Lefthook Fast git hooks manager

CONCEPTS

Hooks are read from lefthook.yml on every git event, so editing the config needs no reinstall.

git commit git event
↓ runs .git/hooks
lefthook run reads lefthook.yml
↓ filters files
Jobs glob, skip, root
↓ exit code
Commit allowed or aborted

Run lefthook install after cloning; the npm package does it for you in a postinstall script.

INSTALL & CLI

brew install lefthook
Install with Homebrew

npm install lefthook --save-dev
Install as a dev dependency

go install github.com/evilmartians/lefthook/v2@latest
Install with Go

gem install lefthook
Install as a gem

lefthook install
Install hooks, create lefthook.yml if missing

lefthook install *<hook>* **...**
Install only some hooks

lefthook install --force
Install even if core.hooksPath is set

lefthook uninstall
Remove hooks lefthook installed

lefthook add *<hook>* **--dirs**
Add a hook and its script directory

lefthook validate
Check the config for errors

lefthook dump
Print the merged configuration

RUNNING

lefthook run *<hook>*
Run a hook's jobs now

RUNNING (cont.)

lefthook run *<hook>* **--all-files**
Treat all tracked files as input

lefthook run *<hook>* **--file** *<path>*
Run on chosen files only

lefthook run *<hook>* **--job** *<name>*
Run only the named job

lefthook run *<hook>* **--tag** *<tag>*
Run only jobs with a tag

LEFTHOOK_VERBOSE=1 lefthook run *<hook>*
Print verbose output

LEFTHOOK_EXCLUDE=*<tag>*,*<name>*
Exclude tags or commands for one run

git commit --no-verify
Bypass pre-commit and commit-msg hooks

LEFTHOOK=0 git *<cmd>*
Disable lefthook for one command

HOOK & JOB OPTIONS

pre-commit:
Hook name as top-level key

jobs: [{name, run}]
Ordered list of tasks in a hook

commands: *<name>***: run:**
Older map form of jobs

run: *<shell command>*
Command to execute (via sh)

parallel: true
Run a hook's jobs concurrently

piped: true
Stop at the first failing job

group: {parallel, jobs}
Nest jobs with their own flow

script: *<file>* **runner: bash**
Run a script from .lefthook/<hook>/

root: *<dir>*
Run the job from a subdirectory

env: {KEY: value}
Set environment variables for a job

FILES & PLACEHOLDERS

{staged_files}
Files staged for this commit

{push_files}
Commits not yet pushed, as files

{all_files}
Every file tracked by git

{files}
Output of the job's files command

{1} / {2}
Arguments git passed to the hook

glob: "*.{js,ts}"
Only run for matching files

glob: ["*.ts", "*.js"]
List of globs

exclude: [*<glob>***]**
Leave matching files out

files: git diff --name-only main
Custom command listing files

stage_fixed: true
git add files after the job (pre-commit)

SKIPPING & CONDITIONS

skip: true
Disable a job or hook

skip: [merge, rebase]
Skip in merge or rebase state

skip: [{ref: main}]
Skip on a branch (globs ok)

skip: [{run: *<cmd>***}]**
Skip if the command succeeds

only: [{ref: dev/*}]
Run only when the condition holds

lefthook-local.yml
Personal overrides, not committed

CONFIG FILE

extends: [*<file>***]**
Merge other YAML files (globs ok)

remotes: [{git_url, ref, configs}]
Share config from another repo