namedtupled Documentation

Release 0.3.2

brennv

November 20, 2016

Contents

namedtuples are immutable, performant and classy. namedtupled is a lightweight wrapper for recursively creating namedtuples from nested dicts, lists, json and yaml. Inspired by hangtwenty.

CHAPTER 1

Installation

```
pip install namedtupled
```

CHAPTER 2

Getting started

```
import namedtupled

data = {'binks': {'says': 'meow'}}
cat = namedtupled.map(data)

cat  # NT(binks=NT(says='meow'))

cat.binks.says  # 'meow'
```

CHAPTER 3

Usage

Create namedtuples with methods: *map*, *json*, *yaml*, *zip*, *env* and helper method *ignore*. Unpack nested namedtuples with *reduce*.

Optionally name namedtuples by passing a 'name' argument to any method, the default name is simply 'NT'.

```
data = {'binks': {'says': 'meow'}}
cat = namedtupled.map(data, name='Cat')

cat  # Cat(binks=NT(says='meow'))
```

3.1 map()

Recursively convert mappings like nested dicts, or lists of dicts, into to namedtuples.

args: mapping, name='NT'

From dict:

```
data = {'binks': {'says': 'meow'}}
cat = namedtupled.map(data)

cat.binks.says  # 'meow'
```

From list:

```
data = [{'id': 'binks', 'says': 'meow'}, {'id': 'cedric', 'says': 'prrr'}]
cats = namedtupled.map(data)

cats[1].says  # 'prrr'
```

3.2 reduce()

Recursively convert nested namedtuples to mappings.

args: obj

```
cat  # NT(binks=NT(says='meow'))

data = namedtupled.reduce(cat)
```

```
data  # {'binks': {'says': 'meow'}}
```

3.3 json()

Map namedtuples from json data.

args: data=None, path=None, name='NT'

Inline:

```
data = """{"binks": {"says": "meow"}}"""
cat = namedtupled.json(data)

cat.binks.says  # 'meow'
```

Or specify path for a json file:

```
cat = namedtupled.json(path='cat.json')

cat.binks.says  # 'meow'
```

3.4 yaml()

Map namedtuples from yaml data.

args: data=None, path=None, name='NT'

Inline:

```
data = """
binks:
  says: meow
"""
cat = namedtupled.yaml(data)

cat.binks.says  # 'meow'
```

Or specify path for a yaml file:

```
cat = namedtupled.yaml(path='cat.yaml')

cat.binks.says  # 'meow'
```

3.5 zip()

Map namedtuples given a pair of key, value lists.

args: keys=[], values=[], name='NT'

Example:

```
keys, values = ['id', 'says'], ['binks', 'prrr']
cat = namedtupled.zip(keys, values)

cat.says  # 'prrr'
```

3.6 env()

Returns a namedtuple from a list of environment variables. If not found in shell, gets input with *input* or *getpass*.

args: keys=[], name='NT', use_getpass=False

In shell:

```
export USERNAME="binks"
export APIKEY="c4tnip!"
```

Then in python:

```
variables = ['USERNAME', 'APIKEY']
env = namedtupled.env(variables)

env.USERNAME  # 'binks'
```

3.7 ignore()

Use ignore to prevent a mapping from being converted to a namedtuple.

args: mapping

Example usage:

```
data = {'binks': namedtupled.ignore({'says': 'meow'})}
cat = namedtupled.map(data)

cat.binks  # {'says': 'meow'}
```

CHAPTER 4

Alternatives

bunch and munch

CHAPTER 5

Development

Source on github. Issues and PRs welcome, tests run with:

```
pip install pytest pytest-cov pytest-datafiles
python -m pytest --cov=namedtupled/ tests
```

Edit the docs here.