

Sử dụng Git và GitHub với RStudio: : CHEATSHEET



Quản lý phiên bản, còn được gọi là **quản lý nguồn**, là việc theo dõi và quản lý các thay đổi trong code của phần mềm.

Hệ thống quản lý phiên bản là các công cụ phần mềm giúp các nhóm phát triển phần mềm quản lý các thay đổi trong mã nguồn theo thời gian.

Git là một phần mềm **mã nguồn mở** để quản lý phiên bản, được phát triển năm 2005 bởi Linus Torvalds, người tạo ra hạt nhân (kernel) hệ điều hành Linux.

Git là một công cụ quản lý phiên bản để theo dõi các thay đổi trong mã nguồn của một dự án.

GitHub là dịch vụ lưu trữ phổ biến nhất trong việc quản lý mã nguồn bằng Git.

Yêu cầu

1. Đã cài đặt R và RStudio
2. Đã cài đặt Git
3. Đăng kí tài khoản GitHub (miễn phí)



Kiểm tra Git đã được cài đặt

Trong Terminal của RStudio, nhập `which git` để yêu cầu đường dẫn đến Git của bạn:

```
which git
## /usr/bin/git
```

và `git --version` để xem phiên bản của Git:

```
git --version
## git version 2.34.1
```

« Giới thiệu » bạn với Git

Mở cửa sổ dòng lệnh (shell) từ RStudio bằng cách vào *Tools > Shell* và nhập từng dòng một, thay thế bằng tên của bạn và địa chỉ email liên kết với tài khoản GitHub của bạn:

```
git config --global user.name 'Jane Doe'
git config --global user.email 'jane@example.com'
```

Thuật ngữ GitHub

[Bảng thuật ngữ](#) này giới thiệu những khái niệm Git và GitHub phổ biến.

Cơ bản

git init <thư mục>	Tạo một Git repository trống trong thư mục được chỉ định.
git clone <repository>	Sao chép một repository tại <repository> trên máy của bạn.
git config user.name <tên-người-dùng>	Xác định tên người dùng sẽ được sử dụng cho tất cả các commit trong repository hiện tại.
git add <thư mục>	Đưa tất cả thay đổi trong <thư mục> vào trạng thái chuẩn bị cho commit tiếp theo.
git commit -m <"thông-diệp">	Commit snapshot đã chuẩn bị, nhưng thay vì mở một trình soạn thảo văn bản, sử dụng <"thông-diệp"> làm thông điệp commit.
git status	Liệt kê các tệp tin đã được chuẩn bị, chưa được chuẩn bị và chưa được theo dõi.
git log	Hiển thị toàn bộ lịch sử commit sử dụng định dạng mặc định.
git diff	Hiển thị các thay đổi chưa được chuẩn bị giữa vùng chứa (index) và thư mục làm việc (working directory) của bạn.

Repository từ xa

git remote add <tên> <url>	Tạo kết nối mới đến một repository từ xa. Sau khi thêm, bạn có thể sử dụng <tên> như một phím tắt cho <url> trong các lệnh khác.
git fetch <remote> <nhánh>	Lấy dữ liệu từ một <nhánh> cụ thể, từ repository. Bỏ trống <nhánh> để lấy tất cả các tham chiếu từ xa.
git pull <remote>	Lấy bản sao của nhánh hiện tại từ repository từ xa đã được chỉ định và ngay lập tức gộp nó vào bản sao trên máy của bạn.
git push <remote> <nhánh>	Đẩy <nhánh> lên <remote>, cùng với các commit và đối tượng cần thiết. Tạo nhánh có tên <nhánh> trong repository từ xa nếu nhánh chưa tồn tại.

Hoàn tác các thay đổi

git revert <commit>	Tạo một commit mới để hoàn tác tất cả các thay đổi đã được thực hiện trong <commit>, sau đó áp dụng cho nhánh hiện tại.
git reset <tệp>	Gỡ bỏ <tệp> khỏi vùng chuẩn bị (staging area) nhưng không thay đổi thư mục làm việc (working directory). Điều này gỡ bỏ tệp ra khỏi trạng thái chuẩn bị mà không ghi đè lên bất kỳ thay đổi nào.
git clean -n	Hiển thị danh sách các tệp sẽ được gỡ bỏ khỏi thư mục làm việc. Sử dụng -f thay cho -n để thực hiện việc gỡ bỏ tệp.

Viết lại lịch sử Git

git commit --amend	Thay thế commit cuối cùng bằng các thay đổi đã chuẩn bị (staged) kết hợp với commit trước đó. Khi không có thay đổi, sử dụng để chỉnh sửa thông điệp của commit cuối cùng.
git rebase <base>	Rebase nhánh hiện tại lên <base>. <base> có thể là ID của commit, tên nhánh, một thẻ (tag) hoặc một tham chiếu tương đối đến HEAD.
git reflog	Hiển thị nhật ký (log) các thay đổi đối với HEAD của repository cục bộ. Thêm --relative-date để hiển thị thông tin về ngày thay đổi hoặc --all để hiển thị tất cả các thông tin tham chiếu.

Các nhánh Git

git branch	Liệt kê tất cả các nhánh trong repository. Dùng git branch <tên-nhánh-mới> để tạo một nhánh mới.
git checkout -b <nhánh>	Tạo và chuyển đến một nhánh mới với tên <nhánh>. Bỏ -b để chuyển đến một nhánh đã tồn tại.
git merge <nhánh>	Gộp <nhánh> vào nhánh hiện tại.