Cấu hình:

git config –global user.name name

git config –global user.email email

Tạo một repository rỗng hoặc khởi tạo lại một repository đã tồn tại.

git init

xem cây trạng thái

git status

|  |  |
| --- | --- |
| Lệnh | Giải thích |
| git config –global user.name name  git config –global user.email email | Cấu hình tên và địa chỉ email |
| git init | Tạo một repository rỗng hoặc khởi tạo lại một repository đã tồn tại. |
| git clone | clone repository trên một thư mục mới |
| git add | Thêm 1 nội dung của các file vào index |
| Git status |  |
| Git diff |  |
| Git commit |  |
| Git reset |  |
| Git rm |  |
| Git mv |  |
|  |  |

**Giới thiệu**

* Version control là hệ tống mà ghi lại các thay đổi của một file hoặc tập các files qua thời gian mà sau này bạn có thể nhớ lại cụ thể các version. Dẫn đến sự ra distributed version control system.

Git là gì ?

* Git là mã nguồn mở distributed version control system miễn phí, được thiết kế để xử lý mọi thứ từ các dự án nhỏ cho đến các dự án lớn với tốc độ và hiệu quả.

Sự khác nhau giữ Git và các hệ thống khác.

* Các hệ thống như CVS, Subversion, Perforce, Bazaar và ... lưu trữ các file và sự thay đổi ở mỗi file theo thời gian.
* Git thì mỗi lần commit nó lưu lại trạng thái của project, nó coi mỗi version như là hệ thống tập file nhỏ.
* Chính vì cách lưu trữ khác nhau này làm cho Git có thể xem xét mọi khía cạnh của version control hơn .

Git có 3 trạng thái: commited, modified và staged:

* Commited : dữ liệu đã được lưu trữ an toàn trên cục bộ.
* Modified : đã thay đổi file nhưng vẫn chữ commited tới data.
* Staged : đánh dấu để sửa 1 file trong phiên bản hiện tại chuẩn bị commited.

Một project trong git có 3 phần chính là : git directory, working directory và staging area.

* Git directory : là nơi git lưu metadata và các đối tượng cơ sở dữ liệu của project. Chính là cái mà clone sang máy tính khác.
* Working directory : sẽ kiểm tra duy nhất một phiên bản của project. Những file này sẽ được láy ra từ git directory và đưa ra ổ đĩa để sử dụng hoặc sửa.
* Staging area : là một file thường chứa trong git directory, lưu trữ về thông tin về những gì bạn commit lần tới. Đôi khi nó được gọi như index.

Workflow trong git :

* Thay đổi ở trong working directory.
* stage các file và thêm snapshots của các file vào staging area.
* Thực hiện commit, nó sẽ lấy những file trong staging area và lưu trữ các snapshot tới git directory.

Nếu một file cụ thể nào đó trong git directory được commit, nếu nó bị sửa và đã được thêm vào stagin area, ở trạng thái staged. Và nếu nó bị thay đổi từ khi nó được kiểm tra (check out) nhưng vẫn được stag, nó đang ở trạng thái modified

**Sử dụng Git**

**Có 2 cách tiếp cận Git:**

* Lấy project đã tồn tại hoặc thư mục rồi import nó vào git

Gõ lệnh : git init

khởi tạo project rỗng.

* Clone project từ server khác.

Gõ lệnh : git clone [url]

Clone project từ địa chỉ url, nó sẽ copy toàn bộ dữ liệu của project mà server có.

Đễ xem trạng của các file trong project sử dụng lệnh : git status. Các file có thể là untracked, umodified, modified, staged.

Để theo dõi 1 file ta sử dụng : git add name files. Nó nói với rằng bạn muốn cập nhật file trong lần commit tiếp theo.

Lệnh git commit dùng để staged snapshot tới lịch sử project. Sử dụng : git commit.

Lệnh git log dùng để xem các snapshots đã được commited. Nó sẽ liệt kê lịch sử của project rồi lọc, cuối cùng là tìm sự thay đổi cụ thể.

Lệnh git checkout : có 3 chức năng

* Checking out files
* Checking out commit
* Checking out branches.

Checking out commit sử dụng để kiểm tra trạng thái cũ của project mà không làm thay đổi trạng thái hiện tại. Sử dụng lệnh : git chekout < commit>

checking out file để xem phiên bản cũ của một file cụ thể. Sử dụng lệnh git checkout <commit> < file>.

Lệnh git revert : nó khôi phục một snapshot đã được commit. Nó không xóa commit khỏi lịch sử của project mà nó biết phải làm như thế nào để khôi phục thay đổi từ một commit và nối commit mới vào lịch sử của project. Sử dụng : git revert <commit>

Lệnh git reset : cũng dùng để khôi phục giống git revert nhưng nó xóa hết các snapshots đã commit. Sử dụng :

* git reset file : xóa file ở staging area, nhưng không thay đổi work dicrectory.
* git reset : khôi phục stagingarea với commit mới nhất , nhưng không thay đổi work dicrectory.
* git reset <commit> :