



🔥CHỌN LỌC TOP NHỮNG KHOÁ HỌC LẬP TRÌNH ONLINE NHIỀU NGƯỜI THEO HỌC TẠI ĐÂY🔥

Có nhiều cách khác nhau để sử dụng Git. Git hỗ trợ nhiều công cụ dòng lệnh và giao diện người dùng đồ họa. Dòng lệnh Git là nơi duy nhất mà bạn có thể chạy tất cả các lệnh Git.

Tập hợp các lệnh sau đây sẽ giúp bạn hiểu cách sử dụng Git thông qua dòng lệnh.

Nội dung chính

Các lệnh Git cơ bản

1. Lệnh cấu hình Git
2. Lệnh Git Init
3. Lệnh Git clone
4. Lệnh Git add
5. Lệnh Git commit
6. Lệnh Git Status
7. Lệnh Git Push
8. Lệnh Git Pull
9. Lệnh Git branch
10. Lệnh Git Merge
11. Lệnh Git log
12. Lệnh Git remote

Các lệnh Git cơ bản

Dưới đây là danh sách các lệnh Git cần thiết nhất được sử dụng hàng ngày.

- Lệnh Git Config
- Lệnh git init
- Lệnh git clone
- Lệnh git add
- Lệnh git commit
- Lệnh git status
- Lệnh git status
- Lệnh git pull
- Lệnh git branch
- Lệnh git merge
- Lệnh git log
- Lệnh git remote

Hãy cùng cafedev hiểu chi tiết từng lệnh này.

Lưu ý: Các lệnh này sẽ có các bài viết hướng dẫn chi tiết với ví dụ cụ thể nha ace.

1. Lệnh cấu hình Git

Lệnh này cấu hình người dùng. Lệnh Git config là lệnh đầu tiên và cần thiết được sử dụng trên dòng lệnh Git. Lệnh này đặt tên tác giả và địa chỉ email sẽ được sử dụng với các commit của bạn. Cấu hình Git cũng được sử dụng trong các trường hợp khác.

Cú pháp

```
git config --global user.name "Cafedevn"  
git config --global user.email "Cafedevn@gmail.com"
```

2. Lệnh Git Init

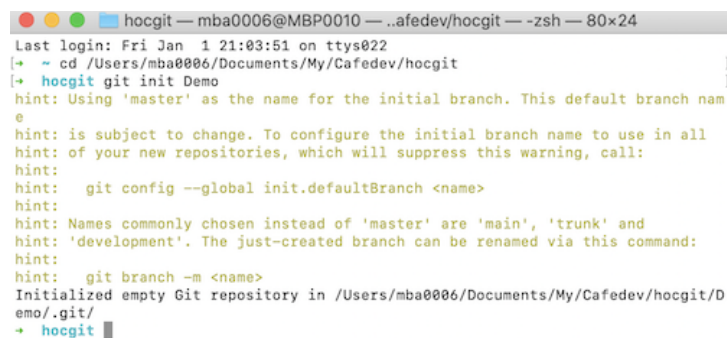
Lệnh này được sử dụng để tạo một kho lưu trữ cục bộ.

Cú pháp

-->

```
git init Demo
```

Lệnh init sẽ khởi tạo một kho lưu trữ trống. Xem ảnh chụp màn hình bên dưới.



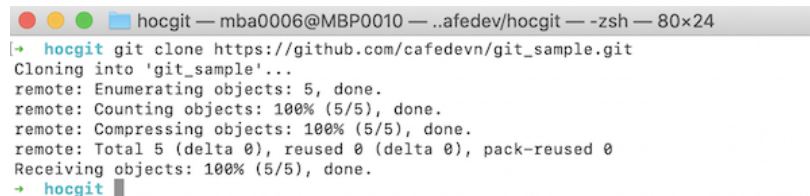
```
hocgit — mba0006@MBP0010 — ..afedev/hocgit — zsh — 80x24
Last login: Fri Jan 1 21:03:51 on ttys022
[+] ~ cd /Users/mba0006/Documents/My/Cafedev/hocgit
[+] hocgit git init Demo
hint: Using 'master' as the name for the initial branch. This default branch nam
e
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/mba0006/Documents/My/Cafedev/hocgit/D
emo/.git/
→ hocgit
```

3. Lệnh Git clone

Lệnh này được sử dụng để tạo bản sao của kho lưu trữ từ một URL hiện có. Nếu tôi muốn có bản sao cục bộ của kho lưu trữ của mình từ GitHub, lệnh này cho phép tạo bản sao cục bộ của kho lưu trữ đó trên thư mục cục bộ của bạn từ URL kho lưu trữ.

Cú pháp

```
git clone URL
```



```
hocgit — mba0006@MBP0010 — ..afedev/hocgit — zsh — 80x24
[+] hocgit git clone https://github.com/cafedevn/git_sample.git
Cloning into 'git_sample'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.
→ hocgit
```

4. Lệnh Git add

Lệnh này được sử dụng để thêm một hoặc nhiều tệp vào vùng dàn (Chỉ mục).

Cú pháp

Để thêm một file

```
git add tên file
```

Để thêm nhiều hơn một file

```
git add *
```

5. Lệnh Git commit

Lệnh commit được sử dụng trong hai trường hợp. Chúng như sau.

git commit -m

Lệnh này thay đổi HEAD. Nó ghi lại hoặc chụp nhanh mọi thay đổi của các file trong lịch sử thay đổi bằng một tin nhắn.

Cú pháp

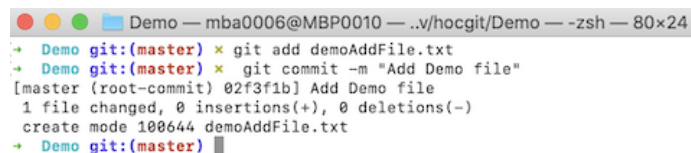
```
git commit -m "Commit Message"
```

Git commit -a

Lệnh này commit bất kỳ file nào được thêm vào kho lưu trữ bằng git add và cũng commit bất kỳ file nào bạn đã thay đổi kể từ đó.

Cú pháp

```
git commit -a
```



```
Demo — mba0006@MBP0010 — ../hocgit/Demo — zsh — 80x24
+ Demo git:(master) * git add demoAddFile.txt
+ Demo git:(master) * git commit -m "Add Demo file"
[master (root-commit) 02f3f1b] Add Demo file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 demoAddFile.txt
+ Demo git:(master) █
```

6. Lệnh Git Status

Lệnh trạng thái được sử dụng để hiển thị trạng thái của thư mục làm việc và vùng origin. Nó cho phép bạn xem những thay đổi nào đã được thực hiện, những thay đổi nào chưa và những file nào không được Git theo dõi. Nó không hiển thị cho bạn bất kỳ thông tin nào về lịch sử dự án đã commit. Đối với điều này, bạn cần sử dụng git log. Nó cũng liệt kê các file bạn đã thay đổi và những file bạn vẫn cần thêm hoặc commit.

Cú pháp

```
git status
```

```
Demo — mba0006@MBP0010 — ../hocgit/Demo — zsh —  
[→ Demo git:(master) git status  
On branch master  
nothing to commit, working tree clean  
→ Demo git:(master) █
```

7. Lệnh Git Push

Nó được sử dụng để tải nội dung của kho lưu trữ cục bộ lên kho lưu trữ từ xa. Đây là một hành động chuyển các commit từ kho lưu trữ cục bộ của bạn đến một kho lưu trữ từ xa. Đó là sự bổ sung cho git fetch, Các nhánh từ xa được cấu hình bằng cách sử dụng lệnh git remote. Push có thể ghi đè các thay đổi và cần thận trọng khi push.

Lệnh Git push có thể được sử dụng như sau.

Git push origin main

Lệnh này sẽ gửi các thay đổi được thực hiện trên nhánh chính tới kho lưu trữ từ xa của bạn.

Cú pháp

```
git push [tên biến] main
```

```
[→ git_sample git:(main) git push origin main  
Username for 'https://github.com': cafedevn  
[Password for 'https://cafedevn@github.com':  
Enumerating objects: 4, done.  
Counting objects: 100% (4/4), done.  
Delta compression using up to 8 threads  
Compressing objects: 100% (2/2), done.  
Writing objects: 100% (3/3), 271 bytes | 271.00 KiB/s, done.  
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0  
remote: Resolving deltas: 100% (1/1), completed with 1 local object.  
To https://github.com/cafedevn/git_sample.git  
18felc3..80db048 main -> main  
→ git_sample git:(main) █
```

Git push -all

Lệnh này đẩy tất cả các nhánh vào kho lưu trữ máy chủ.

Cú pháp

```
git push --all
```

```
[→ git_sample git:(main) git push --all
Everything up-to-date
→ git_sample git:(main) █
```

8. Lệnh Git Pull

Lệnh kéo(pull) được sử dụng để nhận dữ liệu từ GitHub. Nó tìm nạp và kết hợp các thay đổi trên máy chủ từ xa vào thư mục làm việc của bạn.

Cú pháp

```
git pull URL
```

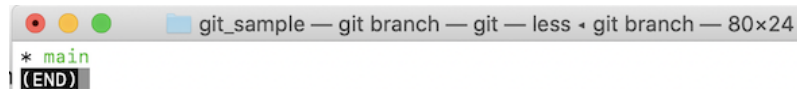
```
[→ git_sample git:(main) git pull https://github.com/cafedevn/git_sample.git
hint: Pulling without specifying how to reconcile divergent branches is
hint: discouraged. You can squelch this message by running one of the following
hint: commands sometime before your next pull:
hint:
hint:   git config pull.rebase false  # merge (the default strategy)
hint:   git config pull.rebase true   # rebase
hint:   git config pull.ff only       # fast-forward only
hint:
hint: You can replace "git config" with "git config --global" to set a default
hint: preference for all repositories. You can also pass --rebase, --no-rebase,
hint: or --ff-only on the command line to override the configured default per
hint: invocation.
From https://github.com/cafedevn/git_sample
 * branch            HEAD       -> FETCH_HEAD
Already up to date.
→ git_sample git:(main) █
```

9. Lệnh Git branch

Lệnh này liệt kê tất cả các nhánh có sẵn trong kho lưu trữ.

Cú pháp

```
git branch
```



```
* main
(END)
```

Nhấn phím q để thoát giao diện danh sách branch

10. Lệnh Git Merge

Lệnh này được sử dụng để hợp nhất lịch sử của nhánh được chỉ định vào nhánh hiện tại.

Cú pháp

```
git merge BranchName
```



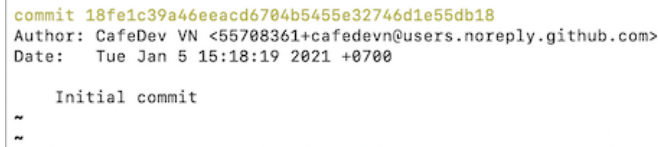
```
git_sample — mba0006@MBP0010 — ..it/git_sample —
[+] git_sample git:(main) git merge main
Already up to date.
-> git_sample git:(main) █
```

11. Lệnh Git log

Lệnh này được sử dụng để kiểm tra lịch sử commit.

Cú pháp

```
git log
```



```
commit 18fe1c39a46eeacd6704b5455e32746d1e55db18
Author: CafeDev VN <55708361+cafedevn@users.noreply.github.com>
Date: Tue Jan 5 15:18:19 2021 +0700

    Initial commit
~
~
```

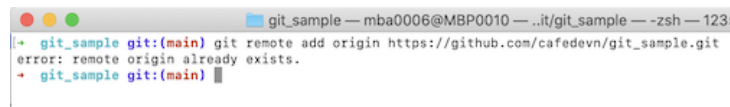
Nhấn phím q để thoát giao diện danh sách xem log

Theo mặc định, nếu không có đối số nào được thông qua, nhật ký Git sẽ hiển thị các commit gần đây nhất trước tiên. Chúng ta có thể giới hạn số lượng mục nhật ký được hiển thị bằng cách chuyển một số làm tùy chọn, chẳng hạn như -3 để chỉ hiển thị ba mục nhập cuối cùng.

```
git log -3
```

12. Lệnh Git remote

Lệnh Git Remote được sử dụng để kết nối kho lưu trữ cục bộ của bạn với máy chủ từ xa. Lệnh này cho phép bạn tạo, xem và xóa các kết nối đến các kho lưu trữ khác. Lệnh này không cung cấp quyền truy cập thời gian thực vào các kho lưu trữ.



```
git_sample — mba0006@MBP0010 — ..it/git_sample — -zsh — 123:
[+] git_sample git:(main) git remote add origin https://github.com/cafedevn/git_sample.git
error: remote origin already exists.
-> git_sample git:(main) █
```

Cài ứng dụng cafedev để dễ dàng cập nhật tin và học lập trình mọi lúc mọi nơi tại đây.

Tài liệu từ cafedev:

- Full series tự học Git từ cơ bản tới nâng cao tại đây nha.
- Ebook về Git tại đây.
- Các series tự học lập trình MIỄN PHÍ khác
- Nơi liên hệ hợp tác hoặc quảng cáo cùng Cafedevn tại đây.
- Giới thiệu chi tiết về Cafedev

Nếu bạn thấy hay và hữu ích, bạn có thể tham gia các kênh sau của cafedev để nhận được nhiều hơn nữa:

- [Group Facebook](#)
- [Fanpage](#)
- [Youtube](#)
- [Instagram](#)
- [Twitter](#)
- [Linkedin](#)
- [Pinterest](#)
- [Trang chủ](#)

Chào thân ái và quyết thắng!

Đăng ký kênh youtube để ủng hộ Cafedev nha các bạn, Thanks you!
