

# Tools in our utility belt

- Text Editor - Atom
- Shell - git-bash
- Version Control - Git

# Why is version control useful?

- **Backup** – coffee + keyboard = 🤪
  - *Undo* – Oops, was that important?
- **Collaborate** – programming is a team sport
  - *Track steps* – What happened while you were sleeping
  - *Branching and merging* – Work in parallel and bring it back together



**A Git repository or “repo” is basically a folder that keeps track of changes to its contents.**

*Imperfect analogy: if Git is a camera, the repo is what's in frame*

# **A little on how Git works**

## **The Git index**

Git keeps a local copy of everything you add to it, even if you delete it later.

*Imperfect analogy: if Git is a camera, the index is a roll of film*

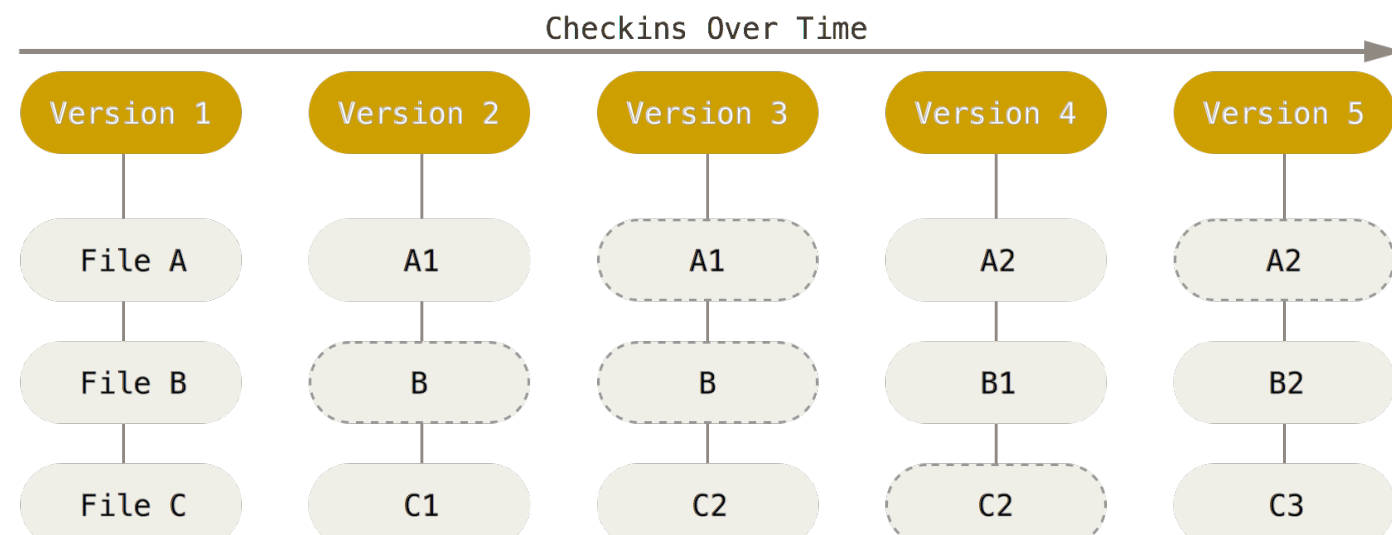
Say it with me.

As a general rule, **don't add sensitive data to Github.**

# Commits: snapshots of your files

When you change things in files, or add or remove files, you commit these changes and Git records the current state of everything

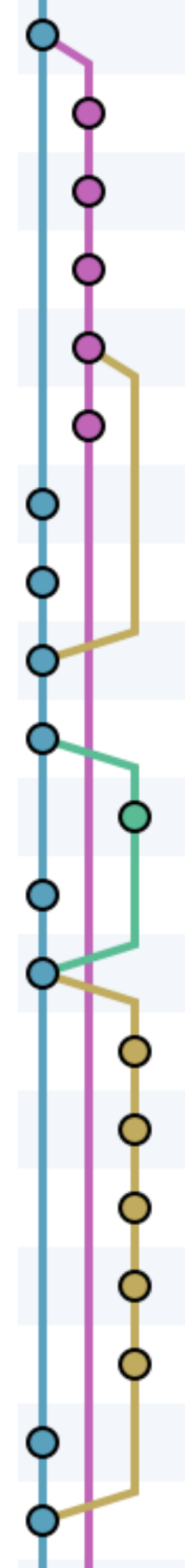
*Imperfect analogy: if Git is a camera, commits are pictures*



# Version history is (kind of) like a family tree

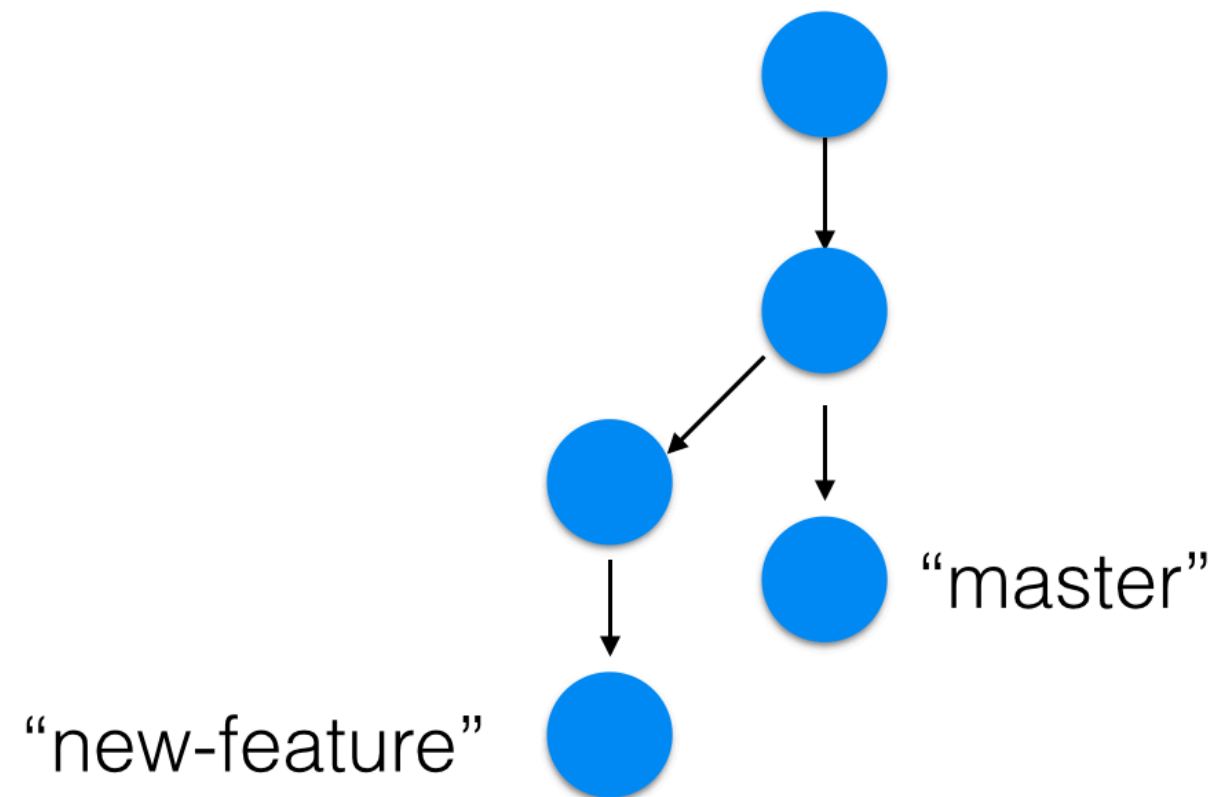
Every commit (except the first one) has a parent (AKA the state of the repo when you started making changes).

If you and someone else start working in parallel from the same starting point, git will try to merge your changes back together later\*



# Branches let you keep things separate

If you want to keep your work separate for a while (say, if you're working on a new feature or something), you can make a *named branch* which will stay separate from the *master branch*. You can merge it back later.





**Remotes are copies of a  
repo on another computer**  
(or on a service like Github).

# **GitHub is a central place to host a repo**

GitHub is a place to store your repo, so others can clone it, push to it, and pull from it.

**CapitalOne** Land™

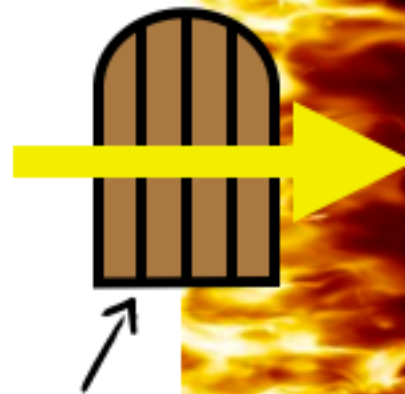
**GitHub** Enterprise

CapitalOne Github  
([github.kdc.capitalone.com](https://github.kdc.capitalone.com))

you



The Proxy



The Great  
CapitalOne  
Firewall



The internet



Public Github  
([github.com](https://github.com))

plebians/hackers



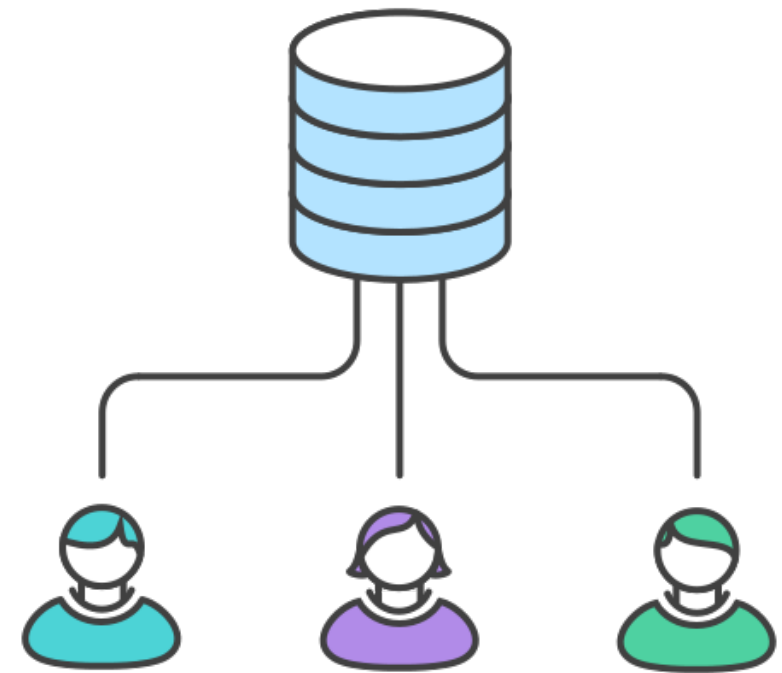
# **Collaborating using Git**

# Centralized workflow

Typically used on small teams

Pros:

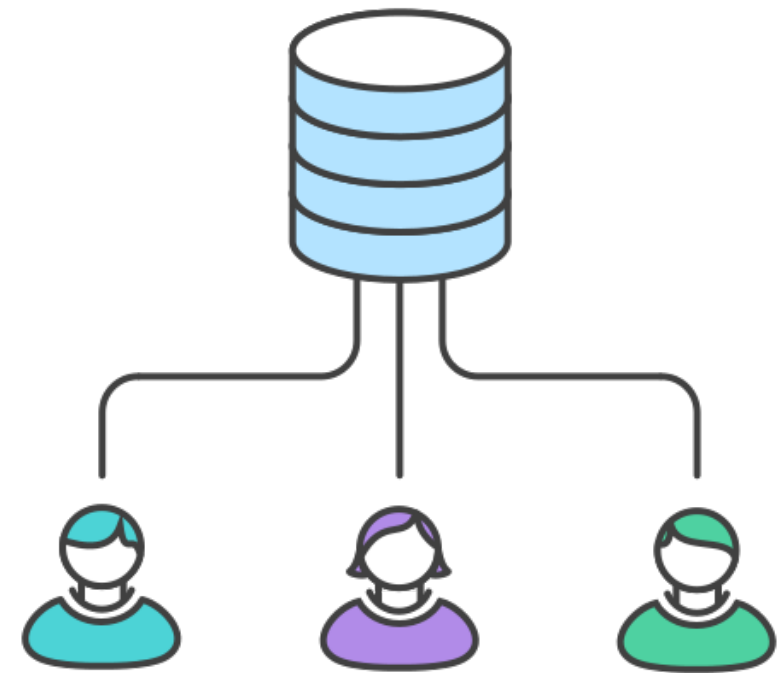
- everyone can do everything, no bottlenecks
- efforts can be organized + coordinated using git branches
- Github not technically required



# Centralized workflow

Cons:

- people can make decisions/ruin code arbitrarily

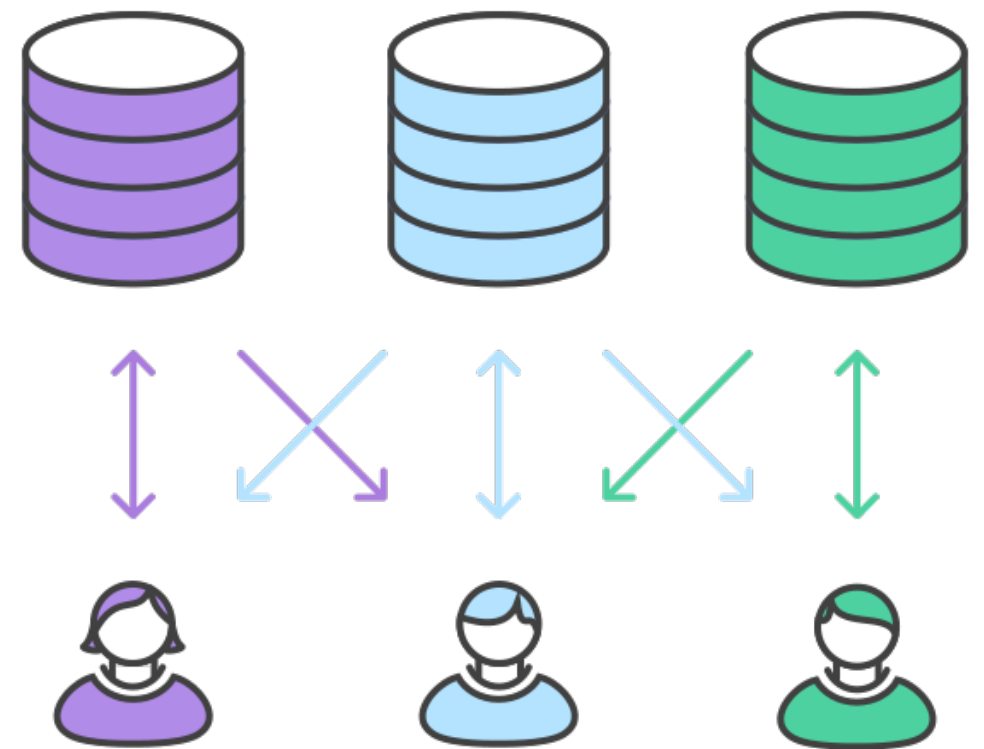


# Fork + Pull request workflow

Typically used on larger open-source projects

Pros:

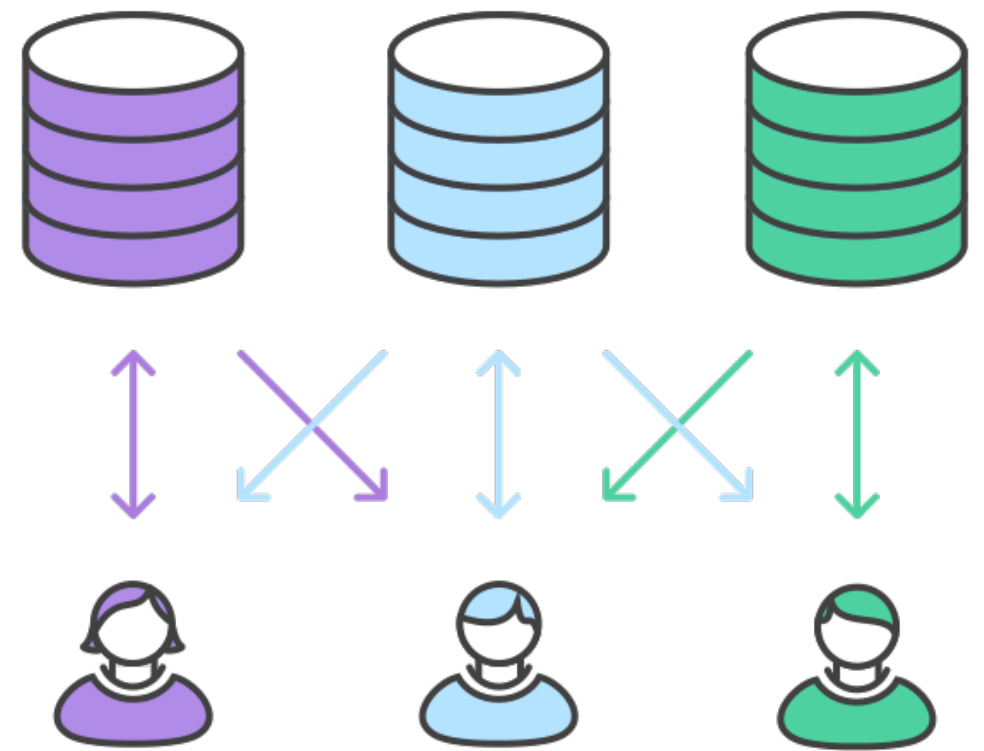
- Pull requests serve as places to review code for quality and discuss architecture
- Forks give people freedom to change the code in a way that suits them



# Fork + Pull request workflow

Cons:

- Changes must be approved by one of the *anointed ones*
- Harder to do without something like Github







**Let's jump in!**