Merge pull request /hợp nhất yêu cầu kéo

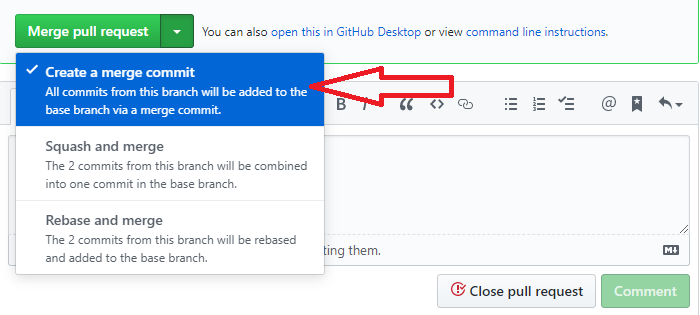
Create a merge commit / Tạo một cam kết hợp nhất từ nhánh này sẽ được thêm vào nhanh cơ sở thông qua 1 cam kết hợp nhất

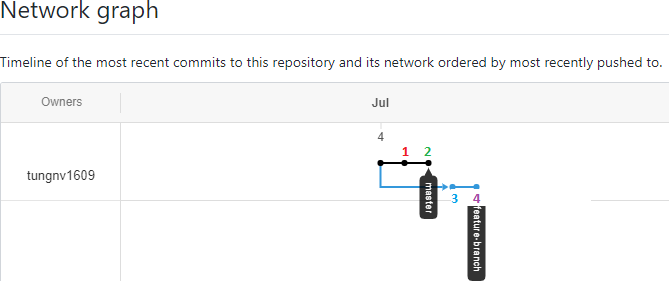
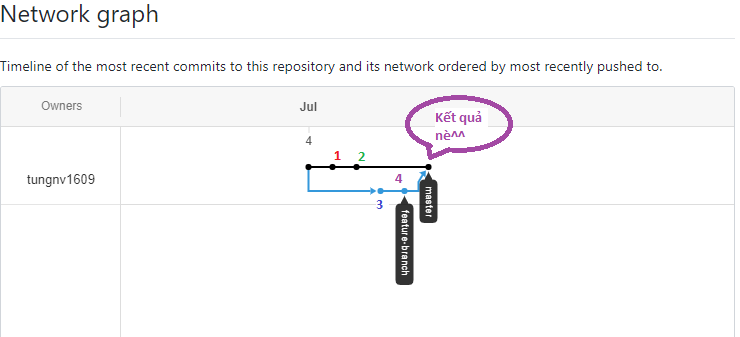
Squash and merge / 3 lần xác nhận từ nhánh này sẽ được kết hợp thành một lần xác nhận trong nhánh cơ sở

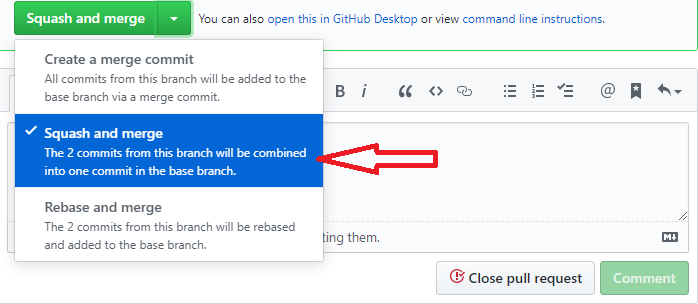
Rebase and merge / 3 lần xác nhận từ nhanh này sẽ bị huỷ bỏ và được thêm vào nhanh cơ sở

Trên Github, việc "merge pull request" được sử dụng để tích hợp các thay đổi từ một nhánh (branch) của dự án vào nhánh chính (thường là nhánh **master**). Khi một pull request được tạo, nó đại diện cho sự đề xuất thay đổi hoặc bổ sung vào dự án. Khi merge pull request, sự khác nhau phụ thuộc vào các tùy chọn và trạng thái của pull request cụ thể. Dưới đây là một số sự khác nhau quan trọng trong việc merge pull request trên Github:

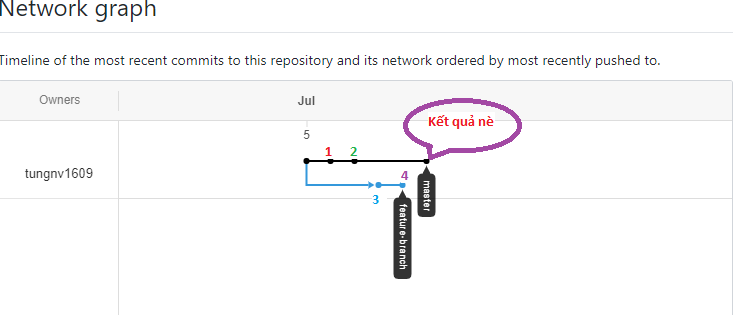
1. *Merge Commit: Khi bạn merge pull request, Github cung cấp tùy chọn để tạo một commit mới (merge commit) trên nhánh chính. Merge commit chứa thông tin về việc merge và các thay đổi từ nhánh khác. Điều này giúp theo dõi lịch sử và quản lý các thay đổi được tích hợp.*
2. *Rebase and Merge: Một tùy chọn khác là "Rebase and Merge". Thay vì tạo merge commit, tùy chọn này sẽ áp dụng các commit từ pull request trực tiếp vào nhánh chính mà không tạo ra commit mới. Quá trình này sẽ tạo ra một lịch sử commit thẳng và sạch hơn, nhưng cũng có thể thay đổi lịch sử commit ban đầu.*
3. *Squash and Merge: Một tùy chọn khác là "Squash and Merge". Khi sử dụng tùy chọn này, tất cả các commit từ pull request sẽ được gộp thành một commit duy nhất trước khi được áp dụng vào nhánh chính. Điều này giúp giữ lịch sử commit ngắn gọn và dễ đọc***.**
4. Conflict Resolution: Trong một số trường hợp, khi có xung đột giữa các thay đổi trong pull request và nhánh chính, Github sẽ hiển thị một xung đột và yêu cầu bạn giải quyết. Bạn có thể xem và chỉnh sửa các tệp bị xung đột để hoàn thành quá trình merge. Sau khi xung đột được giải quyết, pull request sẽ được kết hợp thành công.
5. Trạng thái và quyền hạn: Merge pull request có thể chỉ được thực hiện bởi người có quyền truy cập và quyền hạn tương ứng. Người tạo pull request hoặc các người được chỉ định có thể thực hiện quá trình merge. Ngoài ra, pull request có thể có các trạng thái như "open" (mở) hoặc "closed" (đã đóng) tùy thuộc vào việc

**1.Create a Merge Commit**  
  
**Hình bên dưới là tùy chọn “Create a merge commit”:**  
  
  
*Merge all of the commits into the base branch (merge tất cả commit vào trong nhánh cơ sở)*  
  
Đây là lựa chọn mặc định khi Merge pull request trên Github.  
Để bắt đầu, chúng ta có một nhánh base master với commit ban đầu **(Initial Commit)** và 2 commit mới, kí hiệu là **commit 1,commit 2**.  
Sau đó chúng ta tạo một branch “feature-branch” , tiếp tục tạo 2 commit **commit 3 và commit 4**. Tiến hành push những thay đổi của chúng ta vào **“feature-branch”**.  
Chúng ta vào Github mục **Insights -> Network** và quan sát:

  
  
  
  
Khi sử dụng **“Merge pull request”** (giả sử không có gì xung đột xảy ra trước đó), **commits 3 và 4** sẽ được thêm vào nhánh chính như một commit, gọi là Merge branch feature-branch và trở thành một HEAD mới. Chúng ta cùng quan sát điều gì xảy ra trong **Insights -> Network**:  


**2. Squash and merge**  
  
**Hình bên dưới là tùy chọn “Squash and merge”:**  


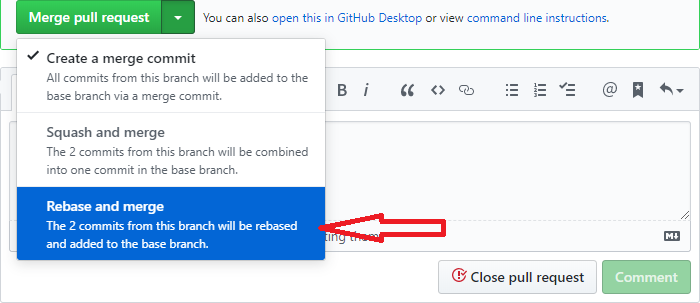
*All commits from this branch will be combined into one commit in the base branch (gộp những commit vào trong một commit duy nhất)*  
Squashing làm việc giống với rebasing một nhánh, chúng ta có thể nhận nhiều commits, pull request và squash - nén chúng vào 1 commit duy nhất. Việc này khá giống với Merge pull request, nhưng tinh tế hơn.

Thay vì thêm **commits 3 và 4** với commit **“Merge branch feature-branch”**, chúng ta gộp 3 và 4 và kết quả cho ra duy nhất commit gộp là 3\*\*(squashed)\*\*.  
  
Chúng ta cùng quan sát điều gì xảy ra trong **Insights -> Network:**  
  
  
  
Tùy chọn squash trên Github UI cho phép chúng ta thay đổi các thông điệp commit mặc định. Các commit dưới đây được “squash” dưới dạng thông điệp mặc định như sau:



***Nếu bạn muốn sử dụng nó , đừng quên edit các thông điệp !***  
  
Chúng ta sử dụng tùy chọn merge này khi 1 feature branch có rất nhiều các commit nhỏ được add. Khi đó, chúng ta có thể nén chúng vào 1 commit. Commit nén có thể chứa một thông điệp với các thông tin mô tả về commit được nén dưới nó.

**3. Rebase and Merge  
  
Hình bên dưới là tùy chọn “Rebase and merge”:**

  
  
Rebase the commits individual onto the base branch (tập hợp tất cả những commit riêng lẻ vào trong một branch cơ sở)  
Trong trường hợp chỉ cần sửa đổi một số lượng nhỏ commit cũng như một đoạn code cụ thể thì rebase and merge sẽ là lựa chọn hàng đầu. Cách làm này sẽ làm cho quá trình merge được đơn giản cũng như giữ lịch sử Git gọn gàng hơn.  
Ta cần tạo 2 commit trên nhánh **“feature-branch”** thêm **commits 1 và 2**. Rebase and Merge sẽ chuyển các commits lên đầu nhánh chính master branch - thay đổi HEAD của master branch mà không merge **“feature-branch”** như khi sử dụng **Merge pull request:**  
  
Chúng ta cùng quan sát điều gì xảy ra trong **Insights -> Network:**  
  
  
