**Fresher: Lâm Diệp Thảo**

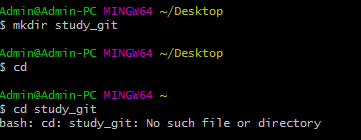
**BÀI TẬP GIT**

**Bài 1.2:**

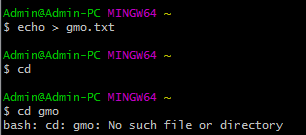
1. Cài đặt Git, xem phiên bản đang sử dụng:



1. Tạo thư mục study\_git:



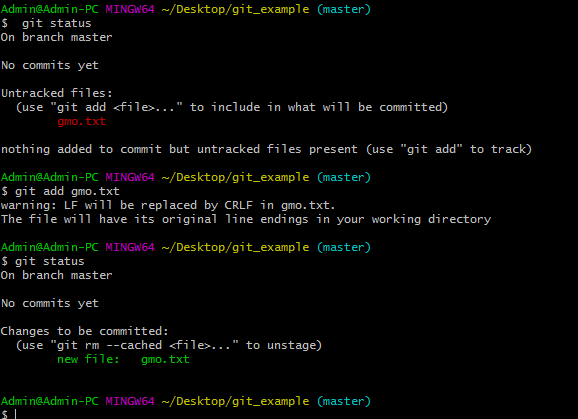
1. Tạo file gmo.txt:



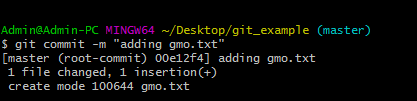
1. Khởi tạo một repository trống:



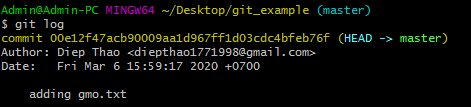
1. Add file gmo.txt vào staging area (index):



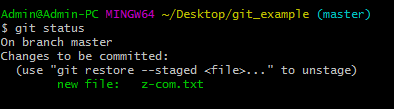
1. Commit file này với message “adding gmo.txt”:



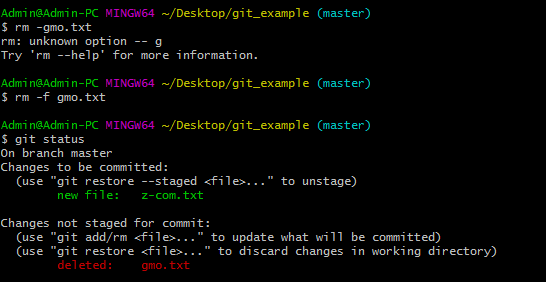
1. Thực hiện check out commit này:



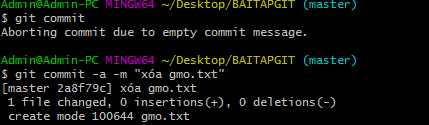
1. Tạo file z-com.txt, sau đó add vào staging area và commit:



1. Xóa file gmo.txt:



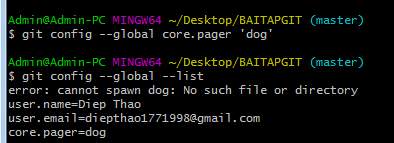
1. Add thay đổi này vào staging area, commit lên với message “xóa gmo.txt”:



1. Thay đổi cấu hình global core.pager=dog:

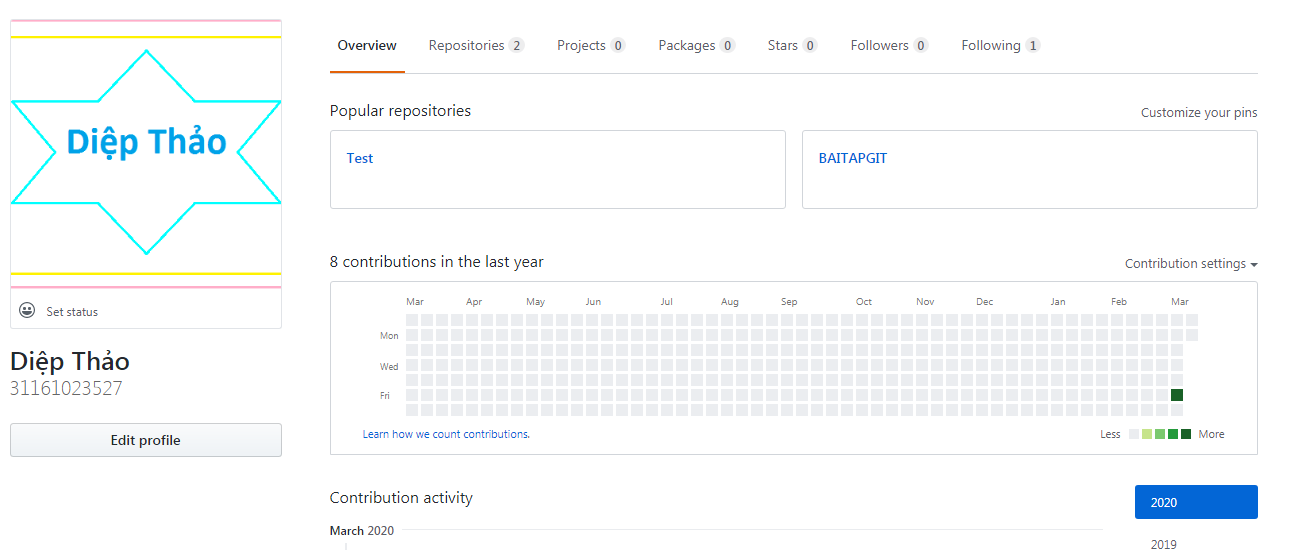


1. Viết lệnh để liệt kê tất cả các cấu hình global cho git.

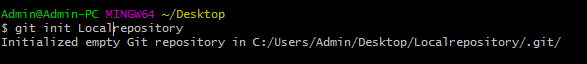


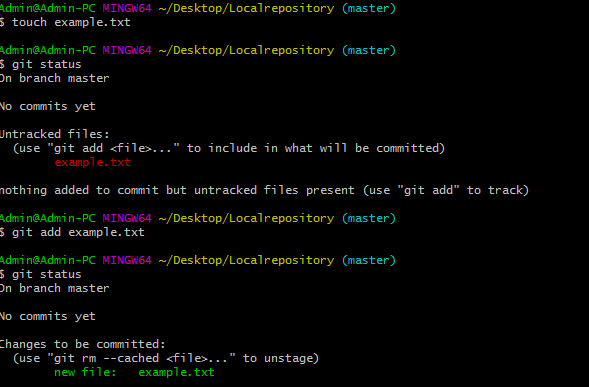
**Bài 2.2:**

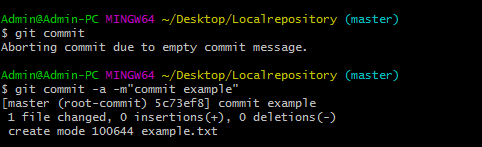
1. Tạo tài khoản GitHub:



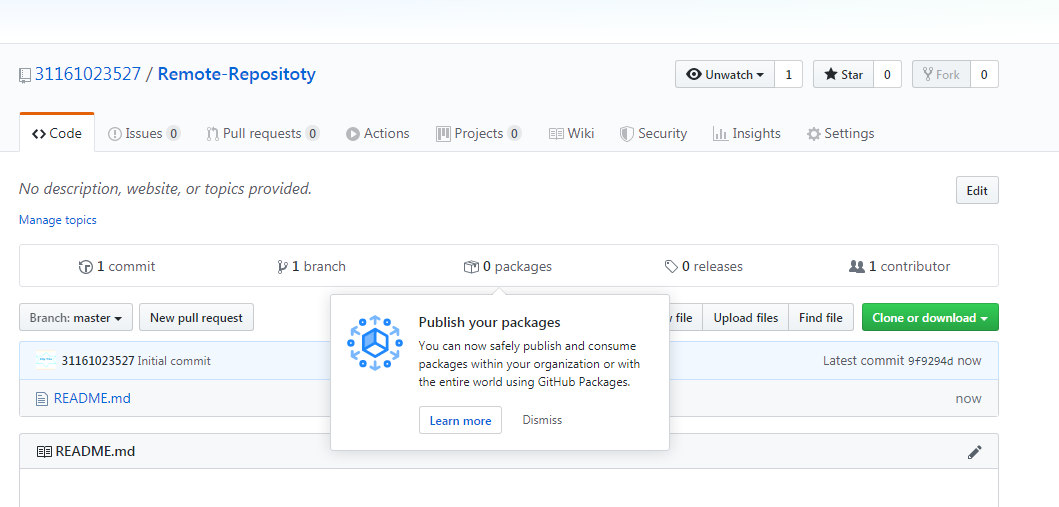
1. Tạo 1 local repository, add, commit 1 số files:

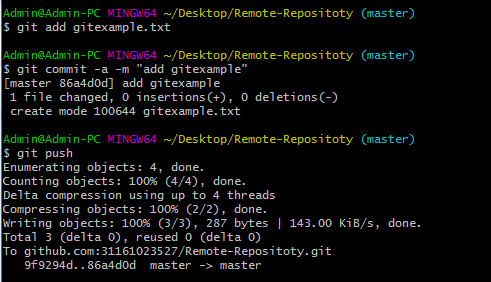


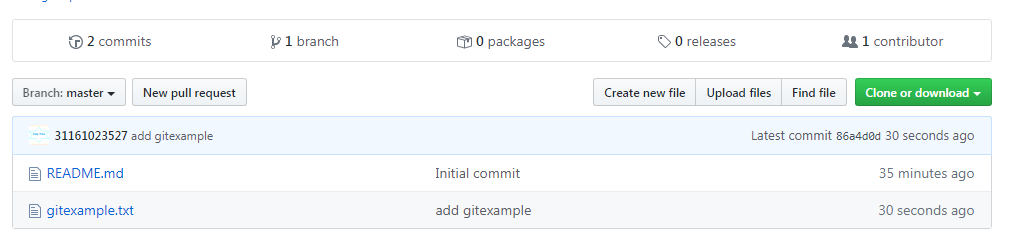




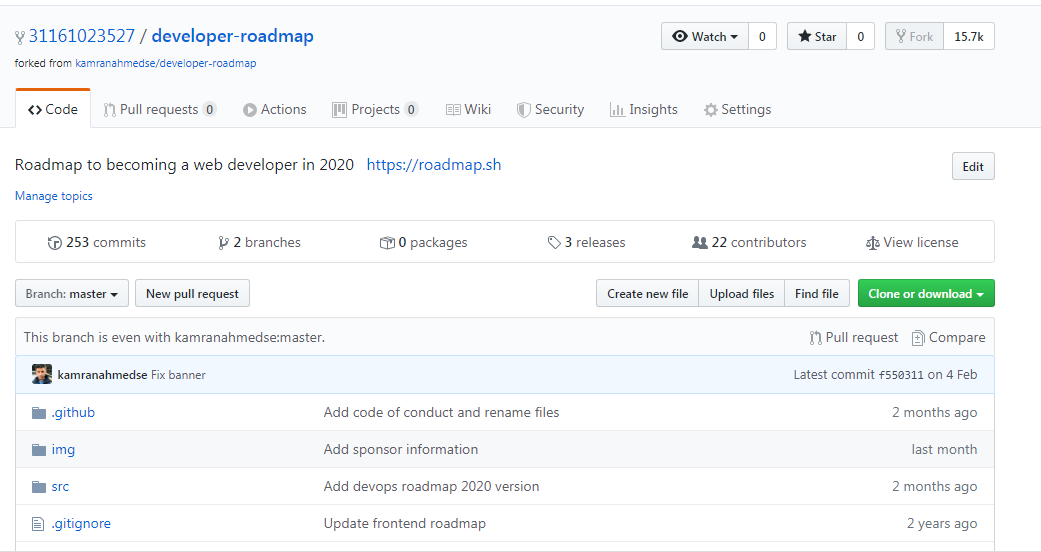
1. Tạo 1 remote repository và push code từ local lên remote:

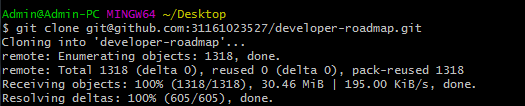


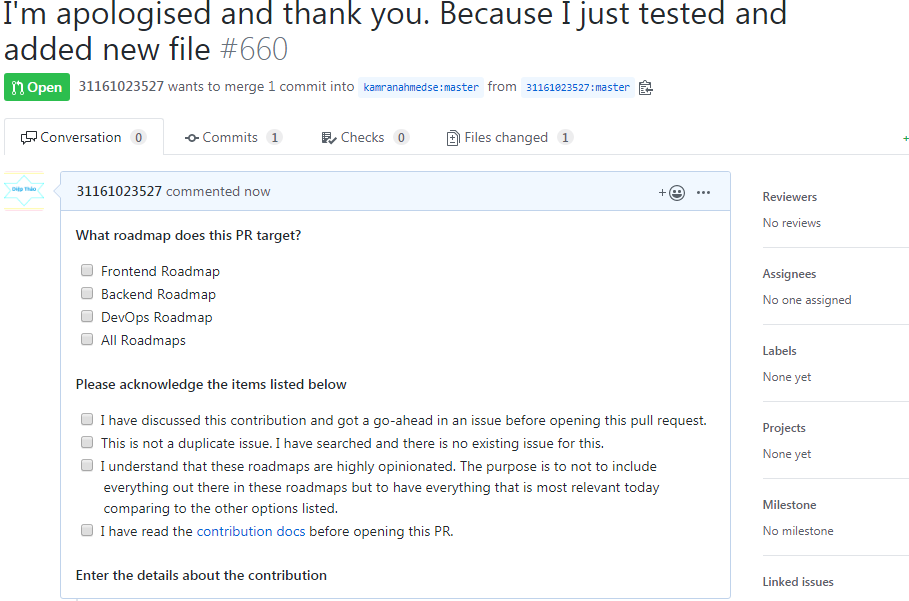




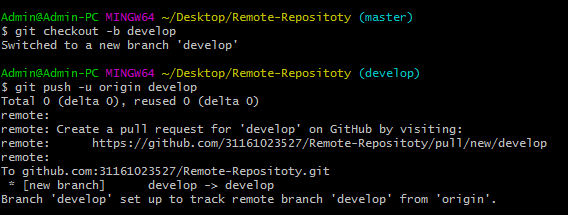
1. Fork 1 repo bất kì nào đó – clone và thực hiện gửi pull request:



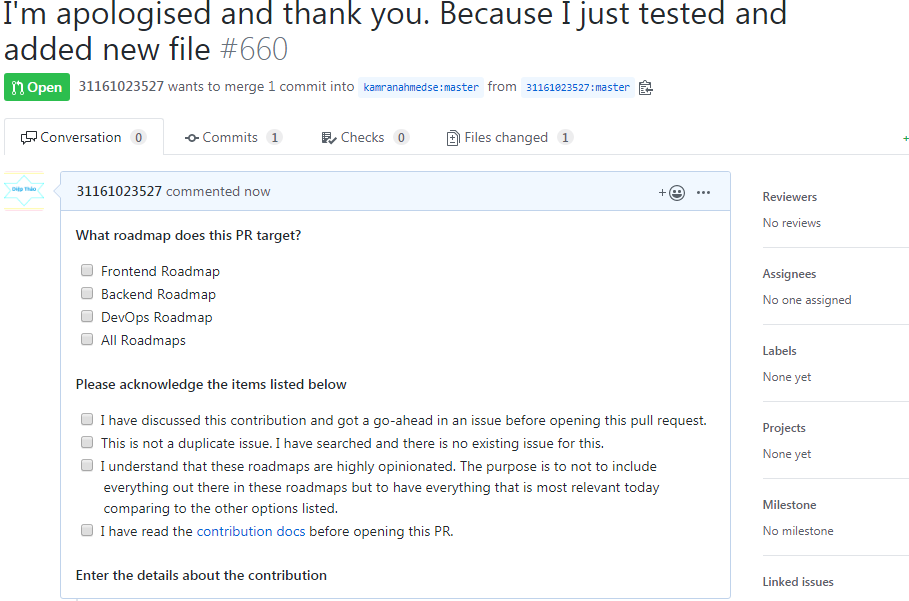




1. Tạo 1 branch mới trên local rồi push lên GitHub:



1. Gửi 1 pull request với nhánh mới đó trên repo vừa fork ở bài 4:



Bài 3.2:

1. **Trả lời các câu hỏi sau vào file “Bai tap Git.docx”:**

- **Lệnh Git clean làm gì?** Dùng để xóa untracked files

- **Flag -d và -f trong git clean để làm gì?** Dùng để xóa cả folder

- **Lệnh nào tạo branch?** Lệnh Git branch <name\_branch>

- **Sự khác nhau giữa 2 phương pháp merge fast forward và recursive?**

+ Merge fast forward: Khi chúng ta merge 2 nhánh với nhau thì nhánh được merge chỉ thêm được cập nhật thêm nhưng file mà nhánh kia không có…  
ví dụ nhánh A có các file a,b,c và nhánh B có các file a,b,c,d, khi chúng ta marge nhánh B vào nhánh A thì nhánh A bây giờ sẻ được cập nhật thêm file d, tức là nhánh A sẻ có file a,b,c,d….   
+ Recursive: Khi chúng ta merge 2 nhánh với nhau thì nhánh được merge sẻ cập nhật các file đã thay đổi trước đó và cập nhật thêm những file mà nhánh được merge không có.

**- Lệnh git nào thay đổi sang nhánh khác?** Lệnh Git checkout <name\_branch>

- **Làm thế nào để bỏ những file đã sửa hoặc xóa ở trong thư mục làm việc?** Git rm –f < name\_file>

- **Lệnh nào xóa branch?** Git branch –D <name\_branch

- **Lệnh git diff làm gì?** Dùng để xem thay đổi ( chưa dc add ) của những file hiện tại

- **Làm thế nào để xóa file từ staging area?** Git rm –cached < name\_file>

- **Conflict xảy ra như thế nào? Làm thế nào khắc phục được?** Conflict xảy ra khi có 2 sự thay đổi trên cùng 1 dòng code, Git không thể xác định đâu là đúng ý coder.

1. Tự tạo conflict bằng cách làm việc với 1 file trên 2 nhánh riêng biệt, sau đó merge 2 nhánh đó lại. Tự sửa lỗi conflict đó.

