

## Complex Git Command (The `git` Agent):

1. Start an interactive rebase for the last two commits:

```
● bash
```

```
git rebase -i HEAD~2
```



2. In the editor that opens, you'll see something like:

```
● Code
```

```
pick abc1234 First commit message
pick def5678 Second commit message

# Commands:
# p, pick <commit> = use commit
# r, reword <commit> = use commit, ↓ edit the message
# s, squash <commit> = use commit, but meld into previous commit
```



3. Change the second `pick` to `squash` (or just `s`):

```
● Code
```

```
pick abc1234 First commit message
squash def5678 Second commit message
```



4. Save and close the editor (`:wq` in vim, `Ctrl+S` then `Ctrl+X` in nano, etc.)

5. A second editor will open with the combined commit message. Edit it to your desired message, then save and close.

6. Done! Your last two commits are now combined into a single commit with your new message.