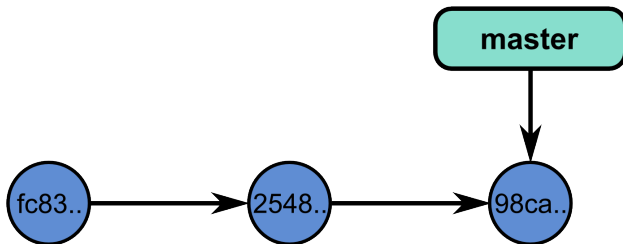


Git: Branching and Merging

What is a branch?

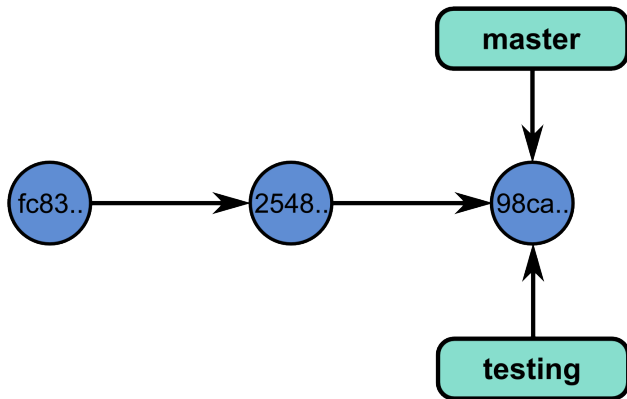
A branch is just a pointer to a commit:



We have been using the `master` branch.

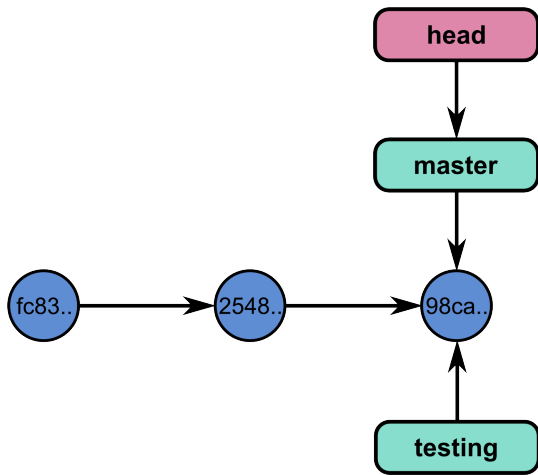
Intro to Branching

We can create a new branch and it will add a new pointer to the current commit:



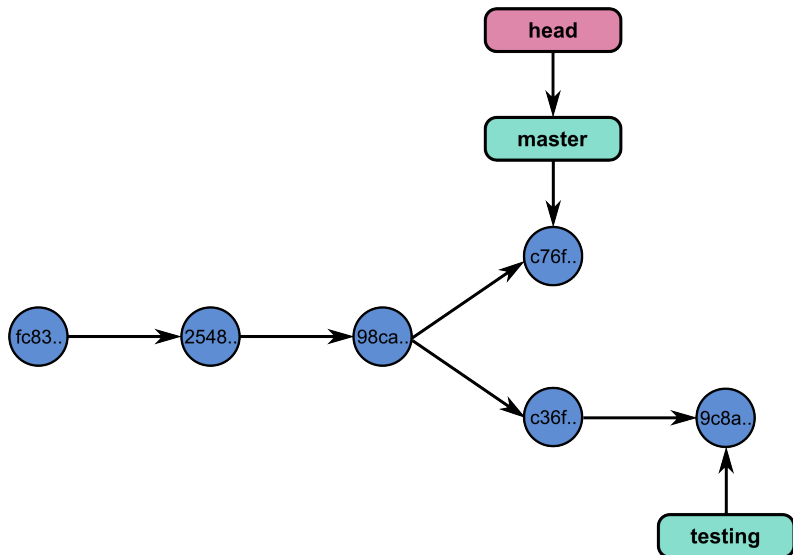
Intro to Branching

How does Git know which branch you are currently on?



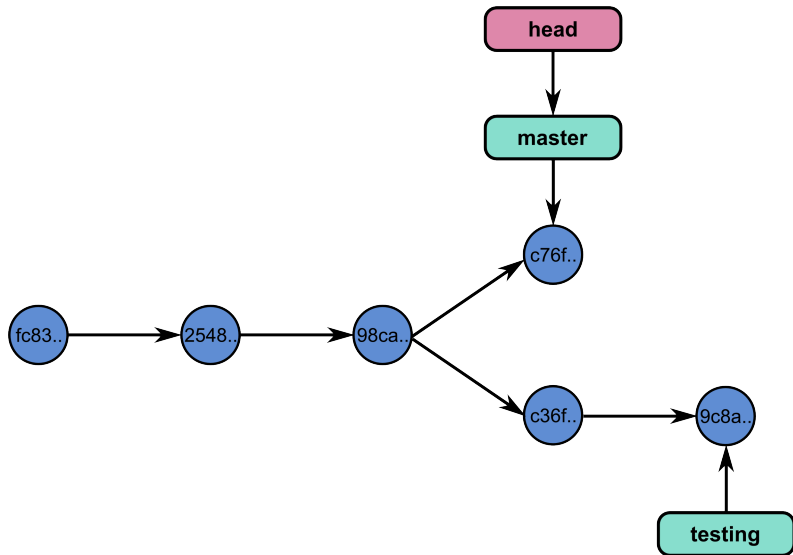
Intro to Branching

If you add commits on both branches, the directories can diverge:



Intro to Branching

Eventually, you might want to merge your changes on your branch back into the master development branch:



Why branch?

Isolation of changes.

Why branch?

Isolation of changes.

Try new things without disrupting main code.

Why branch?

Isolation of changes.

Try new things without disrupting main code.

Usually, there are a few main types of branches:

1. Feature Branch

- ▶ If a particular feature is disruptive enough that you don't want the entire development team to be affected in its early stages, you can create a branch on which to do this work.

2. Fixes Branch

- ▶ While development continues on the main trunk, a fixes branch can be created to hold the fixes to the latest released version of the software.

Now, how do we actually do this?

Resolving Conflicts

Your Turn

Exercise 5 (30 mins)