
coolfig Documentation

Release 3.1.0

Jonathan Stoppani

Aug 23, 2018

Contents

1	Installation	3
2	Quick start	5
3	Extending coolfig	7
3.1	Custom providers	7
3.2	Custom types	7
4	Django integration	9
5	API documentation	11
5.1	Module contents	11
5.2	Submodules	12
	Python Module Index	15

Coolfig is a library to easily write extensible configuration specifications to be fulfilled by various sources, such as config files or environment variables. It was written with the goal to make it easy to apply the [twelve-factor methodology](#).

Coolfig is MIT-licensed. Source code is available on [GitHub](#) and packages are distributed through [PyPI](#).

Contents:

CHAPTER 1

Installation

CHAPTER 2

Quick start

CHAPTER 3

Extending coolfig

3.1 Custom providers

3.2 Custom types

CHAPTER 4

Django integration

5.1 Module contents

Support for working with different sources of configuration values.

```
class DefaultSettings(schema.Settings): SECRET_KEY = schema.Value(str) DEBUG =
    schema.Value(types.boolean, default=False) DB_URL = schema.Value(types.sqlalchemy_url)
    LOCALES = schema.Value(types.list(str))

settings = DefaultSettings( providers.DictConfig(os.environ, prefix='MYAPP_'))

coolfig.computed_value (func)

class coolfig.DictConfig (conf_dict, prefix="")
    Bases: coolfig.providers.ConfigurationProvider

    Loads configuration values from the passed dictionary.

    get (key)

    iterprefixed (prefix)

class coolfig.Dictionary (spec)
    Bases: coolfig.schema.ValueBase

class coolfig.EnvDirConfig (base_path, prefix="")
    Bases: coolfig.providers.ConfigurationProvider

    get (key)

    iterprefixed (prefix)

coolfig.load_django_settings (provider, static_config, base=<class
    'coolfig.django.BaseDjangoSettings'>, apps=None, name=None)

class coolfig.Settings (config_provider)
    Bases: coolfig.schema.SettingsBase
```

classmethod `merge` (**others*)

Merge the *others* schema into this instance.

The values will all be read from the provider of the original object.

class `coolfig.Value` (*type*, *default*=<object object>, *key*=None)

Bases: `coolfig.schema.ValueBase`

5.2 Submodules

5.2.1 coolfig.django module

class `coolfig.django.BaseDjangoSettings` (*config_provider*)

Bases: `coolfig.schema.Settings`

ALLOWED_HOSTS

CACHES

DATABASES

DEBUG

LANGUAGE_CODE

SECRET_KEY

STATIC_URL

TIME_ZONE

USE_I18N

USE_L10N

USE_TZ

install (*name*=None)

load_apps (*apps*=None)

`coolfig.django.get_app_settings_path` (*app_path*)

`coolfig.django.load_django_settings` (*provider*, *static_config*, *base*=<class
'coolfig.django.BaseDjangoSettings'>, *apps*=None,
name=None)

`coolfig.django.make_django_settings` (*static_config*, *base*=<class
'coolfig.django.BaseDjangoSettings'>)

5.2.2 coolfig.providers module

class `coolfig.providers.ConfigurationProvider`

Bases: `object`

get (*key*)

iterprefixed (*prefix*)


```

class coolfig.providers.DictConfig (conf_dict, prefix="")
    Bases: coolfig.providers.ConfigurationProvider

    Loads configuration values from the passed dictionary.

    get (key)

    iterprefixed (prefix)

class coolfig.providers.EnvDirConfig (base_path, prefix="")
    Bases: coolfig.providers.ConfigurationProvider

    get (key)

    iterprefixed (prefix)

class coolfig.providers.FallbackProvider (providers)
    Bases: coolfig.providers.ConfigurationProvider

    get (key)

    iterprefixed (prefix)

```

5.2.3 coolfig.schema module

```

class coolfig.schema.BoundValue (cls, name, value)
    Bases: object

class coolfig.schema.ComputedValue (callable, *args, **kwargs)
    Bases: coolfig.schema.ValueBase

class coolfig.schema.DictValue (type, keytype=<type 'str'>, *args, **kwargs)
    Bases: coolfig.schema.Value

class coolfig.schema.Dictionary (spec)
    Bases: coolfig.schema.ValueBase

exception coolfig.schema.ImproperlyConfigured
    Bases: exceptions.Exception

    Raised when a request for a configuration value cannot be fulfilled.

class coolfig.schema.Reference (key)
    Bases: object

class coolfig.schema.Settings (config_provider)
    Bases: coolfig.schema.SettingsBase

    classmethod merge (*others)
        Merge the others schema into this instance.

        The values will all be read from the provider of the original object.

class coolfig.schema.SettingsBase (config_provider)
    Bases: object

    as_dict ()

    items ()

    keys ()

class coolfig.schema.SettingsMeta (name, bases, clsdict)
    Bases: type

```

```
class coolfig.schema.StaticValue (value)
    Bases: coolfig.schema.BoundValue

class coolfig.schema.Value (type, default=<object object>, key=None)
    Bases: coolfig.schema.ValueBase

class coolfig.schema.ValueBase
    Bases: object

coolfig.schema.bind_values (cls, clsdict)

coolfig.schema.computed_value (func)

coolfig.schema.ref
    alias of coolfig.schema.Reference
```

5.2.4 coolfig.types module

Common types for settings classes.

```
class coolfig.types.LazyCallable (module_path, callable_path)
    Bases: object

    func

coolfig.types.boolean (string)

coolfig.types.dottedpath (string)

coolfig.types.list (inner_type, sep=', ')
```

C

`coolfig`, 11

A

ALLOWED_HOSTS (coolfig.django.BaseDjangoSettings attribute), 12

as_dict() (coolfig.schema.SettingsBase method), 13

B

BaseDjangoSettings (class in coolfig.django), 12

bind_values() (in module coolfig.schema), 14

boolean() (in module coolfig.types), 14

BoundValue (class in coolfig.schema), 13

C

CACHES (coolfig.django.BaseDjangoSettings attribute), 12

computed_value() (in module coolfig), 11

computed_value() (in module coolfig.schema), 14

ComputedValue (class in coolfig.schema), 13

ConfigurationProvider (class in coolfig.providers), 12

coolfig (module), 11

coolfig.django (module), 12

coolfig.providers (module), 12

coolfig.schema (module), 13

coolfig.types (module), 14

D

DATABASES (coolfig.django.BaseDjangoSettings attribute), 12

DEBUG (coolfig.django.BaseDjangoSettings attribute), 12

DictConfig (class in coolfig), 11

DictConfig (class in coolfig.providers), 12

Dictionary (class in coolfig), 11

Dictionary (class in coolfig.schema), 13

DictValue (class in coolfig.schema), 13

dottedpath() (in module coolfig.types), 14

E

EnvDirConfig (class in coolfig), 11

EnvDirConfig (class in coolfig.providers), 13

F

FallbackProvider (class in coolfig.providers), 13

func (coolfig.types.LazyCallable attribute), 14

G

get() (coolfig.DictConfig method), 11

get() (coolfig.EnvDirConfig method), 11

get() (coolfig.providers.ConfigurationProvider method), 12

get() (coolfig.providers.DictConfig method), 13

get() (coolfig.providers.EnvDirConfig method), 13

get() (coolfig.providers.FallbackProvider method), 13

get_app_settings_path() (in module coolfig.django), 12

I

ImproperlyConfigured, 13

install() (coolfig.django.BaseDjangoSettings method), 12

items() (coolfig.schema.SettingsBase method), 13

iterprefixed() (coolfig.DictConfig method), 11

iterprefixed() (coolfig.EnvDirConfig method), 11

iterprefixed() (coolfig.providers.ConfigurationProvider method), 12

iterprefixed() (coolfig.providers.DictConfig method), 13

iterprefixed() (coolfig.providers.EnvDirConfig method), 13

iterprefixed() (coolfig.providers.FallbackProvider method), 13

K

keys() (coolfig.schema.SettingsBase method), 13

L

LANGUAGE_CODE (coolfig.django.BaseDjangoSettings attribute), 12

LazyCallable (class in coolfig.types), 14

list() (in module coolfig.types), 14

load_apps() (coolfig.django.BaseDjangoSettings method), 12

load_django_settings() (in module coolfig), 11

load_django_settings() (in module coolfig.django), 12

M

make_django_settings() (in module coolfig.django), 12

merge() (coolfig.schema.Settings class method), 13

merge() (coolfig.Settings class method), 11

R

ref (in module coolfig.schema), 14

Reference (class in coolfig.schema), 13

S

SECRET_KEY (coolfig.django.BaseDjangoSettings attribute), 12

Settings (class in coolfig), 11

Settings (class in coolfig.schema), 13

SettingsBase (class in coolfig.schema), 13

SettingsMeta (class in coolfig.schema), 13

STATIC_URL (coolfig.django.BaseDjangoSettings attribute), 12

StaticValue (class in coolfig.schema), 13

T

TIME_ZONE (coolfig.django.BaseDjangoSettings attribute), 12

U

USE_I18N (coolfig.django.BaseDjangoSettings attribute), 12

USE_L10N (coolfig.django.BaseDjangoSettings attribute), 12

USE_TZ (coolfig.django.BaseDjangoSettings attribute), 12

V

Value (class in coolfig), 12

Value (class in coolfig.schema), 14

ValueBase (class in coolfig.schema), 14