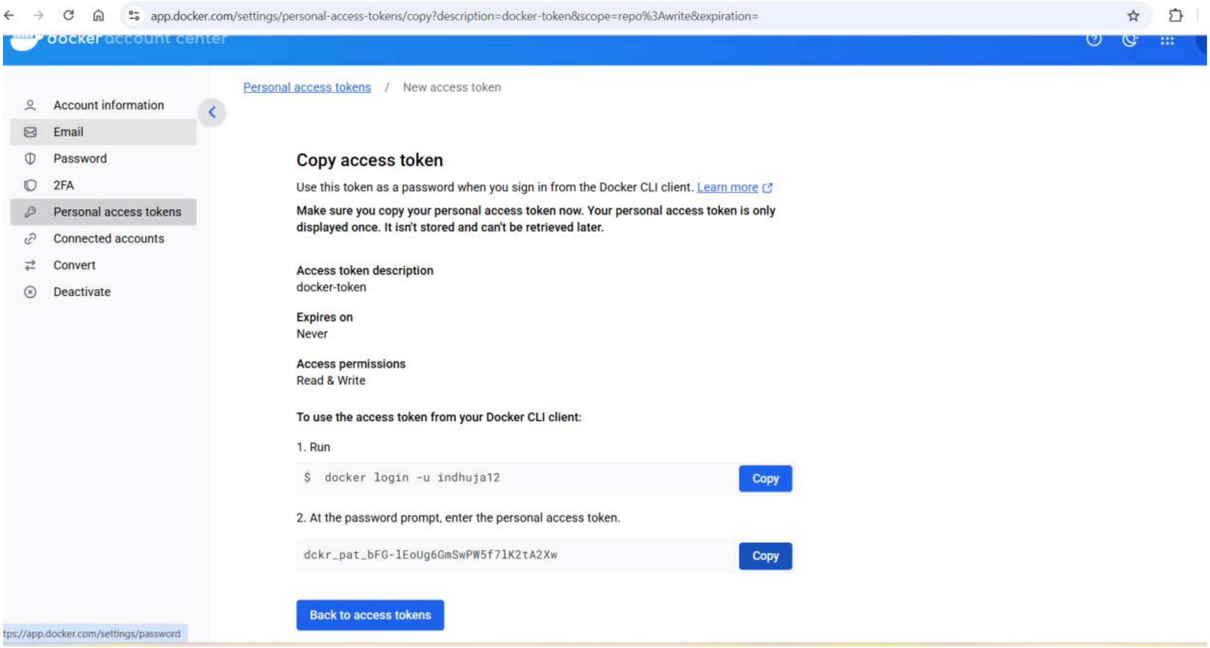
DEVOPS

Task 2


app.docker.com/settings/personal-access-tokens/copy?description=docker-token&scope=repo%3Awrite&expiration=
docker account center
Account information
Email
Password
2FA
Personal access tokens
Connected accounts
Convert
Deactivate
tps://app.docker.com/settings/password

Personal access tokens / New access token

Copy access token

Use this token as a password when you sign in from the Docker CLI client. Learn more

Make sure you copy your personal access token now. Your personal access token is only displayed once. It isn't stored and can't be retrieved later.

Access token description
docker-token

Expires on
Never

Access permissions
Read & Write

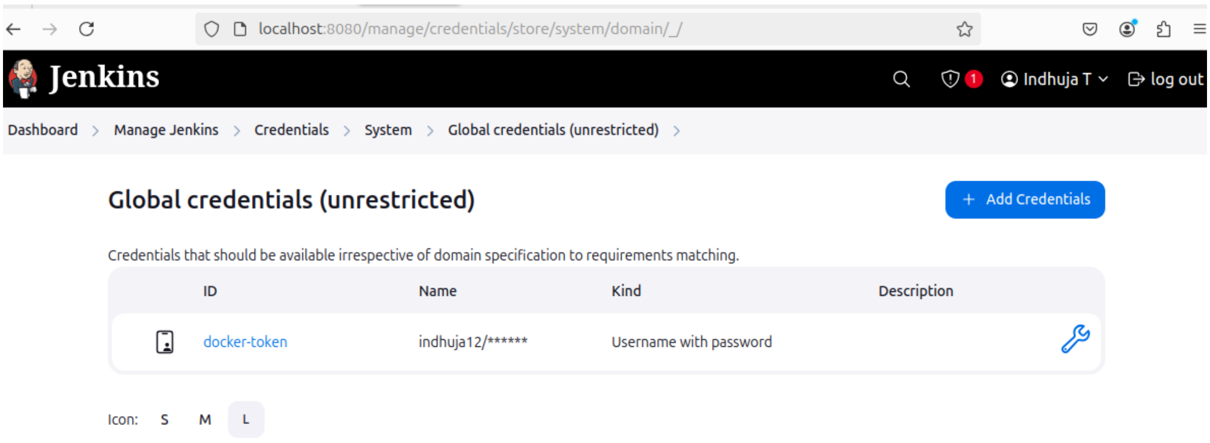
To use the access token from your Docker CLI client:

1. Run

```
$  docker login -u indhuja12
```

2. At the password prompt, enter the personal access token.

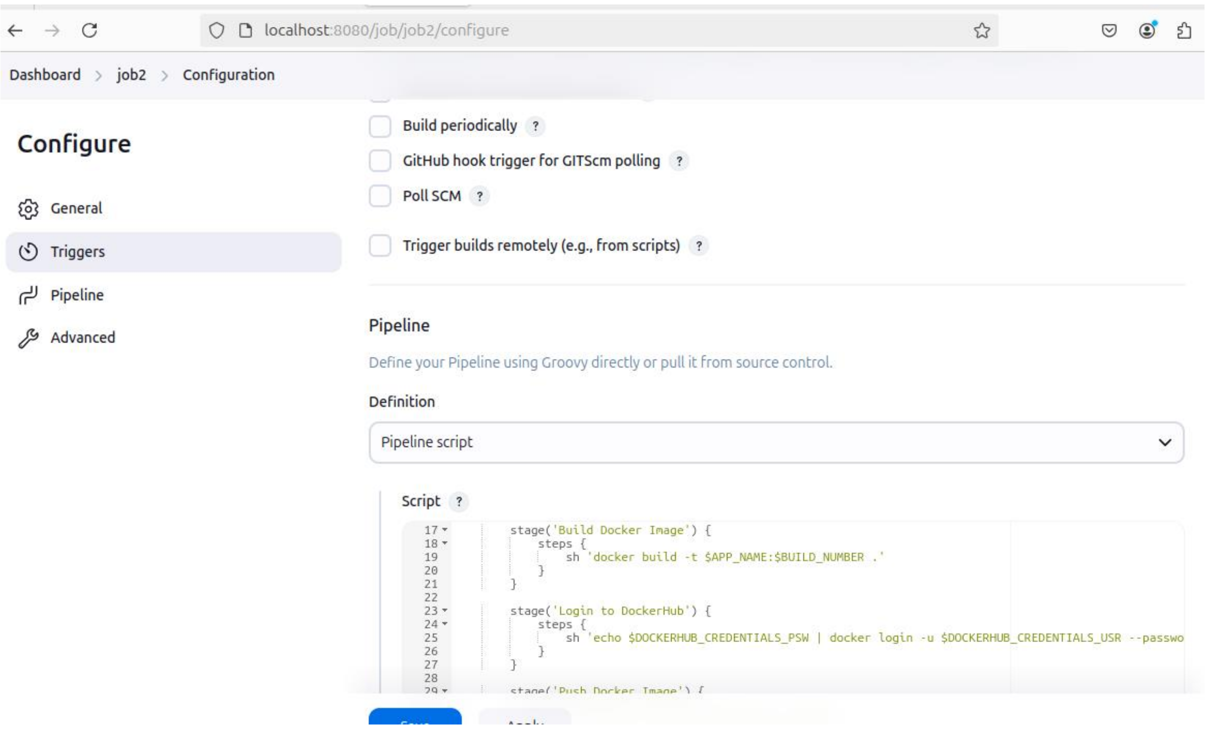
```
dckr_pat_bFG-lEoUg6GmSwPW5f7lK2tA2Xw
```


localhost:8080/manage/credentials/store/system/domain/_/
Jenkins
1
Indhuja T
log out
Dashboard
Manage Jenkins
Credentials
System
Global credentials (unrestricted)
Add Credentials

Global credentials (unrestricted)

Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description
docker-token	indhuja12/******	Username with password	



localhost:8080/job/job2/configure
Dashboard > job2 > Configuration

Configure

General
Triggers
Pipeline
Advanced

□ Build periodically ?

□ GitHub hook trigger for GITScm polling ?

□ Poll SCM ?

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

Pipeline

Define your Pipeline using Groovy directly or pull it from source control.

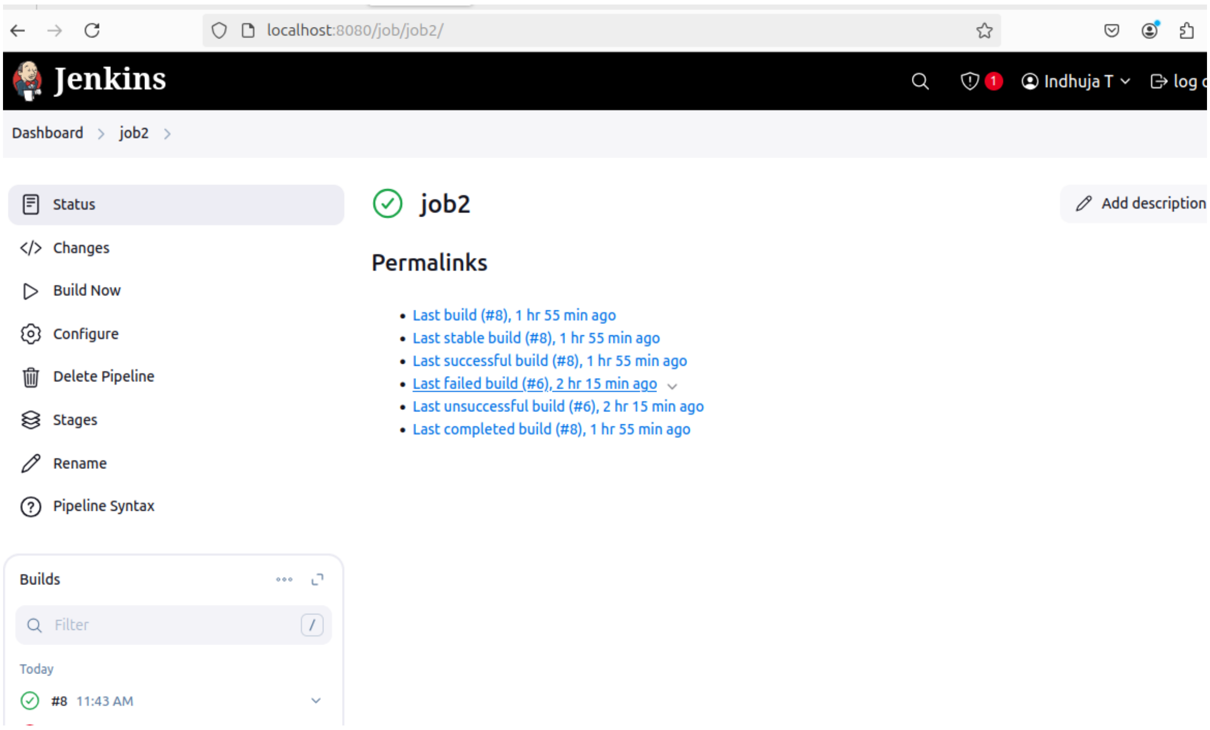
Definition: Pipeline script

Script ?

```
        stage('Build Docker Image') {
            steps {
                sh 'docker build -t $APP_NAME:$BUILD_NUMBER .'
            }
        }

        stage('Login to DockerHub') {
            steps {
                sh 'echo $DOCKERHUB_CREDENTIALS_PSW | docker login -u $DOCKERHUB_CREDENTIALS_USR --passwo
            }
        }

        stage('Push Docker Image') {
```



job2

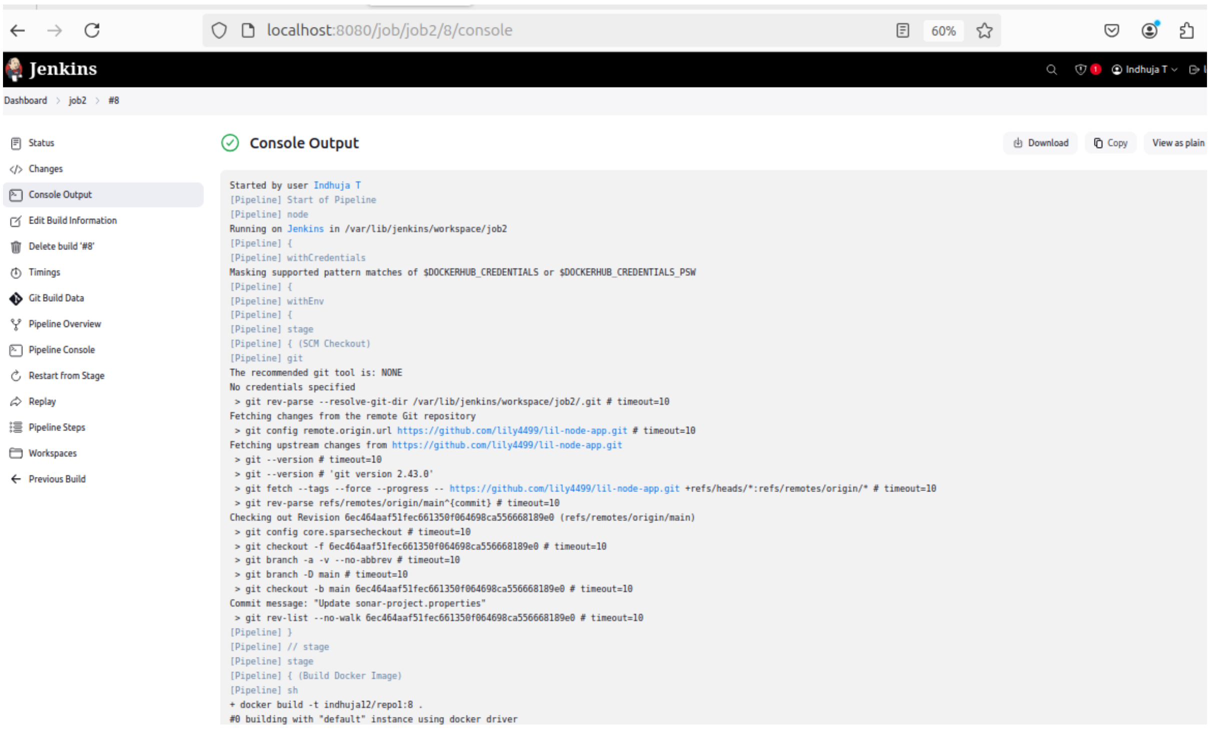
Permalinks

- Last build (#8), 1 hr 55 min ago
- Last stable build (#8), 1 hr 55 min ago
- Last successful build (#8), 1 hr 55 min ago
- Last failed build (#6), 2 hr 15 min ago
- Last unsuccessful build (#6), 2 hr 15 min ago
- Last completed build (#8), 1 hr 55 min ago

Builds

Today

✓ #8 11:43 AM



Console Output

```
Started by user Indhuja T
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/job2
[Pipeline] {
[Pipeline] withCredentials
Masking supported pattern matches of $DOCKERHUB_CREDENTIALS or $DOCKERHUB_CREDENTIALS_PSW
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (SCM Checkout)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/job2/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/lily4499/lil-node-app.git # timeout=10
Fetching upstream changes from https://github.com/lily4499/lil-node-app.git
 > git --version # timeout=10
 > git --version # 'git version 2.43.0'
 > git fetch --tags --force --progress -- https://github.com/lily4499/lil-node-app.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 6ec464aaf51fec661350f064698ca556668189e0 (refs/remotes/origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 6ec464aaf51fec661350f064698ca556668189e0 # timeout=10
 > git branch -a -v --no-abbrev # timeout=10
 > git branch -D main # timeout=10
 > git checkout -b main 6ec464aaf51fec661350f064698ca556668189e0 # timeout=10
Commit message: "Update sonar-project.properties"
 > git rev-list --no-walk 6ec464aaf51fec661350f064698ca556668189e0 # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build Docker Image)
[Pipeline] sh
+ docker build -t indhuja12/repo1:8 .
#0 building with "default" instance using docker driver
```