Fix Simple Git Mistakes

To add extra files/changes or to fix the commit message:

```
git commit --amend
```

To remove files mistakenly added:

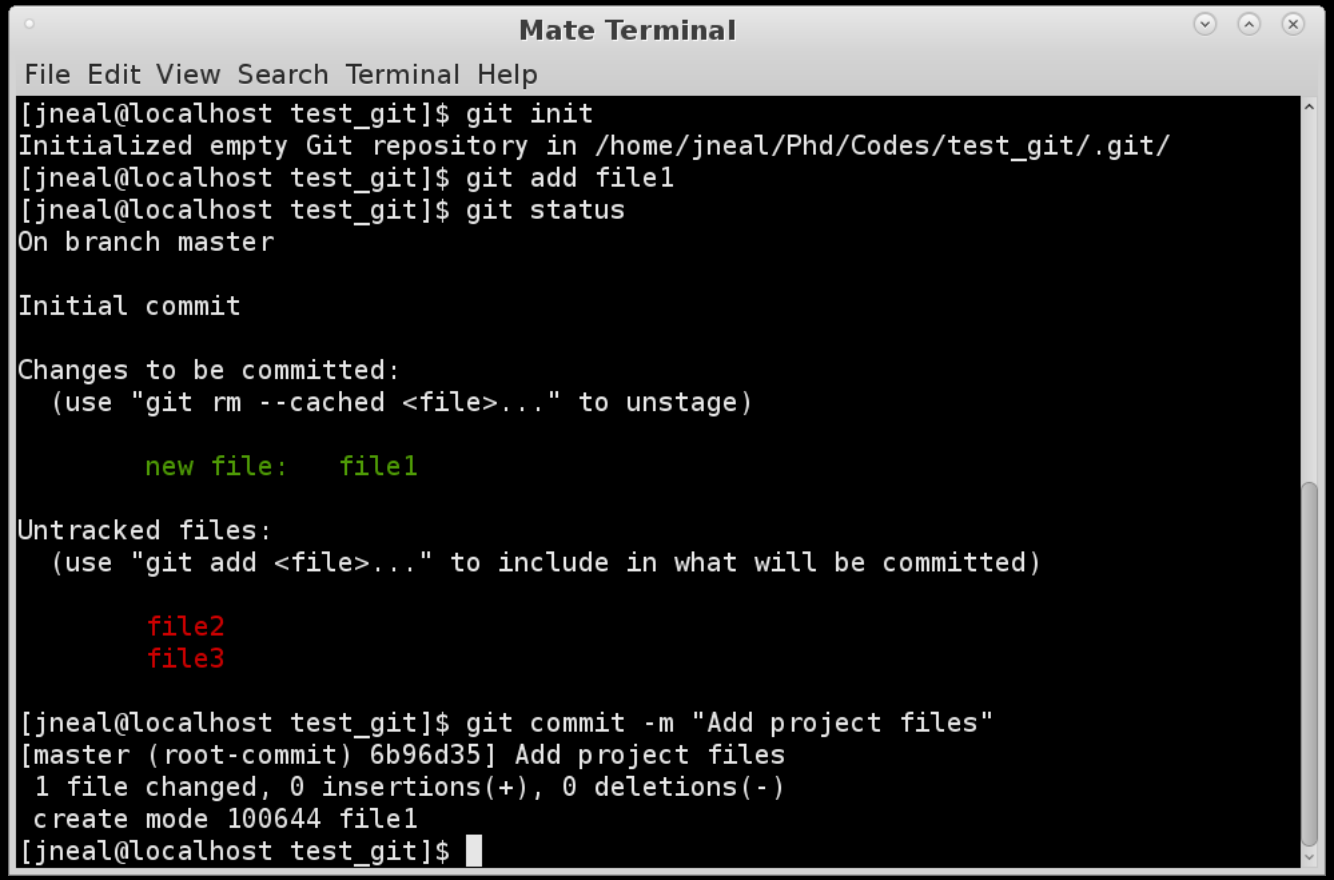
```
git reset --soft HEAD~1
git reset HEAD <file>
```

To undo changes from an entire commit:

```
git revert <SHA>
```

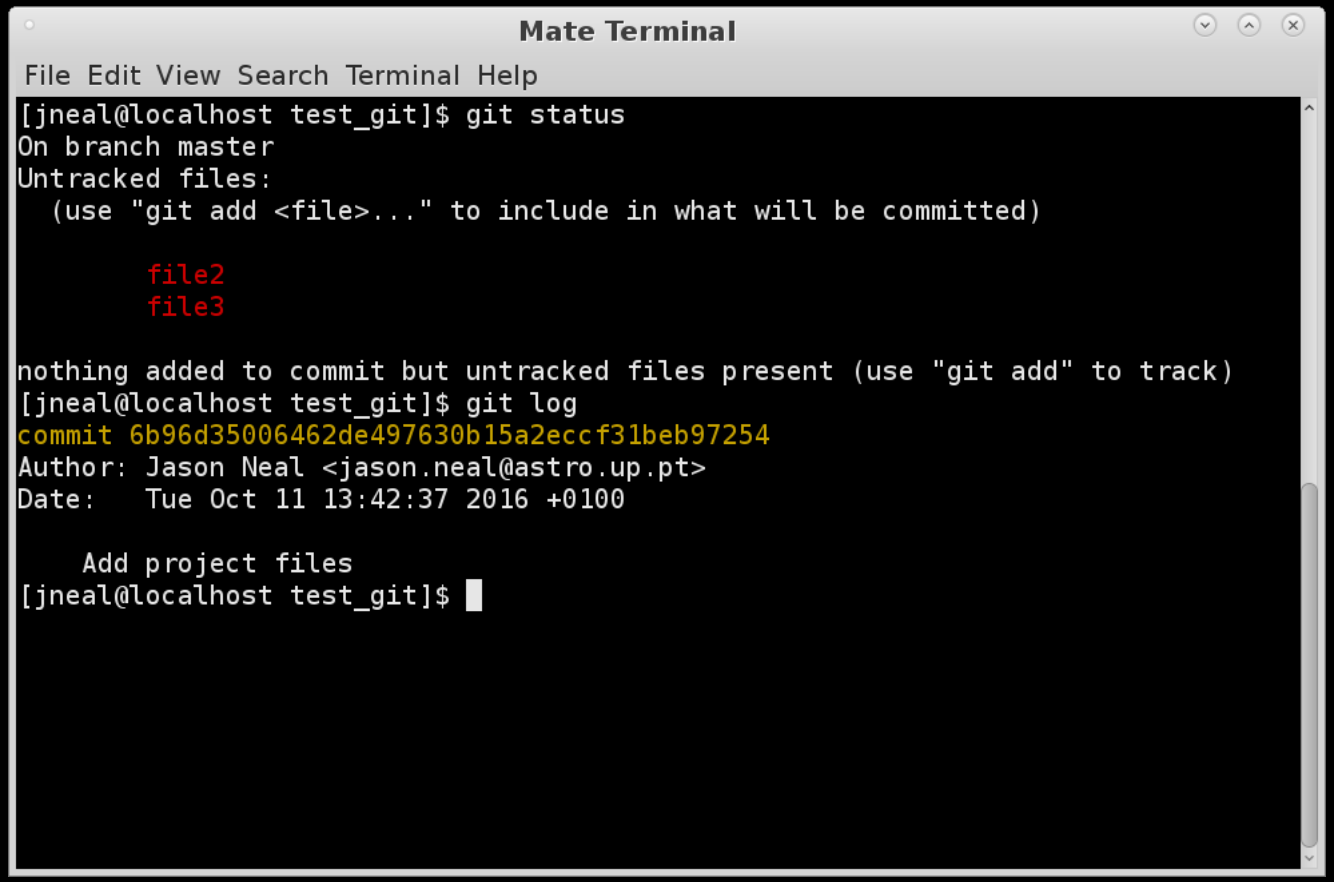
Git commit --amend

- Made a commit but missed files



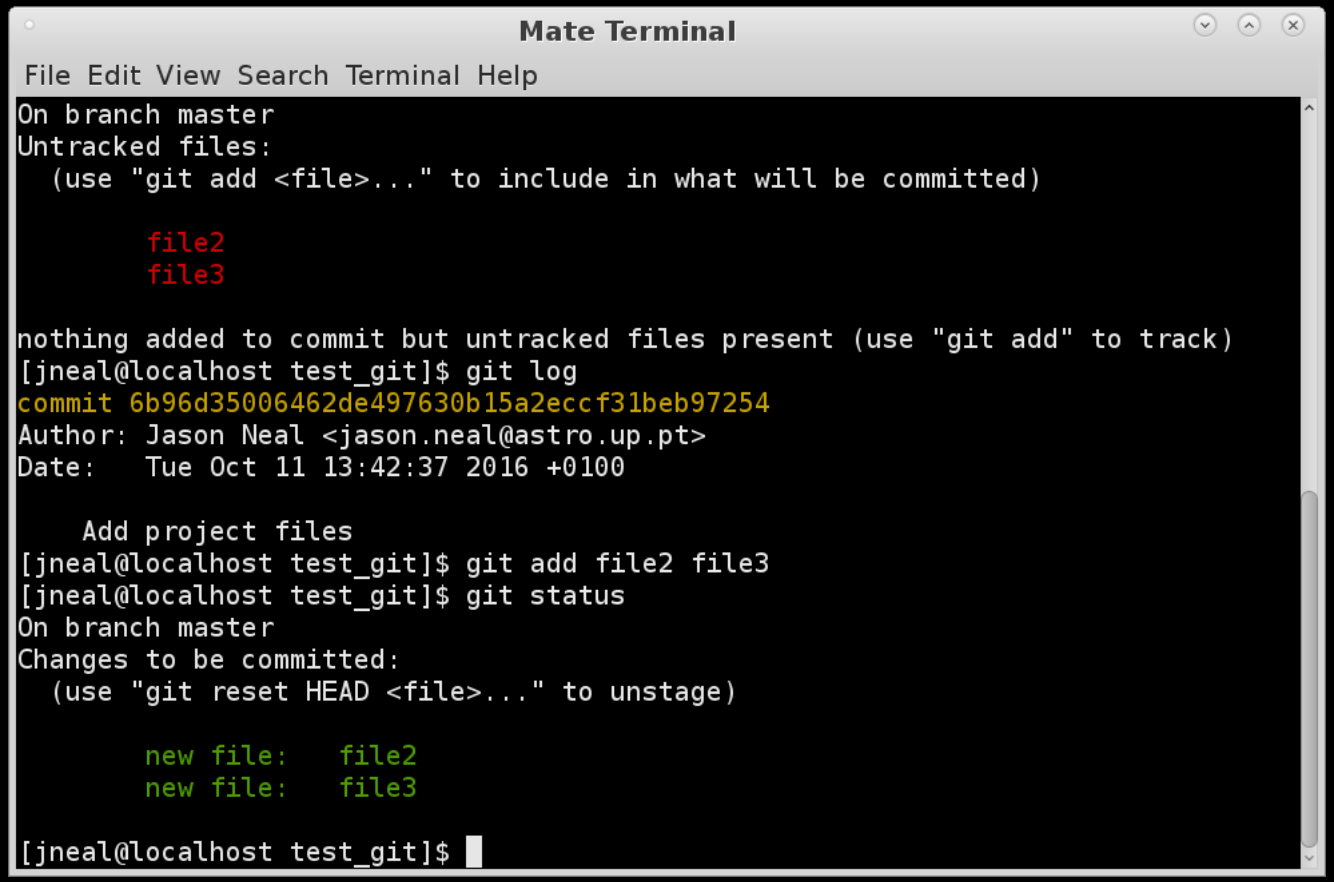
Git commit --amend

- Made a commit but missed files



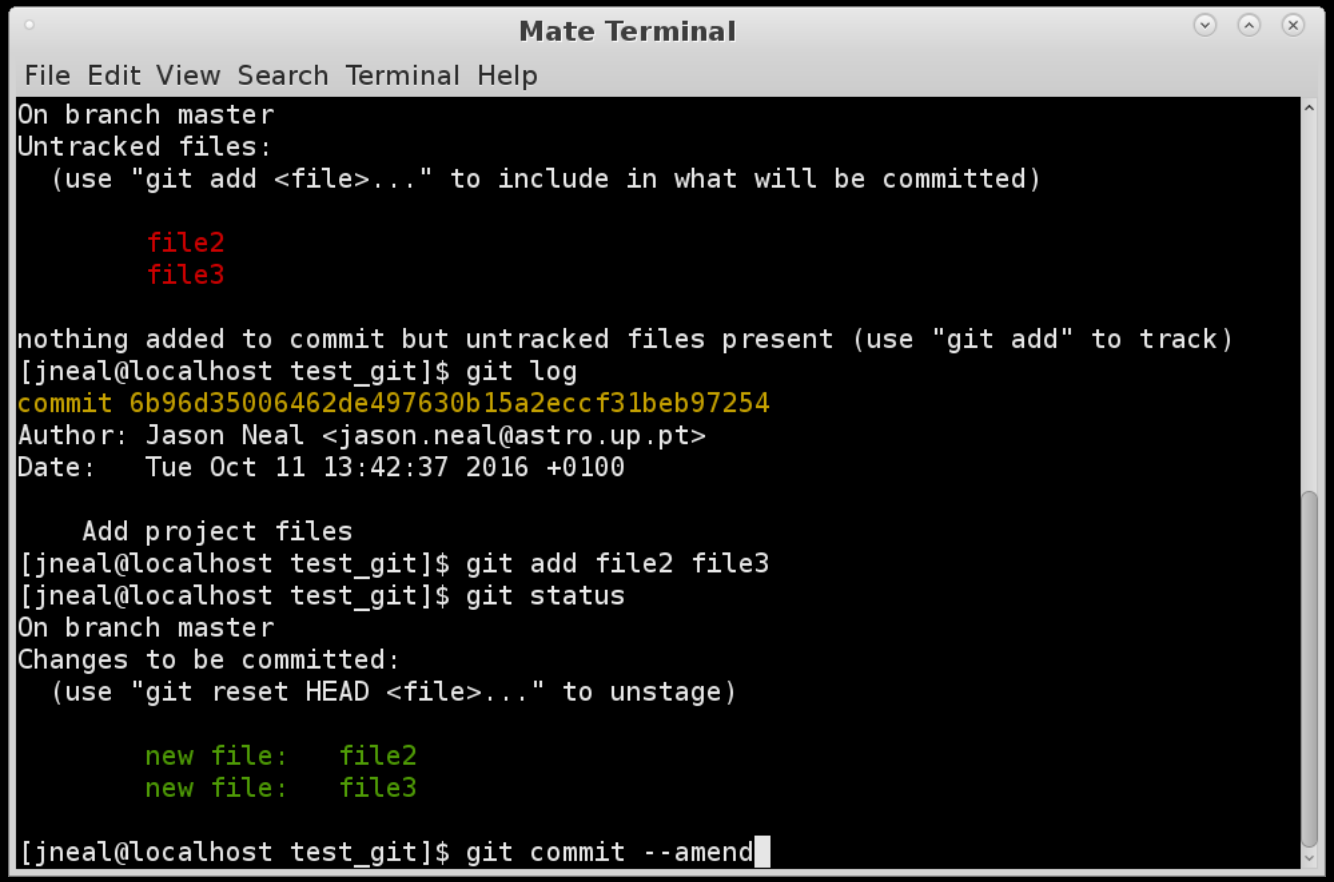
Git commit --amend

- Made a commit but missed files
- Stage the changes you want to add to the last commit



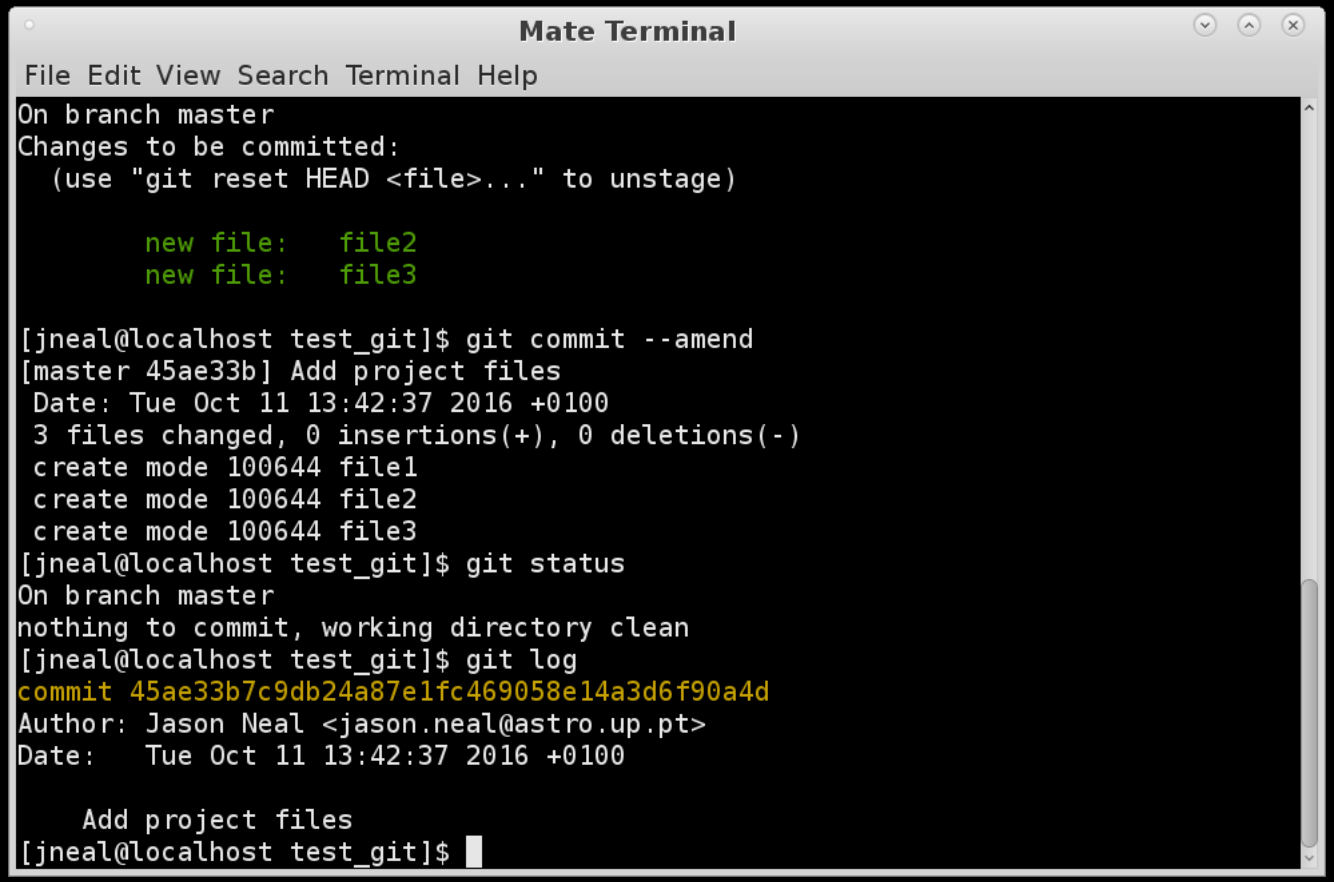
Git commit --amend

- Made a commit but missed files
- Stage the changes you want to add to the last commit
- Amend the commit



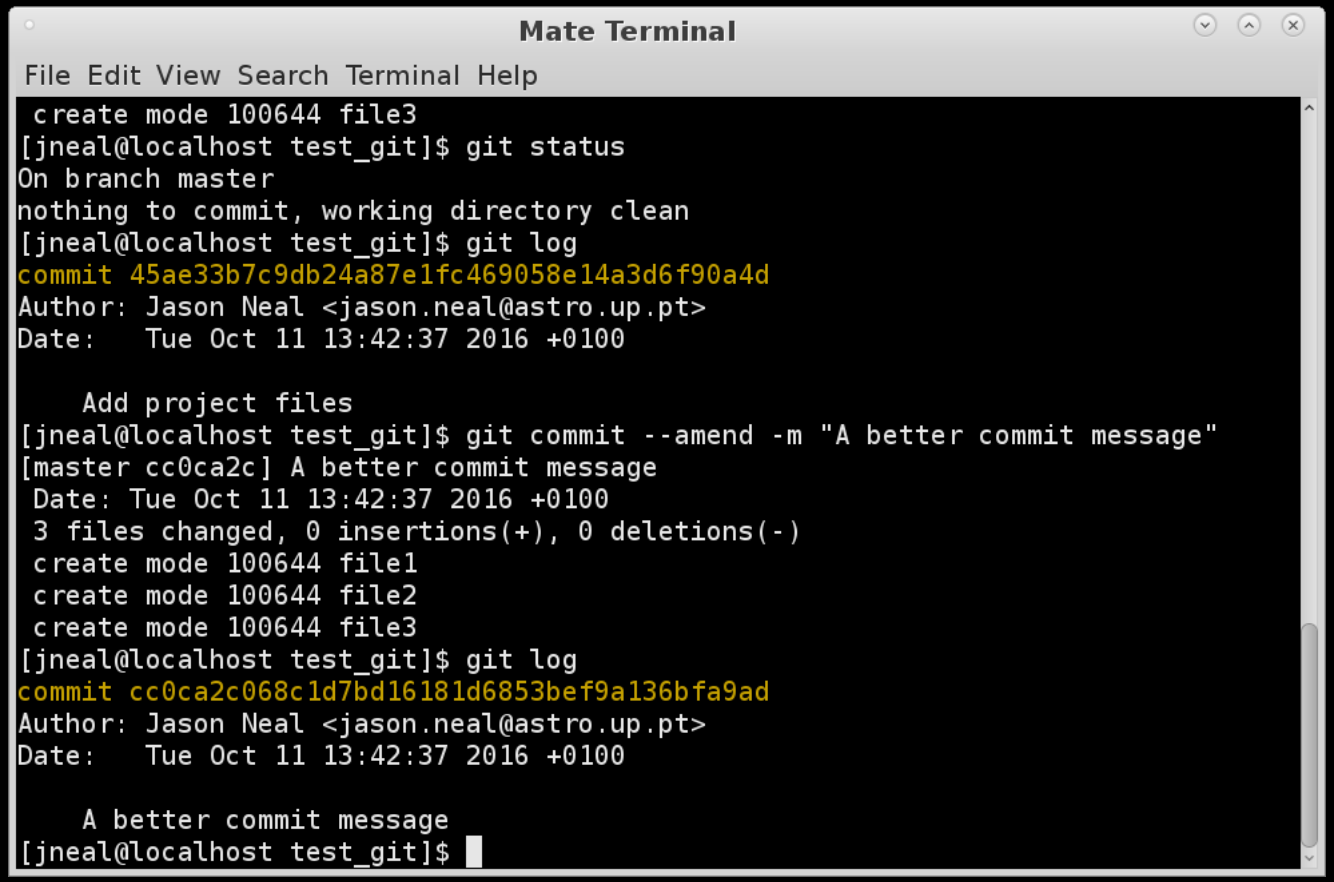
Git commit --amend

- Made a commit but missed files
- Stage the changes you want to add to the last commit
- Amend the commit
- The extra files were added to the last commit



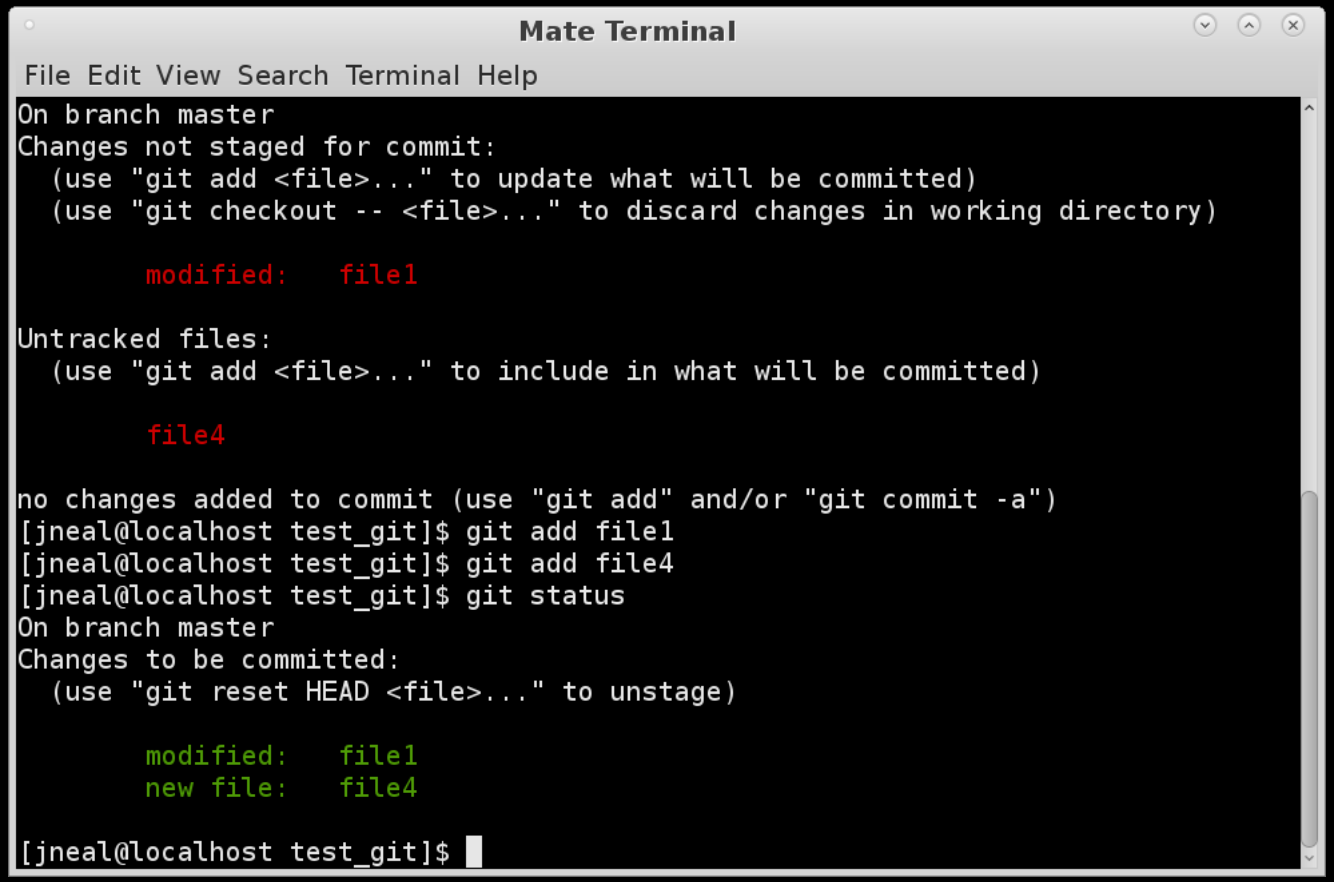
Git commit --amend

- Can also change the last commit message.



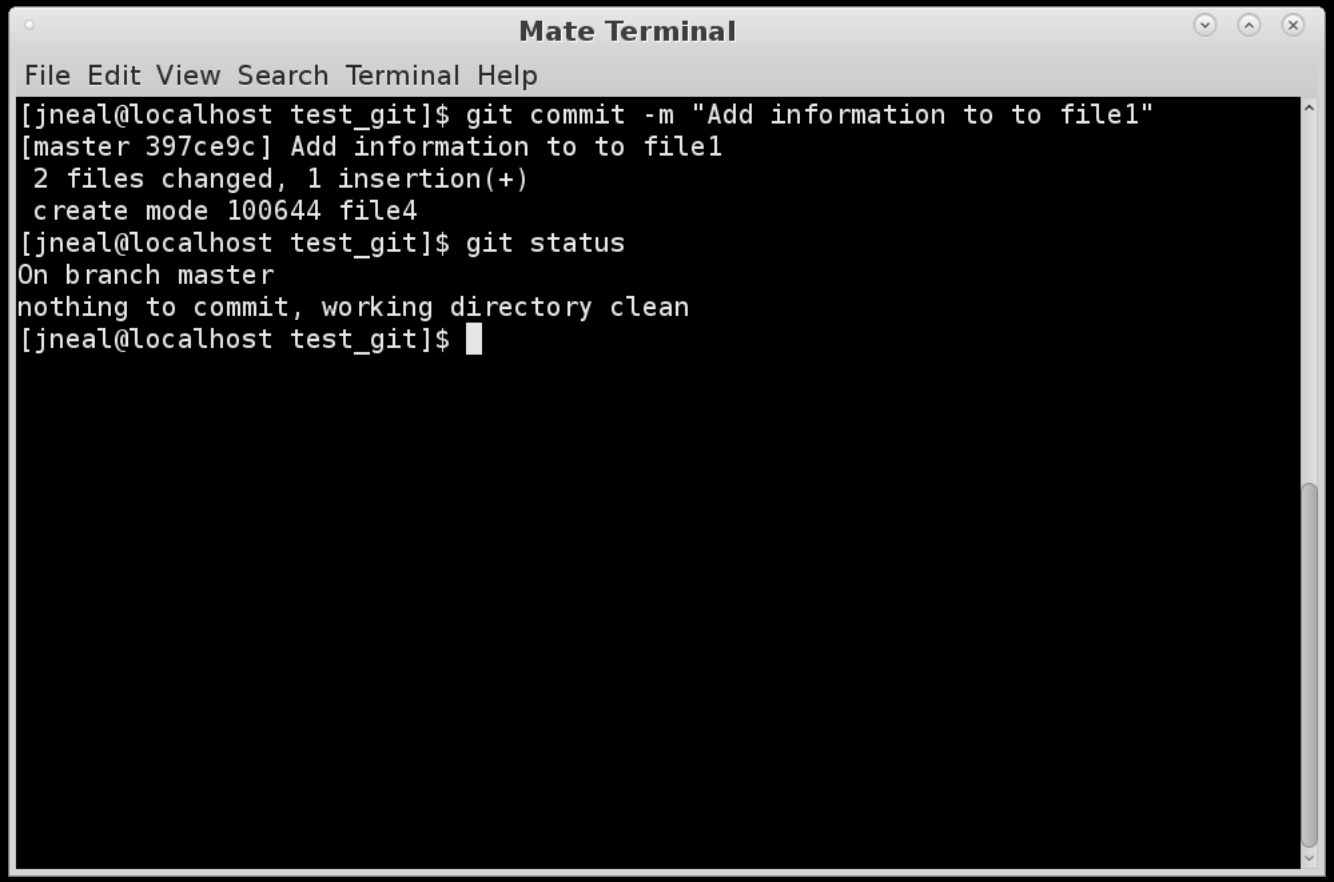
Git reset

- Accidentally added extra files.
- i.e. build files or files with sensitive information



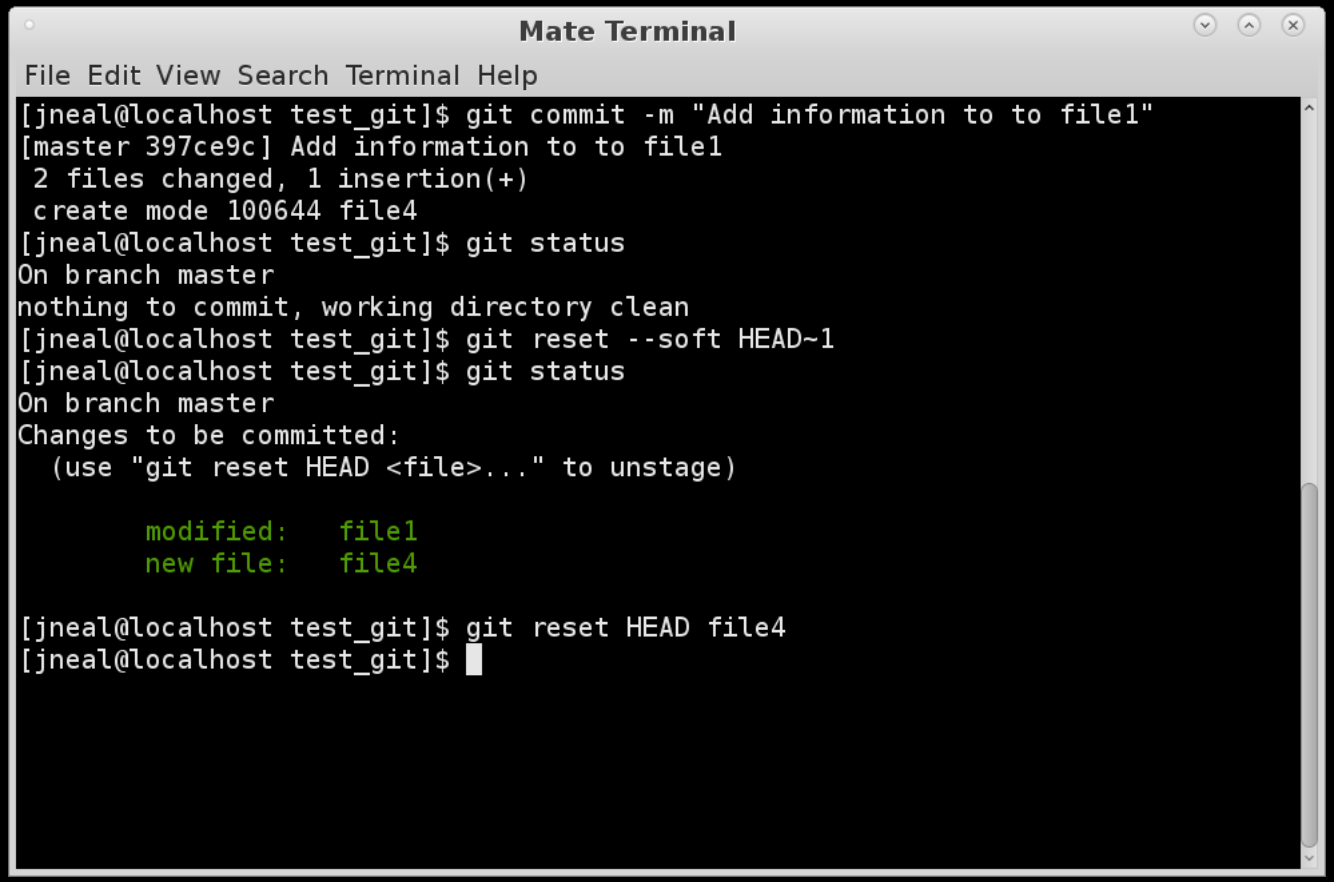
Git reset

- Accidentally added extra files and committed them.
- i.e. build files or files with sensitive information



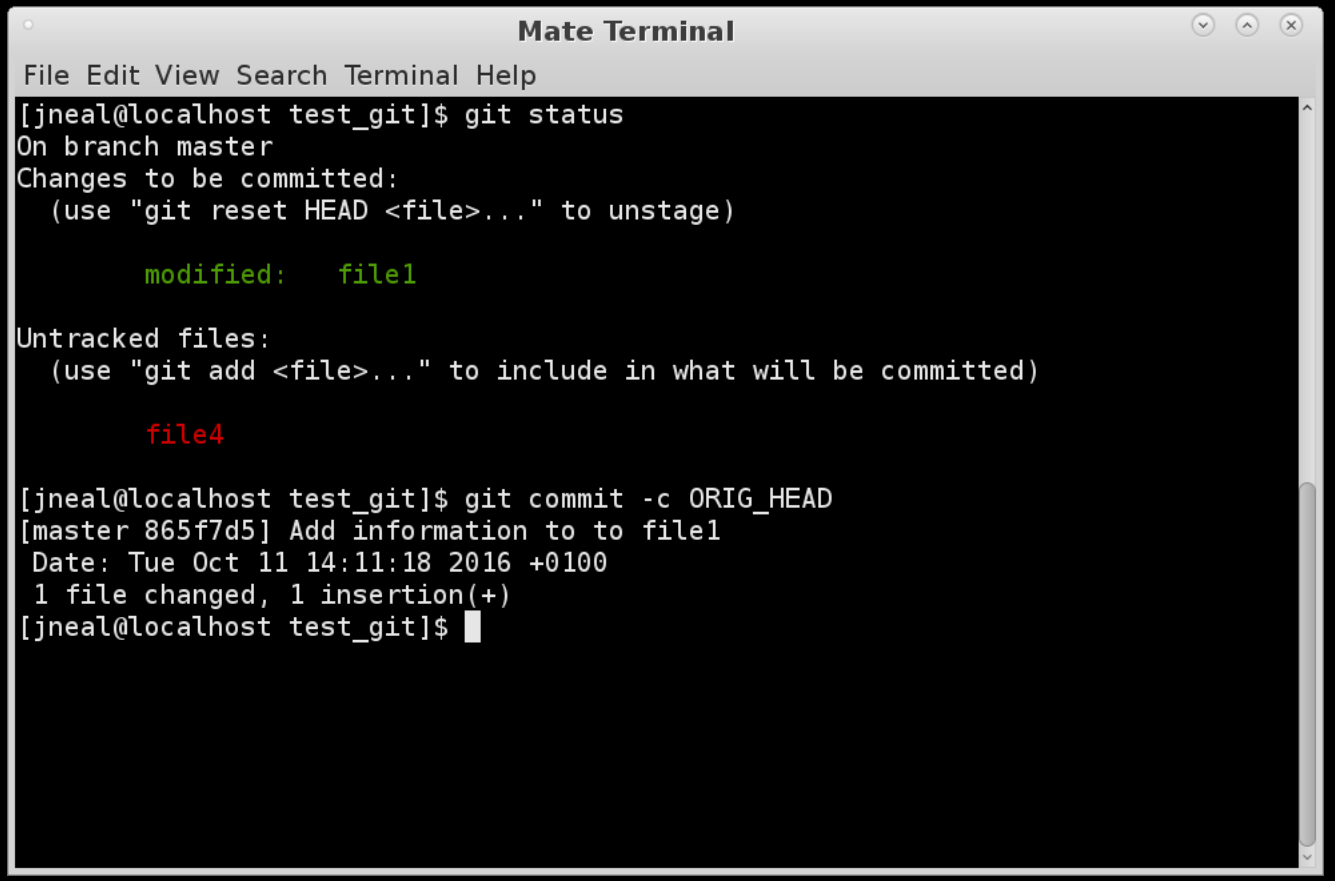
Git reset

- Reset HEAD to the previous commit.



Git reset

- Reset HEAD to the previous commit.
- Unstage files not wanted in this commit.
- Commit with the files removed.

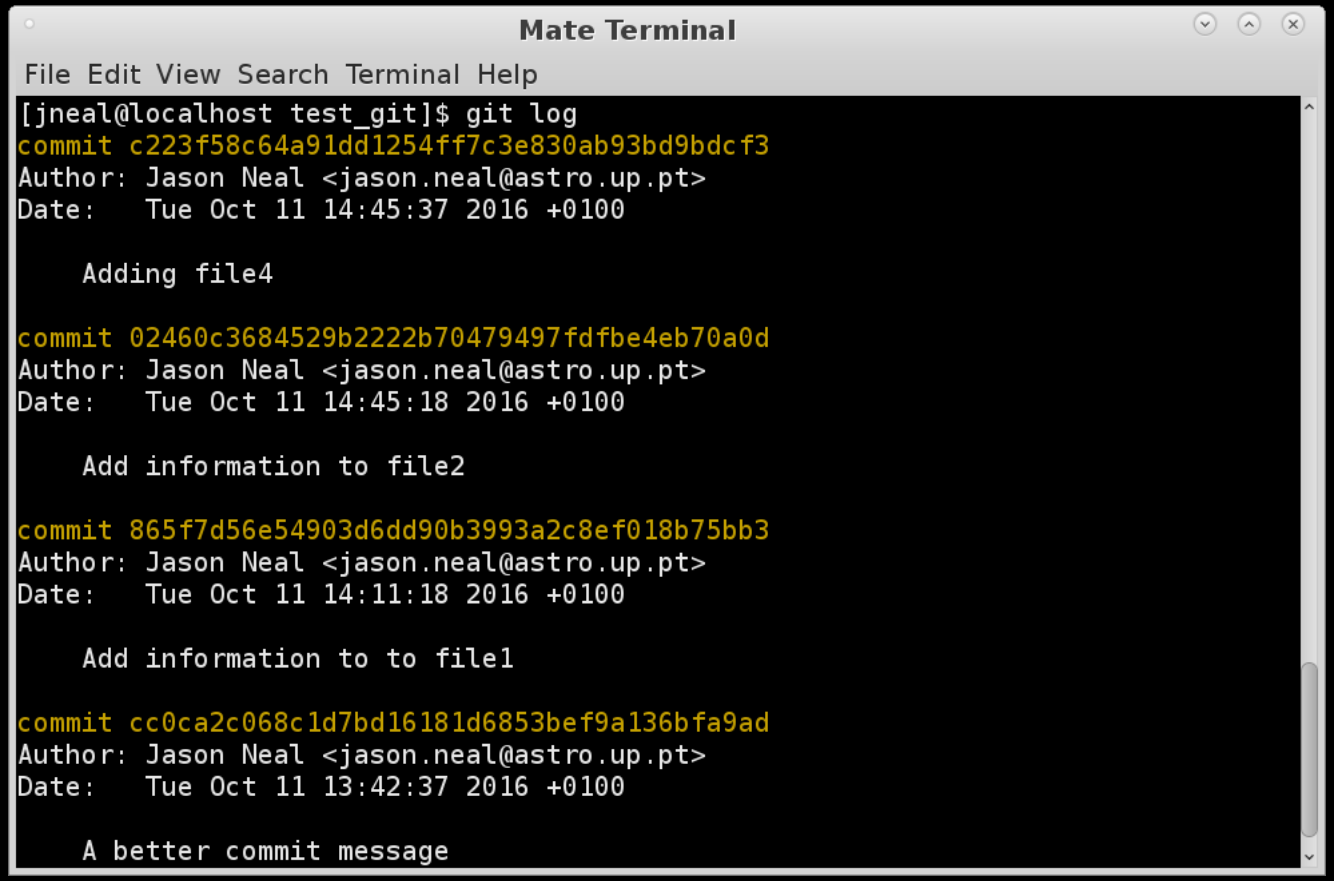


Git revert <SHA>

- Reverses (undoes) all the changes from a previous commit.
- Creates a new commit with the revert applied.
- You can also revert a revert.
- Need to provide the first few digits of the commit hash.
- The changes remain in your commit history! (Doesn't remove the sensitive information)

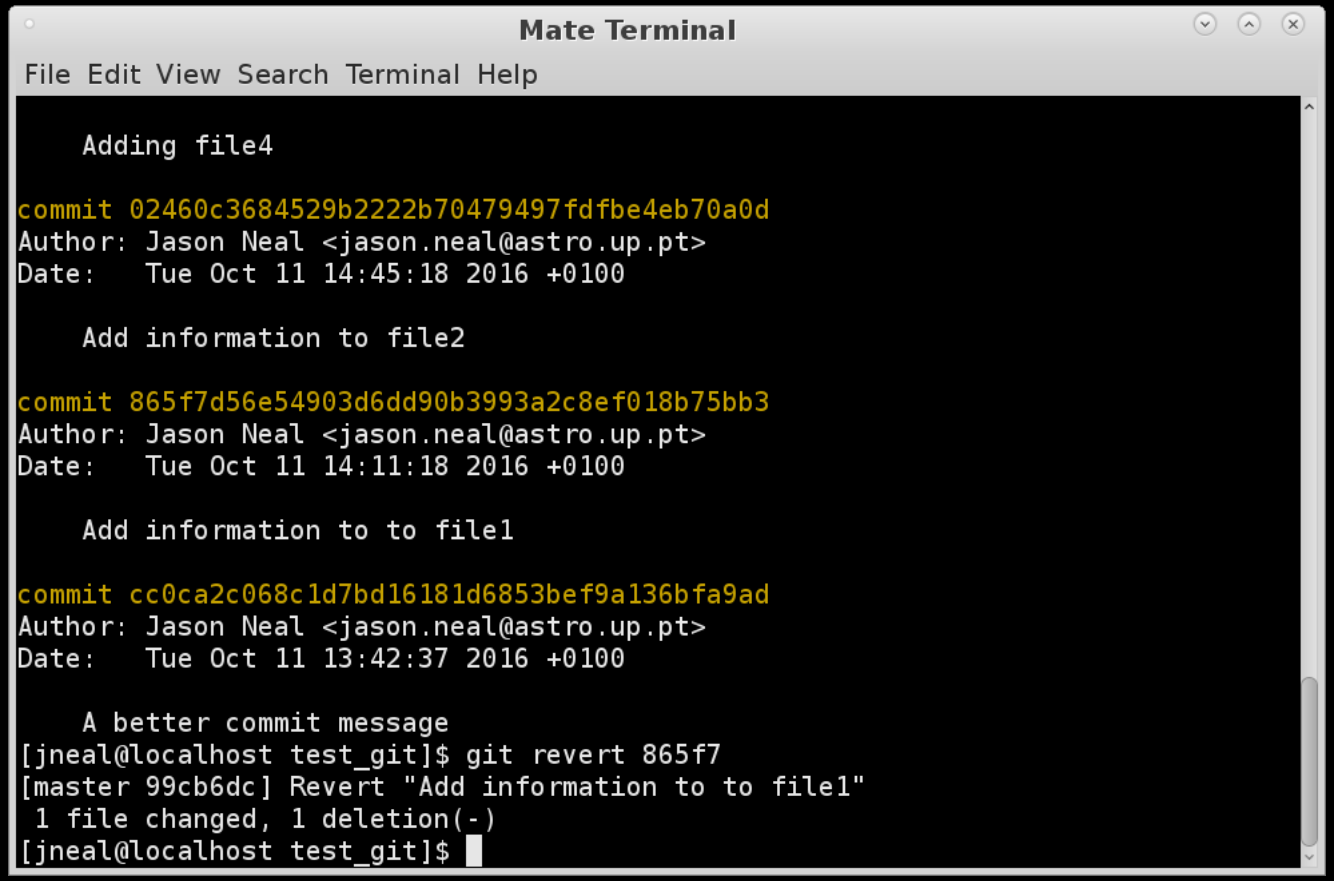
Git revert <SHA>

- Now we have a number of commits.



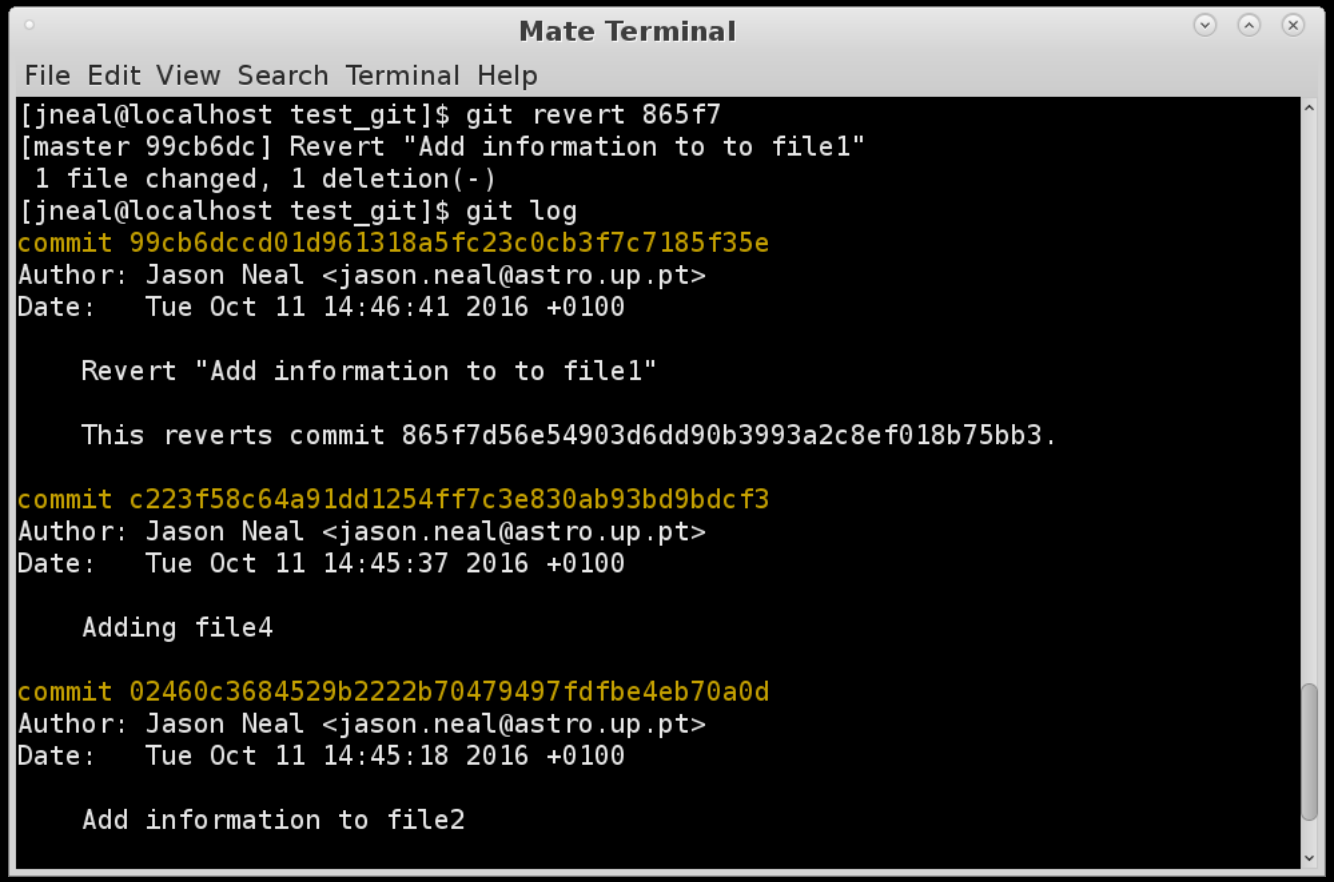
Git revert <SHA>

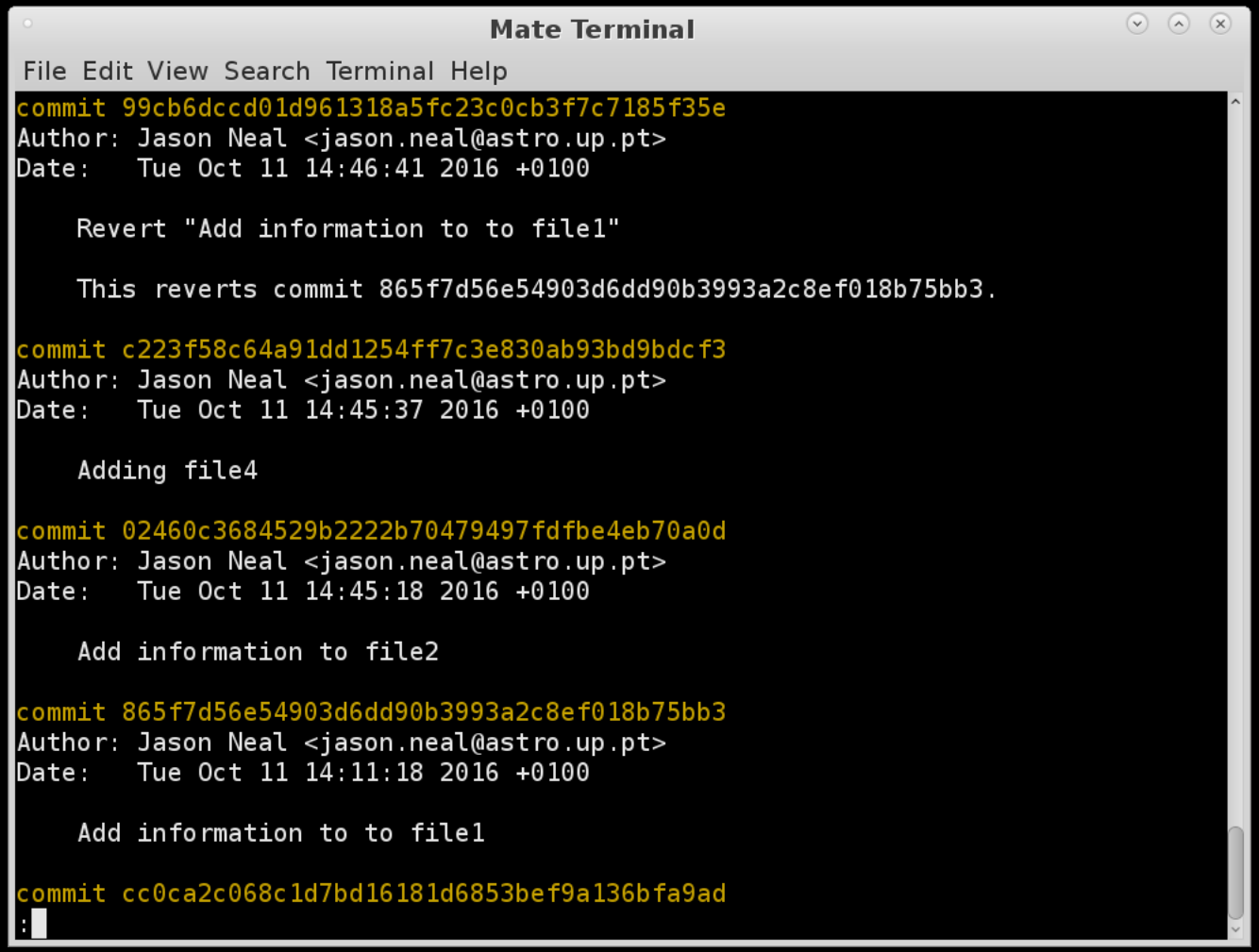
- Now we have a number of commits.
- We revert a previous commit starting 865f7.



Git revert <SHA>

- Now we have a number of commits.
- We revert a previous commit starting 865f7.



Git revert <SHA>

- Now we have a number of commits.
- We revert a previous commit starting 865f7.
- A new commit is created but the old commit remains!