
Enforce Git Message

Aug 09, 2021

Contents:

1	Installation	3
2	Verification	5
3	Notes	7
4	Attribution	9
5	Indices and tables	11



`enforce-git-message` is a tool for enforcing [conventional git commit messages](#) for all new git repositories you create by running `git init`. For existing repositories, you can go to your source directory and simply run:

```
enforce-git-message
```

Examples of valid commit messages:

```
+ 61c8ca9 fix: navbar not responsive on mobile
+ 479c48b test: prepared test cases for user authentication
+ a992020 chore: moved to semantic versioning
+ b818120 fix: button click even handler firing twice
+ c6e9a97 fix: login page css
+ dfdc715 feat(auth): added social login using twitter
```

Examples of invalid commit messages resulting in an error message:

```
- 61c8ca9 fix for navbar not responsive on mobile
- 479c48b prepared test cases for user authentication
- a992020 moved to semantic versioning
- b818120 fixed button click even handler firing twice
- c6e9a97 login page css fix
- dfdc715 added social login auth feature using twitter
```


CHAPTER 1

Installation

```
pip install enforce-git-message
```


CHAPTER 2

Verification

Go to your source folder and try to commit with a non-conventional message like this and it should fail:

```
> git commit -m "added a new feature for xyz"

COMMIT FAILED!

Please enter commit message in the conventional format and try to commit again.
↪Examples:

+ 61c8ca9 fix: navbar not responsive on mobile
+ 479c48b test: prepared test cases for user authentication
+ a992020 chore: moved to semantic versioning
+ b818120 fix: button click even handler firing twice
+ c6e9a97 fix: login page css
+ dfdc715 feat(auth): added social login using twitter
```

After that, try doing the commit with a valid message and it should work:

```
> git commit -m "feat(test): added xyz"
[master (root-commit) 8797aa0] feat(test): added xyz
 6 files changed, 1 insertion(+)
 create mode 100644 test.java
 create mode 100644 foo.txt
 create mode 100644 bar.txt
```


CHAPTER 3

Notes

Read [this article](#) to fully understand how the enforcement actually works by using git hooks.

CHAPTER 4

Attribution

CHAPTER 5

Indices and tables

- `genindex`
- `modindex`
- `search`