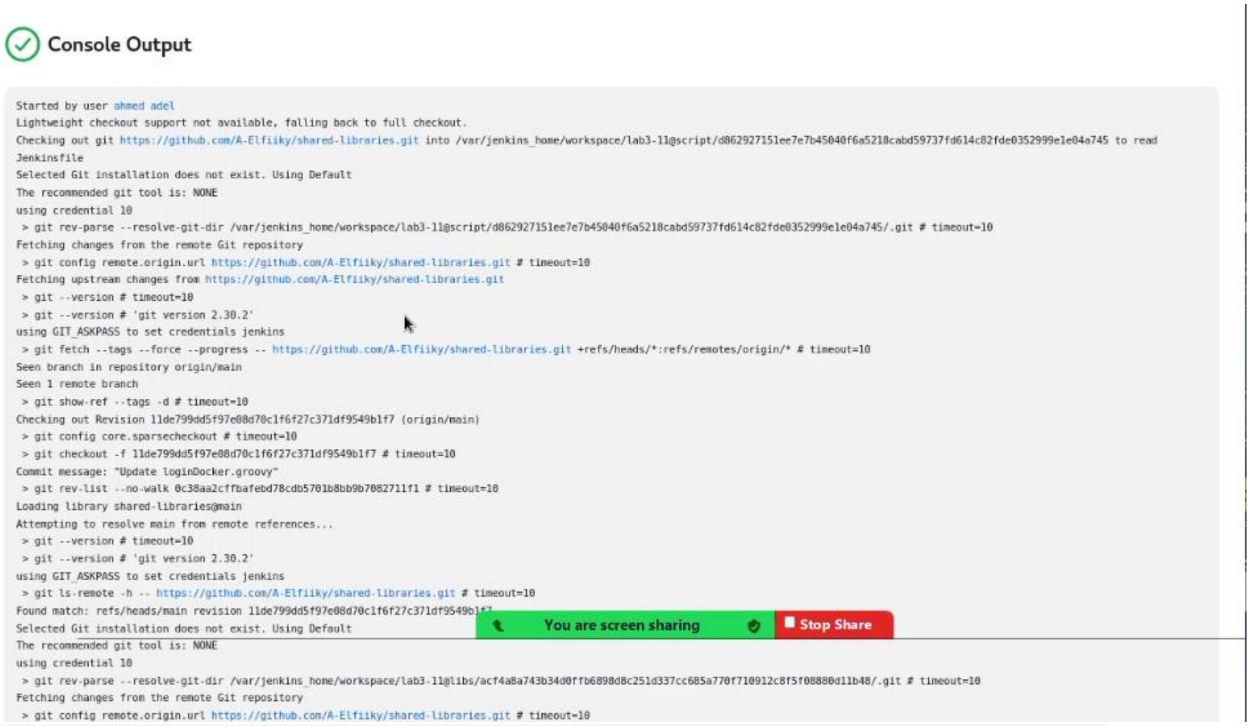
Credentials ?

A-Elfiky/****** (jenkins)

Behaviors

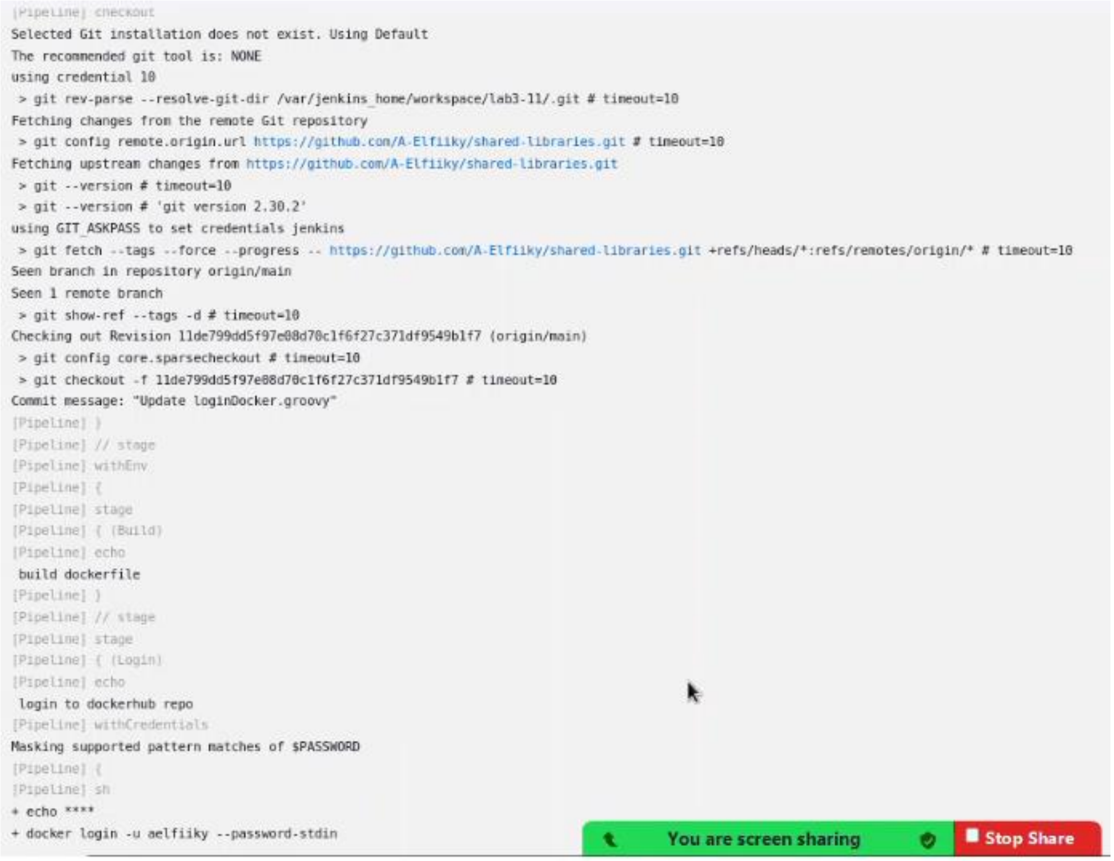
Discover branches ?

Library Path (optional) ?

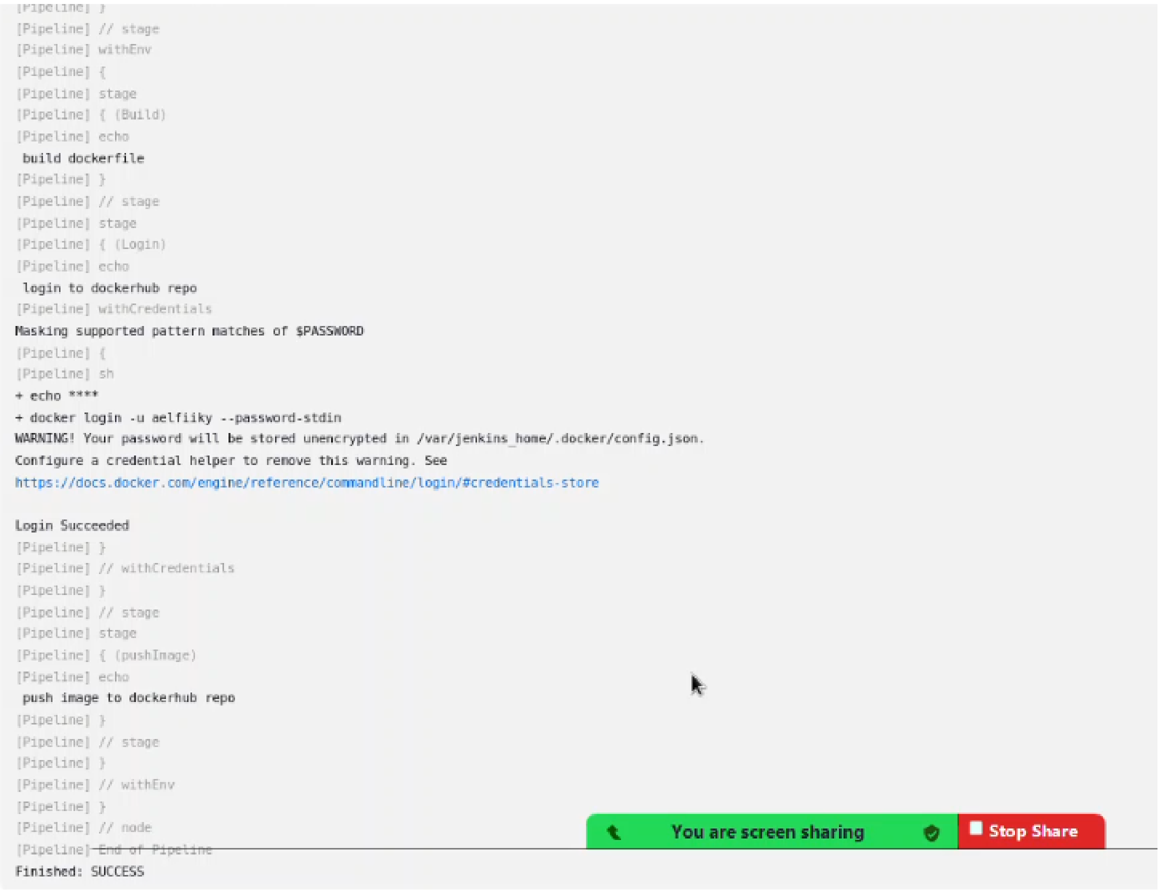


Console Output

```
Started by user ahmed adel
Lightweight checkout support not available, falling back to full checkout.
Checking out git https://github.com/A-Elfiiky/shared-libraries.git into /var/jenkins_home/workspace/lab3-11@script/d862927151ee7e7b45040f6a5218cabd59737fd614c82fde0352999e1e04a745 to read
Jenkinsfile
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential 10
 > git rev-parse --resolve-git-dir /var/jenkins_home/workspace/lab3-11@script/d862927151ee7e7b45040f6a5218cabd59737fd614c82fde0352999e1e04a745/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/A-Elfiiky/shared-libraries.git # timeout=10
Fetching upstream changes from https://github.com/A-Elfiiky/shared-libraries.git
 > git --version # timeout=10
 > git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials jenkins
 > git fetch --tags --force --progress -- https://github.com/A-Elfiiky/shared-libraries.git +refs/heads/*:refs/remotes/origin/* # timeout=10
Seen branch in repository origin/main
Seen 1 remote branch
 > git show-ref --tags -d # timeout=10
Checking out Revision 11de799dd5f97e08d70c1f6f27c371df9549b1f7 (origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 11de799dd5f97e08d70c1f6f27c371df9549b1f7 # timeout=10
Commit message: "Update loginDocker.groovy"
 > git rev-list --no-walk 0c38aa2cffbafebd78cdb5701b8bb9b7082711f1 # timeout=10
Loading library shared-libraries@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 jenkins
 > git ls-remote -h -- https://github.com/A-Elfiiky/shared-libraries.git # timeout=10
Found match: refs/heads/main revision 11de799dd5f97e08d70c1f6f27c371df9549b1f7
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential 10
 > git rev-parse --resolve-git-dir /var/jenkins_home/workspace/lab3-11@libs/acf4a8a743b34d0ffb6898d8c251d337cc685a770f710912c8f5f08880d11b48/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/A-Elfiiky/shared-libraries.git # timeout=10
```



```
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential 10
 > git rev-parse --resolve-git-dir /var/jenkins_home/workspace/lab3-11/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/A-Elfiiky/shared-libraries.git # timeout=10
Fetching upstream changes from https://github.com/A-Elfiiky/shared-libraries.git
 > git --version # timeout=10
 > git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials jenkins
 > git fetch --tags --force --progress -- https://github.com/A-Elfiiky/shared-libraries.git +refs/heads/*:refs/remotes/origin/* # timeout=10
Seen branch in repository origin/main
Seen 1 remote branch
 > git show-ref --tags -d # timeout=10
Checking out Revision 11de799dd5f97e08d70c1f6f27c371df9549b1f7 (origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 11de799dd5f97e08d70c1f6f27c371df9549b1f7 # timeout=10
Commit message: "Update loginDocker.groovy"
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] echo
 build dockerfile
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Login)
[Pipeline] echo
 login to dockerhub repo
[Pipeline] withCredentials
Masking supported pattern matches of $PASSWORD
[Pipeline] {
[Pipeline] sh
+ echo ****
+ docker login -u aelfiiky --password-stdin
```



```
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] echo
 build dockerfile
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Login)
[Pipeline] echo
 login to dockerhub repo
[Pipeline] withCredentials
Masking supported pattern matches of $PASSWORD
[Pipeline] {
[Pipeline] sh
+ echo ****
+ docker login -u aelfiiky --password-stdin
WARNING! Your password will be stored unencrypted in /var/jenkins_home/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (pushImage)
[Pipeline] echo
 push image to dockerhub repo
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```