tạo một repository mới

Để tạo 1 repository mới, bạn hãy mở cửa sổ lệnh và gõ dòng lệnh sau  
git init

sao chép (clone) một repository

để clone 1 repository có sẵn ở trên máy cục bộ, bạn hãy sử dụng dòng lệnh sau:  
git clone /đường-dẫn-đến/repository/  
Nếu repository đó ở máy chủ khác thì bạn hãy gõ dòng lệnh sau:  
git clone tênusername@địachỉmáychủ:/đường-dẫn-đến/repository

quy trình làm việc

thư mục cục bộ của bạn bao gồm ba "trees" được duy trì bởi git. đầu tiên là Thư Mục Đang Làm Việc (Working Directory) có chứa các tập tin hiện tại. cái thứ hai là Chỉ Mục (Index) đóng vai trò như staging area và cuối cùng là HEAD trỏ đến commit gần đây nhất của bạn.



thêm (add) & commit

Bạn có thể đề xuất thay đổi (thêm nó vào chỉ mục **Index**) bằng cách  
git add <tên-tập-tin>  
git add \*  
Đây là bước đầu tiên trong quy trình git cơ bản. Để thật sự commit những thay đổi, bạn sử dụng  
git commit -m "Ghi chú Commit"  
Bây giờ thì tập tin đã được commit đến **HEAD**, nhưng chưa phải trên thư mục remote.

đẩy (push) các thay đổi

Thay đổi của bạn hiện đang nằm tại **HEAD** của bản sao cục bộ đang làm việc. Để gửi những thay đổi đó đến repository remote, bạn thực thi  
git push origin master  
Thay đổi *master* bằng bất cứ nhánh nào mà bạn muốn đầy những thay đổi đến.  
  
Nếu bạn chưa clone một repository hiện có và muốn kết nối repository của bạn đến máy chủ remote, bạn phải thêm nó với  
git remote add origin <máy-chủ>  
Bây giờ bạn đã có thể đẩy các thay đổi của mình vào máy chủ đã chọn

nhánh

Các nhánh (branches) được dùng để phát triển tính năng tách riêng ra từ những nhánh khác. Nhánh *master* là nhánh "mặc định" khi bạn tạo một repository. Sử dụng các nhánh khác tri đang trong giai đoạn phát triển và merge trở lại nhánh master một khi đã hoàn tất.



tạo một nhánh mới và đặt tên là "feature\_x" và chuyển qua nhánh đó (từ master) bằng cách  
git checkout -b feature\_x  
trở lại nhánh master  
git checkout master  
và xóa nhánh feature\_x đó lần nửa  
git branch -d feature\_x  
một nhánh *không có giá trị với các nhánh khác* trừ khi bạn đẩy nhánh đó đến remote repository  
git push origin <nhánh>

cập nhật & trộn (update & merge)

để cập nhật repository cục bộ của bạn và commit mới nhất, thực thi  
git pull  
trong thự mục đang làm việc để *lấy về (fetch)* và *trộn (merge)* các thay đổi ở remote.  
để trộn một nhánh khác vào nhánh đang hoạt động (vd: master), sử dụng  
git merge <nhánh>  
trong cả hai trường hợp, git cố gắng trộn tự động (auto-merge) các thay đổi. Không may, điều này không phải lúc nào cũng làm được và thường dẫn đến *xung đột*. Trách nhiệm của bạn là trộn *các xung đột* đó thủ công bằng cách chỉnh sửa các tập tin được hiển thị bởi git. Sau khi thay đổi, bạn phải đánh dấu chúng là đã được trộn (merged) với lệnh  
git add <tên-tập-tin>  
trước khi trộn các thay đổi, bạn có thể xem trước chúng bằng các  
git diff <nhánh\_nguồn> <nhánh\_mục\_tiêu>

gắn nhãn (tagging)

người ta khuyên nên tạo nhãn (tags) khi phát hành phần mềm. đây là khái niệm được biết đến, đã từng có trên SVN. Bạn tạo tag mới tên là *1.0.0* bằng cách  
git tag 1.0.0 1b2e1d63ff  
chuỗi *1b2e1d63ff* là 10 ký tự đầu tiên của mã commit (commit id) mà bạn muốn tham chiếu đến bằng nhãn của bạn. Bạn có thể lấy mã commit với lệnh  
git log  
bạn cũng có thể sử dụng ít ký tự hơn từ mã commit, nó chỉ cần phải là duy nhất.

thay thế các thay đổi cục bộ

Trong trường hợp bạn làm sai điều gì đó, bạn có thể thay thế các thay đổi cục bộ bằng lệnh  
git checkout -- <tên-tập-tin>  
lệnh này thay thế những thay đổi trong "tree" đang làm việc với nội dung mới nhất của HEAD. Các thay đổi đã được thêm vào chỉ mục, kể cả các tập tin mới, điều này sẽ được giữ lại.

Nếu bạn muốn hủy tất cả thay đổi và commit cục bộ, lấy về (fetch) lịch sử gần đây nhất từ máy chủ và trỏ nhánh master cục bộ vào nó như sau  
git fetch origin  
git reset --hard origin/master

các gợi ý hữu ích

git GUI tích hợp sẵn  
gitk  
sử dụng kết quả git với nhiều màu  
git config color.ui true  
hiện log trên chỉ một dùng mỗi commit  
git config format.pretty oneline  
sử dụng thêm tập tin tương tác  
git add -i