**MAKECHANGES**

1. $git status:

Hiển thị trạng thái của các file sửa đổi

1. $git diff:

Hiển thị tất cả các thay đổi chưa unstaged trong index và thư mục hiện tại

1. $git add[File]:

Để đưa một tập tin vào Staging Area

1. $git diff --staged

Hiển thị sự khác biệt giữa các file trong staging và phiên bản mới nhất

1. $git reset [file]:

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

1. $git commit –m “[Descriptive massage]”

Lệnh trên tạo ra một commit với nội dung lấy từ vùng staging, một điểm trong lịch sử commit được tạo ra với thông tin là dòng thông tin nhập vào, sau này bạn có thể xem lại lịch sử này bằng lệnh git log

**CONFIGURE TOOLING**

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

Đặt tên người dùng bạn muốn để tham gia commit

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

Đặt email người dùng bạn muốn để tham gia commit

1. $git config –global color.ui auto

Cho phép tô sáng dòng lệnh

**GROUP CHANGES**

1. $git branch

Hiển thị tất cả các nhánh

1. $git branch <branchname>

Tạo ra một nhánh mới

1. $git checkout <branchname>

Di chuyển tới một nhánh

1. $git merge <branchname>

Hợp nhất một nhánh khác với nhánh hiện tại của bạn

1. $git branch -d <branchname>

Xóa một nhánh

**CREATE REPOSITORIES**

1. $ git init

Khởi tạo 1 git repository 1 project mới hoặc đã có

1. $git clone (URL)

Copy 1 git repository từ remote source

**REFACTOR FILENAMES**

1. $git rm (file)

Xóa một file và ngừng theo dõi nó

1. $git rm –cached (file)

Untrack file hiện tại

1. $git mv[File-original][file-renamed]

Tự động cập nhật chỉ mục cho cả đường dẫn mới và cũ

**REVIEW HISTORY**

1. $git log

Hiển thị lịch sử thay đổi

1. $git log –follow[File]

Liệt kê lịch sử phiên bản cho một ﬁle, bao gồm cả tên

1. $git diff[first-branch]…[second-branch]

Sử dụng lệnh git diff để biết được mỗi file đã thay đổi bao nhiêu dòng code

1. $git show[commit]

Để hiển thị thông tin về bất kỳ git object nào

**SUPPRESS TRACKING**

1. $git ls-files –other –ignored –exclude-standard

Hiển thị danh sách các file bị ignore

**REDO COMMITS**

1. $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

1. $git reset-hard[commit]

Hủy tất cả các thay đổi đối với một commit

**SAVE FRAGMENTS**

1. $git stash

Bỏ tất cả các file đã sửa đổi

1. $git stash pop

Truy xuất các file đã stashed

1. $git stash list

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

1. $git stash drop

Xóa các file đã stashed

**SYNCHRONIZE CHANGES**

1. $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

1. $git marge[bookmark]/[branch]

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

1. $git push[alias][branch]

Cập nhật các thay đổi lên remote

1. $ git pull

Cập nhật dữ liệu từ remote