Getting Started – Bắt Đầu:

1. Cách Truy Cập?

* <https://github.com/>
* Vào Sign In + điền tài khoản và mật khẩu + Click đăng nhập

1. Cách Tạo Remote Repository?

* Đầu tiên tạo 1 Remote Repository mới trên Github + đặt tên cho Repository mới + viết mô tả nếu muốn + chọn Private nếu muốn không ai biết Repository này trên Github + Click Add A README File nếu không muốn hiện phần hướng dẫn tân thủ + nhấn Create Repository
* Nếu không Click Add A README File thì Remote Repository ban đầu sẽ không có nhánh nào còn Click thì có nhánh Main ban đầu

1. Cách Chuyển Từ Public Repository Sang Private Và Ngược Lại?

* Chọn Repository muốn đổi + vào Tab “Settings” + cuộn xuống phần “Danger Zone” + chọn “Change Visibility” + Click “Change to private” hay “Change to public” + nhấn liên tục cái nút ở dưới cái cửa sổ hiện lên

1. Private Repository Khác Gì Public?

* Bất kể người nào cũng xem được Public Repository của bạn miễn là học truy cập được vào Github, còn Private Repository sẽ được ẩn chỉ mình bạn mới thấy
* Public Repository luôn Clone được, Private Repository khi Clone = HTTPS thì nó bắt mình nhập tài khoản và mật khẩu Github, nhưng thay vì nhập cái này, vì mật khẩu nó tàng hình khó chịu vãi, thì sử dụng khóa SSH

1. Khóa SSH?

* Là 1 cặp chuỗi kí tự dài và khó hiểu vãi, 1 cái Private bạn giữ, 1 cái Public trên Github
* Khóa Private là cái tạo ra khóa Public bằng mã hóa, và mỗi khóa Private sẽ cho độc nhất 1 khóa Public, và mỗi khóa Public chỉ được tạo ra từ độc nhất 1 khóa Private

1. Cách Cài Đặt Khóa SSH Và Sử Dụng?

* Mở Bash lên rồi nhập vào dòng lệnh sau

ssh-keygen -t rsa -b 4096

* -t chỉ định thuật toán mã hóa, ở đây là RSA
* -b chỉ định chiều dài chuỗi kí tự, ở đây là 4096 Bit tương ứng 512 kí tự
* Sau đó nhấn Enter 3 lần, đợi 1 lát rồi thoát Bash
* Sau đó thư mục “.ssh” được tạo ra trong thư mục “C:\Users\<Tên Người Dùng Hiện Tại>” đối với hệ điều hành Windows và “\root\.ssh” đối với Linux, lưu ý tất cả nội dung trong “\root” bị ẩn nên cần nhấn “Ctrl” + “H” để hiện, bên trong “.ssh” sẽ có 2 File, và đéo được đổi tên, 1 File không có phần mở rộng chứa khóa Private bạn giữ, 1 File có phần mở rộng “pub” là khóa Public bạn cần đưa lên Github
* Mở File khóa Public lên + Copy Full nội dung của nó trừ kí tự xuống dòng cuối cùng + lên Github + vào phần “Settings” + vào Tab “SSH and GPG keys” + Click “New SSH key” + điền vào “Title” cái mẹ gì cũng được + Paste cái khóa Public vào ô “Key” + Click “Add SSH key”
* Bạn đã cài đặt thành công khóa SSH, bây giờ là sử dụng
* Khi Clone thì Copy đường dẫn SSH chứ đừng Copy đường dẫn HTTPS, nếu không nó lại bắt nhập mật khẩu mệt, khi làm như vậy, Git sẽ tự Check thư mục “C:\Users\<Tên Người Dùng>\.ssh” File chứa khóa Private của bạn, sau đó đối chiếu với từng khóa Public trên Github, nếu khóa Private của bạn đéo mã hóa ra khóa Public nào thì nó sẽ báo lỗi, còn không thì Clone như bình thường, lúc mới Clone lần đầu tiên thì nó sẽ hỏi cái quần què gì thì cứ gõ “yes” rồi Enter

1. Cách Tải 1 Thư Mục Trong Remote Repository?

* Copy đường Link tới thư mục, trên thanh tìm kiếm trình duyệt + nhập từ khóa Down Git trên trình duyệt + vào trang Web đầu tiên + Paste vào ô + Click “Download” + chờ tải

1. Branch?

* 1 Repository sẽ bao gồm nhiều bản thể khác nhau của dự án, gọi là các Branch, trong đó có 1 Branch chính, các Branch khác khi cần thì có thể được gộp vào Branch chính

Git:

1. Cách Tải?

* <https://git-scm.com/download/win>
* Nhấn Click Here To Download + chờ nó Download + mở cái File Exe vừa mới Download + nhấn Next liên tục cho đến khi nó hỏi về Editor thì chọn VS Code + nhấn Next liên tục cho tới khi nó hỏi về tên mặc định của Branch khi khởi tạo Repository cục bộ mới + gõ tên mặc định theo ý muốn + nhấn Next liên tục cho tới khi nó cài đặt + xóa Tick khỏi View Release Notes + Click Finish + xóa tệp Exe ban đầu
* Sau khi làm xong, trong thư mục “C:\Program Files” sẽ xuất hiện thư mục “Git”, trong biến hệ thống Path sẽ có thêm đường dẫn “C:\Program Files\Git\cmd” chứa File lệnh “git.exe”

1. Cách Xóa Hoàn Toàn Khỏi Máy?

* Gỡ cài đặt + xóa thư mục “C:\Program Files\Git”

1. Cách Hoạt Động Của Repository?

* Khi 1 nhóm làm việc, mỗi người sẽ tạo 1 thư mục riêng gọi là các Repository cục bộ và làm phần việc của mình trên Repository này, đồng thời cả nhóm sẽ có 1 Remote Repository chung ở trên Github
* 1 Repository cục bộ có thể kết nối nhiều Remote Repository, lí do là để tiện cập nhật Code lên nhiều nền tảng 1 lúc như Github, Gitlab, …

1. Cấu Hình Của Git?

* Đường dẫn mặc định tới File chứa cầu hình toàn cục của Git là

“C:\Users\<Tên Người Dùng>\.gitconfig”

* Có thể dùng VS Code để chỉnh sửa

1. Tại Sao Lại Cần Cấu Hình Tên Người Dùng Và Email Toàn Cục?

* Để khi Commit thì nó sẽ kèm theo tên và Email của người chỉnh sửa, thuận lợi trong việc xem ai đã sửa Code
* Nhập lệnh sau vào CMD để cấu hình

git config --global user.email <Địa Chỉ Email Của Mình>

git config --global user.name <Tên Mẹ Gì Cũng Được>

1. Cách Gộp Nhiều Câu Lệnh Thành 1?

git config --global alias.<Lệnh Gộp> "!<Lệnh 1> && <Lệnh 2> && …"

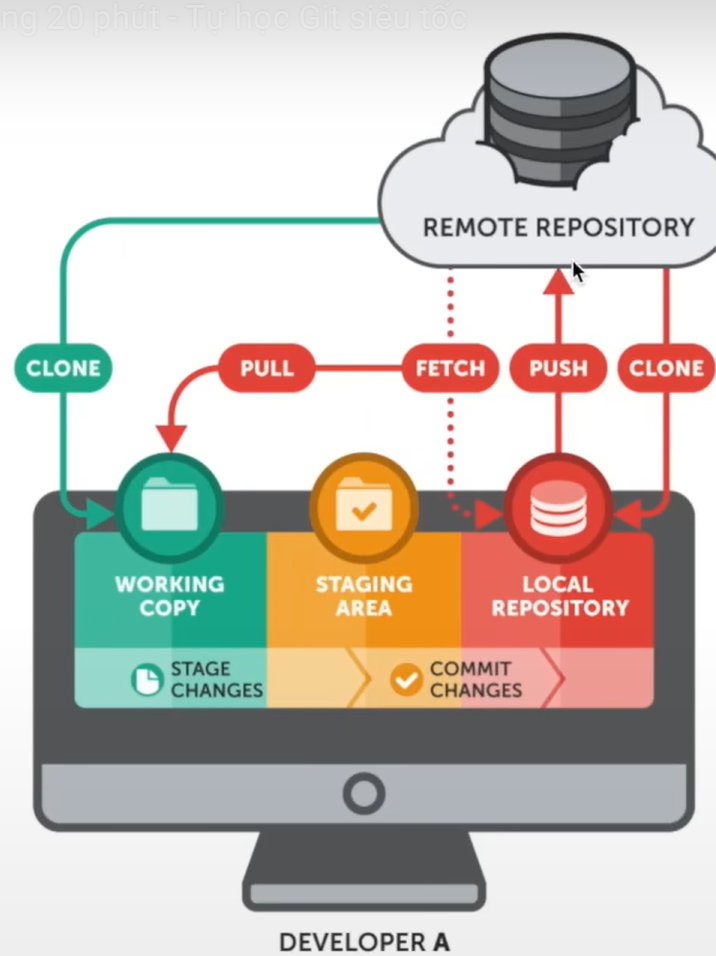
* Ví dụ

git config --global alias.foo "!git add . && git commit –m 'hell no'"

* Bây giờ chỉ cần bạn nhập “git foo” lên CMD là nó sẽ thực hiện “git add .” rồi

“git commit –m ‘hell no’”

1. Các Giai Đoạn Đẩy Code?



* Trong máy bạn, sẽ có 3 khu vực lưu trữ, thứ nhất là Working Copy, là nơi bạn đang chỉnh sửa File, tiếp theo là Staging Area, bạn sẽ có nhiệm vụ đẩy các thay đổi từ Working Copy sang đây, lưu ý đây là các thay đổi, không phải các File, cuối cùng là Local Repository, là nơi chứa File chuẩn bị đẩy lên Remote Repository, bạn sẽ đẩy những thay đổi từ Staging Area vào đây
* Dễ thấy khi bạn mới Clone Remote Repository, thì nó chỉ Copy Remote Repository vào Working Copy và Local Repository, Staging Area sẽ rỗng, ..., các hành động khác bạn có thể suy ra trong hình

Command – Lệnh:

1. Cách Khởi Tạo 1 Local Repository Mới Trong 1 Thư Mục?

* Mở CMD trong thư mục muốn tạo Repository, nhập lệnh

git init

* Câu lệnh này sẽ tạo 1 thư mục ẩn có tên “.git”, nơi chứa dữ liệu như Remote Origin, …

1. Cách Kết Nối Local Repository Với 1 Remote Repository?

git remote add <Định Danh> <Đường Dẫn Đến Repository Trên Github>

* <Định Danh> được coi như 1 biến lưu trữ thông tin của Remote Repository được kết nối
* <Đường Dẫn Đến Repository Trên Github> lấy được = cách Click vào nút Code trong Repository rồi Copy dòng địa chỉ ở Tab HTTPS
* Ví dụ

git remote add origin https://github.com/nguyenduc13475/AIConquer.git

1. Hiện Tên Của Remote Repository Trên Github Đang Được Kết Nối Với Local Repository Ứng Với Định Danh Nào Đó?

git remote show <Định Danh>

1. Xóa Liên Kết Với 1 Remote Repository Thông Qua Định Danh Của Nó?

git remote remove <Định Danh>

1. Chỉ Hiện Tên Của Branch Hiện Tại Trong Local Repository?

git branch ––show–current

1. Hiện Tên Của Branch Hiện Tại Trong Local Repository Cùng Với Những Thông Tin Khác Như Các File Trong Staging Area?

git status

1. Đổi Tên Của Branch Hiện Tại Trong Local Repository?

git branch –m <Tên Muốn Đổi Thành>

1. Liệt Kê Tên Các Branch Trong Local Repository?

git branch

1. Đẩy Code Từ 1 Nhánh Của Local Repository Lên 1 Nhánh Cùng Tên Của 1 Remote Repository?

git push <Định Danh> <Tên Nhánh>

* <Định Danh> là biến lưu thông tin Remote Repository muốn đẩy Code lên
* <Tên Nhánh> là 1 nhánh ở Local Repository, nếu trên Remote Repository chưa có nhánh nào trùng tên với nhánh này thì nó sẽ tạo mới
* Ví dụ

git push mainrepo nhanh1

* Tuy nhiên, nếu sử dụng cú pháp này thì mỗi lần chúng ta đẩy Code lên lại phải nhập lại <Định Danh> và <Tên Nhánh>, rất phiền khi chúng ta chỉ làm việc cố định trên 1 nhánh, do đó lần đầu tiên khi đẩy Code lên thì sử dụng lệnh sau

git push –u <Định Danh> <Tên Nhánh>

* –u sẽ chỉ định kể từ nay khi muốn đẩy Code chỉ cần sử dụng lệnh sau

git push

* Câu lệnh này sẽ mặc định đẩy Code từ nhánh có tên là <Tên Nhánh> lên nhánh cùng tên ở Remote Repository có biến lưu trữ thông tin là <Định Danh>

1. Tại Sao Gặp Lỗi Khi Cố Gắng Đẩy Code Lên Remote Repository?

* Vì chưa có Commit nào, phải Commit trước kể cả khi không có thay đổi gì thì mới đẩy được Code

1. Đẩy Các Thay Đổi Trong Các File Từ Working Copy Sang Staging Area?

git add <Liệt Kê Đường Dẫn Các File Có Phần Mở Rộng>

* Ví dụ

git add foo.txt bar/bob.py

* Ta có thể dùng lệnh sau để đẩy tất cả trong Working Copy

git add .

1. Đẩy Tất Cả Thay Đổi Từ Staging Area Sang Local Repository?

git commit -m <Tin Nhắn>

* Ví dụ

git commit -m "bo dang commit"

1. Xóa Tất Cả Các Thay Đổi Trong Staging Area?

git restore --staged <Liệt Kê Đường Dẫn Các File Có Phần Mở Rộng>

* Để xóa hết

git restore --staged .

1. Tải Xuống Remote Repository Lưu Vào Nơi Tạm Bợ?

git fetch <Định Danh> <Tên Nhánh>

* Giống git push, ta có thể thêm Flag -u để từ sau chỉ cần dùng lệnh

git fetch

1. Đẩy Tất Cả File Từ Nơi Tạm Bợ Vào Working Copy?

git pull

* Ta có thể kết hợp git fetch và git pull như sau

git pull <Định Danh> <Tên Nhánh>

* Giống git push, ta có thể thêm Flag -u để từ sau chỉ cần dùng lệnh

git pull

1. Copy Ra Branch Mới Trong Local Repository?

git branch <Tên Branch Mới>

1. Đổi Branch Hiện Tại Sang Branch Khác?

git checkout <Tên Branch Muốn Chuyển Sang>

* Để vừa tạo Branch mới vừa chuyển sang Branch này luôn thì thêm Flag -b
* Ví dụ

git checkout -b foo

1. Áp 1 Local Repository Của 1 Branch Vào Branch Hiện Tại?

git merge <Tên Branch Áp>

1. Hiện Lịch Sử Commit Của Branch Hiện Tại?

* Để hiện chi tiết từng Commit

git log

* Để hiện mỗi Commit đúng 1 dòng súc tích

git log --oneline

Docker:

1. Cách Cài Đặt?

* Vào Link
* <https://www.docker.com/products/docker-desktop/>
* Click mũi tên bên phải “Download for Windows” + chờ tải + chạy File vừa tải và cài đặt mặc định + khởi động lại máy tính
* Khi này trong thư mục “C:\Program Files” sẽ xuất hiện thư mục “Docker”, trong đây có File EXE “Docker Desktop” là phần mềm Docker
* Lần đầu chạy Docker + Click “Accept” + Click “Finish”