**LinK: https://www.youtube.com/watch?v=QV0kVNvkMxc&list=PL4cUxeGkcC9goXbgTDQ0n\_4TBzOO0ocPR&index=8**

**Why Use Git**

Git là một hệ thống điều khiển các phiên bản phân tán

Cơ bản nó có nghĩa là:

+ Nó là hệ thống ghi lại các thay đổi của các file dữ liệu theo thời gian

+ Chúng ta có thể kiểm tra lại các phiên bản cụ thể của các file dữ liệu đó tại bất cứ thời điểm nào

+ Nhiều người có thể dễ dàng hợp tác trong một dự án và có phiên bản các file dự án riêng của họ trên máy tính của mình.



Tại sao sử dụng Git:

+Lưu trữ các sửa đổi trong một lịch sử dữ án chỉ trong một thư mục

+Xem lại bất kì các sửa đổi nào của dự án mà bạn muốn

+Làm việc trên các tính năng mới mà không làm rối loạn codebase chính

+Dễ dàng hợp tác với các lập trình viên khác

**GitHub**

+Dịch vụ trực tuyến để tổ chức các dự án

+Chia sẻ code với những người khác

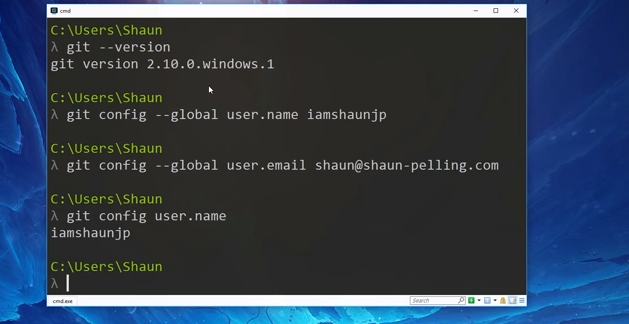
+Có thể tải và làm việc với code đó

+Có thể upload sau khi sửa và hợp nhất chúng với codebase

**Install Git**

Link: <https://git-scm.com/>

Kiểm tra:



**How Git Works**

Respositories

+ Là một container cho project bạn muốn theo dõi với Git

+ Có thể có nhiều respo khác nhau cho nhiều project khác nhau trong máy tính của bạn

+ Giống như một thư mục, Git theo dõi nội dung chúng ta

Modifield : Thay đổi files nhưng ko commit

Staging : Thêm bất cứ changed files tới staging , cái mà bạn muốn commit

Commited: Bất kì files trong staging đã đc thêm tới commit

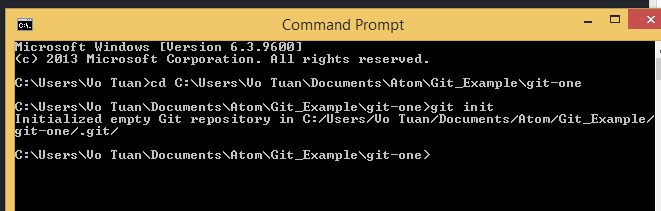
**Create a Respository**

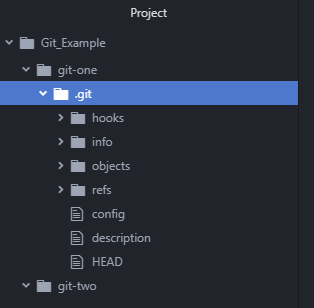
B1. Tạo project mới, và tạo 2 thư mục git-one, git-two

Copy path của git-one tới cmd

Lệnh : Cd + path

Lệnh : Git init =>

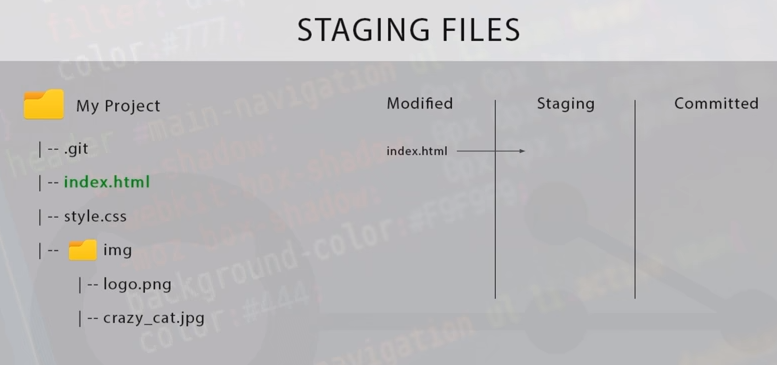




B2: Tao file mới trong git-one

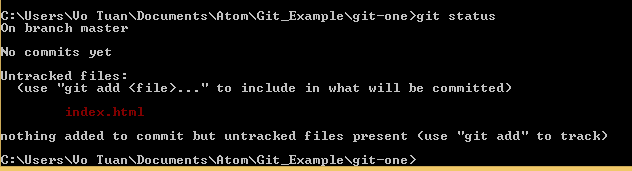
Touch index.html

**Staging files**



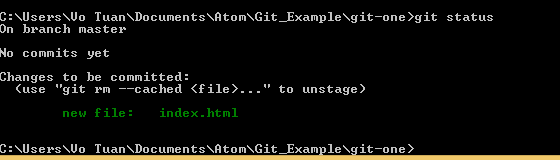
Để commit đc, ta cần chuyển file đã thay đổi tới staging

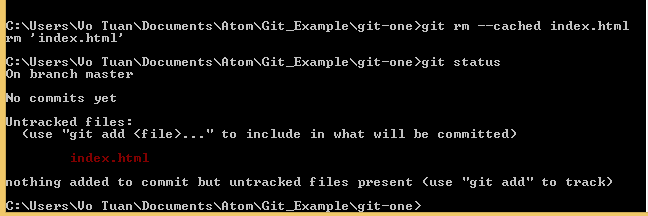
Git-status: Kiểm tra các file có đc track ko



Thêm file đó tới staging

Git add index.html

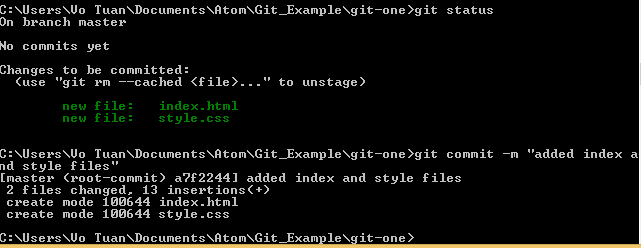




Sử dụng git add . để add tất cả file tới trạng thái staging

**Making Commits**

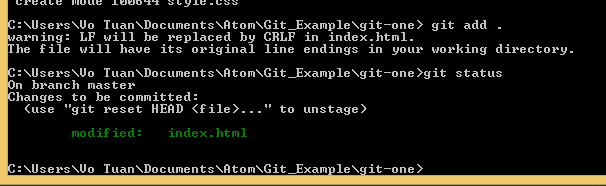


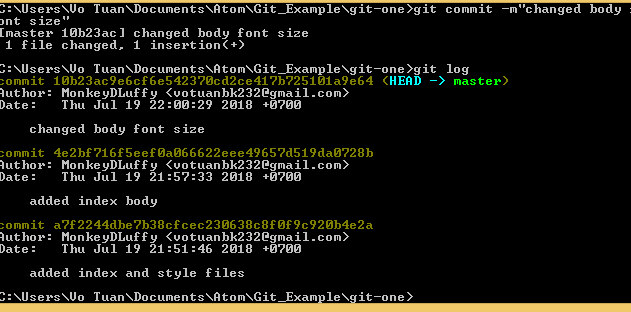
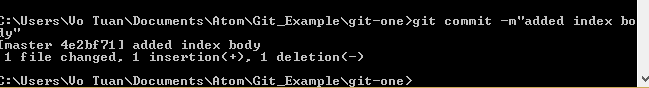


Master branch và id commit là a7f2244j

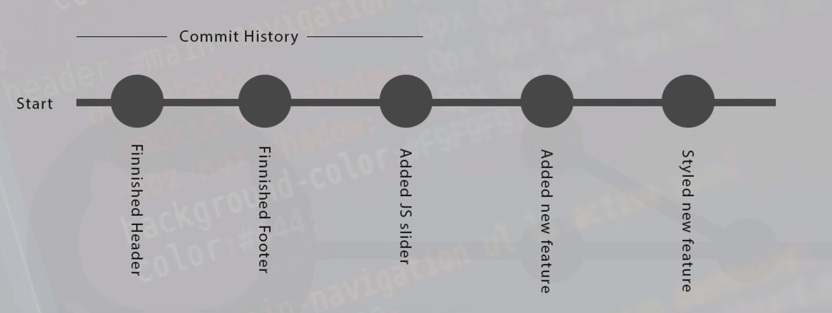
Có 2 files đc thay đổi và 13 dòng

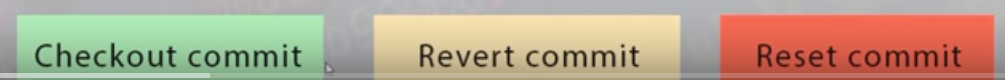
+Sửa file index.html





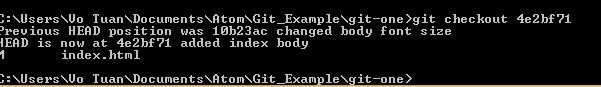
**Undoing Things**





**Sử dụng git checkout:**

=> để quay trở về trạng thái mong muốn

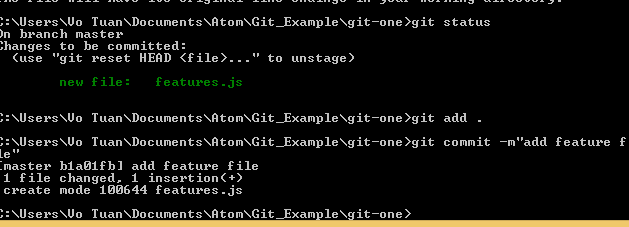


Git checkout master=> trở về trạng thái commit ban đầu

**Sử dụng git revert:**

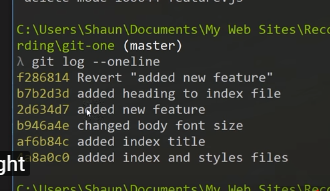
+ Tạo 1 file js và commit nó

+git revert id\_commit: file js đó sẽ bị mất



Sử dụng git revert id\_commit của add features

Nhấn shift + : và ấn WQ



**Tổng hợp lệnh :**

1.git status: kiểm tra trạng thái file

2.git add index.html => chuyển file index về trạng thái stage

3.git add . =>> chuyển tất cả file về dạng stage

4.git rm –cached index.html => chuyển file index về dạng unstage

5.git commit –m”tên của commit” => commit thay đổi

6.git log => xem lịch sử commit

7.git –version

8.git config –global user.name MonkeyDLuffy=> cấu hình tên

9.git config - - global user.email [votuanbk232@gmail.com](mailto:votuanbk232@gmail.com)=> cấu hình email

10.git config user.name=> xem tên

11.git log –oneline=> hiển thị id và message của commit

12.get checkout id\_commit => quay trở về trạng thái commit đó