CẤU HÌNH TOOLING

Định cấu hình thông tin người dùng cho tất cả các kho lưu trữ địa phương

$ git config - global user.name "[tên]"

Đặt tên bạn muốn đính kèm vào các giao dịch cam kết

$ git config - người dùng toàn cầu.email "[email address]"

Đặt email bạn muốn đính kèm với các giao dịch cam kết

$ git config - toàn cầu color.ui tự động

Cho phép tạo màu hữu ích cho đầu ra dòng lệnh

Làm thay đổi

Xem lại các chỉnh sửa và tạo giao dịch cam kết

$ git status

Liệt kê tất cả các tệp mới hoặc đã sửa đổi được cam kết

$ git diff

Hiển thị các tập tin văn bản chưa được hiển thị

$ git add [file]

Chụp nhanh tệp tin để chuẩn bị cho phiên bản

$ git diff - staged

Hiển thị tập tin giữa các giai đoạn và phiên bản tập tin cuối cùng

$ git reset [file]

Hủy tệp, nhưng lưu trữ nội dung của tệp

$ git commit -m "[thông báo mô tả]"

Ghi lại ảnh chụp nhanh vĩnh viễn trong lịch sử phiên bản

Git init [ten du an]

Tao kho luu tru cuc bo moi voi ten duoc chi dinh

Git clone [url]

Tai xuong mot du an va toan bo lich su cua no

$ git rm [file]

Xoá tệp khỏi thư mục làm việc và xóa các tập tin

$ git rm --cached [file]

Xóa tệp khỏi kiểm soát phiên bản nhưng lưu tệp cục bộ

$ git mv [file-original] [file-renamed]

Thay đổi tên tệp và chuẩn bị cho cam kết

$ git ls-files - khác --ignored - không bao gồm tiêu chuẩn

Liệt kê tất cả các tệp bị bỏ qua trong dự án này

$ git log

Liệt kê lịch sử phiên bản cho chi nhánh hiện tại

$ git log - theo [file]

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

$ git khác [chi nhánh đầu tiên] ... [nhánh thứ hai]

Hiển thị nội dung giữa hai nhánh

$ git show [cam kết]

Xuất ra siêu dữ liệu và thay đổi nội dung của cam kết được chỉ định

$ git reset [cam kết]

Khong lam tất cả các cam kết sau khi [cam kết] , giữ thay đổi cục bộ

$ git reset --hard [cam kết]

Loại bỏ tất cả lịch sử và thay đổi trở lại cam kết đã chỉ định