

I. Git là gì ?

Git là một hệ thống quản lý phiên bản phân tán (Distributed Version Control System – DVCS), nó là một trong những hệ thống quản lý phiên bản phân tán phổ biến nhất hiện nay. Git cung cấp cho mỗi lập trình viên kho lưu trữ (repository) riêng chứa toàn bộ lịch sử thay đổi.

II. Cài đặt git

Cài đặt trên các hệ điều hành khác nhau dựa vào tài liệu chính thức của git:

<http://git-scm.com/downloads>

III. Một số thuật ngữ trong git

1. Branch

Các Branch (nhánh) đại diện cho các phiên bản cụ thể của một kho lưu trữ tách ra từ project chính của bạn

2. Commit

Một commit đại diện cho một thời điểm cụ thể trong lịch sử dự án của bạn.

Sử dụng lệnh commit kết hợp với lệnh git add để cho git biết những thay đổi bạn muốn lưu vào local repository.

3. Checkout

Sử dụng lệnh git checkout để chuyển giữa các branch.

4. Fetch

Lệnh git fetch tìm nạp các bản sao và tải xuống tất cả các tệp branch vào máy tính của bạn. Sử dụng nó để lưu các thay đổi mới nhất vào kho lưu trữ của bạn.

5. Fork

Một fork là một bản sao của một kho lưu trữ (repository).

6. Head

Các commit ở đầu của một branch được gọi là head. Nó đại diện cho commit mới nhất của repository mà bạn hiện đang làm việc.

7. Index

Bất cứ khi nào bạn thêm, xóa hoặc thay đổi một file, nó vẫn nằm trong chỉ mục cho đến khi bạn sẵn sàng commit các thay đổi. Nó như là khu vực tổ chức (staging area) cho Git. Sử dụng lệnh git status để xem nội dung của index của bạn.

8. Master

Master là nhánh chính của tất cả các repository của bạn. Nó nên bao gồm những thay đổi và commit gần đây nhất.

9. Merge

Lệnh git merge kết hợp với các yêu cầu kéo (pull requests) để thêm các thay đổi từ nhánh này sang nhánh khác.

10. Origin

Origin là phiên bản mặc định của repository. Origin cũng đóng vai trò là bí danh hệ thống để liên lạc với nhánh chính.

Lệnh git push origin master để đẩy các thay đổi cục bộ đến nhánh chính.

11. Pull

Pull requests thể hiện các đề xuất thay đổi cho nhánh chính. Nếu bạn làm việc với một nhóm, bạn có thể tạo các pull request để yêu cầu người bảo trì kho lưu trữ xem xét các thay đổi và hợp nhất chúng.

12. Push

Lệnh git push được sử dụng để cập nhật các nhánh từ xa với những thay đổi mới nhất mà bạn đã commit.

13. Rebase

Lệnh git rebase cho phép bạn phân tách, di chuyển hoặc thoát khỏi các commit. Nó cũng có thể được sử dụng để kết hợp hai nhánh khác nhau.

14. Remote

Một Remote (kho lưu trữ từ xa) là một bản sao của một chi nhánh. Remote giao tiếp ngược dòng với nhánh gốc (origin branch) của chúng và các Remote khác trong kho lưu trữ.

15. Repository

Kho lưu trữ Git chứa tất cả các tệp dự án của bạn bao gồm các branch, tags và commit.

IV. Các lệnh cơ bản của git

1. git config

Tác dụng: Để set user name và email của bạn trong main configuration file.

Sử dụng:

Để kiểm tra tên và kiểu email trong cấu hình dùng git config--global user.name và git config --global user.email.

Để set email hoặc tên mới git config--global user.name = “Hoàng Nguyễn” và git config--global user.email = “hoangnguyen@gmail.com”

2. git init

Tác dụng: Khởi tạo 1 git repository mới.

Sử dụng: git init trong thư mục gốc của dự án

3. git clone

Tác dụng: Copy 1 git repository từ remote source.

Sử dụng: `git clone <:clone git url:>`

4. git status

Tác dụng: Để check trạng thái của những file bạn đã thay đổi trong thư mục làm việc.

Sử dụng: `git status` trong thư mục làm việc.

5. git add

Tác dụng: Thêm thay đổi đến stage/index trong thư mục làm việc.

Sử dụng: `git add *`

6. git commit

Tác dụng: commit nghĩa là một action để Git lưu lại một snapshot của các sự thay đổi trong thư mục làm việc.

Sử dụng: `git commit -m "Đây là message"`

7. git push/git pull

Tác dụng: Push hoặc Pull các thay đổi đến remote. Nếu bạn đã added và committed các thay đổi và bạn muốn đẩy nó lên hoặc remote của bạn đã update và bạn apply tất cả thay đổi đó trên code của mình.

Sử dụng: `git pull <:remote:> <:branch:>` and `git push <:remote:> <:branch:>`

8. git branch

Tác dụng: liệt kê tất cả các branch (nhánh).

Sử dụng: `git branch`

9. git checkout

Tác dụng: Chuyển sang branch khác.

Sử dụng: `git checkout <: branch:>`

10. git stash

Tác dụng: Lưu thay đổi mà bạn không muốn commit ngay lập tức.

Sử dụng: `git stash` trong thư mục làm việc của bạn.

11. git merge

Tác dụng: Merge 2 branch lại với nhau.

Sử dụng: Chuyển tới branch bạn muốn merge rồi dùng `git merge <:branch_ban_muon_merge:>`

12. git reset

Tác dụng: Bạn đã đưa một tập tin nào đó vào Staging Area nhưng bây giờ bạn muốn loại bỏ nó ra khỏi đây để không phải bị commit theo.

Sử dụng: `git reset HEAD tên_file`

13. git remote

Tác dụng: Để check remote/source bạn có hoặc add thêm remote.

Sử dụng: `git remote` để kiểm tra và liệt kê. Và `git remote add <: remote_url:>` để thêm.

14. git add

Tác dụng: Để đưa một tập tin vào Staging Area.

Sử dụng: `git add tên_file` hoặc muốn thêm hết file của thư mục thì `git add all`