CS3110 Level Up

Git Advanced

Rachit Nigam (rn359) & Timmy Zhu (tz277)

1 Some review

What we did last week

- Learn the basic workflow of Git
- Learn how to push your code
- Learn how to pull others' code changes
- Learn to reverse mistakes
- Learn how to view the status of your git repository
- Learn how to view the history of your git logs

What we will do today

- Learn how to fix merge conflicts (!!!)
- Learn how to make independent pieces of development (a.k.a branches)
- Learn more ways to reverse mistakes
- Learn how to make pull requests

2

Fixing Conflicts

push rejections
merge conflicts

Demo: Merge Conflicts

Merge conflicts

```
<<<<<<< HEAD
code1
=======
code2
>>>>>>> 8b30ed75d5e00ba70d34766c7588e01c5acbeb5f
```

Merge conflicts

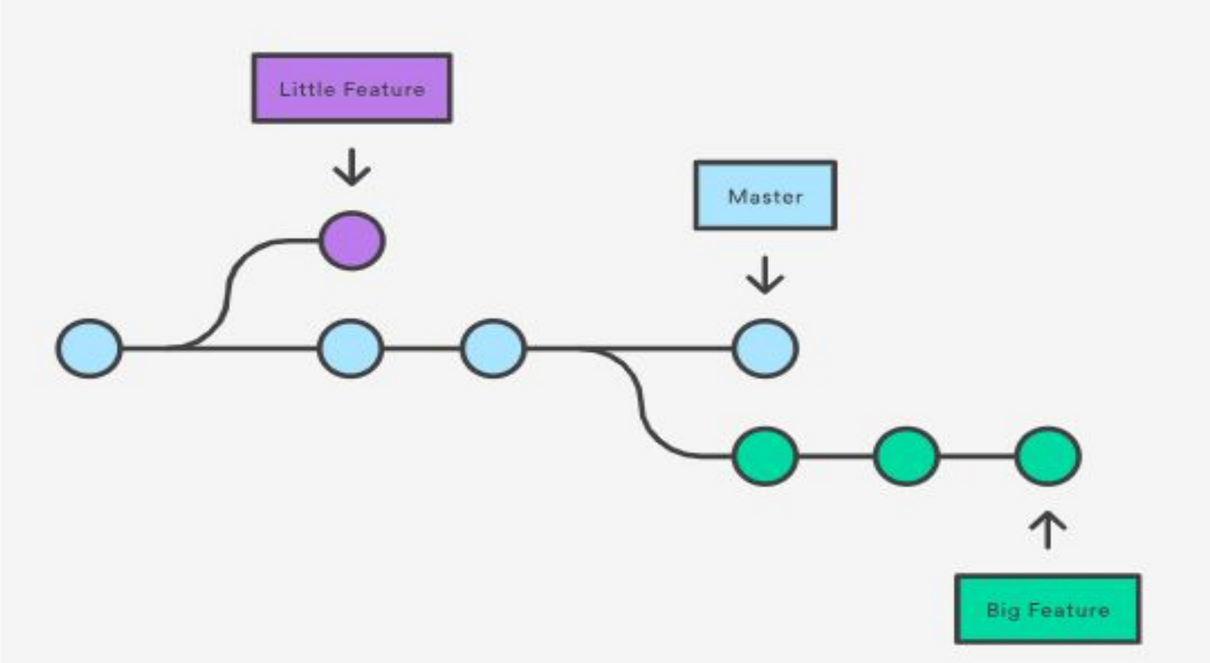
- Simply keep the code you want
- Re-push

3

Branches

```
git branch
git branch <NEWBRANCH>
git checkout <NEWBRANCH>
```

Branching



What is a branch?

- An independent line of development
- A way to have a new staging area,

Branch commands

git branch

- list all branches

git branch <branchName>

- create a new branch

git branch -d <branchName>

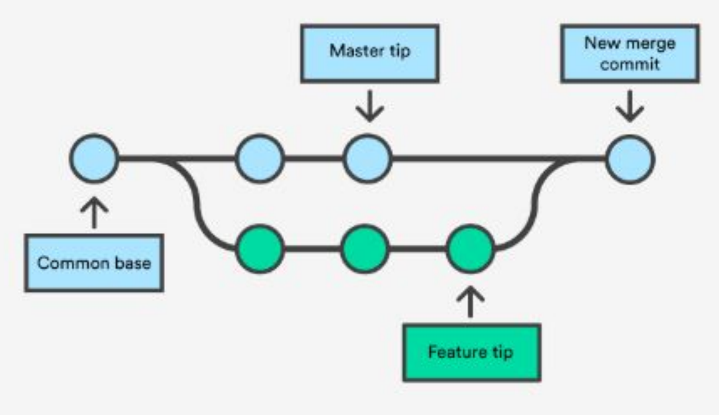
- delete a merged branch
- (use -D to delete regardless)

git checkout <branchName>

- switch to branchName

git merge

- combine branches



Demo: Branching

4

Cherry Picking

git cherry-pick

Cherry Picking

- Choose a commit from one branch
- Apply it to the branch you’re on
-
- i.e. you want to push a commit somewhere in your log to master

Demo: Cherry Picking

5

Rebasing

git rebase

Demo: Rebasing

6

Pull Requests

i.e. teammates, please pull my changes!

Demo: Pull Requests on Github

Thanks!

Any *questions* ?

You can email me at

- amf269@cornell.edu