:pencil2: Contributor please read this

By default the command suggestion will generate a file named based on your commit. That's generally ok as long as you add the file to your commit. Someone can reorganize it later.

:warning: The command is written for posix shells. You can copy the contents of each `perl` command excluding the outer ' marks and dropping any '"/"' quotation mark pairs into a file and then run `perl file.pl` from the root of the repository to run the code. Alternatively, you can manually insert the items...

If the listed items are:

- ... **misspelled**, then please *correct* them instead of using the command.
- ... *names*, please add them to `.github/actions/spelling/allow/names.txt`.
- ... APIs, you can add them to a file in `.github/actions/spelling/allow/`.
- ... just things you're using, please add them to an appropriate file in `.github/actions/spelling/expect/`.
- ... tokens you only need in one place and shouldn't *generally be used*, you can add an item in an appropriate file in `.github/actions/spelling/patterns/`.

See the `README.md` in each directory for more information.

:microscope: You can test your commits **without** *appending* to a PR by creating a new branch with that extra change and pushing it to your fork. The check-spelling action will run in response to your **push** – it doesn't require an open pull request. By using such a branch, you can limit the number of typos your peers see you make. :wink:

:clamp: If you see a bunch of garbage

If it relates to a ...

well-formed pattern

See if there's a pattern that would match it.

If not, try writing one and adding it to a `patterns/{file}.txt`.

Patterns are Perl 5 Regular Expressions - you can test yours before committing to verify it will match your lines.

Note that patterns can't match multiline strings.

binary-ish string

Please add a file path to the `excludes.txt` file instead of just accepting the garbage.

File paths are Perl 5 Regular Expressions - you can test yours before committing to verify it will match your files.

`^` refers to the file's path from the root of the repository, so `^README\.md$` would exclude README.md (on whichever branch you're using).