



A More Sophisticated Example

To get a better understanding of hooks, examine a more sophisticated example.

We'll cover the following ^

- Banning keywords

The previous pre-commit scripts were fairly limited in their usefulness. But just to give a flavor of what's possible, you're going to give an example that is able to choose whether to allow or reject a commit based on its content.

Banning keywords

Imagine that you've decided against allowing any mention of politics in your code. The following hook will reject any mention of politics (or any word beginning with "politic").

```
1  echo 'a political comment' >> file1
2  cat > .git/hooks/pre-commit << EOF
3  > if grep -rni politic *
4  > then
5  >   echo 'no politics allowed!'
6  >   exit 1
7  > fi
8  > echo OK
9  > exit 0
10 > EOF
11 git commit -am 'Political comment'
```

Terminal 1



Terminal





Click to Connect...

Again, the commit should have been rejected. If you change the content to something else that doesn't mention politics, it will commit and push just fine.

```
12 echo 'a boring comment' > file1
13 git commit -am 'Boring comment'
14 git push
```

Even more sophisticated scripts are possible but require a deeper knowledge of bash (or other scripting languages), which is out of scope. You will, however, look at one much more realistic example in the last section of this chapter.

Before you go on, remove the local Git hook so it doesn't affect any more work:

```
15 rm .git/hooks/pre-commit
```

← Back

Next →

A 'pre-commit' Hook

Additional Information on Hooks

☒ Mark as Completed





Report an Issue



Ask a Question

(https://discuss.educative.io/tag/a-more-sophisticated-example__git-hooks__learn-git-the-hard-way)