
django cache extension Documentation

Release 1.0.0

damao lee

July 27, 2016

1	content	3
1.1	Quick Start	3
2	Indices and tables	5

django cache extension is a extension package for cache use in django. django cache extension is now works with django 1.9 and python3.

1.1 Quick Start

1.1.1 config

1. Install cache extension by pip:

```
pip install django-cache-extension
```

2. For redis backend use cache like this:

```
config your cache file backend to cache_extension:

CACHES={
    "default": {
        'BACKEND': 'cache_extension.backends.redis.ExtensionRedisBackend',
        'LOCATION': 'redis://redis:6379/0',
        'TIMEOUT': '172800',
        'KEY_PREFIX': 'cache_extension',
        'OPTIONS': {
            'DB': 0,
            'CLIENT_CLASS': 'django_redis.client.DefaultClient',
            'PARSER_CLASS': 'redis.connection.HiredisParser',
            'PICKLE_VERSION': 2,
        }
    }
},
```

1.1.2 methods

get_model (*model*, *pk=None*, *cache_exc=False*, ***kwargs*)

Return django model in cache, query database when cache miss hit, raise Model.DoesNotExist when miss database.

Set cache_exc=True where cache model.DoesNotExist in cache, and return None.

get_model_list (*model*, ***kwargs*)

Get multiple models with filter on fields, return a list of models.

clear_model (*model*, **args*)

clear model cache on args, usually use id.

clear_model_list (*model*, **args*)

Clear model cache using field name.

clear_model_cache (*model*, **args*, ***kwargs*)

Call clear_model or clear_model_list dynamically according to params passed.

Indices and tables

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

C

`clear_model()`, 3
`clear_model_cache()`, 4
`clear_model_list()`, 3

G

`get_model()`, 3
`get_model_list()`, 3