



## BLOCK NONCE #0: Git Like a Surgeon

### What?

Git isn't just for saving work - it's a time machine for code!

Every commit is a snapshot of your thinking.

Every branch is a playground for your ideas.

Every PR carries your work toward a shared creation.

### Why?

Master Git. Develop fearlessly.

### How?

- **Commit Surgery:**

Just committed a change, but realized it should be two separate commits ?

```
# Store the current commit hash in a variable.
CMT=$(git rev-parse HEAD)
# Go back one commit.
git reset --hard HEAD~1
# Go over the changes in $CMT and pick the ones you want.
# Explore the options in `git checkout -p`, such as e (edit) and s
(split).
git checkout -p $CMT
# Commit the first part
git commit
# Stage and commit the rest
git checkout -p $CMT
git commit
```



**TIP**💡 Don't just follow these steps blindly, understand what each of them does.

There are many ways to split a commit, this is just one example.

- **Recovery:**

Did you use `git commit --amend` instead of `git commit` ?

```
# reflog shows the history of the HEAD pointer
git reflog
# The original commit will likely be HEAD@{1}
git reset --soft HEAD@{1}
# Commit again, now without "--amend"
git commit
```

**Call For Action!**   Try it yourself! Check `git status` and `git log` after each command. Try running `git rev-parse HEAD` and check the output.