not$_{grep}$

Release 1.0.0

Jun 02, 2020

Project Operations

`not-grep` is kind of like grep, but different.

CHAPTER 1

WAT?

If you have ever needed to inspect a file for particular patterns, you probably used `grep`.

```
grep FooClassName file.py
```

If you needed to do that for a lot of files, you might have combined it with `find`.

```
find . -type f -name "*.py" -exec grep -n FooClassName {} /dev/null \;
```

This works great for one-off checks but less great if you need to do those checks repeatedly, if you need to do lots of such checks, if you need to do those checks somewhere that you don't have access to `grep`, or if you need to do things that `grep` cannot do.

CHAPTER 2

Not Grep?

`not-grep` is designed for static use, not ad-hoc use. For example, as part of a continuous integration test suite. This is why it gets its configuration from a config file, not the CLI. Because of this, the `not-grep` CLI is very simple: the only things you can specify are the config file and verbosity.

```
not-grep --config config.toml -vv
```

Inside the config file, things start to get interesting.

`not-grep` is built around checker plugins. Each plugin takes a map as input: the file glob pattern for the files you want to check and a value that tells the plugin what to do with that file.

The config file is a collection of TOML tables. The table name identifies the plugin and the table members are the input to that plugin.

```
# The "include" checker will error unless the specified value is include.
[include]
"src/**/*.py" = "__all__"

# The "exclude" checker will error if the specified value is include.
[exclude]
"src/**/*.py" = "FooClassName"
```

The output shows you, for each plugin, whether each matched file met or failed the plugin requirements. In lower verbosity levels, `not-grep` only shows failed checks.

```
$ not-grep --config config.toml -vv
================Running include checks================
-----------Checking src/**/*.py for pattern-----------
__all__
******************************************************
src/foo/__init__.py............................. PASS
src/foo/bar.py.................................. FAIL

================Running exclude checks================
```

(continues on next page)

(continued from previous page)

```
-----------Checking src/**/*.py for pattern-----------
FooClassName
**************************************************
src/foo/__init__.py............................. PASS
src/foo/bar.py.................................. PASS
```

CHAPTER 3

Awesome! Can I use it in GitHub Actions?

Yes. Yes you can.

```yaml
- uses: mattsb42/not-grep@master
  with:
    # If you don't set config-file the action uses ".github/not-grep.toml".
    config-file: ./github/config/check-things.toml
    # If you don't set debug, passing checks will be hidden.
    debug: true
```

3.1 Versioning

not-grep releases follow Semantic Versioning.

3.2 API

3.2.1 Public API

not_grep.REPLACEME
not_grep.REPLACEME
not_grep.REPLACEME

3.2.2 Internal Resources

Warning: This documentation is provided for information purposes only. No guarantee is provided on the modules and APIs described in here remaining consistent. Directly reference at your own risk.

not_grep._internal.REPLACEME
not_grep._internal.REPLACEME
not_grep._internal.REPLACEME

3.3 Changelog

Versions follow Semantic Versioning.

Changes or the upcoming release can be found in the "changelog.d" directory.

CHAPTER 4

1.0.0 – 2020-05-16

GitHub Action release.

CHAPTER 5

0.1.1 – 2020-05-16

Pre-release to test PyPI automation.

CHAPTER 6

0.1.0 – 2020-05-16

Initial release.