Git II
Goin' Old School

Lecture this week will use the whiteboard during class.
These slides capture the lecture notes / plan.
We will also post some supplemental material on the course homepage
Repo Sandbox: In the Beginning...

Activity

1. Create empty repo, add commits 1, 2, 3 and record hashes
2. Check out .git/HEAD
   ○ What is "HEAD"?
Repo Sandbox: A New Branch

1 <- 2 <- 3  master
\                                
  <- 3 <- 4  no_two

Activity

1. git checkout <commit 1 hash>
   ✅ What is 'detached HEAD' state?
2. git branch no_two
3. git checkout no_two, add commits 3 & 4 and record hashes
   ✅ HEAD is reattached
4. Explore .git/refs/heads/...
   ✅ What is a branch?
Repo Sandbox: The First Merge

Activity

1. A new alias!
2. `git checkout master`
   - look around
3. `git merge no_two`
   - 'merge' means 'merge into'
   - look around
Repo Sandbox: Fast Forward

Activity

1. `git checkout -b fast_five`, add commit 5, record hash
   - look around
2. `git checkout master`
3. `git merge fast_five`
   - What does 'fast forward' mean?
   - look around
Repo Sandbox: More Branches

Activity

1. `git checkout -b add_six`
   - look around
2. `git branch add_seven master`
   - look around
3. Add commit 6, record hash
   - look around
4. `git checkout add_seven`, add commit 7, record hash
   - look around
Repo Sandbox: Merge en Trois

```
<- 6 <- add_six
    /     \
1 <- 2 <- 3 <- M1 <-- 5 <-- ---- <-- M2 master
    \       /       \
       /         \       /
      /           \     /  \
     /             \   /    \
    <- 3 <- 4 <- no_two <- 7 <- add_seven
```

Activity

1. `git branch -d fast_five`
2. `git checkout master`
3. `git merge add_six add_seven`
   - Can merge $n$ branches at once (octopus!)
   - Creates a 'merge commit', why?
Remotes

1. Show how the graph varies based on machine
   ○ Sync'ing is all about syncing graph objects
2. Open GitLab, explain what it is
3. Push demo to GitLab
4. Pull down a clone
5. Make changes
6. Push up
7. Fetch, then merge
8. Repeat with pull
9. Pushing, pulling, and tracking branches
Rewriting History: Squashing, Rebasing


2. Create a feature branch, several commits, squash