
kaptan Documentation

Release 0.5.8

Emre Yilmaz

May 25, 2016

1	installation	3
1.1	FreeBSD port	3
2	usage	5
3	exporting configuration	7
4	cli	9
5	running tests	11
6	contributors	13
6.1	API Reference	13
6.2	Command Line Interface	14
6.3	History	15
6.4	Roadmap	15
	Python Module Index	17

configuration parser.

installation

```
$ pip install kaptan
```

1.1 FreeBSD port

devel/py-kaptan package.

- To install the port: `cd /usr/ports/devel/py-kaptan/ && make install clean`
- To add the package: `pkg install devel/py-kaptan`

usage

supported handlers

- dict
- json
- yaml
- .ini
- python file

default (dict) handler

```
config = kaptan.Kaptan()
config.import_config({
    'environment': 'DEV',
    'redis_uri': 'redis://localhost:6379/0',
    'debug': False,
    'pagination': {
        'per_page': 10,
        'limit': 20,
    }
})

print config.get("pagination.limit")

# output: 20
```

json handler

```
config = kaptan.Kaptan(handler="json")
config.import_config('{"everything": 42}')

print config.get("everything")

# output: 42
```

yaml handler

```
config = kaptan.Kaptan(handler="yaml")
config.import_config("""
product:
  price:
    value: 12.65
    currency_list:
      1. TL
```

```
    2. EURO
    """)
print config.get("product.price.currency_list.0")
# output: TL
```

or you can get from directly from the filename:

```
config.import_config("configuration.yaml")
```

.ini handler

config.ini

```
[development]
database_uri = mysql://root:123456@localhost/posts

[production]
database_uri = mysql://poor_user:poor_password@localhost/poor_posts
```

```
config = kaptan.Kaptan(handler="ini")
config.import_config('config.ini')

print config.get("production.database_uri")
# output: mysql://poor_user:poor_password@localhost/poor_posts
```

file handler

config.py

```
DATABASE = 'mysql://root:123456@localhost/posts'
DEBUG = False
PAGINATION = {
    'per_page': 10,
    'limit': 20,
}
```

```
config = kaptan.Kaptan(handler="file")
config.import_config('config')

print config.get("DEBUG")
# output: False
```

exporting configuration

```
config = kaptan.Kaptan(handler="file")
config.import_config({
    'environment': 'DEV',
    'redis_uri': 'redis://localhost:6379/0',
    'debug': False,
    'pagination': {
        'per_page': 10,
        'limit': 20,
    }
})

print config.export("yaml")
```

output:

```
debug: false
environment: DEV
pagination: {limit: 20, per_page: 10}
redis_uri: redis://localhost:6379/0
```

```
print config.export("json")
```

outputs unindented json. `.export` accepts kwargs which pass into `json.dumps`.

```
print config.export("json", indent=4)
```

output:

```
{
  "environment": "DEV",
  "debug": false,
  "pagination": {
    "per_page": 10,
    "limit": 20
  },
  "redis_uri": "redis://localhost:6379/0"
}
```

`config.export('yaml')` also supports the kwargs for `pyyaml`.

New in Version 0.5.7: `config.export('yaml', safe=True)` will use `.safe_dump`.

exporting (defaults to json)

```
$ echo "environment: DEV" > config.yaml
$ kaptan config.yaml --export json > config.json
$ cat config.json
{"environment": "DEV"}
```

getting a value

```
$ kaptan config.yaml --key environment
DEV
```

specifying the handler

```
$ mv config.yaml config.settings
$ kaptan config.settings:yaml --export json
{"environment": "DEV"}
```

config from stdin

```
$ echo '{"source": "stdin"}' | kaptan -
{"source": "stdin"}
$ echo 'source: stdin' | kaptan -:yaml
{"source": "stdin"}
```

merging configs

```
$ echo "environment: PROD" > config.settings
$ echo '{"source": "stdin"}' | kaptan - config.json config.settings:yaml
{"environment": "PROD", "source": "stdin"}
```

setting default handler

```
$ echo "source: stdin" | kaptan --handler yaml - config.settings
{"environment": "PROD", "source": "stdin"}
```

writing json with yaml

```
$ kaptan -:yaml -e json
<type yaml here>
<Ctrl + D>
<get json here>
```

running tests

With *py.test*:

```
$ py.test
```

contributors

- Cenk Altı
- Wesley Bitter
- Mark Steve
- Tony Narlock
- Berker Peksag

Explore:

6.1 API Reference

6.1.1 kaptan

configuration parser.

copyright

3. 2013 by the authors and contributors (See AUTHORS file).

license BSD, see LICENSE for more details.

class `kaptan.Kaptan` (*handler=None*)

Bases: `object`

HANDLER_MAP = {`u'yaml'`: <class 'kaptan.handlers.yaml_handler.YamlHandler'>, `u'json'`: <class 'kaptan.handlers.json_handler.JsonHandler'>

upsert (*key, value*)

__is_python_file (*value*)

Return True if the *value* is the path to an existing file with a `.py` extension. False otherwise

import_config (*value*)

__get (*key*)

get (*key=None, default=<object object>*)

export (*handler=None, **kwargs*)

__Kaptan__handle_default_value (*key, default*)

`kaptan.get_parser` ()

Create and return argument parser.

Return type `argparse.ArgumentParser`

Returns CLI Parser

`kaptan.main()`

class `kaptan.Kaptan` (*handler=None*)

Bases: `object`

`__is_python_file` (*value*)

Return True if the *value* is the path to an existing file with a `.py` extension. False otherwise

6.1.2 Handlers

class `kaptan.handlers.BaseHandler`

Bases: `object`

Base class for data handlers.

`dump` (*data*)

`load` (*data*)

class `kaptan.handlers.dict_handler.DictHandler`

Bases: `kaptan.handlers.BaseHandler`

`dump` (*data*)

`load` (*data*)

class `kaptan.handlers.ini_handler.IniHandler`

Bases: `kaptan.handlers.BaseHandler`

`dump` (*file_*)

`load` (*value*)

class `kaptan.handlers.json_handler.JsonHandler`

Bases: `kaptan.handlers.BaseHandler`

`dump` (*data*, ***kwargs*)

`load` (*data*)

class `kaptan.handlers.yaml_handler.YamlHandler`

Bases: `kaptan.handlers.BaseHandler`

`dump` (*data*, *safe=False*, ***kwargs*)

`load` (*data*)

6.2 Command Line Interface

```
usage: kaptan [-h] [--handler HANDLER] [-e EXPORT] [-k KEY]
             [config_file [config_file ...]]
```

Positional arguments:

`config_file` file/s to load config from

Options:

`--handler=json` set default handler

`-e=json, --export=json` set format to export to
`-k, --key` set config key to get value of

6.3 History

Here you can find the recent changes to kaptan

6.3.1 CURRENT

- Changelog created
- Sphinx documents added

6.4 Roadmap

See <https://github.com/emre/kaptan/issues>.

k

kaptan, 15

Symbols

`_Kaptan__handle_default_value()` (kaptan.Kaptan method), 13

`_get()` (kaptan.Kaptan method), 13

`_is_python_file()` (kaptan.Kaptan method), 13, 14

B

BaseHandler (class in kaptan.handlers), 14

D

DictHandler (class in kaptan.handlers.dict_handler), 14

`dump()` (kaptan.handlers.BaseHandler method), 14

`dump()` (kaptan.handlers.dict_handler.DictHandler method), 14

`dump()` (kaptan.handlers.ini_handler.IniHandler method), 14

`dump()` (kaptan.handlers.json_handler.JsonHandler method), 14

`dump()` (kaptan.handlers.yaml_handler.YamlHandler method), 14

E

`export()` (kaptan.Kaptan method), 13

G

`get()` (kaptan.Kaptan method), 13

`get_parser()` (in module kaptan), 13

H

HANDLER_MAP (kaptan.Kaptan attribute), 13

I

`import_config()` (kaptan.Kaptan method), 13

IniHandler (class in kaptan.handlers.ini_handler), 14

J

JsonHandler (class in kaptan.handlers.json_handler), 14

K

Kaptan (class in kaptan), 13, 14

kaptan (module), 13, 15

L

`load()` (kaptan.handlers.BaseHandler method), 14

`load()` (kaptan.handlers.dict_handler.DictHandler method), 14

`load()` (kaptan.handlers.ini_handler.IniHandler method), 14

`load()` (kaptan.handlers.json_handler.JsonHandler method), 14

`load()` (kaptan.handlers.yaml_handler.YamlHandler method), 14

M

`main()` (in module kaptan), 14

U

`upsert()` (kaptan.Kaptan method), 13

Y

YamlHandler (class in kaptan.handlers.yaml_handler), 14