

GitHub SSH + Git Setup Guide (Windows, PowerShell & Git Bash)

A complete end-to-end guide to setting up Git, GitHub, SSH keys, Git LFS, and pushing repositories using SSH on Windows.

0) Install the Tools

- Install Git for Windows from <https://git-scm.com/download/win> (includes Git Bash & Credential Manager).
- Optionally install Git LFS: winget install -e --id GitHub.GitLFS
- Install Python module git-filter-repo if needed: python -m pip install git-filter-repo

1) Configure Git Globally

- git config --global user.name "Your Name"
- git config --global user.email "you@example.com"
- git config --global init.defaultBranch main

2) Create an SSH Key

- PowerShell: ssh-keygen -t ed25519 -C "you@example.com" -f "\$env:USERPROFILE\.ssh\id_ed25519"
- Git Bash: ssh-keygen -t ed25519 -C "you@example.com" -f ~/.ssh/id_ed25519

3) Start SSH-Agent and Add Key

- PowerShell: Get-Service ssh-agent | Set-Service -StartupType Automatic
- Start-Service ssh-agent
- ssh-add "\$env:USERPROFILE\.ssh\id_ed25519"

4) Copy Public Key to Clipboard

- PowerShell: Get-Content -Raw "\$env:USERPROFILE\.ssh\id_ed25519.pub" | Set-Clipboard
- Git Bash: clip < ~/.ssh/id_ed25519.pub

5) Add Key to GitHub

- On GitHub → Settings → SSH and GPG Keys → New SSH key.
- Paste the copied key (starts with ssh-ed25519) and save.

6) Test SSH Access

- ssh -T git@github.com → should show success message.

7) Create a Repository and Push

- git init
- git add -A
- git commit -m "Initial commit"

- git branch -M main
- git remote add origin git@github.com:USERNAME/REPO.git
- git push -u origin main

8) Handle Large Files (Git LFS)

- git lfs install
- git lfs track "*.mov" "*.mp4" "*.exr"
- git add .gitattributes
- git commit -m "Enable Git LFS"

9) Common .gitignore Rules

- *.mov, *.mp4, *.exr, *.zip, *.rar, *.bak, backup/*, ffmpeg/

10) Troubleshooting

- Permission denied (publickey): ensure key added and ssh-agent running.
- Large file (>100MB): use git-filter-repo to remove and force push.
- origin missing: git remote add origin git@github.com:USER/REPO.git

Author: Martin Heigan (mh_tools) — A compact reference for maintaining private repositories efficiently on GitHub.