



4  
Thích  
Chia sẻ

# NHỮNG LỆNH GIT CƠ BẢN CẦN NHỚ

[Trang chủ](#) > [Blog](#) > [HyShare - IT Tips](#) > NHỮNG LỆNH GIT CƠ BẢN CẦN NHỚ

Nội dung chính [ẩn]

1. Cơ bản về Git

2. Flow cơ bản khi sử dụng Git

3. Những câu lệnh cơ bản thường sử dụng

Thiết lập chứng thực cá nhân

Tạo một kho chứa Git

Sao chép một kho chứa đã tồn tại

Nhánh trong git

Chuyển nhánh

Cập nhật thay đổi

Cập nhật lên server

Gộp nhánh

Xem lại lịch sử commit

Xem thay đổi trước khi push

Gộp commit

Pull từ remote repository

Hybrid Technologies gửi tới các bạn một bài viết giới thiệu khái quát về Git và đưa ra những câu lệnh cơ bản thường xuyên sử dụng. Mong bài viết này có thể giúp ích cho các bạn mới bắt đầu sử dụng Git có thể tham khảo, hiểu được một phần nào đó về Git và vận dụng tốt vào việc sử dụng của các bạn.

## 1. Cơ bản về Git

Git là tên gọi của một Hệ thống quản lý phiên bản phân tán (Distributed Version Control System – DVCS) là một trong những hệ thống quản lý phiên bản phân tán phổ biến nhất hiện nay. **DVCS** nghĩa là hệ thống giúp mỗi máy tính có thể lưu trữ nhiều phiên bản khác nhau của một mã nguồn được nhân bản (clone) từ một kho chứa mã nguồn (repository), mỗi thay đổi vào mã nguồn trên máy tính sẽ có thể ủy thác (commit) rồi đưa lên máy chủ nơi đặt kho chứa chính. Và một máy tính khác (nếu họ có quyền truy cập) cũng có thể clone lại mã nguồn từ kho chứa hoặc clone lại một tập hợp các thay đổi mới nhất trên máy tính kia. Trong Git, thư mục làm việc trên máy tính gọi là Working Tree.

## 2. Flow cơ bản khi sử dụng Git

Dưới đây là flow cơ bản khi bạn sử dụng Git:

- Clone project từ server về Local Repository
- Check-out 1 nhánh từ Local Repository về Working Space
- Bạn sẽ làm việc (thêm, sửa, xoá tại Working Space)
- Add : xác nhận sự thay đổi của các files (đưa đến vùng Staging Area)
- Commit: cập nhật sự thay đổi lên Local Repository

Về cơ bản đến đây là bạn đã hoàn thành 1 chu trình sử dụng Git. Lúc này, nếu như bạn muốn cập nhật sự thay đổi này lên server thì bạn sẽ dùng lệnh push để đẩy chúng lên server.

### 3. Những câu lệnh cơ bản thường sử dụng

#### Thiết lập chứng thực cá nhân

```
$ git config --global user.name "User Name"
$ git config --global user.email "username@gmail.com"
```

Lưu ý: `--global` được sử dụng để áp dụng cho tất cả các projects. Nếu bạn ko sử dụng `--global` thì settings sẽ chỉ dùng cho riêng project đó.

#### Tạo một kho chứa Git

```
$ git init
```

Nếu như bạn muốn theo dõi một dự án cũ trong Git, bạn cần ở trong thư mục của dự án đó. Lệnh này sẽ tạo một thư mục mới có tên `.git`, thư mục này chứa tất cả các tập tin cần thiết cho kho chứa.

#### Sao chép một kho chứa đã tồn tại

```
$ git clone https://github.com/user/repository.git
```

Câu lệnh trên sẽ tạo một thư mục mới có tên giống trên của Repo.

#### Nhánh trong git

Khi sử dụng Git, bạn có thể tạo ra nhiều nhánh (branch) khác nhau. Câu lệnh Git này dùng để kiểm tra branch hiện tại:

```
$ git branch
```

Để tạo mới một branch:

```
$ git branch <name_branch>
```

Để chuyển và tạo mới:

```
$ git branch -b <name_branch>
```

#### Chuyển nhánh

Trước khi muốn thay đổi source code, điều đầu tiên mà bạn cần phải làm là checkout một nhánh. Để checkout một nhánh, bạn dùng câu lệnh Git sau:

```
$ git checkout <name_branch>
```

#### Cập nhật thay đổi

Sau khi bạn thay đổi source code: thêm mới, sửa, xóa files,... Bạn cần phải cập nhật lên Staging Area. Để cập nhật hết các files:

```
$ git add .
```

Sau lệnh add, bạn cần sử dụng câu lệnh Commit để đẩy thông tin thay đổi lên Local Repository:

```
$ git commit -m "Message"
```

#### Cập nhật lên server

Sau câu lệnh Commit, thông tin mới chỉ được cập nhật lên Local Repository. Nếu muốn cập nhật lên server thì bạn phải sử dụng câu lệnh push:

```
$ git push origin <name_branch>
```

Ngoài ra, nếu chưa tồn tại remote trên server thì bạn cần phải add mới một remote trước rồi mới push:

```
$ git remote add origin <remote_url>
$ git push origin <name_branch>
```

#### Gộp nhánh

Sau một thời gian cập nhật các file và push lên git trên branch mới, bây giờ mình cần ghép (merge) code lại vào nhánh gốc (master). Trước tiên, cần phải checkout ra khỏi branch hiện tại cần gộp để vào branch master, sau đó thì dùng lệnh merge để ghép branch mới vào master:

```
$ git checkout master
$ git merge <new_branch>
```

Xem lại lịch sử commit

```
$ git log
```

Lệnh git log sẽ cho bạn biết về người commit, ngày giờ, message của những lần commit đó.

Xem thay đổi trước khi push

```
$ git diff
```

Lệnh này giúp bạn biết những gì đã được thay đổi giữa nhánh hiện tại và nhánh trước nó.

Gộp commit

```
$ git rebase -i HEAD~
```

Sau dấu ~ là số commit bạn muốn gộp. Sau khi gõ lệnh này một cửa sổ trình soạn thảo hiện ra. Thay đổi ký tự pick của dòng các dòng sau dòng đầu thành s rồi lưu lại/kết thúc. Khi đó, trình soạn thảo để chỉnh sửa giải thích commit thiết lập cho commit sau khi đã tổng hợp sẽ được hiển thị, nên hãy chỉnh sửa lưu lại/kết thúc.

Pull từ remote repository

```
$ git pull origin master
```

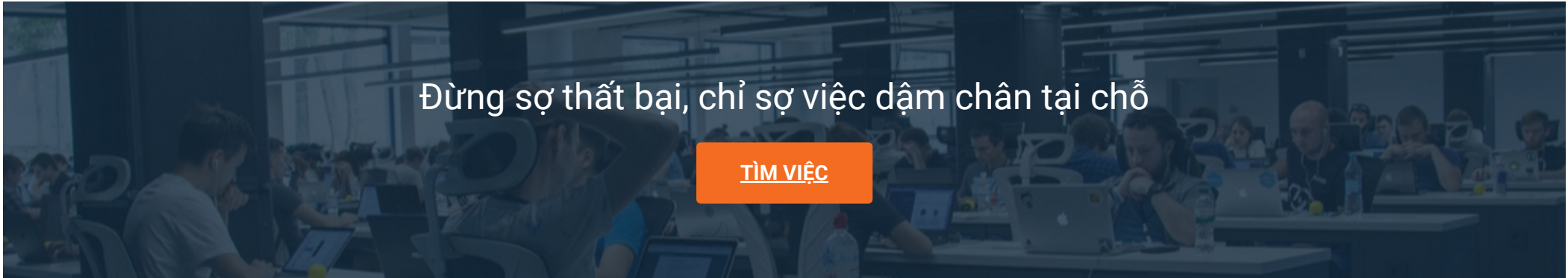
Lệnh trên sẽ gộp những thay đổi mới kéo về từ máy chủ từ xa với nhánh hiện tại trên máy local.

*Hãy cùng comment những thắc mắc và góp ý ngay bên dưới để cùng xây dựng một cộng đồng cung cấp kiến thức hữu ích về chuyên ngành công nghệ cùng Hybrid Technologies nhé!*

Nguồn: Viblo.

Bạn thích bài viết này chứ?  
Đăng ký để nhận những bài viết thú vị như thế hàng tuần.

★★★★★  
4.6 (92%) 10 votes



Bình luận

0 bình luận Sắp xếp theo Cũ nhất



Viết bình luận...

[Plugin bình luận trên Facebook](#)