- git reset: Phục hồi dữ liệu và trở về lần commit trước đó

+ git reset --sotf HEAD~1 --> Undo commit cuối trên git, và chuyển nó sang trạng thái

Staged và không xóa nó như --hard

+ git reset --hard HEAD~1 -->Phục hồi dữ liệu lại lần commit trước đó

+ git reset -- . --> Hủy bỏ đưa file vào vùng Staged

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git commit:

+ git commit -m”description”

+ git commit --amend -m”description” --> Thực hiện commit nhưng nó không tạo ra lần commit mới, thông báo commit ở lần trước đó

+ git commit -a -m”description” -> vừa add vừa commit

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git --version

- git config list

- git status

- git init

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git log: Xem lịch sử commit

+ git log

+ git log --oneline --> Viết tắt nội dung các lần commit

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git restore: Hủy không cho nó trong trạng thái Staged

+ git restore indext.html

+ git restore --staged

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git diff: So sánh dữ liệu

+ git diff --> So sánh file vừa Add với file đang làm việc trên máy

+ git diff --staged --> So sánh file vừa Add với file commit trên Git trước đó

+ git diff sdf23434 sdsdf2423 --> So sánh giữa 2 lần commit

+ git diff sdf23434 sdsdf2423 --stat --> So sánh giữa 2 lần commit Cách 2

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git add:

+ git add .

+ git add indext.html

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git checkout: Phục hồi dữ liệu

+ git checkout d095ec7 --indext.html --> *Phục hồi dữ liệu với mã commit trước đó, đồng thời nó cũng chuyển sang trạng thái Staged sẵng sàng cho lần commit tiếp theo*

+ git checkout -b tên\_nhánh\_mới --> Tạo một nhánh mới

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git branch: Kiểm tra đang làm việc trên nhánh nào

+ git branch:  *Kiểm tra đang làm việc trên nhánh nào, đồng thời xuất danh sách nhánh*

+ git checkout *alpha --> Chuyển đối nhánh làm việc sang alpha*

*+* git branch -D tên\_nhánh\_cần xóa *--> Xóa một nhánh*

*+* git bran tên\_nhánh\_cần\_tạo *--> Tạo một nhánh mới*

*+ git bran -a --> Kiểm tra các nhánh*

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git remote : Xem kho chứa

+ git remote -v *--> Xem các kho chứa trên Git*

+ git remote add tên\_kho\_chứa URL *--> Thêm một kho chứa*

+ git push origin main *--> Đẩy code lên Git*

*+* git pull *--> Lấy code về máy localhost*

+ git remote rm tên\_remote **--> Xóa một kho chứa**

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git tag: Tạo tên cho commit

+ git tag -a “V1.temp” -m”Phien ban temp 1” mã\_commit *(nếu không để mã commit thì nó sẽ lấy mã commit cuối )*

+ git show V0.temp *--> hiển thị nội dung tag*

+ git tag -d V0.temp *--> Xóa một tag*

+ git checkout V0.temp *--> Qua về lần commit trước*

*+ git push origin\_Git V1.temp --> Nó tạo ra một file .Zip trên Git để ta tải xuống*

*+ git push origin\_Git --tag --> Đẩy tất cả các tag lên Git*

*+ git tag -d V3.temp -- Xóa tag ở trên local trước*

*+ git push --delete origin\_Git V3.temp -- Tiếp theo là xóa trên remote*

*+ git push --all -- Đẩy code lên hết*

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git fetch : tải dữ liệu về nhưng không ảnh hưởng đến file ở local

*+ git fetch origin master --> Tải thông tin một nhánh cụ thể*

*+ git fetch orrigin*

*+ git fetch --all*

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

- git push:

+ git push orrigin

+ git push --delete origin tên\_nhánh --> Xóa một nhánh trên remote

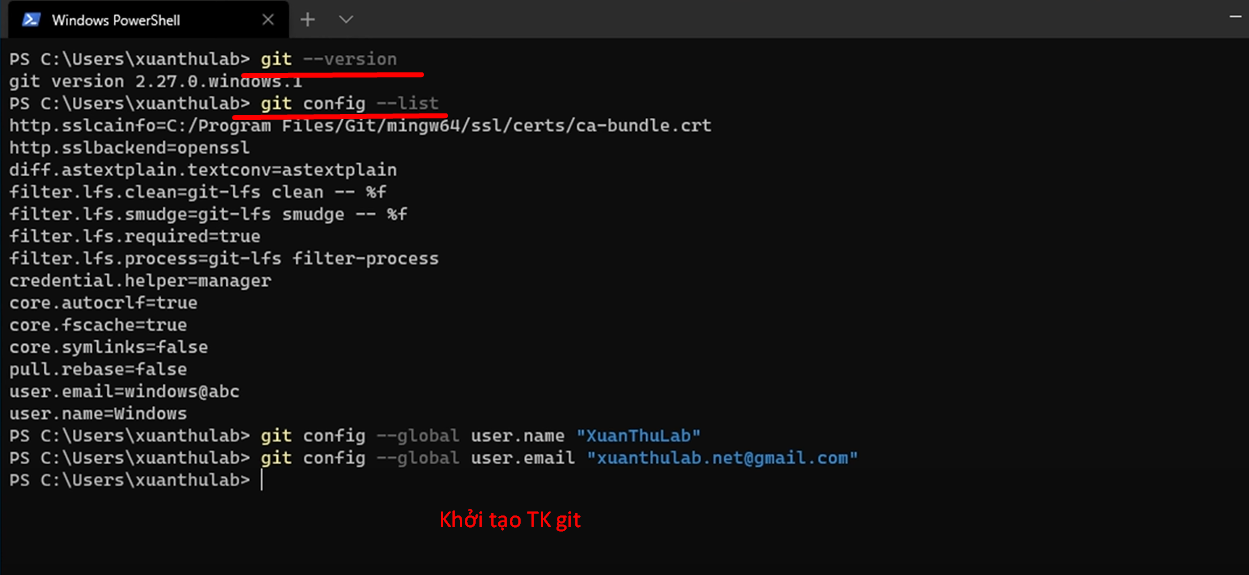
- gitk

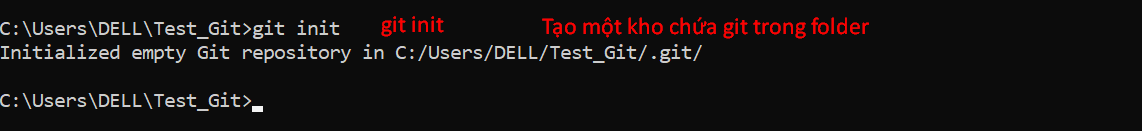
*+ gitk dùng để xem đồ họa thay đổi commit*

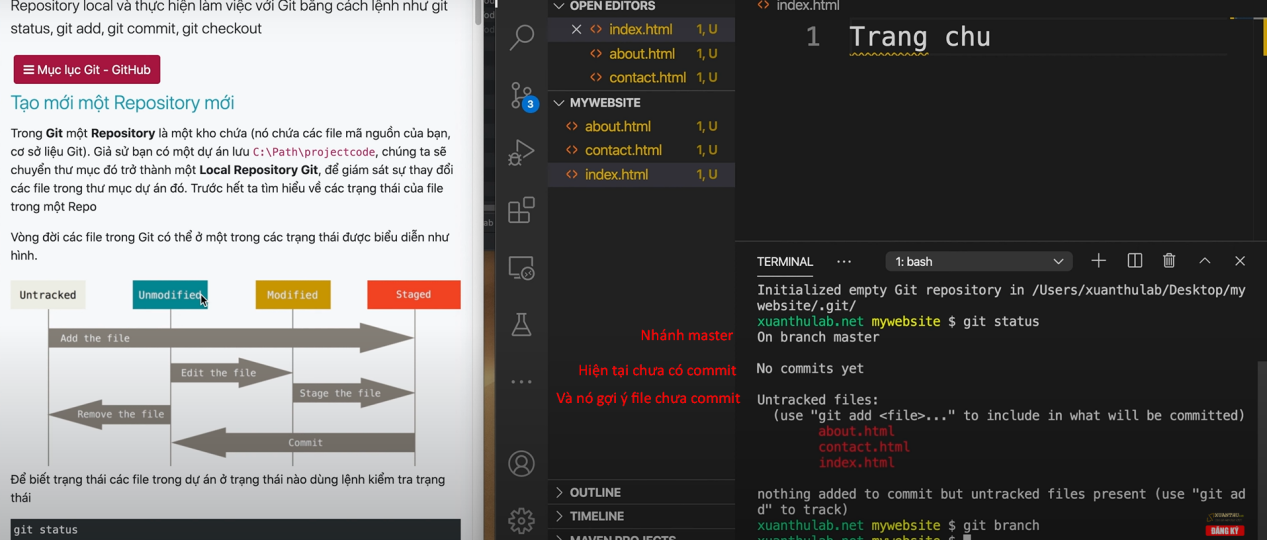
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

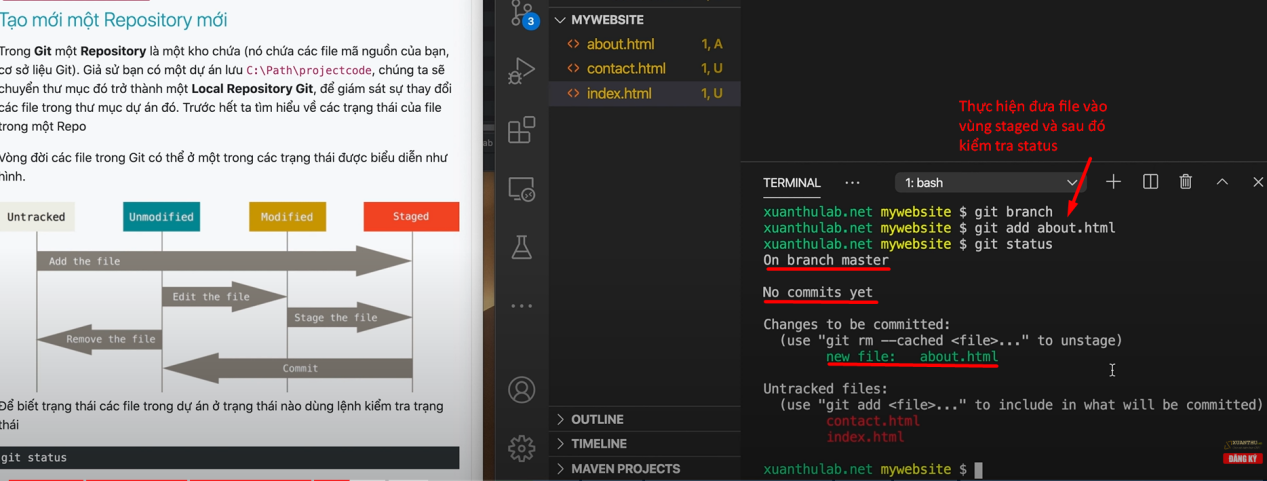
*Link youtobe học git*

*https://www.youtube.com/watch?v=pC1s6JOwglE&list=PLncHg6Kn2JT6nWS9MRjSnt6Z-9Rj0pAlo&index=3*

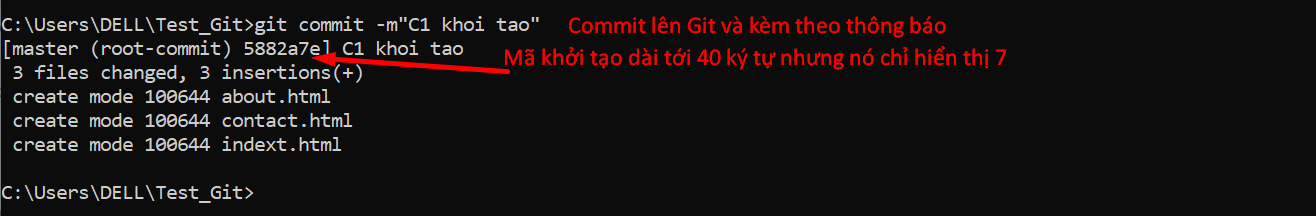


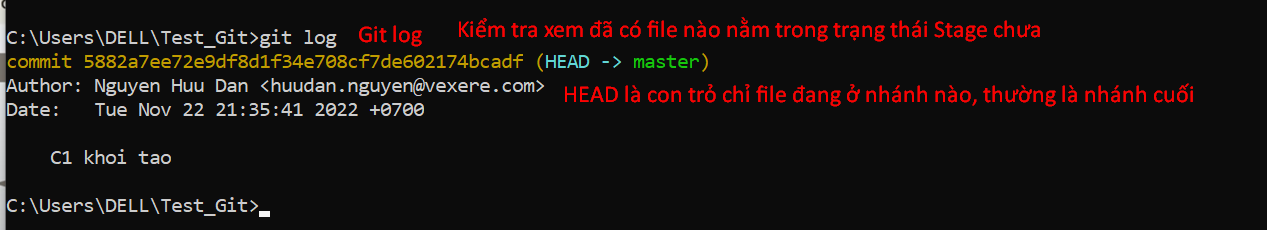


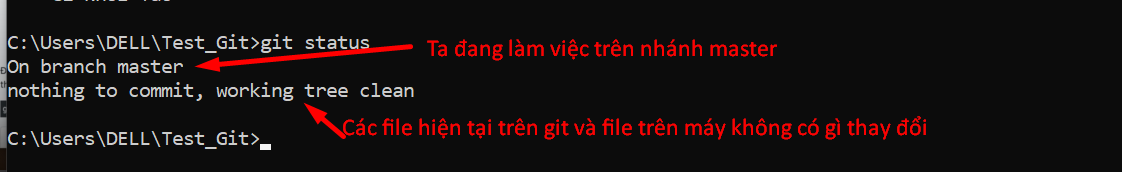


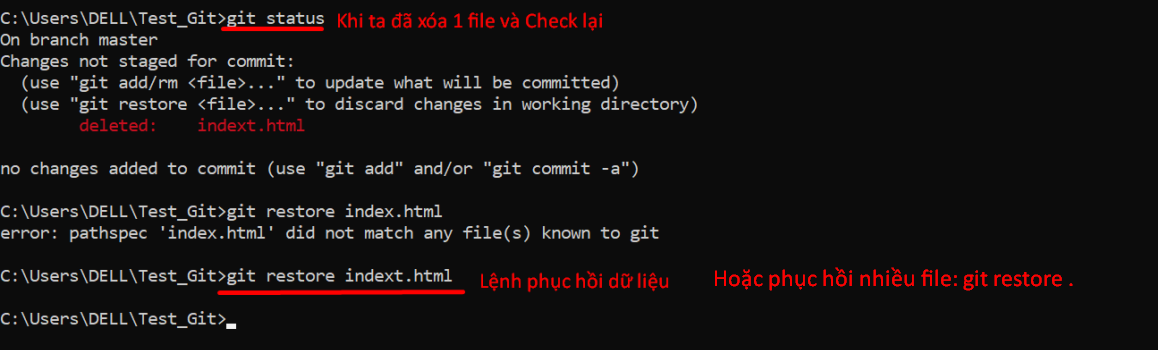


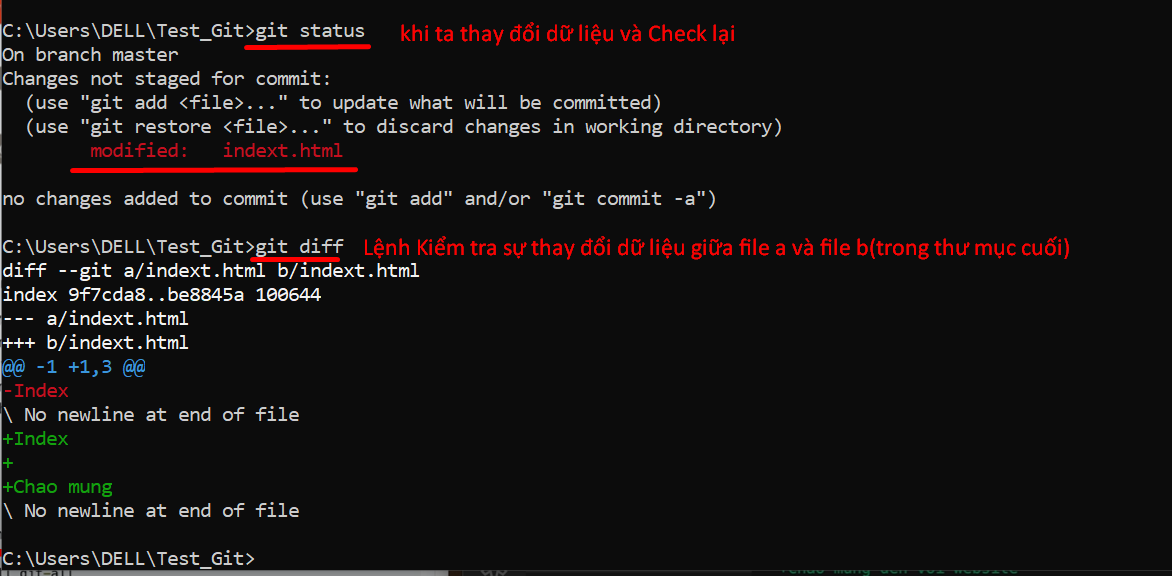
-- Hoặc sử dụng git add . để đưa tất cả các file ở Untracked hoặc file đang ở trạng thái Modify vào Staged

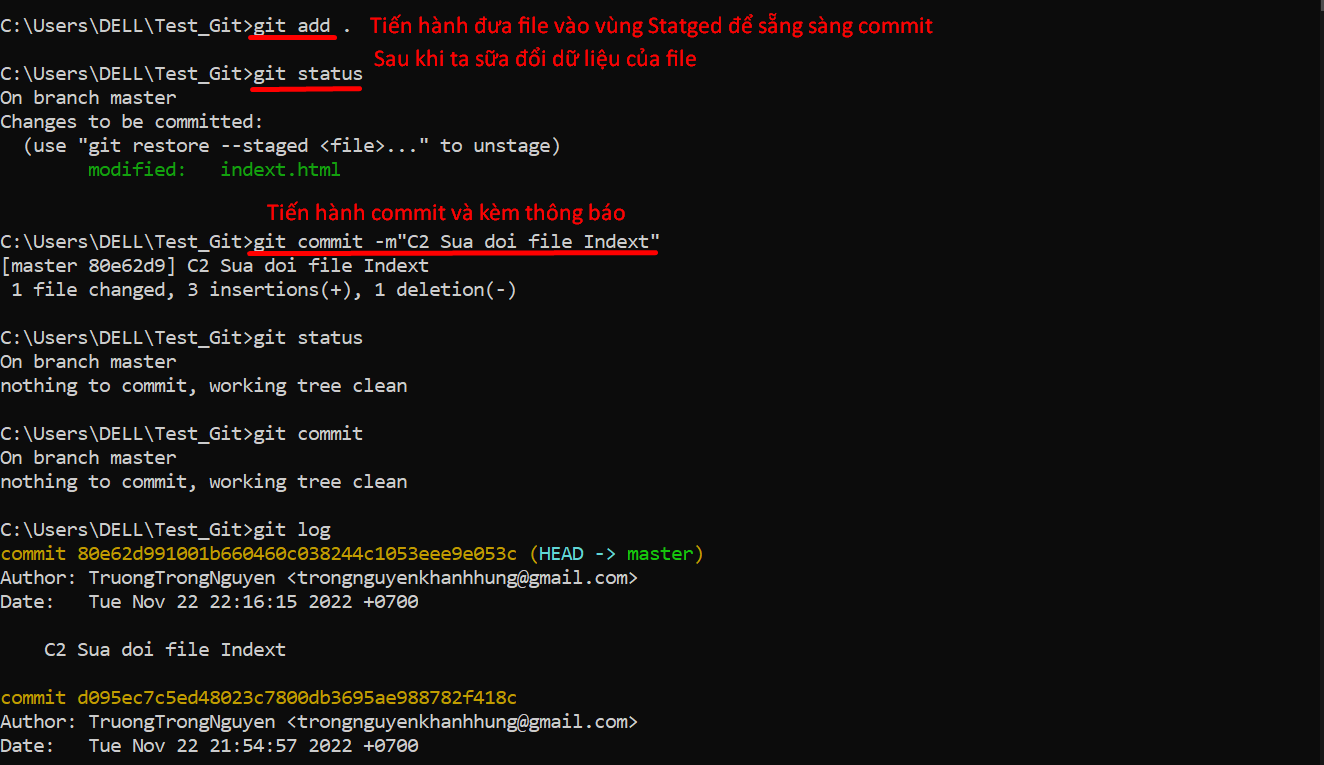


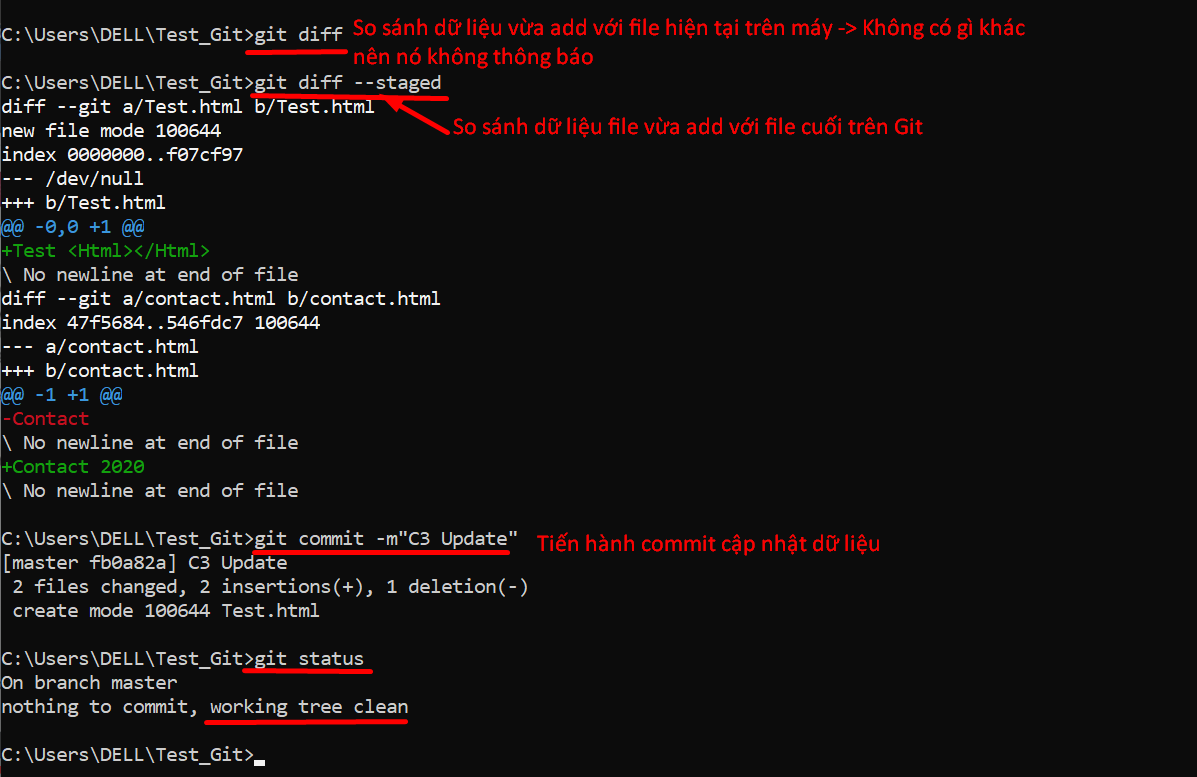


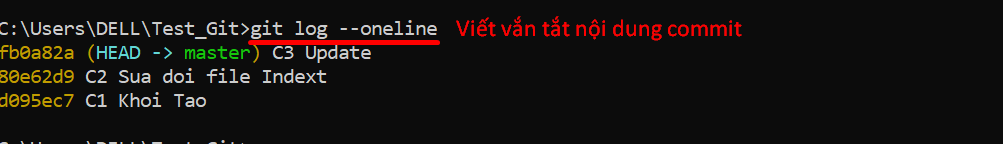


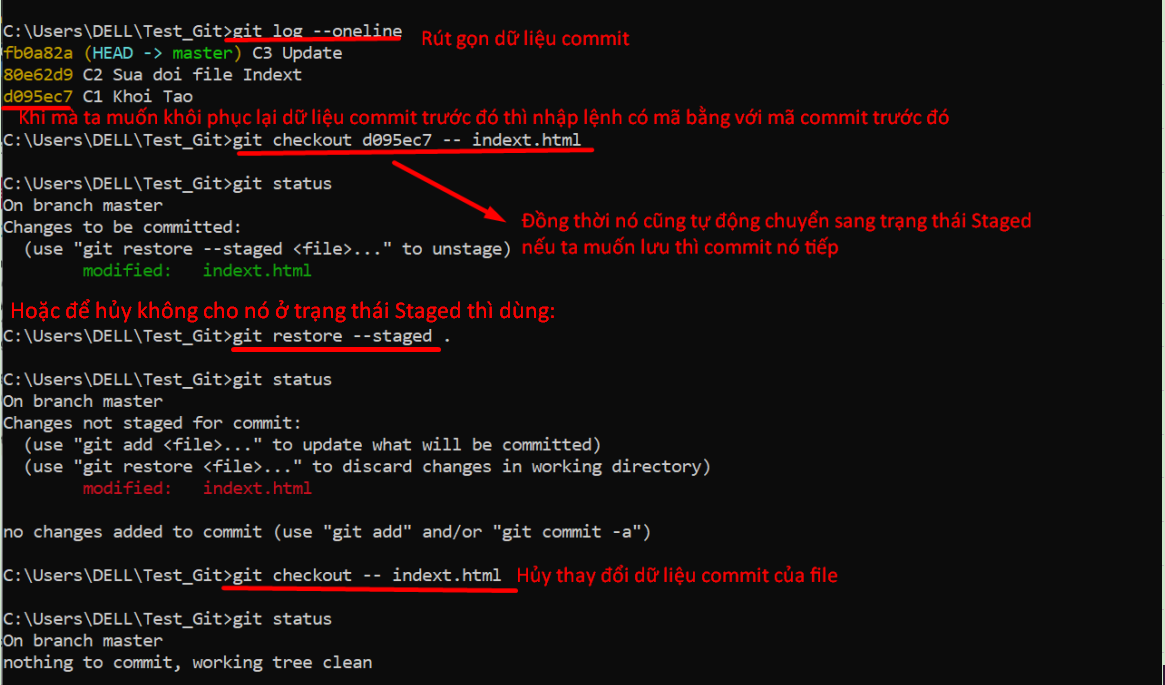


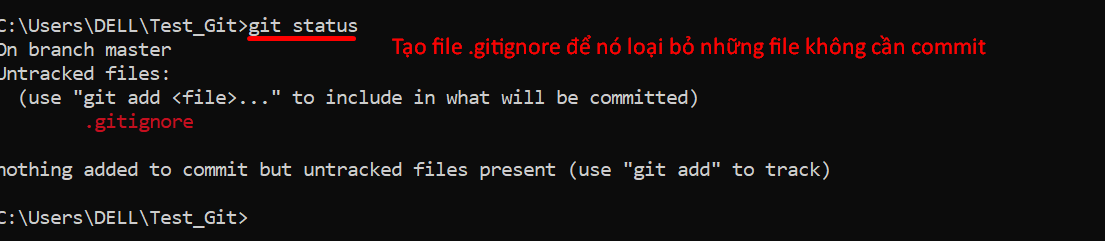












git