
dicts Documentation

Release 0.3.1

Miguel Olivares

March 13, 2012

CONTENTS

1	First steps	3
1.1	Installation	3
2	User Guide	5
2.1	Dictionaries Usage	5
2.2	Coming Soon	6
	Python Module Index	7

Dicts is a collection of useful dictionaries that will make your life easier!

FIRST STEPS

1.1 Installation

To install dicts form Pypi:

```
easy_install dicts
```

or

```
pip install dicts
```

if you want to install it from source code:

```
python setup.py install
```


USER GUIDE

2.1 Dictionaries Usage

2.1.1 Dict

Dict.join(*dic*)

Add dic pairs to self.data

```
>>> d1 = Dict({1 : 7, 2 : [1,2], 3 : 'a'})
>>> d1.join({1 : 2, 2 : [3], 3 : 'b'})
{1: 9, 2: [1, 2, 3], 3: 'ab'}
```

Dict.map(*callable*)

Apply 'callable' function over all values.

```
>>> d = Dict({'1' : 1, '2' : 1})
>>> d.map(lambda x: x/3.0)
>>> d
{'1': 0.3333333333333333, '2': 0.3333333333333333}
```

classmethod Dict.fromrepetitions(*iterable*)

Create a dict whose keys are the members of the iterable and values are the number of times the key appears in the iterable.

```
>>> Dict.fromrepetitions([1,1,1,1,2])
{1: 4, 2: 1}
```

Dict.relookup(*pattern*)

Dictionary lookup with a regular expression. Return pairs whose key matches pattern.

```
>>> d = Dict({'1' : 1, '2' : 1, 'a' : 10, '11' : 20})
>>> d.relookup('d+')
[('1', 1), ('2', 1), ('11', 20)]
```

2.1.2 SortedDict

class dicts.SortedDict(*data={}, cmp=None, key=None, reverse=False*)

Dictionary that iterates over its elements.

```
>>> sd = SortedDict({'c' : (1,'b'), 'a' : (6,'c'), 'b' : (5,'a')}, key=lambda x: x[1][1])
>>> sd.items()
[('b', (5, 'a')), ('c', (1, 'b')), ('a', (6, 'c'))]
```

2.1.3 KeySortedDict

class dicts.**KeySortedDict** (*data={}, reverse=False*)
Dictionary sorted by key

```
>>> ksd = KeySortedDict({'c' : 1, 'a' : 6, 'b' : 5})
>>> ksd.items()
[('a', 6), ('b', 5), ('c', 1)]
>>> ksd.keys()
['a', 'b', 'c']
>>> ksd.values()
[6, 5, 1]
```

2.1.4 ValueSortedDict

class dicts.**ValueSortedDict** (*data={}, reverse=False*)
Dictionary sorted by value

```
>>> vsd = ValueSortedDict({'c' : 1, 'a' : 6, 'b' : 5})
>>> vsd.items()
[('c', 1), ('b', 5), ('a', 6)]
>>> vsd.keys()
['c', 'b', 'a']
>>> vsd.values()
[1, 5, 6]
```

2.1.5 NoCaseDict

class dicts.**NoCaseDict** (*data*)
Keys of the dictionary are case insensitive.

```
>>> ncd = NoCaseDict({'exAmPlE' : 1})
>>> 'example' in ncd
True
```

2.1.6 RegexpDict

class dicts.**RegexpDict** (*data={}*)
Keys are regular expressions.

```
>>> rd = RegexpDict({'\d+' : 1, '\w+' : 2})
>>> rd['1']
[1, 2]
>>> rd['a']
[2]
```

2.2 Coming Soon

2.2.1 BiDict

Keys and values are uniques.

PYTHON MODULE INDEX

d

`dicts`, 5