

---

# **charms.templating.jinja2** **Documentation**

*Release 0.1*

**Cory Johns**

**Jul 26, 2018**



---

## Contents

---

<b>1 API Documentation</b>	<b>3</b>
<b>Python Module Index</b>	<b>5</b>



This module provides helpers for charms to work with Jinja2 templates.



## Reference

`charms.templating.jinja2.render` (*source=None, target=None, context=None, template=None, owner='root', group='root', perms=292, encoding='UTF-8', filters=None, tests=None, templates\_dir=None, template\_loader=None*)

Render a template.

Example usages:

```
render('app.conf.j2', '/etc/app.conf', {
    'my_var': 'my_val',
})

output = render(
    'tpl.j2',
    tests={
        'isnumeric': lambda s: s.isnumeric(),
    })

from jinja2 import Template
output = render(
    template='{{ config["my-opt"]|my_upper }}',
    filters={
        'my_upper': lambda s: s.upper(),
    })
```

### Parameters

- **source** (*str*) – Template path, relative to `templates_dir`.
- **target** (*str*) – Target path. Should be absolute path, or `None` (in which case the output will not be written, and only returned).

- **context** (*dict*) – Map of additional context variables to be passed to the template. Templates will always be given the following variables:
  - `config` A mapping of all of the charm config options.
- **owner** (*str*) – Name of the user that should own the target file.
- **group** (*str*) – Name of the group that should own the target file.
- **perms** (*int*) – Permissions of the target file.
- **filters** (*dict*) – Custom filters to be given to the template. Templates will always be given the following filters:
  - `map_format` Exactly the same as the built-in `format` filter, but with the order of the args rearranged to work with `map`.
- **templates\_dir** (*str*) – Directory in which to look for templates. Defaults to `$CHARM_DIR/templates`.
- **encoding** (*str*) – Defaults to `UTF-8`.
- **template\_loader** (*class*) – Template loader class to use instead of `FileSystemLoader`.

**Returns** The rendered template is returned as well as written to the target file (if not `None`).



**C**

`charms.templating.jinja2`, 3



## C

`charms.templating.jinja2` (module), 3

## R

`render()` (in module `charms.templating.jinja2`), 3