Học lại:

* git init: Khởi tạo git trong dự án
* git add . || git add + tên\_file: Thêm những thay đổi vào trong staging area để commit
* git commit -m “content\_commit”: commit tất cả file được git add, có nghĩa là đóng gói lại với nội dung “content\_commit”.
* git status: hiển thị trạng thái của git (đang ở branch nào, những file được tạo mà chưa được git add, những file đã được git add mà chưa commit, … ). Nếu tất cả đã được commit thì sẽ báo nothing to commit, working tree clean.
* git log: hiển thị lịch sử commit, dòng vàng “commit id\_commit” là id của commit đó.
* git show + id\_commit: Hiển thị chi tiết của commit đó.

Note: Những file mới tạo mà chưa git add khi git status sẽ hiện là untracked file, còn những file đã tạo mà có thay đổi code thì sẽ hiện modified: tên\_file.

* git diff: xem sự khác nhau của những file modified
* có 3 trường làm việc trong git:
  + working directory: nơi viết code
  + staging area: nơi chứa sau khi các file được add
  + git repository: nơi chứa những file sau khi commit
* git checkout -- tên\_file: để loại bỏ phần bị thay đổi trong file. Nếu thay đổi trong working directory thì lệnh này sẽ khôi phục tệp tin về phiên bản trong taging area. Nếu thay đổi trong taging area thì lệnh sẽ khôi phục về phiên bản trong git repository.
* git reset HEAD tên\_file: đưa file từ staging area về working directory, hoặc loại bỏ thay đổi sau khi git add. (đưa file về trạng thái trước khi git add || đưa về phần commit ở đầu của dãy commit).
* git branch: hiển thị tất cả tất cả nhánh đang có trong máy.
* git branch tên\_branch: tạo 1 branch mới trong máy.
* git checkout tên\_branch: chuyển sang làm việc với branch có tên là tên\_branch.
* git checkout -b tên\_branch: tạo và chuyển sang làm việc với branch vừa tạo

Note: khi tạo 1 branch mới, trạng thái của branch này sẽ giống với trạng thái của lần commit cuối cùng.

* git merge tên\_branch: lấy thay đổi của tên\_branch sang branch hiện tại.

Ví dụ: đang ở branch A -> git merge branch B -> branch A sẽ thay đổi theo branch B.

* git branch -D tên\_branch: xóa branch.
* git reset --soft + id\_commit: đưa branch về trạng thái commit của id\_commit và các commit đã được commit sau commit đó sẽ về staging area.
* git reset --mixed + id\_commit: đưa branch về trạng thái commit của id\_commit và các commit đã được commit sau commit đó sẽ về working directory.
* git reset --hard + id\_commit: đưa branch về trạng thái commit của id\_commit và các commit đã được commit sau commit đó sẽ bị xóa.
* git revert + id\_commit: tạo 1 commit mới và đảo ngược commit có id\_commit.
* .gitignore: tạo file chứa tên các file sẽ không được commit khi commit
* git remote add origin + link\_repository: link repository local với remote repository.
* git remote -v: hiện ra danh sách remote.
* git push -u origin master: push branch master lên remote repository. Những lần sau đó chỉ cần git push.
* git clone + link\_repository: clone project về máy. Clone sẽ có đủ hết các lịch sử commit, các nhánh và tệp tin từ remote repository. 2 repository sẽ được liên kết với nhau, git default đặt tên repository là “origin”.
* git pull: tải bản mới nhất của remote repository về nhánh hiện tại trên local repository.
* git push origin + tên\_branch: push branch lên remote repository
* git fetch + tên\_branch: lấy code của 1 branch trên remote repository về máy. Sau đó git checkout + tên\_branch đó để sang branch đó làm việc.

// Cũ

git init: tạo 1 repositories trong máy

git clone + link github: lấy từ trên mạng về

git pull: đồng bộ từ trên mạng về

git add: set up để thêm, sửa file vào repositories trong máy

git commit -m “comment”: commit file vừa add vào repositories, với comment là mục đích commit

git push: đồng bộ từ máy lên mạng

git log, git status: check trạng thái folder repositories trong máy

* git init
* git add .
* git commit -m "first commit" – lưu thay đổi code trong repositories
* git branch -M main – chọn branch trong máy để push code
* git remote add origin https://github.com/DucTien2003/test.git - chọn repositories trên github để push code
* git push -u origin main – push code lên github

// document

git branch + nameBranch: sau khi tạo sẽ có 1 branch master và các branch phụ đc tạo

git checkout + nameBranch: chuyển sang code branch phụ

git checkout -b + nameBranch: tự tạo branch + checkout

git merge + nameBranch: merge branch vào master, sau đó push code lên mạng

- git init: khởi tạo một kho chứa Git mới (Git Repo) ở local

- git status: xem trạng thái của repository

- git add {tên file}: chọn file để đẩy

- git add .: đẩy toàn bộ file

- git reset: hủy commit cuối hoặc staging

- git commit -m "":

- git log: xem lại lịch sử các commit trên git

- git log --oneline: hiện commit ngắn gọn

- git checkout: chọn branch

- git branch: kiểm tra branch

- git branch {branch name}: trở lại về thời điểm hiện tại

- git checkout {branch name}: vào branch đó

- git checkout -b {branch name}:tạo một branch mới

- git merge {branch name}: tổng hợp lại các branch

- git branch -d {branch name}: xóa đi một branch

- git remote add origin {repository url}:

- git push {link repository} {tên branch}: đẩy lên local repository

- git push origin {branch name}:

- git clone {repository url}: lấy về

- git fetch origin: lấy branch từ repo về

- git checkout -b {branch name} origin/{branch name}:

TẠO BRANCH

- git checkout -b feature <tên branch>

- gid add .

- git commit -m "$số và tên commit"

- git push

Lấy code mới về

- Vào branch cần pull

- git pull origin tên branch