

---

**authors**

**Jul 25, 2021**



---

## Contents:

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Installing authors</b>	<b>3</b>
<b>3</b>	<b>Using authors</b>	<b>5</b>
<b>4</b>	<b>Configuring authors</b>	<b>7</b>
<b>5</b>	<b>Testing authors</b>	<b>9</b>
<b>6</b>	<b>authors</b>	<b>11</b>
<b>7</b>	<b>Indices</b>	<b>13</b>
	<b>Python Module Index</b>	<b>15</b>
	<b>Index</b>	<b>17</b>



# CHAPTER 1

---

## Introduction

---

Thank the people who contribute to your Git/GitHub project by creating an AUTHORS file.



## CHAPTER 2

---

### Installing authors

---

authors is available on GitHub at <https://github.com/critical-path/authors>.

If you do not have pip version 18.1 or higher, then run the following command from your shell.

```
[user@host ~]$ sudo pip install --upgrade pip
```

To install authors with test-related dependencies, run the following command from your shell.

```
[user@host ~]$ sudo pip install --editable git+https://github.com/critical-path/
↪authors.git#egg=authors[test]
```

To install it without test-related dependencies, run the following command from your shell.

```
[user@host ~]$ sudo pip install git+https://github.com/critical-path/authors.git
```

(If necessary, replace pip with pip3.)





## CHAPTER 3

---

### Using authors

---

First, select a project whose contributors you want to thank and change to its working directory (the directory in which the `.git` subdirectory resides).

```
[user@host ~]$ cd example-project
```

Then, run `git` with the `log` command and the `--format` option, redirecting standard output to `authors`.

```
[user@host example-project]$ git log --format=%an | authors
```

Finally, add and commit the newly-created `AUTHORS` file.

```
[user@host example-project]$ git add AUTHORS
[user@host example-project]$ git commit
```



## CHAPTER 4

---

### Configuring authors

---

To configure authors, create a file named `.authors.yml` in your project's working directory. `.authors.yml` should be valid YAML and contain five key-value pairs.

authors recognizes the following five keys.

- `name`: The name of the output file generated by authors.
- `kind`: The format of the output file, where `adoc`, `html`, `md`, `rst`, and `txt` are valid values.
- `heading`: The heading string.
- `opening`: The opening string.
- `closing`: The closing string.

If authors cannot find `.authors.yml`, or if it detects any missing or invalid values, then it will use the following default values.

```
name: AUTHORS
kind: md
heading: Authors
opening: Thank you to all of our contributors.
closing: This project would not be possible without you.
```



## CHAPTER 5

---

### Testing authors

---

To conduct testing, run the following commands from your shell.

```
[user@host authors]$ flake8 --count authors
[user@host authors]$ pytest --cov --cov-report=term-missing
```

If any of the tests fail due to a `PermissionError`, then run the following command from your shell.

```
[user@host authors]$ sudo $(which pytest) --cov --cov-report=term-missing
```

On a final note, if you edit any of the existing templates, then be sure to make the corresponding changes to the unit tests. Otherwise, they will fail.



## 6.1 authors package

### 6.1.1 authors.lib module

This module contains functions used by authors.

`authors.lib.read_configuration_file` (*file*='authors.yml')

Reads a configuration file, usually *authors.yml*.

**Parameters** *file* (*str*) – The name of a file with user-defined configuration settings.

The default value is *authors.yml*.

**Returns** The contents of a configuration file or an empty str.

**Return type** str

`authors.lib.parse_yaml_to_dict` (*contents*)

Parses YAML to a dict.

**Parameters** *contents* (*str*) – The contents of a file with user-defined configuration settings.

**Returns** Configuration settings (one key-value pair per setting) or an empty dict.

**Return type** dict

`authors.lib.validate_configuration` (*configuration*)

Validates configuration settings.

**Parameters** *configuration* (*dict*) – The unvalidated configuration settings.

**Returns** *configuration* – The validated configuration settings, using default values in the case of missing or invalid values.

**Return type** dict

`authors.lib.read_standard_input` ()

Reads from standard input.

**Returns** **authors** – A unique, sorted list of authors.

**Return type** list

`authors.lib.render_template(authors, configuration)`

Renders a template in *adoc*, *html*, *md*, *rst*, or *txt* format.

**Parameters**

- **authors** (*list*) – The authors to include in the rendered template.
- **configuration** (*dict*) – Configuration settings relevant to the rendered template (*heading*, *opening*, and *closing*).

**Returns** The rendered template.

**Return type** str

`authors.lib.write_authors_file(contents, configuration)`

Writes the rendered template to a file, usually *AUTHORS*.

**Parameters**

- **contents** (*str*) – The rendered template.
- **configuration** (*dict*) – Configuration settings relevant to the AUTHORS file (*name*).

`authors.lib.main(file='.authors.yml')`

Generates an AUTHORS file.

**Parameters** **file** (*str*) – The name of a file with user-defined configuration settings.

The default value is *.authors.yml*.

## 6.1.2 Module contents

Thank the people who contribute to your Git/GitHub project by creating an AUTHORS file.

`authors.main(file='.authors.yml')`

Generates an AUTHORS file.

**Parameters** **file** (*str*) – The name of a file with user-defined configuration settings.

The default value is *.authors.yml*.



## CHAPTER 7

---

### Indices

---

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



### **a**

authors, [12](#)

authors.lib, [11](#)



## A

`authors` (*module*), [12](#)  
`authors.lib` (*module*), [11](#)

## M

`main()` (*in module authors*), [12](#)  
`main()` (*in module authors.lib*), [12](#)

## P

`parse_yaml_to_dict()` (*in module authors.lib*), [11](#)

## R

`read_configuration_file()` (*in module authors.lib*), [11](#)  
`read_standard_input()` (*in module authors.lib*),  
[11](#)  
`render_template()` (*in module authors.lib*), [12](#)

## V

`validate_configuration()` (*in module authors.lib*), [11](#)

## W

`write_authors_file()` (*in module authors.lib*), [12](#)