# Removing a file added in the most recent unpushed commit

**git rm --cached unwanted\_file**

# Stage our unwanted\_file for removal, but leave it on disk

**git commit --amend -CHEAD**

# Amend the previous commit with your change

# Simply making a new commit won't work, as you need

# to remove the file from the unpushed history as well

**git push**

# Push our rewritten, smaller commit

**The push command has a -n/--dry-run option which will compute what needs to be pushed but not actually do it.**

# How to view the committed files you have not pushed yet?

Assuming you're on local branch master, which is tracking origin/master:

**git diff --stat origin/master..**

The above assumes, of course, that "origin" is the name of your remote tracking branch (which it is if you've used clone with default options).

Can also use ---

**git log origin..HEAD**

Can also use ---

Say you made one commit that you haven't pushed...

First find the last two commits...

git log -2

This shows the last commit first, and descends from there...

[jason:~/git/my\_project] git log -2

commit ea7937edc8b10

Author: xyz

Date: Wed Jul 27 14:06:41 2016 -0500

Made a change in July

commit 52f9bf7956f0

Author: xyz

Date: Tue Jun 14 14:29:52 2016 -0500

Made a change in June

Now just use the two commit hashes (which I abbreviated) to run a diff:

git diff 52f9bf7956f0 ea7937edc8b10