
yadm

Release 1.3.1

February 21, 2017

1	Requirements	3
2	Quick start	5
2.1	CHANGES	6
2.1.1	1.3.1 (2017-02-21)	6
2.1.2	1.3.0 (2017-02-19)	6
2.1.3	1.2.1 (2017-01-19)	6
2.1.4	1.2.0 (2016-12-27)	6
2.1.5	1.1.4 (2016-08-20)	6
2.1.6	1.1.3 (2016-07-23)	6
2.1.7	1.1 (2016-04-26)	6
2.1.8	1.0 (2015-11-14)	7
3	API documentation	9
3.1	API	9
3.1.1	Database	9
3.1.2	Documents	9
3.1.3	Serializers and deserializers	9
3.1.4	Queryset	9
3.1.5	Bulk queries	9
3.1.6	Mongo Aggregation Framework	9
3.1.7	Join	9
3.1.8	Fields	9
	Base fields	9
	Simple fields	9
	Datetime field	9
	Decimal field	9
	Embedded documents fields	9
	Reference field	9
	Containers fields	9
	List fields	9
	Set field	10
	Map field	10
	Geo fields	10

It's small and simple ODM for use with MongoDB.

Requirements

YAMD support MongoDB version 3.x only. MongoDB 2.x is not supported.

Minimal version of python is 3.4.

Quick start

```
import pymongo
from yadm import Database, Document, fields

# Create model
class BlogPost(Document):
    __collection__ = 'blog_posts'

    title = fields.StringField()
    body = fields.StringField()

# Create post
post = BlogPost()
post.title = 'Small post'
post.body = 'Bla-bla-bla...'

# Connect to database
client = pymongo.MongoClient("localhost", 27017)
db = Database(client, 'test')

# Insert post to database
db.insert(post)

# Query posts
qs = db.get_queryset(BlogPost).find({'title': {'$regex': '^s.*'}})
assert qs.count() > 0

for post in qs:
    assert post.title.startswith('s')

# Query one post
post = db.get_queryset(BlogPost).find_one({'title': 'Small post'})

# Change post
post.title = 'Bla-bla-bla title'

# Save changed post
db.save(post)
```

CHANGES

1.3.1 (2017-02-21)

- Change raw data for Money;

1.3.0 (2017-02-19)

- **Add currency support to Money:**
 - Totally rewrite Money type. Now it is not subclass of Decimal;
 - Add storage for currencies: `yadm.fields.money.currency.DEFAULT_CURRENCY_STORAGE`;

1.2.1 (2017-01-19)

- Add `QuerySet.find_in` for \$in queries with specified order;

1.2.0 (2016-12-27)

- Drop MongoDB 2.X support;
- Objects for update and remove results;
- Use Faker instead fake-factory;

1.1.4 (2016-08-20)

- **Add some features to `:py:module:'Bulk <yadm.bulk>'`:**
 - `Bulk.update_one(document, **kw)`: method for add update one document in bulk;
 - `Bulk.find(query).update(**kw)`: update many documents by query;
 - `Bulk.find(query).upsert().update(**kw)`: upsert document;
 - `Bulk.find(query).remove(**kw)`: remove documents;

1.1.3 (2016-07-23)

- **Add `QuerySet.ids`** method for get only documents id's from queryset;
- **Add `Money.total_cents`** method and `Money.total_cents` classmethod;

1.1 (2016-04-26)

- **Add cacheing on queryset level and use it for** `ReferenceField`;
- Add mongo aggregation framework support;
- **Add `exc` argument to** `QuerySet.find_one` for raise specified exception if not found;
- **Add `multi` argument to** `QuerySet.remove`;

- Deprecate `QuerySet.find_one`
- Refactoring.

1.0 (2015-11-14)

- **Change document structure. No more bad `BaseDocument.__data__` attribute:**
 - `BaseDocument.__raw__`: raw data from mongo;
 - `BaseDocument.__cache__`: cached objects, casted with fields;
 - `BaseDocument.__changed__`: changed objects.
- **Changes api for custom fields:**
 - Not more need create field descriptors for every field;
 - `prepare_value` called only for setattr;
 - `to_mongo` called only for save objects to mongo;
 - `from_mongo` called only for load values from `BaseDocument.__raw__`;
 - Remove `Field.default` attribute. Use `Field.get_default` method;
 - Add `get_if_not_loaded` and `get_if_attribute_not_set` method;
 - By default raise `NotLoadedError` if field not loaded from projection;
- **Changes in `ReferenceField`:**
 - Raise `BrokenReference` if link is broken;
 - Raise `NotBindingToDatabase` if document not saved to database;
- `smart_null` keyword for `Field`;
- Fields in document must be instances (not classes!);
- Remove `ArrayContainer` and `ArrayContainerField`;
- Remove old `MapIntKeysField` and `MapObjectIdKeysField`. Use new `MapCustomKeysField`;
- Add `Database.update_one` method for run simple update query with specified document;
- Add `QuerySet.distinct`;
- `serialize.from_mongo` now accept *not_loaded* sequence with filed names who must mark as not loaded, *parent* and *name*;
- `serialize.to_mongo` do not call `FieldDescriptor.__set__`;
- **Fakers!** Subsystem for generate test objects;
- Tests now use `pytest`;
- And more, and more...

API documentation

API

API documentation

Database

Documents

Serializers and deserializers

Queryset

Bulk queries

Mongo Aggregation Framework

Join

Fields

Base fields

Simple fields

Datetime field

Decimal field

Embedded documents fields

Reference field

Containers fields

List fields

Set field

Map field

Geo fields