**GIT CHEAT**

MAKE CHANGES

* $ git status : Lệnh git status hiện thị thông tin khác nhau (do thêm mới, xóa đi, sửa đổi các file) giữa các file trong các trường hợp:.
* $ git diff : Lệnh git diff hiện thị thông tin thay đổi giữa thư mục làm việc và vùng index (staging) hoặc với commit cũ, thông tin thay đổi giữa index(staging) và commit, thông tin thay đổi giữa hai nhánh
* $ git add [file] : Lệnh git add sẽ đưa file, thư mục vào staging - nếu file/thư mục chưa từng được giám sát bởi git nó sẽ bắt đầu giám sát và tạo snapshot là toàn bộ file/thư mục mới. Nếu đã từng giám sát - thì snapshot là nội dung thay đổi so với commit trước.
* $ git diff –staged : kiểm tra sự thay đổi của index với commit cuối.
* $ git reset [file] : hủy một file nào đó trong vùng staging chứ không phải toàn bộ
* $ git commit -m "[descriptive message]" : Lệnh commit cơ bản, đơn giản nhất là thực hiện với tham số -m để kèm dòng thông tin về commit

**CONFIGURE TOOLING**

* $git config --global user.name "name"

Đặt tên người dùng bạn muốn để tham gia commit

* $git config --global user.email “Email address”

Đặt email người dùng bạn muốn để tham gia commit

* $git config –global color.ui auto

Cho phép tô sáng dòng lệnh

**GROUP CHANGES**

* $git branch

Hiển thị tất cả các nhánh

* $git branch <branchname>

Tạo ra một nhánh mới

* $git checkout <branchname>

Di chuyển tới một nhánh

* $git merge <branchname>

Hợp nhất một nhánh khác với nhánh hiện tại của bạn

* $git branch -d <branchname>

Xóa một nhánh

**CREATE REPOSITORIES**

* $ git init

Khởi tạo 1 git repository 1 project mới hoặc đã có

* $git clone (URL)

Copy 1 git repository từ remote source

**REFACTOR FILENAMES**

* $ git rm [file] : xóa các file bị xóa trên local
* $ git rm –cache [file] : Bạn có thể chọn xóa một tệp hoặc xóa toàn bộ thư mục đang làm việc.
* $ git mv [file-original] [file-renamed] : Tự động cập nhật chỉ mục cho cả đường dẫn mới và cũ

**REVIEW HISTORY**

* $ git log : Lệnh git log giúp bạn xem lại thông tin lịch sự commit, nhằm giám sát sự thay đổi của dự án. Lệnh git log có nhiều tham số để xuất ra, định dạng các thông tin hiện thị theo cách mong muốn. Bạn có thể định dạng cách các thông tin mỗi commit được in ra khi xem, cũng như có thể lọc thông tin nào đó.
* $ git log --follow [file] :

**SUPPRESS TRACKING**

\*.log

build/

temp-\*

Tệp văn bản có tên .gitignore ngăn chặn việc tạo phiên bản ngẫu nhiên của các tệp và đường dẫn khớp với các mẫu được chỉ định

$ git ls-files --other --ignored --exclude-standard : Lệnh liệt kê tất cả các tệp bị bỏ qua trong dự án này

**REDO COMMITS**

* $git reset[conmmit]

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

* $git reset-hard[commit]

Hủy tất cả các thay đổi đối với một commit

**SAVE FRAGMENTS**

* $git stash

Bỏ tất cả các file đã sửa đổi

* $git stash pop

Truy xuất các file đã stashed

* $git stash list

Bạn có thể kiểm tra ngăn xếp stash với tham số list

* $git stash drop

Xóa các file đã stashed

**SYNCHRONIZE CHANGES**

* $git fetch[bookmark]

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. Nó có thể tìm nạp nhiều branch cùng một lúc

* $git marge[bookmark]/[branch]

Kết hợp chi nhánh của dấu trang vào chi nhánh cục bộ hiện tại

* $git push[alias][branch]

Cập nhật các thay đổi lên remote

* $ git pull

Cập nhật dữ liệu từ remote