Esc :q ;

Nhiều khi viết commit message hay bị viết sai, viết thiếu và lỡ commit luôn. Để sửa lại commit vừa mới ở đằng trước, ta có thể đơn giản gọi lại câu lệnh commit với amend

***git commit –-amend***

Nếu nội dung muốn thay đổi không chỉ là message mà gồm cả thiếu file, thì đơn giản có thể add lại filevà dùng amend như trên

git add file-name

git commit –amend

**Undo**

Muốn "undo" thay đổi trên một file cố định trước khi commit, có thể dùng câu lệnh reset

git reset file-name

Nếu muốn undo hẳn một commit (do đã lỡ commit xong rồi) thì cần chỉ định mục tiêu

git reset --soft HEAD~1

Ở đây HEAD~1 nghĩa là trước 1 commit. Mình dùng soft để lưu lại những thay đổi chưa commit và chỉ bỏ đi phần đã commit từ lần trước

Nếu muốn bỏ cả phần đã commit từ lần trước và phần chưa commit thì đổi soft thành hard

git reset --hard HEAD~1

## Clean

Khi branch đang làm việc có quá nhiều file mới (chưa được add), và mình muốn clean hết, thì có thể dùng một câu lệnh

git clean –force

## Thay đổi tên tác giả của commit

Hồi mới đầu mình hay bị lỗi này. Khi chưa setup email và password cho Git thì lúc commit sẽ bị username đổi thành username của OS.   
Để sửa lại những commit bị sai tên mình làm như sau

git config user.name "new name"

git config user.email newmail@abc.com

git commit --amend --reset-author

## Rebase

## xdgfdsdsfdsfs

**Xoa tập tin khỏi thư mục theo dõi**

Git rm –cached <ten file >

**Loại bỏ tập tin sau khi thực hiện git add .**

Git reset HEAD <ten file >

**Cấu hình cách viết tắt trong git**

$ git config --global alias.co checkout

$ git config --global alias.br branch

$ git config --global alias.ci commit

$ git config --global alias.st status

**Để biết nhánh nào đã được tích hợp vào nhánh hiện tại, bạn có thể sử dụng**

git branch –merged

**Để xem cách nhánh chứa các công việc/thay đổi chưa được tích hợp vào, bạn có thể chạy lệnh**

git branch --no-merged

**Lệnh này lại hiện thị các nhánh khác. Bởi vì chúng bao gồm các công việc mà bạn chưa tích hợp vào, xóa nó đi bằng lệnh git branch -d sẽ báo lỗi:**

$ git branch -d testing

error: The branch 'testing' is not an ancestor of your current HEAD.

If you are sure you want to delete it, run 'git branch -D testing'.

Nếu bạn thực sự muốn xóa nó đi và chấp nhận mất các thay đổi, bạn có thể bắt buộc bằng cách sử dụng tham số -D, như hướng dẫn trong thông báo trên.

### [Xóa Nhánh Trung Tâm](https://git-scm.com/book/vi/v1/Ph%C3%A2n-Nh%C3%A1nh-Trong-Git-Nh%C3%A1nh-Remote#X%C3%B3a-Nh%C3%A1nh-Trung-T%C3%A2m)

Giả sử bạn và đồng nghiệp đã hoàn thành một chức năng nào đó và đã tích hợp nó vào nhánh mastertrung tâm (hoặc bất kỳ nhánh nào khác sử dụng cho việc lưu trữ các phiên bản ổn định). Bạn có thể xóa một nhánh trung tâm đi sử dụng cú pháp sau git push [remotename] :[branch]. Nếu bạn muốn xóa nhánh serverfix trên máy chủ, bạn có thể chạy lệnh sau:

$ git push origin :serverfix

To git@github.com:schacon/simplegit.git

- [deleted] serverfix

Vậy là đã xong, nhánh đó đã bị xóa khỏi máy chủ. Có thể bạn muốn đánh dấu trang này lại, vì bạn sẽ cần đến câu lệnh này và có thể bạn sẽ quên cú pháp của nó. Một cách để nhớ lệnh này là xem lại cú pháp chúng ta đã nhắc tới trước đó git push [remotename] [localbranch]:[remotebranch]. Nếu bạn bỏ qua phần [localbranch], thì cơ bản bạn đang thực hiện "Không sử dụng gì từ phía nội bộ để tạo nhánh [remotebranch]."