



- [首页](#)
- [入门教程](#)
- [练习题](#)
- [python教程](#)
- [django教程](#)
- [seo应用](#)
- [linux](#)
- [测试应用](#)
- [书籍推荐](#)

Search : 提交 热门知识点: [python 字符串](#)

[python](#)>>[软件测试自动化.python 自动化测试框架](#)>>python unittest单元测试方法和用例

python unittest单元测试方法和用例

python内部自带了一个单元测试的模块，pyUnit也就是我们说的：unittest

先介绍下unittest的基本使用方法：

1. import unittest
2. 定义一个继承自unittest.TestCase的测试用例类
3. 定义setUp和tearDown，在每个测试用例前后做一些辅助工作。
4. 定义测试用例，名字以test开头。
5. 一个测试用例应该只测试一个方面，测试目的和测试内容应很明确。主要是调用assertEqual、assertRaises等断言方法判断程序执行结果和预期值是否相符。
6. 调用unittest.main()启动测试
7. 如果测试未通过，会输出相应的错误提示。如果测试全部通过则不显示任何东西，这时可以添加-v参数显示详细信息。

下面是unittest模块的常用方法：

```
assertEqual(a, b)      a == b
assertNotEqual(a, b)   a != b
assertTrue(x)          bool(x) is True
assertFalse(x)         bool(x) is False
assertIs(a, b)         a is b      2.7
assertIsNot(a, b)      a is not b  2.7
assertIsNone(x)        x is None   2.7
assertIsNotNone(x)     x is not None 2.7
assertIn(a, b)         a in b      2.7
assertNotIn(a, b)      a not in b   2.7
assertIsInstance(a, b) isinstance(a, b) 2.7
assertNotIsInstance(a, b) not isinstance(a, b) 2.7
```

下面看具体的代码应用：

首先写了一个简单应用：

```
import random
import unittest

class TestSequenceFunctions(unittest.TestCase):

    def setUp(self):
        self.seq = range(10)

    def test_shuffle(self):
        # make sure the shuffled sequence does not lose any elements
        random.shuffle(self.seq)
        self.seq.sort()
```

```

self.assertEqual(self.seq, range(10))

# should raise an exception for an immutable sequence
self.assertRaises(TypeError, random.shuffle, (1,2,3))

def test_choice(self):
    element = random.choice(self.seq)
    self.assertTrue(element in self.seq)

def test_error(self):
    element = random.choice(self.seq)
    self.assertTrue(element not in self.seq)

if __name__ == '__main__':
    unittest.main()

```

下面是写了一个简单的应用，测试下面4个网址返回的状态码是否是200。

```

import unittest
import urllib

class TestUrlHttpcode(unittest.TestCase):
    def setUp(self):
        urlinfo =
['http://www.baidu.com', 'http://www.163.com', 'http://www.sohu.com', 'http://www.cnpythoner.com']
        self.checkurl = urlinfo

    def test_ok(self):
        for m in self.checkurl:
            httpcode = urllib.urlopen(m).getcode()
            self.assertEqual(httpcode, 200)

if __name__ == '__main__':
    unittest.main()

```

如果有的网址打不开，返回404的话，测试则会报错

如果有的网址打不开，返回404的话，测试则会报错

```

ERROR: test_ok (__main__.TestUrlHttpcode)
-----
Traceback (most recent call last):
  File "jay.py", line 12, in test_ok
    httpcode = urllib.urlopen(m).getcode()
  File "/usr/lib/python2.7/urllib.py", line 86, in urlopen
    return opener.open(url)
  File "/usr/lib/python2.7/urllib.py", line 207, in open
    return getattr(self, name)(url)
  File "/usr/lib/python2.7/urllib.py", line 462, in open_file
    return self.open_local_file(url)
  File "/usr/lib/python2.7/urllib.py", line 476, in open_local_file
    raise IOError(e.errno, e.strerror, e.filename)
IOError: [Errno 2] No such file or directory: 'fewfe.com'
-----

```

Ran 1 test in 1.425s

FAILED (errors=1)

也有其他的unittest方法，用于执行更具体的检查，如：

Method	Checks that	New in
assertAlmostEqual(a, b)	round(a-b, 7) == 0	
assertNotAlmostEqual(a, b)	round(a-b, 7) != 0	
assertGreater(a, b)	a > b	2.7
assertGreaterEqual(a, b)	a >= b	2.7

```
assertLess(a, b)      a < b      2.7
assertLessEqual(a, b) a <= b     2.7
assertRegexpMatches(s, re)      regex.search(s)      2.7
assertNotRegexpMatches(s, re)    not regex.search(s)    2.7
assertItemsEqual(a, b)           sorted(a) == sorted(b) and works with unhashable objs 2.7
assertDictContainsSubset(a, b)   all the key/value pairs in a exist in b      2.7
assertMultiLineEqual(a, b)       strings      2.7
assertSequenceEqual(a, b)        sequences    2.7
assertListEqual(a, b)            lists        2.7
assertTupleEqual(a, b)           tuples       2.7
assertSetEqual(a, b)             sets or frozensets 2.7
assertDictEqual(a, b)           dicts         2.7
assertMultiLineEqual(a, b)       strings      2.7
assertSequenceEqual(a, b)        sequences    2.7
assertListEqual(a, b)            lists        2.7
assertTupleEqual(a, b)           tuples       2.7
assertSetEqual(a, b)             sets or frozensets 2.7
assertDictEqual(a, b)           dicts         2.7
```

你可以用unittest模块的更多方法来做自己的单元测试。

平均得分4.77分（共13次打分）



相关文章：

发布：2013-5-30 21:49:29 | 分类：[软件测试自动化.python 自动化测试框架](#) | 评论：0 | 引用：0 | 浏览：

上一篇：[软件测试方法](#) 下一篇：[函数应用习题](#)

分享到：[QQ空间](#) [新浪微博](#) [腾讯微博](#) [人人网](#) [网易微博](#) 2

评论列表

发表评论：

名称(*)

邮箱

网站链接

验证(*) 19050

正文(*) (留言最长字数:1000)

☐ 记住我, 下次回复时不用重新输入个人信息

[URL] [URL2] [EMAIL] [EMAIL2] [B] [I] [U] [S] [QUOTE] [显示UBB表情>>](#)

©欢迎参与讨论, 请在这里发表您的看法、交流您的观点。

关于网站



扫描二维码：即可获得python精品教程，并且不定期地分享python, django 实战技巧

最近发表

- [函数关键字参数](#)
- [函数返回值](#)
- [cooklib](#)
- [如何学好python](#)
- [mysql连接代码教程](#)
- [exception 异常](#)
- [缩进格式](#)
- [apply函数](#)
- [gui](#)
- [django select related\(\) 外键查询](#)
- [python is](#)
- [dict 操作](#)
- [django 操作数据库](#)
- [获取第一个字符和最后一个字符](#)
- [提取文章的高频词](#)
- [字符串格式化输出](#)
- [列表迭代](#)
- [startswith endswith](#)
- [python 模块](#)
- [django template 语法](#)
- [动态创建类](#)
- [引用和拷贝](#)
- [re正则表达式](#)
- [mechanize自动登录程序](#)
- [str转int](#)
- [django template html](#)
- [编码设置](#)
- [重载运算符](#)
- [三引号](#)
- [base64](#)
- [web开发必备](#)
- [python 2.5](#)
- [序列化数据](#)
- [python运行](#)
- [ping](#)
- [基础知识](#)
- [list判断](#)
- [python 网络](#)
- [python 基础教程](#)
- [python 安装](#)
- [fabric 安装配置](#)
- [fabric 批量自动化运维管理](#)

老王python ® 版权所有 备案号：湘ICP备10012179号

[站长统计](#)