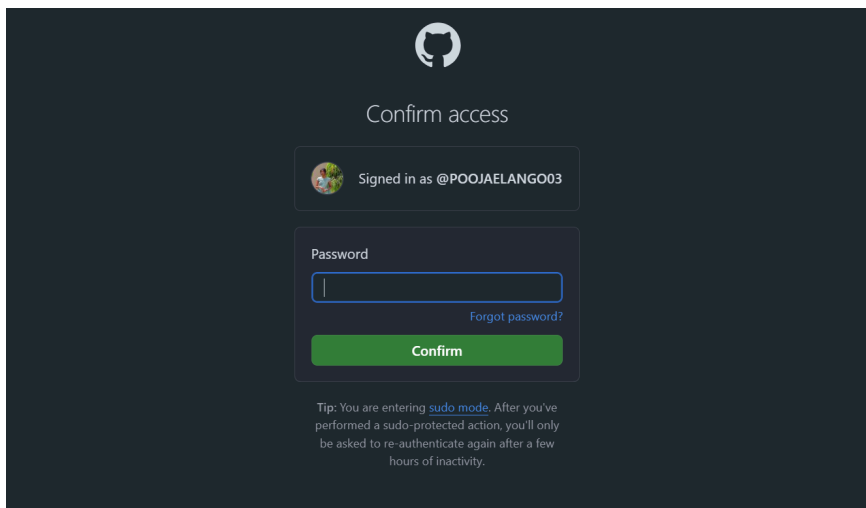
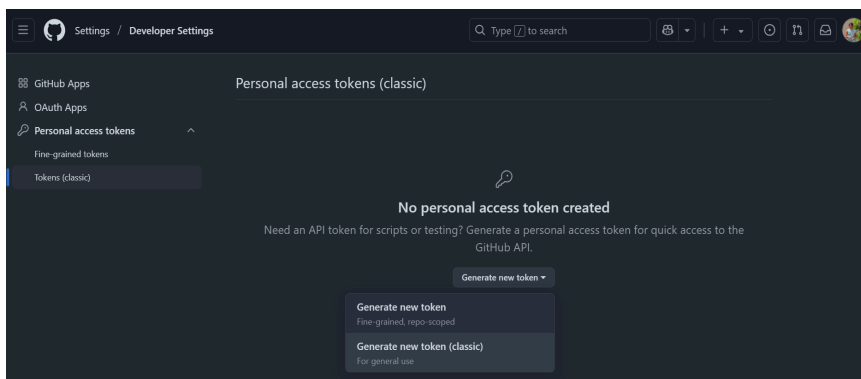
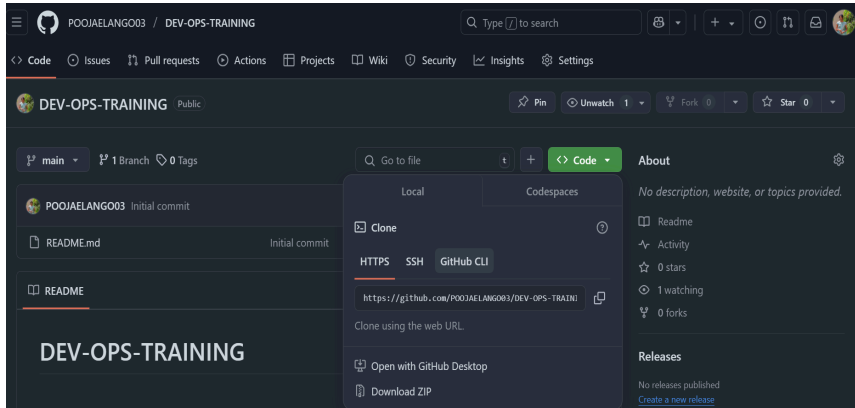


# DEVOPS

## STEPS TO CREATE AND SETUP:



GitHub Apps

OAuth Apps

Personal access tokens

Fine-grained tokens

Tokens (classic)

Search

Type (/) to search

Profile

+

+

+

+

+

+

Avatar

New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

What's this token for?

Expiration

30 days (Apr 18, 2025)

The token will expire on the selected date.

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

☐ repo

☐ repo:status

☐ repo\_deployment

☐ public\_repo

☐ repo:invite

Full control of private repositories

Access commit status

Access deployment status

Access public repositories

Access repository invitations

GitHub Apps

OAuth Apps

Personal access tokens

New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

GitHub Apps

OAuth Apps

Personal access tokens

Fine-grained tokens

Tokens (classic)

Search

Type (/) to search

Profile

+

+

+

+

+

+

Avatar

New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

DAY 3 DEVOPS TRAINING

What's this token for?

Expiration

30 days (Apr 18, 2025)

The token will expire on the selected date.

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

☐ repo

☐ repo:status

☐ repo\_deployment

☐ public\_repo

☐ repo:invite

Full control of private repositories

Access commit status

Access deployment status

Access public repositories

Access repository invitations

Settings / Developer Settings

Search

Profile

+

+

+

+

+

+

Avatar

Some of the scopes you've selected are included in other scopes. Only the minimum set of necessary scopes has been saved.

GitHub Apps

OAuth Apps

Personal access tokens

Fine-grained tokens

Tokens (classic)

Personal access tokens (classic)

Generate new token

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

ghp\_NlCA0B9wS8q0r9bVVECKSdVlSrVPYz1F2xQf

Delete

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

© 2025 GitHub, Inc.

Terms

Privacy

Security

Status

Docs

Contact

Manage cookies

Do not share my personal information

## STARTING AND VERIFYING JENKINS SERVICE ON LINUX:

The terminal session showing the process of managing a Jenkins server on a Linux system using 'systemctl', a command-line tool for controlling the systemd system and service manager.

### 1. Enable Jenkins:

- Command: `sudo systemctl enable jenkins`
- This command enables the Jenkins service to start automatically at boot.

### 2. Start Jenkins:

- Command: `sudo systemctl start jenkins`
- This command starts the Jenkins service immediately.

### 3. Check Jenkins Status:

- Command: `sudo systemctl status jenkins`
- Displays the current status of the Jenkins service, indicating that it is "active (running)" and shows additional details about the service, including its main process ID (PID) and memory usage.

### The Summary:

The session demonstrates the successful initialization and management of a Jenkins Continuous Integration server, indicating its readiness for use. The service was started successfully and is currently running smoothly. It also provides insight into the internal processes Jenkins follows during startup.

```
poojz@zZz:~$ sudo systemctl enable jenkins
[sudo] password for poojz:
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
poojz@zZz:~$ sudo systemctl start jenkins
poojz@zZz:~$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-03-19 04:02:30 UTC; 24min ago
     Main PID: 173 (java)
        Tasks: 52 (limit: 4624)
       Memory: 393.9M ()
      CGroup: /system.slice/jenkins.service
              └─173 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins
Mar 19 04:02:26 zZz jenkins[173]: 2025-03-19 04:02:26.421+0000 [id=36] INFO jenkins.InitReactorRunner$1#o
Mar 19 04:02:26 zZz jenkins[173]: 2025-03-19 04:02:26.426+0000 [id=32] INFO jenkins.InitReactorRunner$1#o
Mar 19 04:02:27 zZz jenkins[173]: 2025-03-19 04:02:27.180+0000 [id=37] INFO h.p.b.g.GlobalTimeOutConfigur
Mar 19 04:02:29 zZz jenkins[173]: 2025-03-19 04:02:29.809+0000 [id=36] INFO jenkins.InitReactorRunner$1#o
Mar 19 04:02:29 zZz jenkins[173]: 2025-03-19 04:02:29.813+0000 [id=33] INFO jenkins.InitReactorRunner$1#o
Mar 19 04:02:29 zZz jenkins[173]: 2025-03-19 04:02:29.901+0000 [id=33] INFO jenkins.InitReactorRunner$1#o
Mar 19 04:02:29 zZz jenkins[173]: 2025-03-19 04:02:29.928+0000 [id=33] INFO jenkins.InitReactorRunner$1#o
Mar 19 04:02:29 zZz jenkins[173]: 2025-03-19 04:02:29.990+0000 [id=30] INFO jenkins.InitReactorRunner$1#o
Mar 19 04:02:30 zZz jenkins[173]: 2025-03-19 04:02:30.051+0000 [id=24] INFO hudson.lifecycle.Lifecycle#on
Mar 19 04:02:30 zZz systemd[1]: Started jenkins.service - Jenkins Continuous Integration Server.
lines 1-19/19 (END)
```

NEXT STEPS TO SETUP:

Dashboard > DEVOPS DAY 2 > Configuration

Configure

General

Triggers

Pipeline

Advanced

Pipeline

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

Definition

Pipeline script from SCM

SCM ?

None

Script Path ?

Jenkinsfile

☒ Lightweight checkout ?

[Pipeline Syntax](#)

Save

Apply

Jenkins

Search, Alerts, PoojaElango, log out

Dashboard >

+ New Item

Build History

Manage Jenkins

My Views

Build Queue

No builds in the queue.

Build Executor Status

0/2

All +

S	W	Name ↓	Last Success	Last Failure	Last Duration	
☹	☀	DEVOPS DAY 2	N/A	N/A	N/A	▶
✔	☀	DEVOPS DAY1	25 min #1	N/A	9.3 sec	▶

Icons: S M L

Add description

REST API

Jenkins 2.492.2

## COMMITTING AND PUSHING CHANGES TO GITHUB IN A DEVOPS PROJECT:

The terminal session where the user is working with a Git repository named DEV-OPS-TRAINING. The steps performed include:

1. Listing the repository files, including Jenkinsfile, README.md, docker-compose.yml, dockerfile, and requirements.txt.
2. Editing the Jenkinsfile using **nano**.
3. Staging changes with **git add ..**
4. Committing the changes with the message "updated".
5. Pushing the changes to a remote GitHub repository using a personal access token for authentication.

The process successfully pushes updates to the main branch on GitHub.

```
poojz@zz:~/devops/DEV-OPS-TRAINING$ ls
Jenkinsfile README.md app.py docker-compose.yml dockerfile requirements.txt
poojz@zz:~/devops/DEV-OPS-TRAINING$ nano Jenkinsfile
poojz@zz:~/devops/DEV-OPS-TRAINING$ git add .
poojz@zz:~/devops/DEV-OPS-TRAINING$ git commit -m "updated"
[main 1da264c] updated
1 file changed, 1 insertion(+)
poojz@zz:~/devops/DEV-OPS-TRAINING$ git push https://POOJAELANGO03:ghp_NcA08Mw58qoDr9bVVECKsdVMSrvPYz1F2xQf@github.com/POOJAELANGO03/DEV-OPS-TRAINING.git
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 309 bytes | 30.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/POOJAELANGO03/DEV-OPS-TRAINING.git
5a43f41..1da264c main -> main
```

The screenshot shows the GitHub web interface for the repository 'DEV-OPS-TRAINING' by user 'POOJAELANGO03'. The repository is public and has 1 branch (main) and 0 tags. The commit history shows a recent update by 'POOJAELANGO03' 5 hours ago, with 7 commits in total. The commit details table lists the following files and their commit history:

File	Commit Message	Time Ago
Jenkinsfile	updated	5 hours ago
README.md	Initial commit	13 hours ago
app.py	first commit	10 hours ago
docker-compose.yml	first commit	10 hours ago
dockerfile	first commit	10 hours ago
requirements.txt	first commit	10 hours ago

The right sidebar shows repository statistics: 0 stars, 1 watching, and 0 forks. There are no releases published yet.

## GRANTING JENKINS DOCKER ACCESS AND RESTARTING SERVICE:

The terminal session where the user is managing Jenkins and Docker permissions. The following commands are executed:

1. **sudo usermod -aG docker jenkins** – Adds the Jenkins user to the Docker group, allowing it to run Docker commands without requiring sudo privileges.
2. **sudo systemctl restart jenkins** – Restarts the Jenkins service to apply the changes made to its user permissions.

These steps are typically done to enable Jenkins to interact with Docker seamlessly in a CI/CD pipeline.

```
poojz@zZz:~/devops/DEV-OPS-TRAINING$ sudo usermod -aG docker jenkins
poojz@zZz:~/devops/DEV-OPS-TRAINING$ sudo systemctl restart jenkins
```

## JENKINS DASHBOARD OVERVIEW:

The screenshot shows the Jenkins dashboard with the 'Console Output' tab selected for the build 'DEVOPS DAY 2' (#14). The output text is as follows:

```
Started by user PoojaElango
Obtained Jenkinsfile from git https://github.com/POOJAELANG03/DEV-OPS-TRAINING.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/DEVOPS DAY 2
[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
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/DEVOPS DAY 2/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/POOJAELANG03/DEV-OPS-TRAINING.git # timeout=10
Fetching upstream changes from https://github.com/POOJAELANG03/DEV-OPS-TRAINING.git
> git --version # timeout=10
> git --version # 'git version 2.43.0'
> git fetch --tags --force --progress -- https://github.com/POOJAELANG03/DEV-OPS-TRAINING.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse origin/main^{commit} # timeout=10
Checking out Revision c243b61c0375863171347d76cf9b091684cf5a1 (origin/main)
> git config core.sparsecheckout # timeout=10
```

The screenshot shows the Jenkins dashboard with the 'Build History' tab selected. The table displays the following data:

S	W	Name	Last Success	Last Failure	Last Duration
✓	☁	DEVOPS DAY 2	4 hr 4 min #14	4 hr 13 min #13	1 min 8 sec
✓	☀	DEVOPS DAY1	12 hr #1	N/A	9.3 sec

Below the table, there is a 'Build Queue' section showing 'No builds in the queue.' and a 'Build Executor Status' section showing '0/2'.

DOCKER HUB REPOSITORY OVERVIEW FOR poojaelango/docker-app:

poojaelango

Docker Personal

Repositories

Settings

Default privacy

Notifications

Billing

Usage

Pulls

Storage

Repositories / docker-app / General

poojaelango/docker-app

Last pushed about 3 hours ago

Add a description

Add a category

General

Tags

Image Management

Collaborators

Webhooks

Settings

Tags

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
latest		Image	less than 1 day	about 3 hours

See all

buildcloud

Build with Docker Build Cloud

Accelerate image build times with access to cloud-based builders and shared cache.

Docker Build Cloud executes builds on optimally-dimensioned cloud infrastructure with dedicated per-organization isolation.

Get faster builds through shared caching across your team, native multi-platform support, and accelerated data transfer, all without

Using 0 of 1 private repositories. Get more

Docker commands

To push a new tag to this repository:

docker push poojaelango/docker-app:tagname

Public view