GitLab CLI GitLab from the terminal

CONCEPTS

Inside a clone, glab finds the project from the git remote. Use -R <group>/<project> to target another one.

Branch local commits
↓ `glab mr create`
Merge request review and approval
↓ `glab mr merge`
Merged target branch
↓ `glab ci status`
Pipeline CI/CD jobs

Merge requests are !123 and issues are #123 on GitLab; glab takes the bare number, a branch name or a URL.

AUTH AND SETUP

`glab auth login`
Sign in interactively (OAuth or token)

`glab auth status`
Show which account and host you use

`glab auth login --hostname <host>`
Sign in to a self-managed instance

MERGE REQUESTS

`glab mr create --fill`
Open an MR with title from commits

`glab mr create --fill --push --yes`
Push, create, skip all prompts

`glab mr create -t "<title>" -b <branch> --draft`
Draft MR into a target branch

`glab mr create -l <label> --reviewer <user>`
Add labels and a reviewer

`glab mr list`
List open MRs

`glab mr list --assignee=@me`
MRs assigned to you

`glab mr list --reviewer=@me`
MRs waiting for your review

`glab mr view <id>`
Show an MR

MERGE REQUESTS (cont.)

`glab mr view <id> --comments`
Include comments and activity

`glab mr view <id> --web`
Open the MR in the browser

`glab mr checkout <id>`
Check out an MR branch

`glab mr diff <id>`
Show the MR diff

`glab mr approve <id>`
Approve an MR

`glab mr note <id> -m "<text>"`
Comment on an MR

`glab mr merge`
Merge the MR of the current branch

`glab mr merge <id> --squash --remove-source-branch`
Squash-merge and delete the branch

`glab mr update <id> --ready`
Mark a draft ready

ISSUES

`glab issue list`
List open issues

`glab issue create -t "<title>" -d "<text>"`
File an issue

`glab issue view <id>`
Show an issue

`glab issue list --assignee=@me`
Issues assigned to you

`glab issue note <id> -m "<text>"`
Comment on an issue

`glab issue close <id>`
Close an issue

CI/CD

`glab ci status`
Pipeline status of this branch

`glab ci status --live`
Follow the pipeline until it ends

`glab ci status --wait`
Block until done, then print

`glab ci view`
Interactive pipeline view

CI/CD (cont.)

`glab ci trace <job-name>`
Stream a job's log

`glab ci run`
Start a pipeline on this branch

`glab ci run -b <branch> --variables <KEY>:<value>`
Run with CI variables

`glab ci retry <job-name>`
Retry a job

`glab ci lint`
Validate .gitlab-ci.yml

`glab ci list`
List recent pipelines

REPOSITORIES

`glab repo clone <group>/<project>`
Clone a project

`glab repo fork --clone`
Fork the current project and clone

`glab repo view --web`
Open the project in the browser

`glab repo create <name>`
Create a project

CI VARIABLES

`glab variable list`
List project CI/CD variables

`glab variable set <KEY> <value>`
Create a variable

`glab variable set <KEY> --masked --protected`
Mask in logs, protected refs only

API

`glab api projects/:fullpath/releases`
GET an endpoint for this project

`glab api projects/:id/issues --paginate`
Fetch every page

`glab api <endpoint> -f <key>=<value>`
Send a string field (POST)

`glab mr list -F json --jq '.[].title'`
JSON output, filtered with jq