

# GIT: Advanced Commands

## GIT Commit with Amend Activity: A simple exercise using git commit with amend

Brian Gorman, Author/Instructor/Trainer

©2017 - MajorGuidanceSolutions



# Introduction

---

Git makes it possible for us to “change” a commit. Sometimes we want to change the commit message. Other times we might want to actually make a small change, perhaps to fix a bug or remove some commented code that we don’t want in the repo.

In the case where we want to fix something and rewrite the commit history, we can do so with the git amend command.

Let’s gets started!

## Step 1: Amend a commit to change the commit message.

- a) Make sure you have any repository up to date and are working on any branch.

[clone repo]

Or

[git checkout master]

[git fetch origin]

[git pull origin master]

[git checkout -b GitAmendDemo]

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (master)
$ git checkout master
Already on 'master'
Your branch is up-to-date with 'origin/master'.

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (master)
$ git fetch origin

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (master)
$ git pull origin master
From https://github.com/majorguidancesolutions/SimpleActivityRepo
* branch      master      -> FETCH_HEAD
Already up-to-date.

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (master)
$ git checkout -b GitAmendDemo
Switched to a new branch 'GitAmendDemo'
```

Make a small change, add and commit.

[code info.txt]

[git status -s]

[git commit -am "This is my commit message"]

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo
$ code info.txt

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo
$ git status -s
M info.txt

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo
$ git commit -am "This is my commit message"
[GitAmendDemo d4e799d] This is my commit message
1 file changed, 2 insertions(+)

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo
```

### Notes

---

---

---

---

---

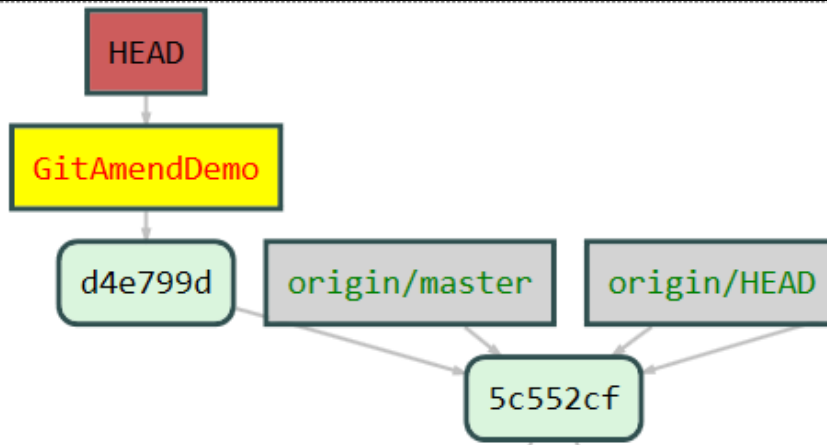
---

---

---

---

---



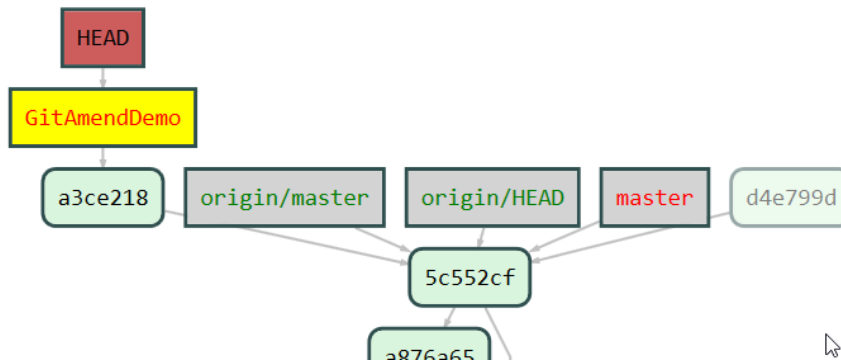
Note the commit id [d4e799d]

- b) The message had a typo, let's fix it.

[git commit --amend -m "This is an important change to the code"]

```

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ git commit --amend -m "This is an important change to the code"
[GitAmendDemo a3ce218] This is an important change to the code
Date: Sat Sep 23 21:45:21 2017 -0500
1 file changed, 2 insertions(+)
  
```



Note that the commit id is different, so we would need to be careful on a public branch if we are using an amend. We can also see the old commit is still showing as unreachable. Therefore we should do a quick cleanup.

## Step 2: Clean up the unreachable commits.

- a) Use reflog to expire unreachable and then cleanup with the garbage collector

To clean up the commits we just need to make sure we have the reflog set to expire our commits and then run the garbage collector. I have these commands aliased, but in case you don't and you want to run these [or want the commands for later reference], here they are:

[git reflog expire --expire-unreachable=now --all]

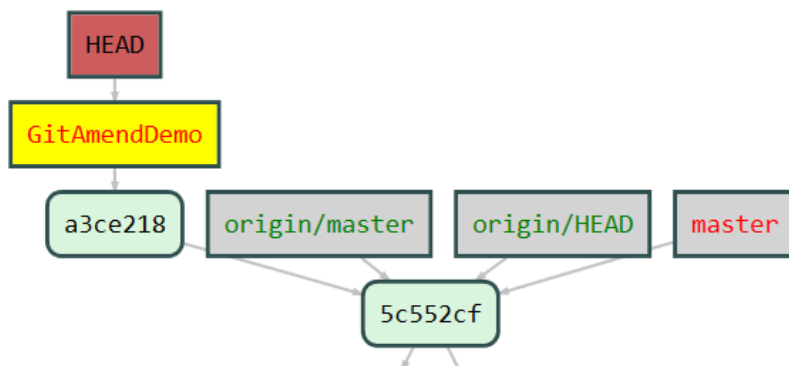
And

[git gc --prune=now]

And here are my aliases in my global config [check out the aliasing activity for more info about aliasing]:

```
alias.expireunreachablenow=reflog expire --expire-unreachable=now --all  
alias.gcunreachablenow=gc --prune=now  
fetch-prune=true
```

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)  
$ git expireunreachablenow  
  
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)  
$ git gcunreachablenow  
Counting objects: 25, done.  
Delta compression using up to 8 threads.  
Compressing objects: 100% (13/13), done.  
Writing objects: 100% (25/25), done.  
Total 25 (delta 13), reused 20 (delta 11)  
  
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)  
$
```



### Step 3: Amend with some actual changes.

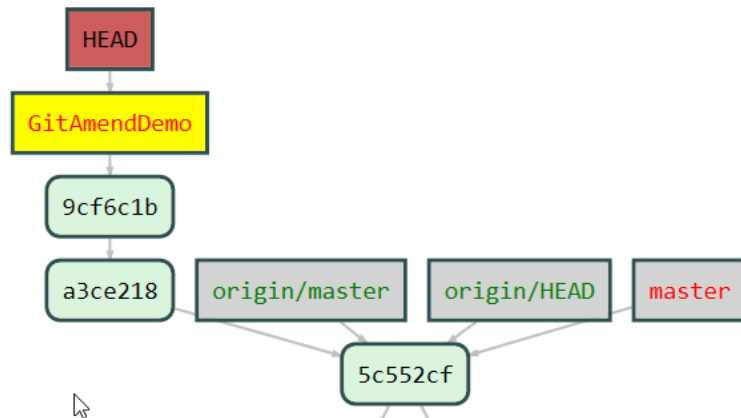
- a) Create some changes to info.txt and commit

```
[code info.txt]
```

```
[git status -s]
```

```
[git commit -am "Some changes for the next release"]
```

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)  
$ code info.txt  
  
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)  
$ git status -s  
M info.txt  
  
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)  
$ git commit -am "Some changes for the next release"  
[GitAmendDemo 9cf6c1b] Some changes for the next release  
1 file changed, 2 insertions(+), 1 deletion(-)
```



Make note of the commit id [9cf6c1b]

b) Add a new file to the repo

[touch readme.txt]

[code readme.txt]

[git status -s]

```

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ touch readme.txt

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ code readme.txt

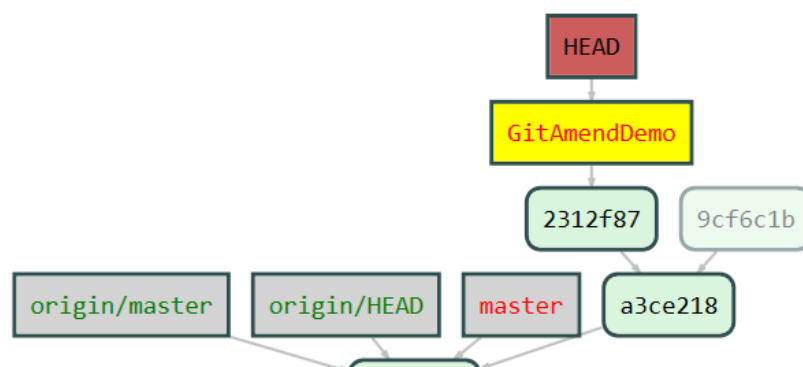
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ git status -s
?? readme.txt
  
```

c) Commit with amend to put the new file into the same commit as the changes for info.txt

[git commit --amend -m "Changed info.txt and added readme.txt"]

```

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ git commit --amend -m "Changed info.txt and added readme.txt"
[GitAmendDemo 2312f87] changed info.txt and added readme.txt
Date: Sat Sep 23 22:33:01 2017 -0500
1 file changed, 2 insertions(+), 1 deletion(-)
  
```

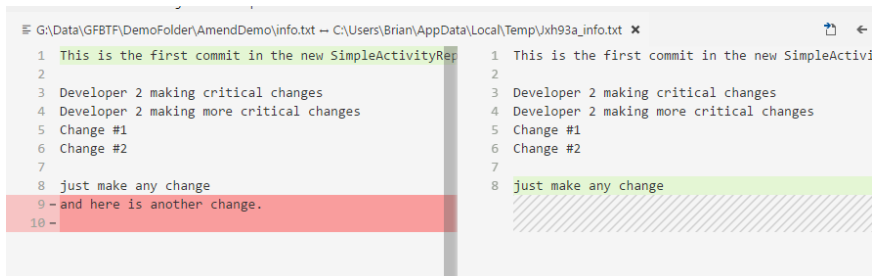


```
[git log --oneline]
```

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ git log --oneline
2312f87 (HEAD -> GitAmendDemo) changed info.txt and added readme.txt
a3ce218 This is an important change to the code
5c552cf (origin/master, origin/HEAD, master) Merge pull request #2 from m
dancesolutions/rebasing-demo-1
a876a65 more changes on my local branch
```

```
[git difftool 2312f87 a3c3218]
```

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo
$ git difftool 2312f87 a3c3218
```



Using the difftool I noticed I forgot to add the new file!!!

```
[git status -s]
```

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo
$ git status -s
?? readme.txt
```

#### d) First add, then commit with amend

```
[git add .]
```

```
[git commit --amend -m "Changed info.txt and added readme.txt"]
```

```
[git status -s]
```

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ git add .

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ git commit --amend -m "Changed info.txt and added readme.txt"
[GitAmendDemo 0cac76b] changed info.txt and added readme.txt
Date: Sat Sep 23 22:33:01 2017 -0500
2 files changed, 3 insertions(+), 1 deletion(-)
create mode 100644 readme.txt

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ git status -s
```

```
[git log --oneline]
```

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ git log --oneline
0cac76b (HEAD -> GitAmendDemo) changed info.txt and added readme.txt
a3ce218 This is an important change to the code
5c552cf (origin/master, origin/HEAD, master) Merge pull request #2 from m
dancesolutions/rebasing-demo-1
```

[git difftool 0cac76b a3ce218]

```
G:\Data\GFBTF\DemoFolder\AmendDemo\info.txt - C:\Users\Brian\AppData\Local\Temp\q3LeNb_info.txt x
1 This is the first commit in the new SimpleActivityRep
2
3 Developer 2 making critical changes
4 Developer 2 making more critical changes
5 Change #1
6 Change #2
7
8 just make any change
9 - and here is another change.
10 -

1 This is the first commit in the new SimpleActivityR
2
3 Developer 2 making critical changes
4 Developer 2 making more critical changes
5 Change #1
6 Change #2
7
8 just make any change

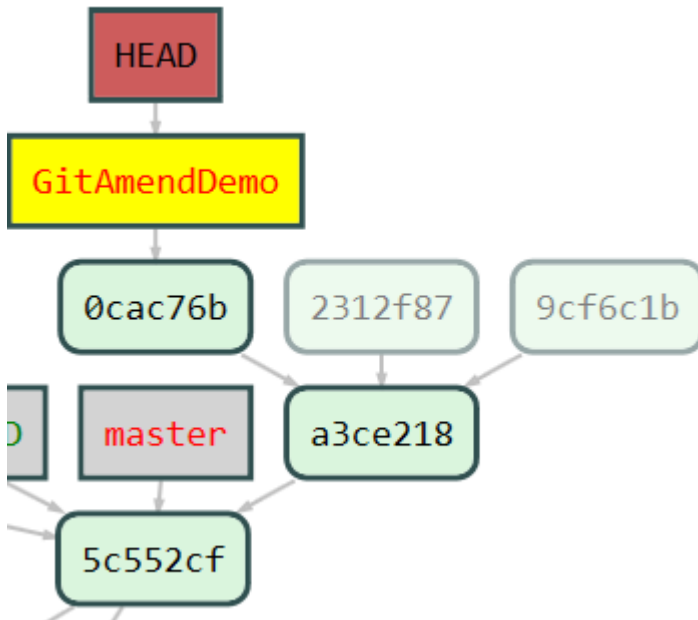
Selection View Go Debug Tasks Help

readme.txt x
1 This is the readme file...
```

Now we are on track.

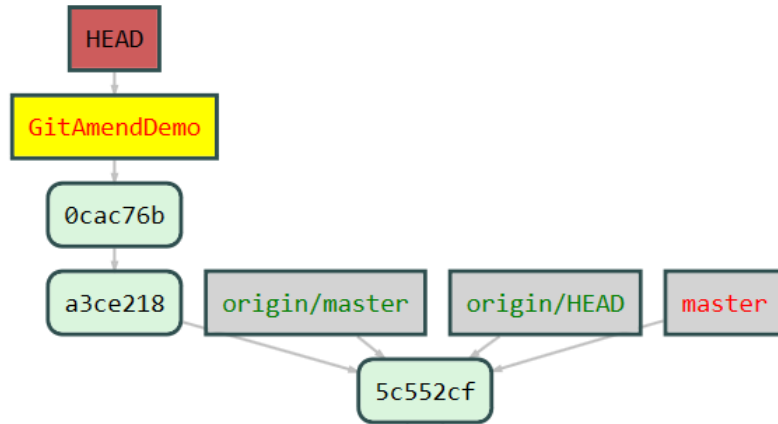
*Step 3: Cleanup the local unreachables, push to GitHub and merge changes into master.*

a) Repeat Step 2 for cleanup.



[commands for cleanup as in step 2]





## b) Push to GitHub

[git push -u origin <branchname>]

```

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (GitAmendDemo)
$ git push -u origin GitAmendDemo
Counting objects: 7, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (7/7), 662 bytes | 0 bytes/s, done.
Total 7 (delta 4), reused 5 (delta 2)
remote: Resolving deltas: 100% (4/4), completed with 2 local objects.
To https://github.com/majorguidancesolutions/SimpleActivityRepo.git
 * [new branch]      GitAmendDemo -> GitAmendDemo
Branch GitAmendDemo set up to track remote branch GitAmendDemo from origin.
  
```

## Create a Pull Request and Merge at GitHub

Your recently pushed branches:

GitAmendDemo (1 minute ago)

Compare & pull request

## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

base: master ... compare: GitAmendDemo ✓ Able to merge. These branches can be automatically merged.



git amend demo

Write Preview

Leave a comment

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

Create pull request

Review

No rev

Assign

No one

Labels

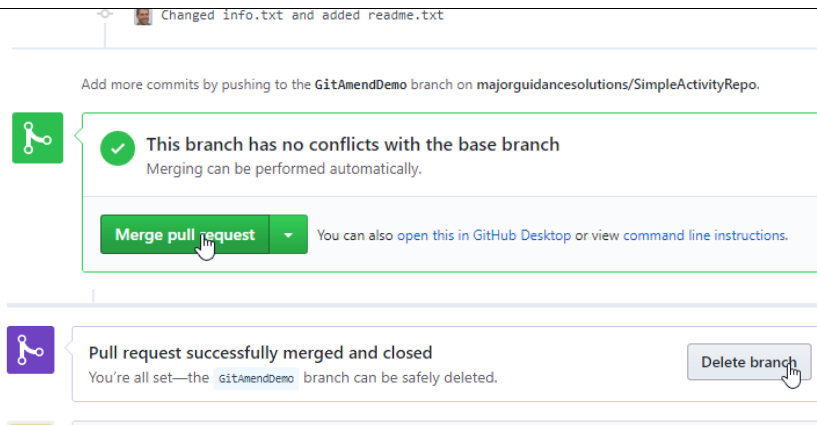
None

Project

None

Milestones

No milestones



c) Get the latest from github into the local repo

[git checkout master]

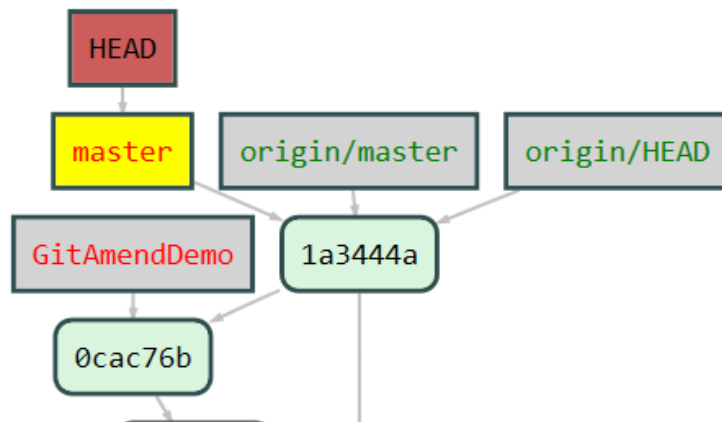
[git fetch origin]

[git pull origin master]

```
Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (
$ git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (
$ git fetch origin
From https://github.com/majorguidancesolutions/SimpleActivi
- [deleted]          (none)      -> origin/GitAmendDemo
remote: Counting objects: 1, done.
remote: Total 1 (delta 0), reused 1 (delta 0), pack-reused
Unpacking objects: 100% (1/1), done.
5c552cf..1a3444a  master      -> origin/master

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/AmendDemo (
$ git pull origin master
From https://github.com/majorguidancesolutions/SimpleActivi
* branch            master      -> FETCH_HEAD
Updating 5c552cf..1a3444a
Fast-forward
 info.txt | 3 +++          I
 readme.txt | 1 +
 2 files changed, 4 insertions(+)
 create mode 100644 readme.txt
```



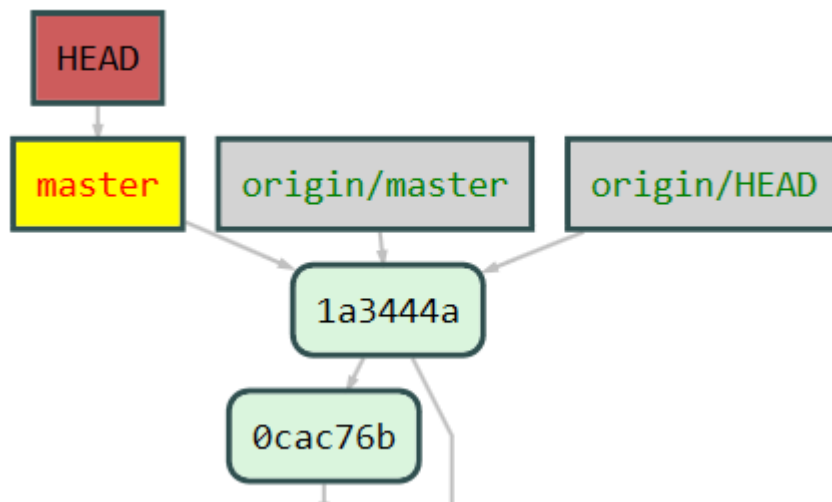
[git branch -d GitAmendDemo]

[git status]

```

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/Ame
$ git branch -d GitAmendDemo
Deleted branch GitAmendDemo (was 0cac76b).

Brian@SENTINEL MINGW64 /g/Data/GFBTF/DemoFolder/Ame
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean
  
```



And that concludes our git commit with amend activity.

## Closing Thoughts

In this activity we saw a couple of scenarios where we were able to change the commit history and amend changes into the most recent commit.

The first amend was a simple change to the commit message. The second change was a full commit with added files amended into the commit.

With the amend, we are able to change the most recent commit, but again we want to keep the history in tact if this was a public branch. Since this is a private branch, we were able to do what we wanted.

The other nice thing about the amend is that all of our changes only made one commit into the repository.

Take a few minutes to make some notes about the various commands we've learned about in this activity, and practice using them.

### Notes