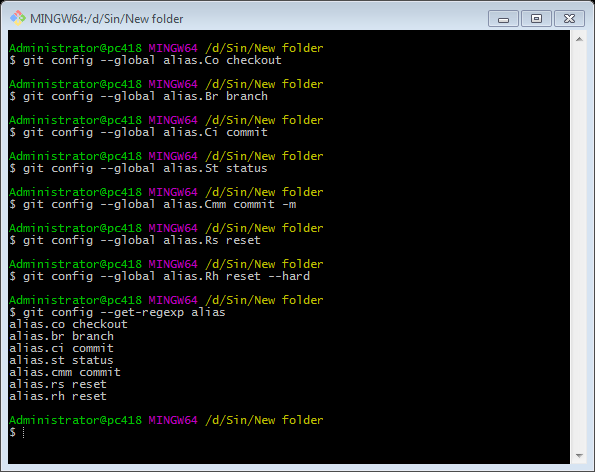
**Thực Hành Lab-04**

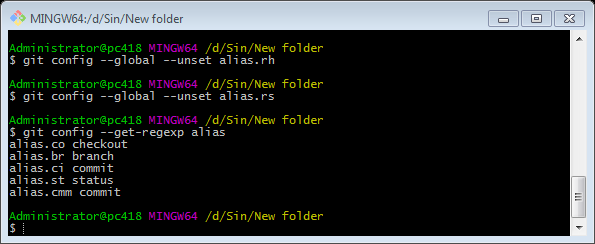
Họ Và Tên: Võ Thị A Sin – 1711061697

BÀI TẬP

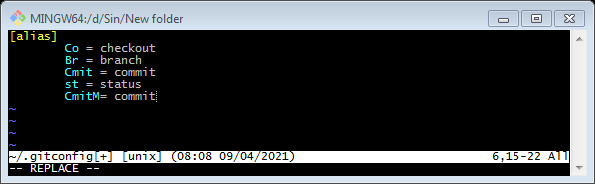
Set up các Alias:

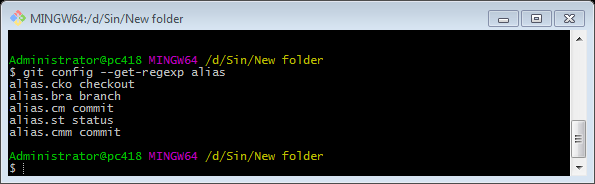


Xóa 2 alias

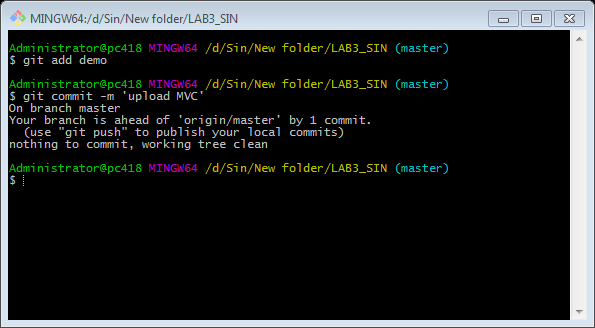


Sửa 2 alias

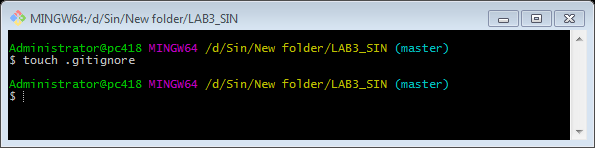


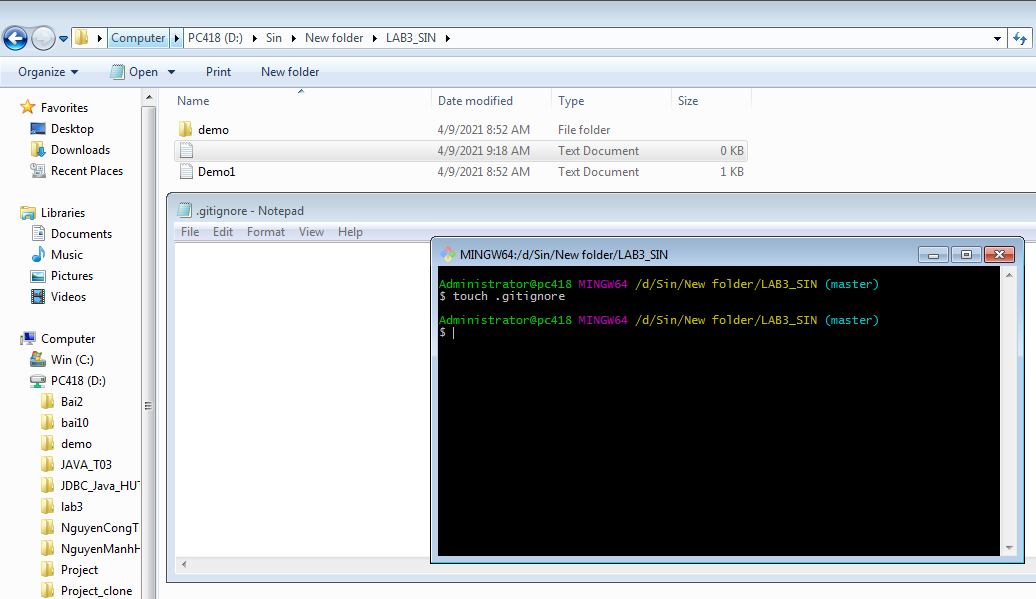


Tạo Project MVC và commit nội dung của project

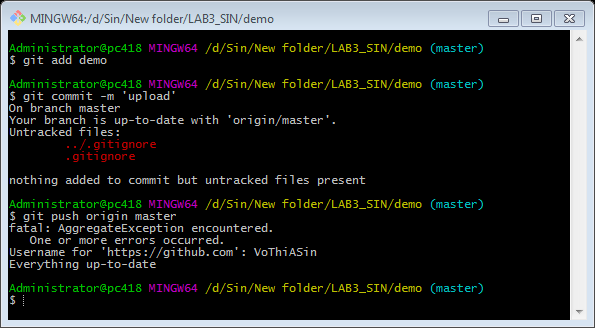


Đánh dấu file có đuôi sln sao cho file đó quay về trạng thái Unstaged

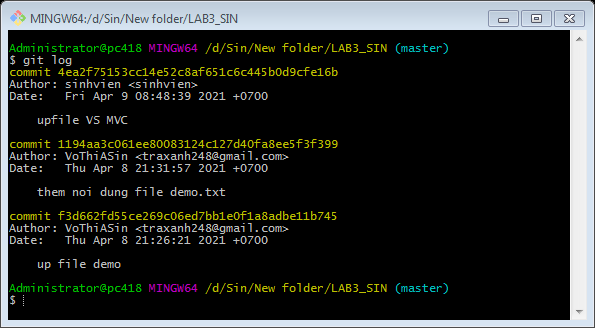




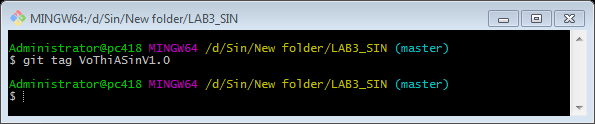
Push project lên github

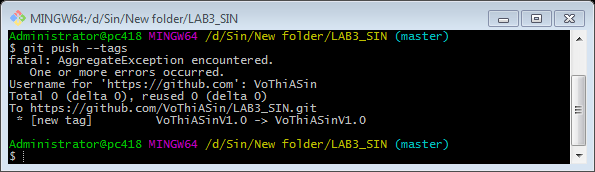


Log lịch sử commit

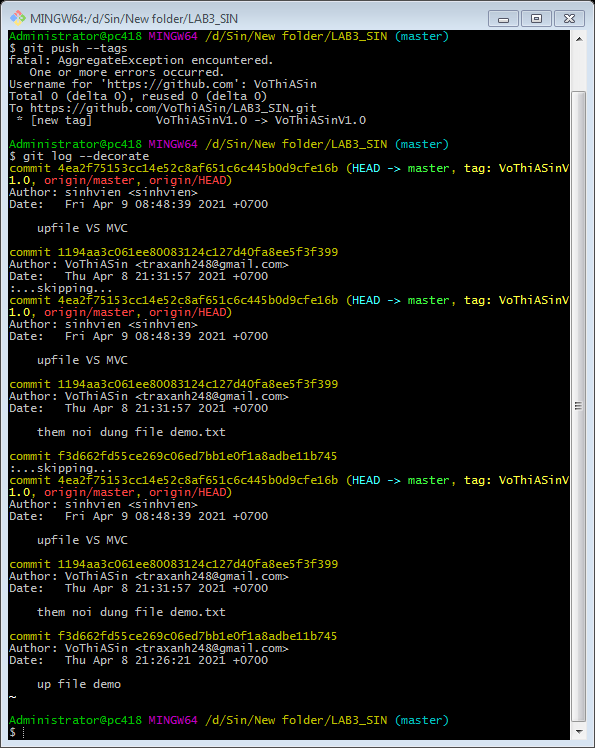


Thêm 1 Tag có tên là tên của sinh viên viết liền không dấu + V1.0 (vd : NguyenThanhTungV1.0) và push lên

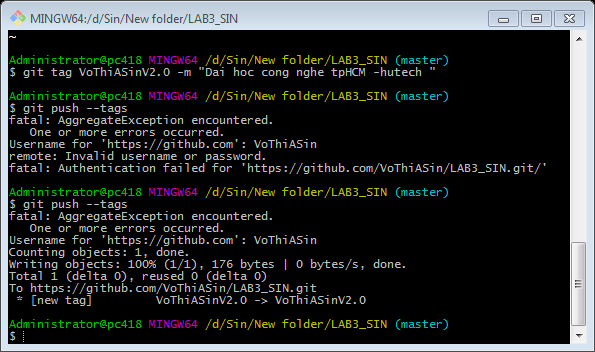


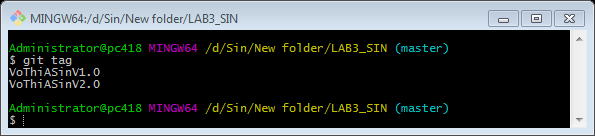


Sử dụng lệnh “log --decorate” và chụp lại kết quả

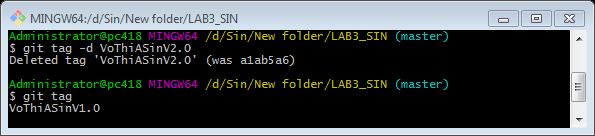


Thêm 1 Tag có tên như Tag trước nhưng là V2.0 và thêm bình luận “Truong dai hoc cong nghe tp hcm - hutech” push lên và chụp lại kết quả

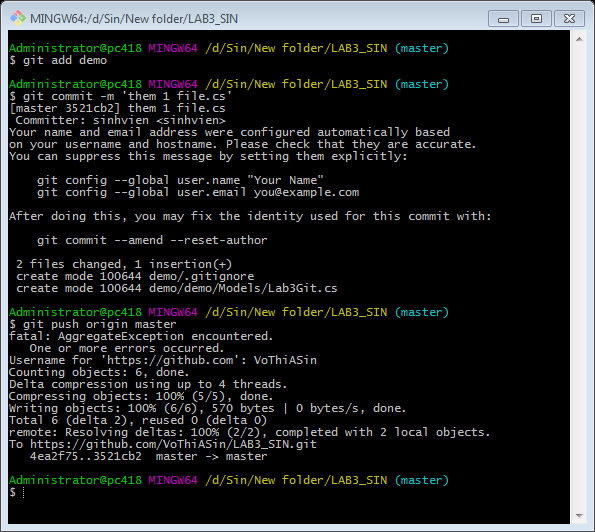




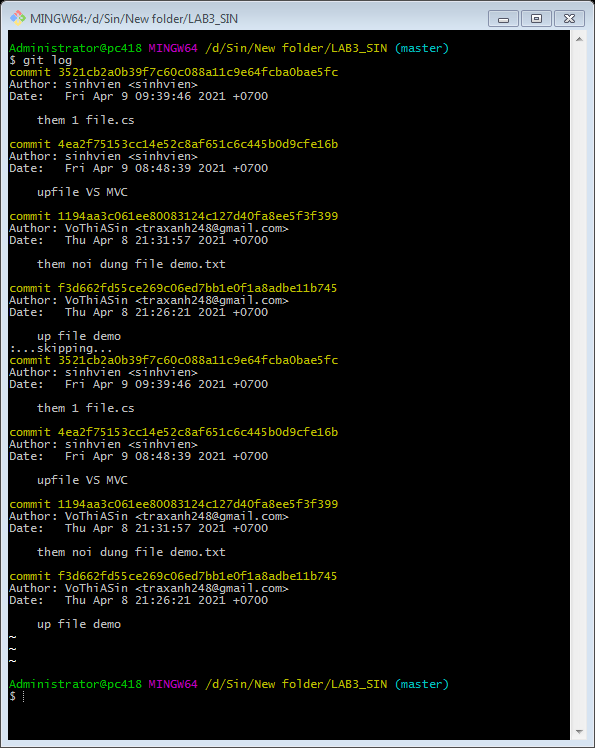
Xóa Tag thứ 2 đi và chụp lại kết quả



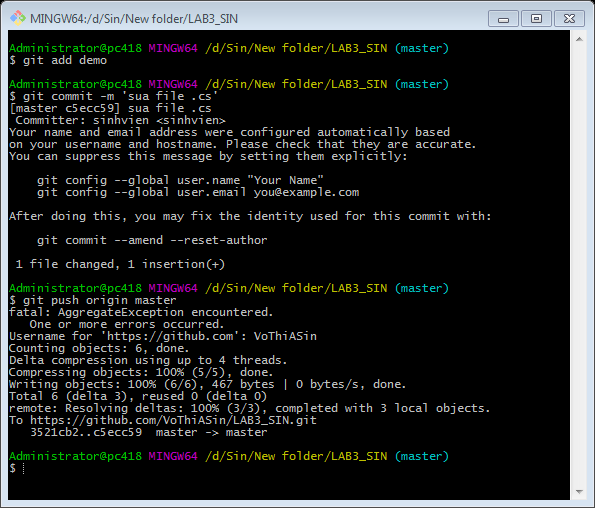
Thêm 1 class co ten Lab3Git.cs và commit với nội dung comment:”Them 1 class” push lên master và chụp lại kết quả



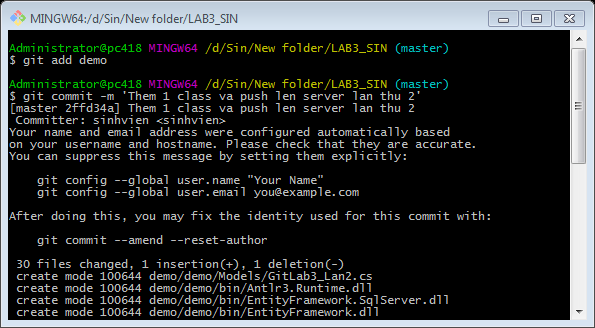
Log lịch sử của class trên

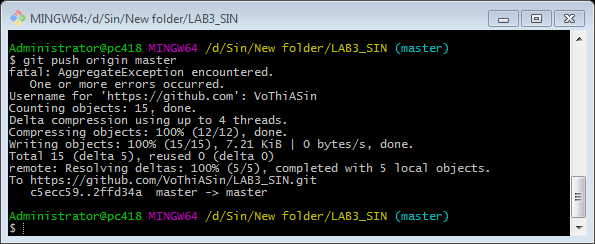


Sửa file LabGit3.cs và sửa commit cũ thành “Them 1 class va push len server” và chụp lại kết quả

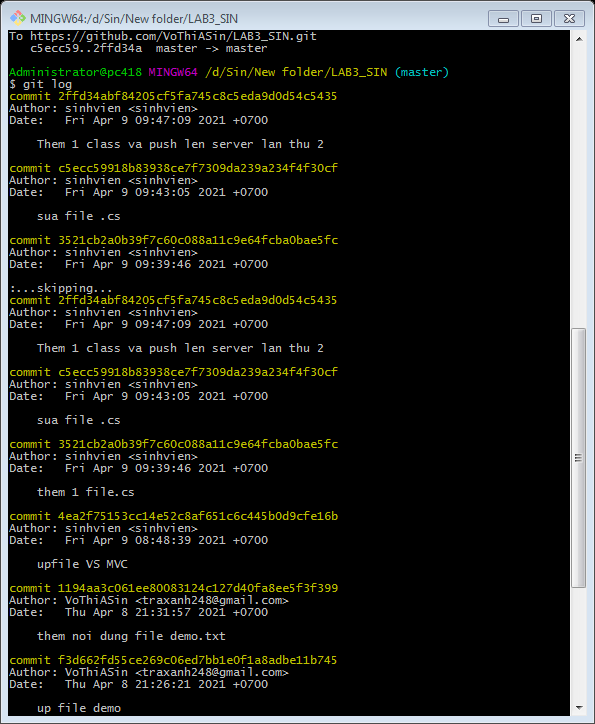


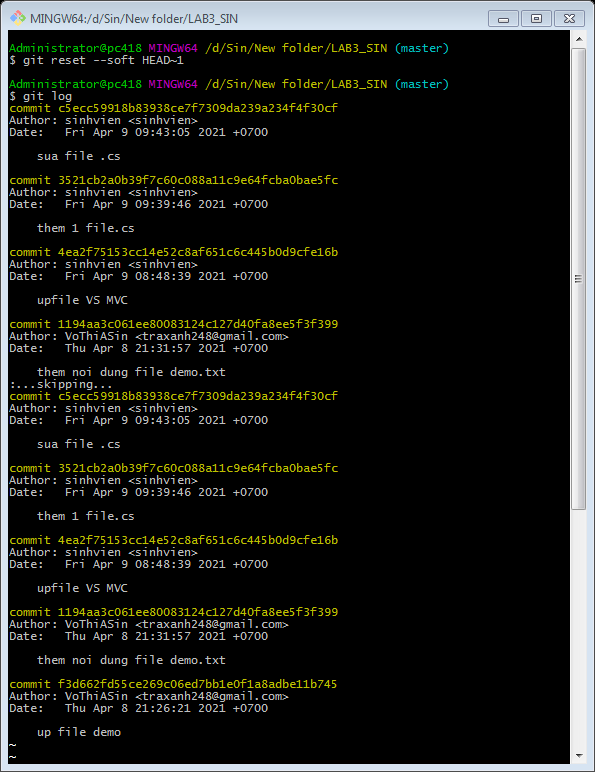
Thêm 1 commit “Them 1 class va push len server lan thu 2”



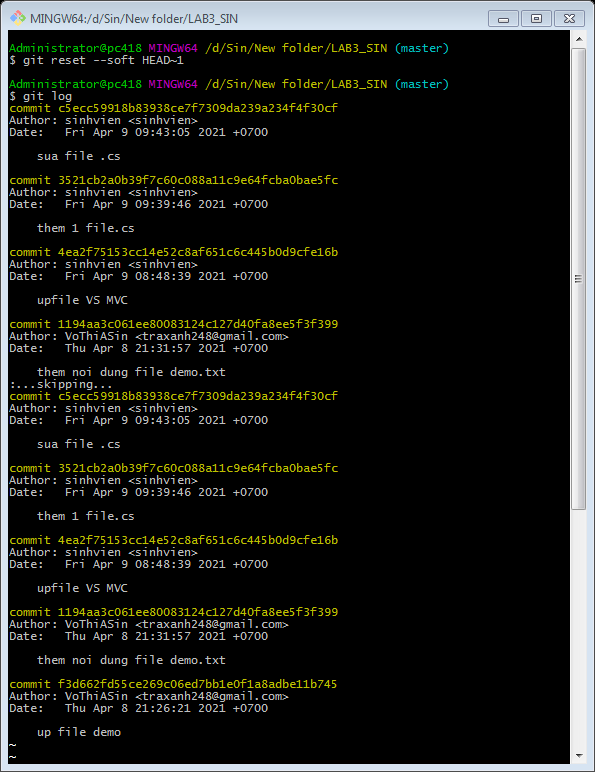


Hủy bỏ commit “Them 1 class va push len server lan thu 2” và chụp lại kết quả

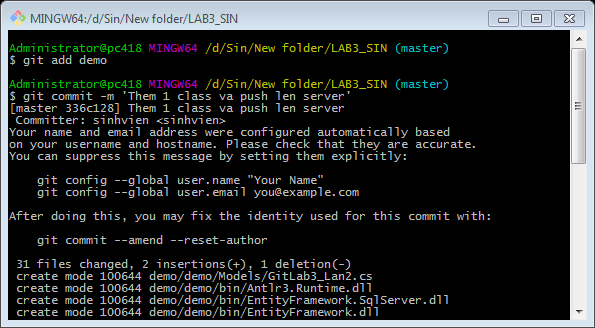


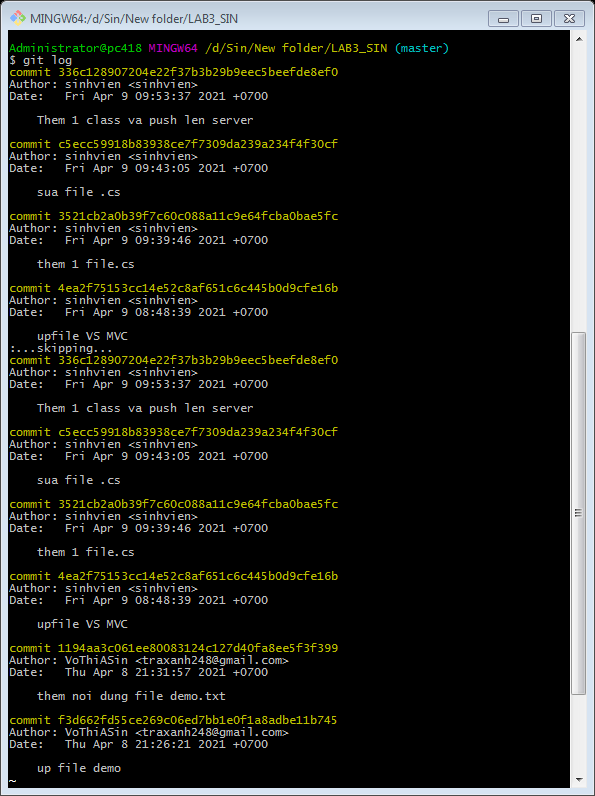


Tổng hợp commit từ của project



Tiếp tục sửa lại file “LabGit3.cs” và chỉnh sửa commit “Them 1 class va push len server” sau đó log và chụp lại kết quả.





Giải thích các lệnh đã sử dụng.

* $ git config --global alias.[chữ viết tắt] [lệnh]: Git không tự động suy luận được câu lệnh của bạn mỗi khi bạn gõ nhầm hay chỉ gõ một phần của câu lệnh. Nếu bạn không muốn gõ toàn bộ câu lệnh của Git. Bạn hoàn toàn có thể thiết lập Git Alias bằng cách sử dụng git config. Ví dụ

$ git config --global alias.ci commit

có nghĩa, thay vì phải gõ git commit, bạn chỉ cần gõ git ci là có thể add 1 commit message lên feature branch mà bạn đang làm việc .

* $ git config --get-regexp alias: dùng để xem tất các alias bạn đã tạo
* $ git config --global -- unset alias.[chữ viết tắt]: Xóa một alias
* $ git config --global --edit: sửa các alias
* $ git add [ten file]: Lệnh git add sử dụng để đánh chỉ mục (index) các nội dung mới, mới cập nhật trong thư mục làm việc, nó chuẩn bị nội dung sắp xếp cho lần commit tiếp theo. Khái niệm đánh chỉ mục ở trên có nghĩa là lưu lại ảnh chụp (snapshot) thông tin thay đổi của thư mục làm việc so với lần commit trước (hoặc so với snapshot chưa commit), snapshot lưu ở khu vực gọi là staging (sắp xếp, chuẩn bị). Bạn có thể thực hiện lệnh git add nhiều lần để tạo tạo ra một snapshot cuối cùng trước khi thực hiện commit.
* $ git commit –m ‘nội dung’: Để lưu lại những gì đã làm vào branch thì bạn sử dụng lệnh commit với cú pháp này. Trong đó nội dung là tin nhắn bạn muốn lưu cho lần commit đó.
* $ tough .gitignore: Tệp văn bản có tên .gitignore ngăn chặn việc tạo phiên bản ngẫu nhiên của
* các tệp và đường dẫn phù hợp với các paterns được chỉ định
* $ git push origin master: đẩy file từ local lên nhánh master của server.
* $ git log: Lệnh git log giúp bạn xem lại thông tin lịch sự commit, nhằm giám sát sự thay đổi của dự án. Lệnh git log có nhiều tham số để xuất ra, định dạng các thông tin hiện thị theo cách mong muốn. Bạn có thể định dạng cách các thông tin mỗi commit được in ra khi xem, cũng như có thể lọc thông tin nào đó. Mặc đinh thi hành git log nó liệt kê các commit theo thứ tự từ mới nhất đến cũ nhất, mỗi commit có các thông tin gồm: mã hash của commit, dòng thông báo, người tạo commit và ngày tạo commit.
* $ git tag [Tên tag]: Tạo một tag mới
* $ git push --tags: đẩy tất cả các tag lên server.
* $ git log --decorate: xem các nhánh nào đang trỏ tới từng commit nào.
* $ git tag [tên tag] –m ‘nội dung’: tạo một tag mới với nội dung commit cùng là ‘nội dung’.
* $ git tag: hiển thị danh sách tất cả các tag đã được tạo
* $ git tag –d [tên tag]: xóa một tag với [tên tag] là tên tag cần xóa
* $ git reset --soft HEAD~1: hủy commit cuối, con trỏ HEAD sẽ chuyển về commit cha. Đồng thời những thay đổi của commit cuối được chuyển vào vùng staging nhằm để có cơ hội commit lại hoặc sửa đổi.