



Inspiring Brilliance

Intro to Git

What is Git?



It's a Version Control System(VCS) for code

What does a VCS do?



Record Changes made

What was the change? Who made it?



Record Reason for change

When and why was the change made?



Recover from mistakes

How do we get back to a working state?

Why do we need a VCS?

- All contributors see a unified and consistent view of a project
- Everyone sees who's working on what
- Team members can work together, separately
 - See a transparent history of changes
 - Who made them
 - How they contribute to the development of a project

Git Concepts



Commit

A snapshot of all the files in the project at a point in time



Branch

A collection of commits



Merge

Combine two branches together



Hands-On

<https://github.com/octocat/Spoon-Knife>

Hands-On Flow

- Fork
- Clone
- Checkout new branch
- Git status
- Stage changes
- Commit
- Push
- Make PR



One more thing...

One more thing

- Every repo has a different set of rules to write code
- Go through README and CONTRIBUTING.md files
- Some examples
 - One commit per PR
 - Format of commit messages
 - License agreements to be accepted
 - Sign off on commits
 - Verified (Crypto-signed) commits

Thanks!

