

# GIT Command

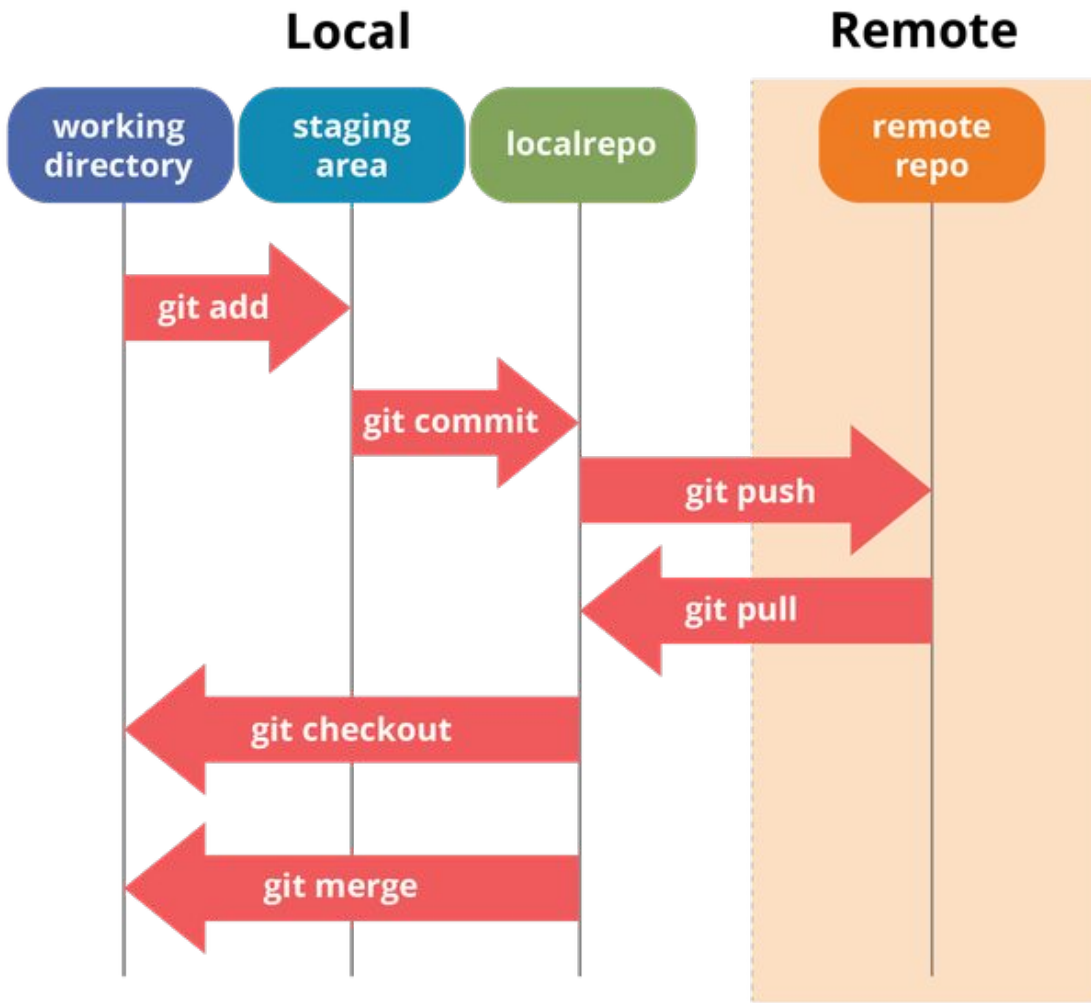
KẾT NỐI  
THẤU HIỂU  
CHIA SẺ

# Overview

1. Cấu trúc stage
2. Git flow
3. Branch (Nhánh)
4. Commit (Lịch sử thay đổi)
5. Rollback code (Reset vs Revert)
6. Merge code (Merge vs Rebase)
7. Git stash (Lưu thay đổi trong bộ nhớ tạm)
8. Git cherry-pick (Copy commit)

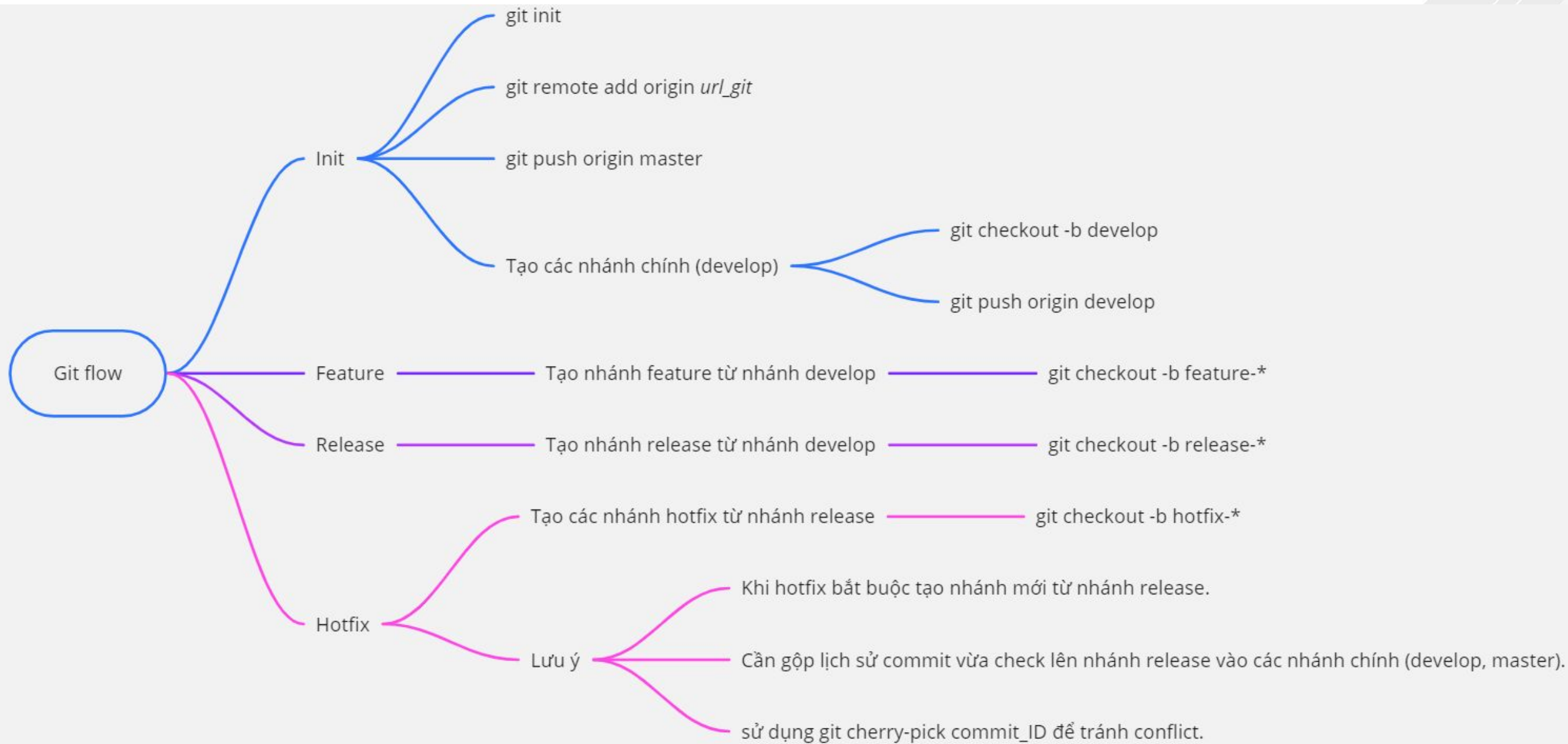


# 1. Cấu trúc Stage

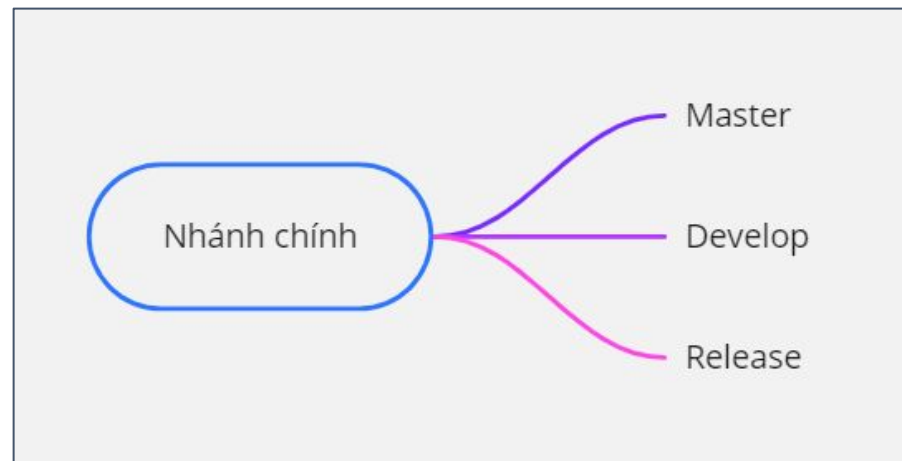


- `git status` — trạng thái project.
- `git add .` — đẩy các thay đổi lên môi trường staging area.
- `git commit -m "message"` {
  - lưu lại các lịch sử thay đổi trong repository.
  - sau khi commit, tự động sinh một mã commitID.
- `git push origin branch_name` — đẩy các thay đổi lên môi trường remote repository.
- `git pull origin branch_name` — kéo source từ remote repository về local repository.

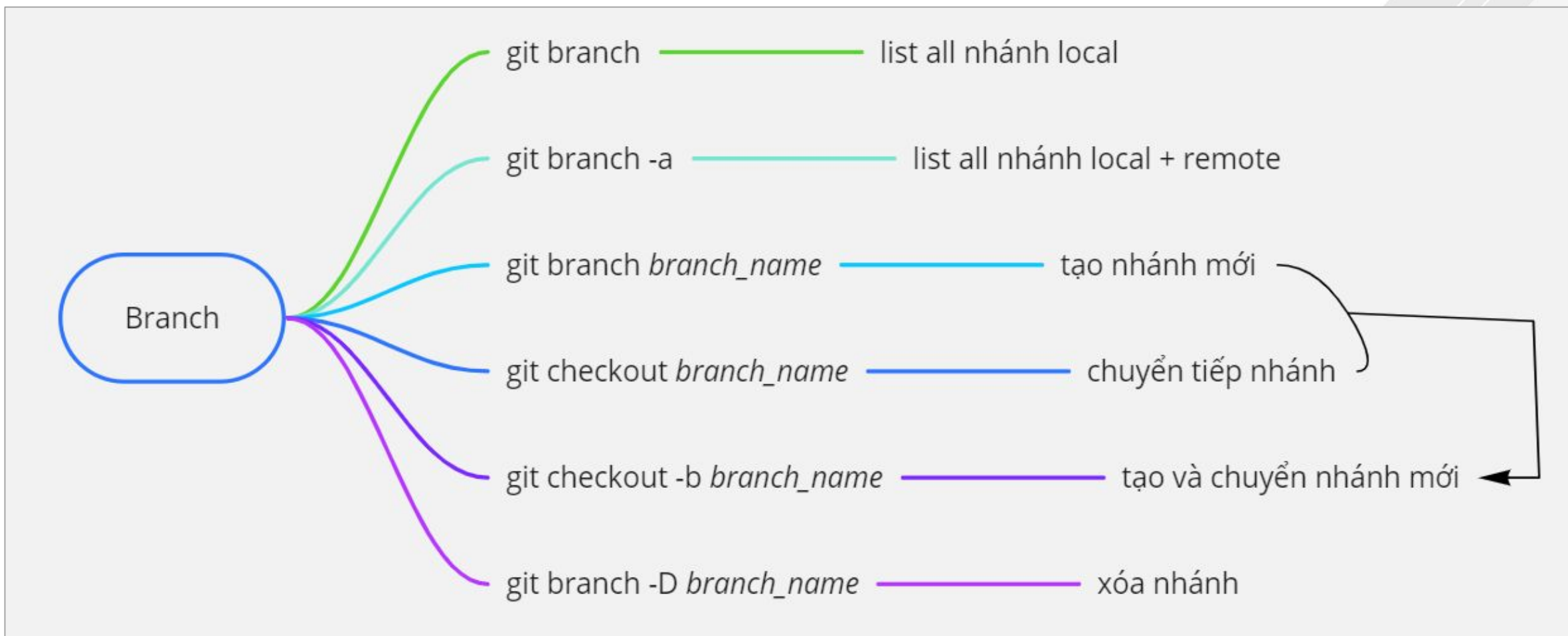
`git fetch origin` - kéo all (nhánh, source, ...) từ remote về local



### 3. Branch (Nhánh)

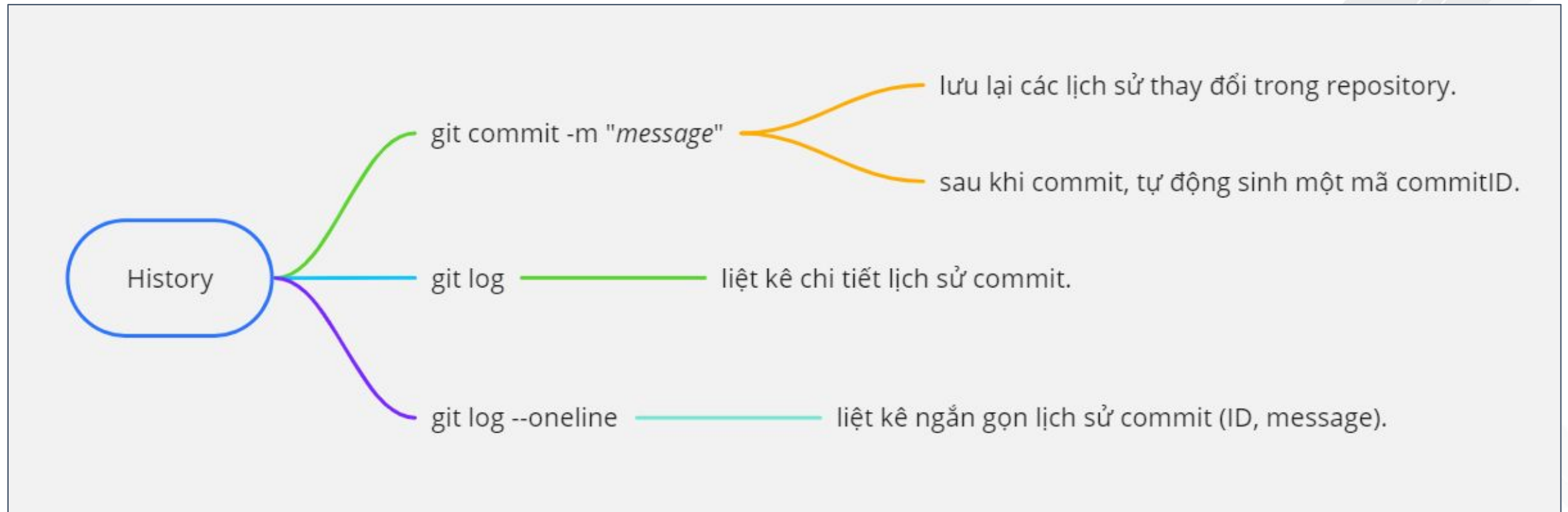


### 3. Branch (Nhánh)

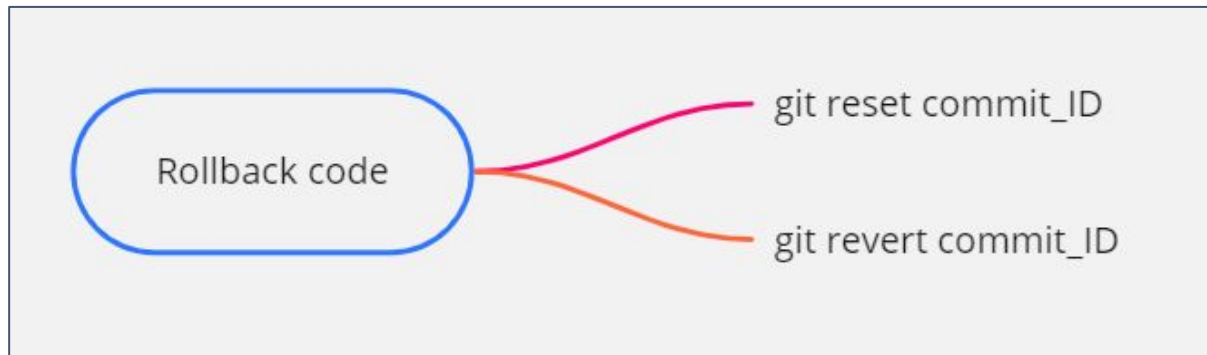




## 4. Commit (Lịch sử thay đổi)

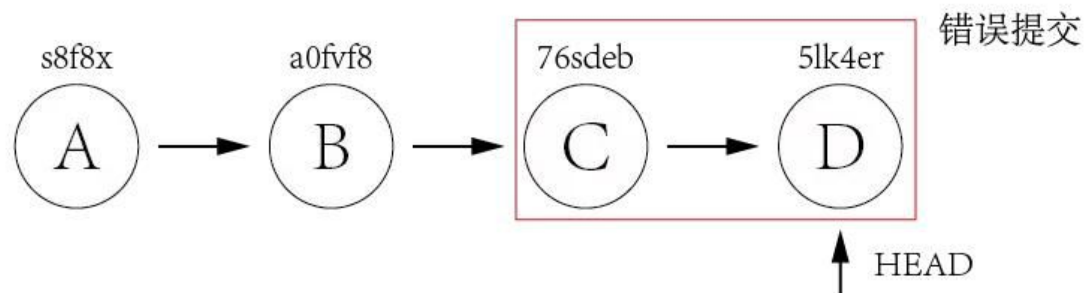


## 5. Rollback code (Reset vs Revert)



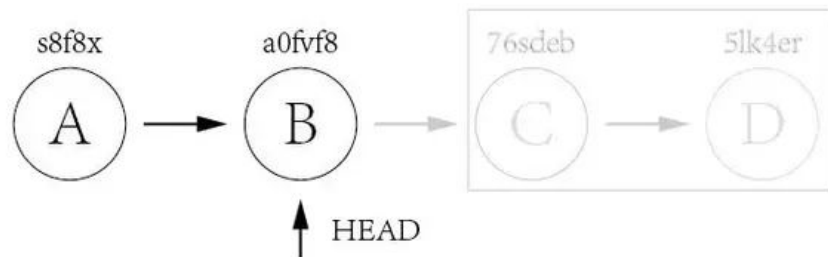


## 5. Rollback code (Reset)

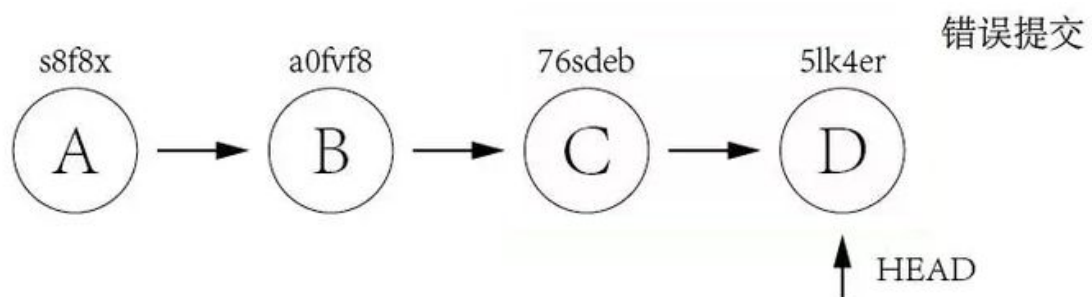


`git reset a0fvf8`

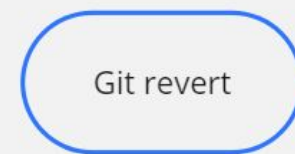
Kết quả:



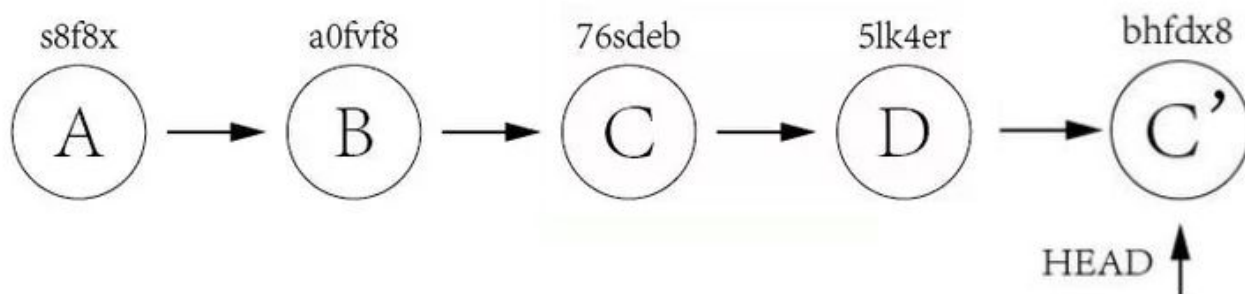
## 5. Rollback code (Revert)



git revert 76sdeb



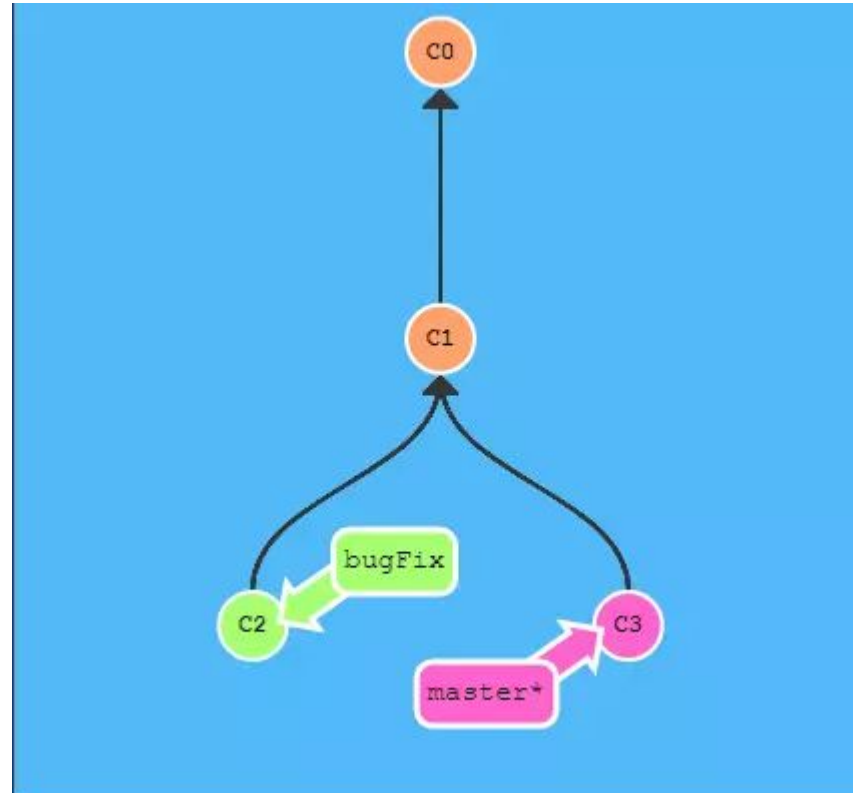
Kết quả:



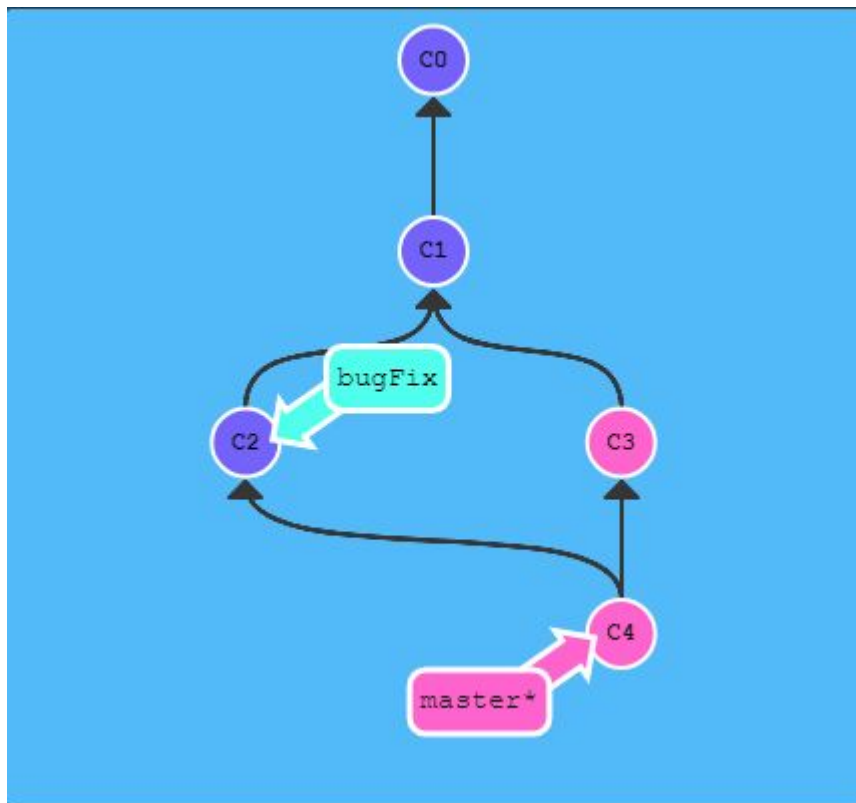
## 6. Merge code (Rebase và Merge)



## 6. Merge code (Rebase và Merge)



## 6. Merge code (Merge)



Merge: Đứng từ nhánh chính (master) để gộp code từ nhánh bugFix

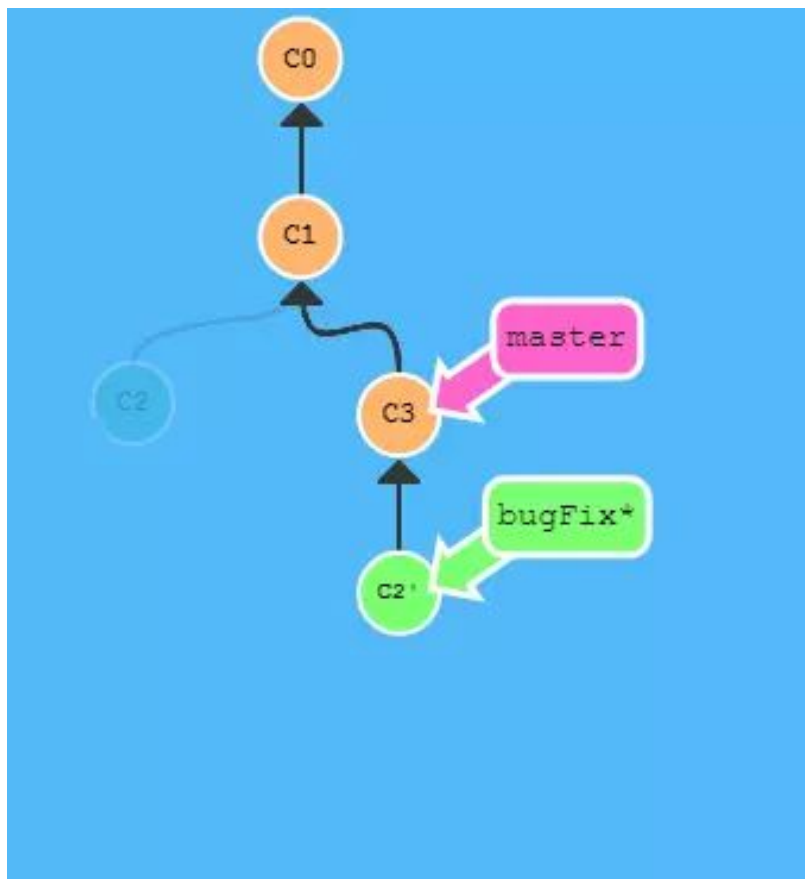
```
git checkout master  
git merge bugFix
```



Tạo 1 commit lịch sử mới

Làm những nhánh tồn tại không bị thay đổi

## 6. Merge code (Rebase)



Rebase: Đứng từ nhánh cần gộp (bugFix) để gộp code từ nhánh bugFix vào Master

```
git checkout bugFix  
git rebase master
```

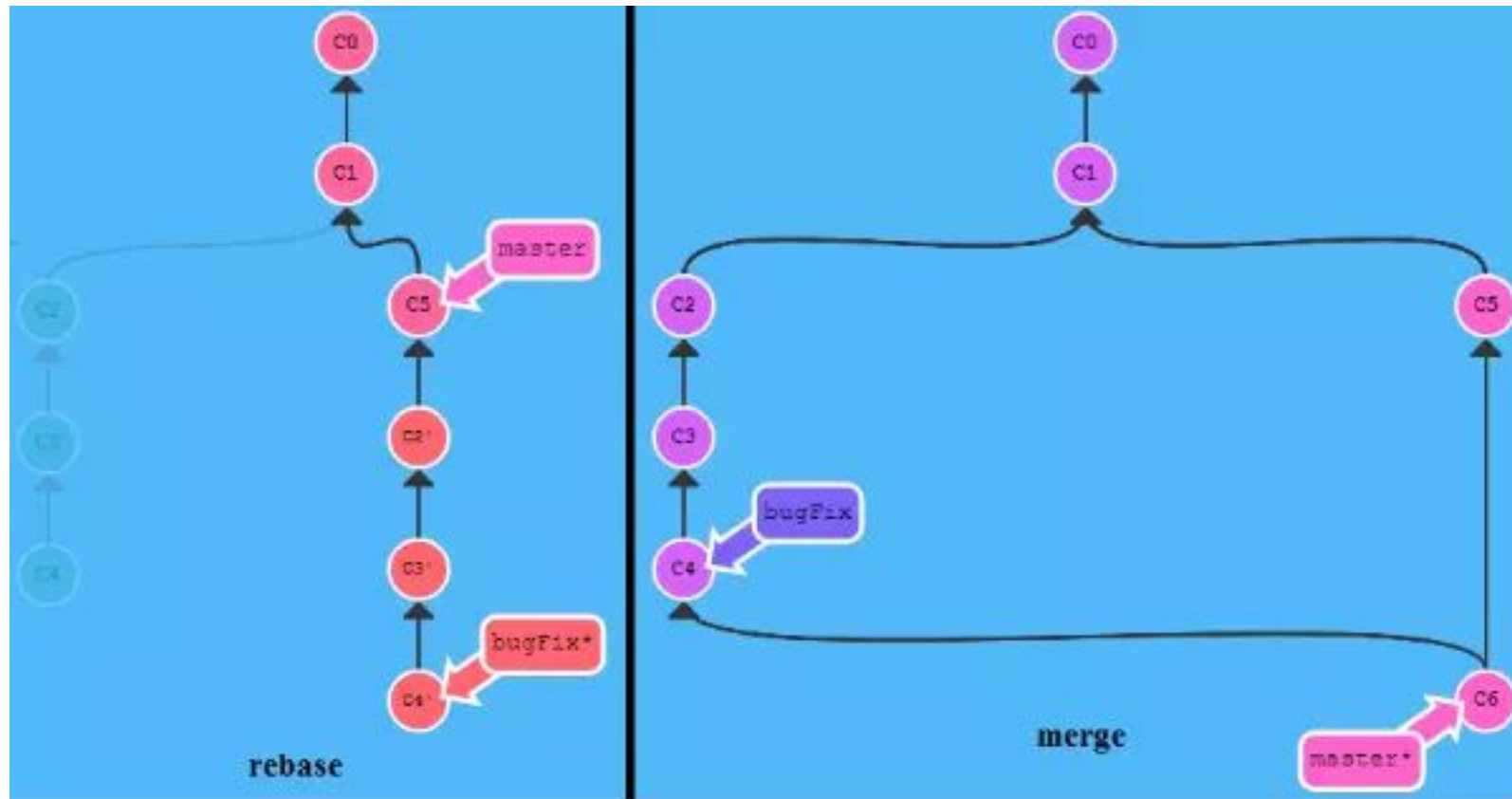
Rebase

Giúp loại bỏ lịch sử commit không cần thiết

Tạo lịch sử commit dạng tuyến tính



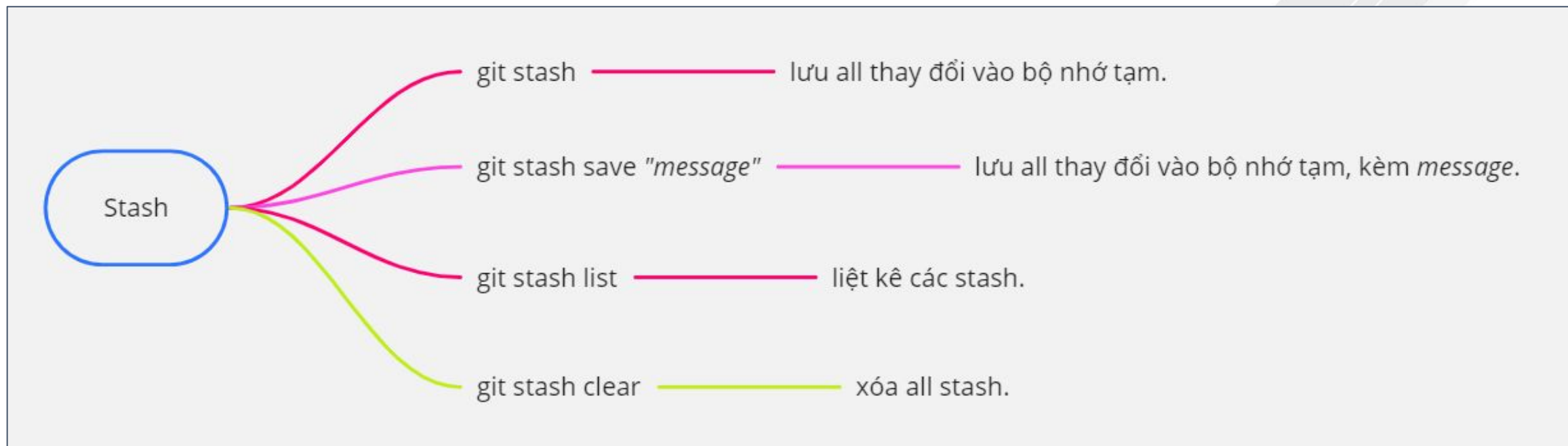
## 6. Merge code (Rebase và Merge)



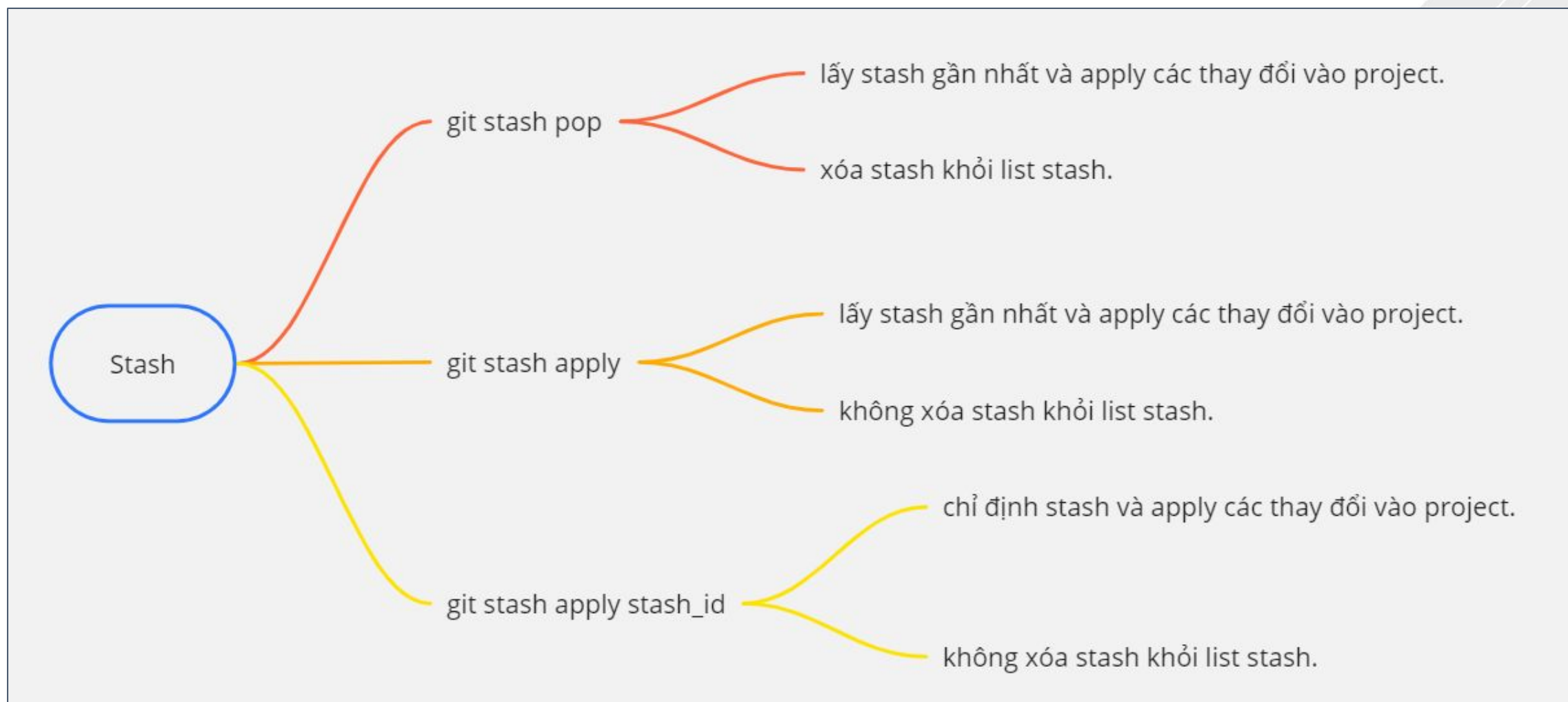
## 7. Git stash (Lưu thay đổi vào bộ nhớ tạm)

Case:

- Đang code PBI mới ở nhánh feature, cần chuyển sang nhánh release để hotfix.
- Khi chuyển sang release sẽ báo: cần commit các thay đổi trên nhánh feature thì mới chuyển sang được.



## 7. Git stash (Lấy thay đổi từ bộ nhớ tạm)

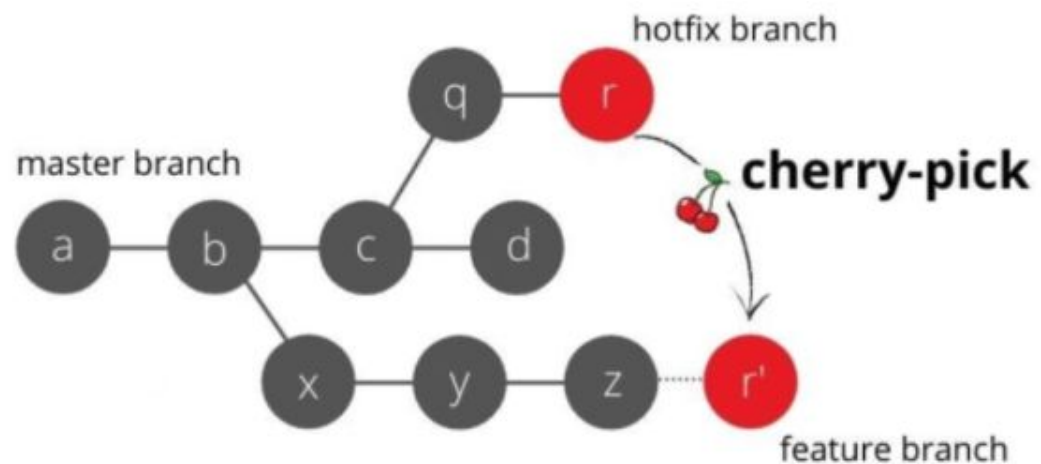


## 8. Git cherry-pick (Copy commit)

Case hotfix:

- Tạo nhánh hotfix từ nhánh release => tạo pull request.
- Rebase nhánh hotfix vào nhánh feature => tạo pull request.

Nếu sử dụng rebase, khả năng cao conflict xảy ra là rất lớn.



`git cherry-pick commitID`



# Q&A?

 **THANK YOU**  
For Your Attention