

蛇小狼

使用 **PYTHON** 的 **UNITTEST** 做测试
Posted on 2011-08-04 22:32 蛇小狼 阅读(4108) 评论(1) 编辑 收藏

文档地址<http://docs.python.org/library/unittest.html>

第一个例子直接引用文档中的例子，其他的简略的写了写结论性的东西，具体的参考python_testing_cookbook一书。

一、简单的例子





```
1 import random
2 import unittest
3
4 class TestSequenceFunctions(unittest.TestCase):
5
6     def setUp(self):
7         self.seq = range(10)
8
9     def test_shuffle(self):
10         # make sure the shuffled sequence does not lose any elements
11         random.shuffle(self.seq)
12         self.seq.sort()
13         self.assertEqual(self.seq, range(10))
14
15         # should raise an exception for an immutable sequence
16         self.assertRaises(TypeError, random.shuffle, (1,2,3))
17
18     def test_choice(self):
19         element = random.choice(self.seq)
20         self.assertTrue(element in self.seq)
21
22     def test_sample(self):
23         with self.assertRaises(ValueError):
24             random.sample(self.seq, 20)
25         for element in random.sample(self.seq, 5):
26             self.assertTrue(element in self.seq)
27
28 if __name__ == '__main__':
29     unittest.main()
```



实现一个继承unittest.TestCase的类，使用unittest.main()来测试该类中所有以test开头的测试用例。

简单的assert方法有如下：
self.assertEqual(100, value)
self.assertTrue(value == 0)
self.assertFalse(value > 0)
self.assertRaises(TypeError, value.convert_to_decimal)

在写测试用例的时候尽量使用assertEquals而不是assertTrue、assertFalse，当assertEquals失败的时候，会打印出比较的两方值更直观

self.fail([msg])会无条件的导致测试失败，不推荐使用。

下面的例子的写法是要避免发生：

< 2016年5月 >

日	一	二	三	四	五	六
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

昵称：蛇小狼
园龄：5年8个月
粉丝：9
关注：8
[+加关注](#)

搜索

- 常用链接
- [我的随笔](#)
[我的评论](#)
[我的参与](#)
[最新评论](#)
[我的标签](#)
- 我的标签

[mysql\(1\)](#)
[python djaong nginx fastcgi\(1\)](#)
[python 算法\(1\)](#)

随笔分类

[android\(1\)](#)
[Crimson Moon\(2\)](#)
[django\(1\)](#)
[my_work_2011\(11\)](#)
[my_work_2012\(1\)](#)
[mysql\(1\)](#)
[other\(4\)](#)
[python\(12\)](#)
[python谜题](#)
[欧拉计划](#)
[算法相关\(11\)](#)
[随笔\(4\)](#)

随笔档案

[2014年12月 \(1\)](#)
[2013年12月 \(1\)](#)
[2013年1月 \(9\)](#)
[2012年12月 \(1\)](#)



```
1 import unittest
2 class BadTest(unittest.TestCase):
3     def test_no_roman_numeral(self):
4         value = RomanNumeralConverter(None)
5         try:
6             value.convert_to_decimal()
7             self.fail("Expected a TypeError")
8         except TypeError, e:
9             pass
```



这个时候推荐使用assertRaises

二、使用setUp和tearDown

每次运行test方法时，先用setUp初始化程序，然后运行test方法，最后使用tearDown清理程序
def setUp(self):

```
    print "Creating a new RomanNumeralConverter..."
    self.cvt = RomanNumeralConverter()
```

def tearDown(self):

```
    print "Destroying the RomanNumeralConverter..."
    self.cvt = None
```

如果不同的方法需要不同的setUp和tearDown方法，那么只好写过个测试用例了。

三、使用比unittest.main()更详细的运行测试用例的方法

```
if __name__ == "__main__":
    suite = unittest.TestLoader().loadTestsFromTestCase( \
        RomanNumeralConverterTest)
    unittest.TextTestRunner(verbosity=2).run(suite)
```

python使用unittest.TestLoader().loadTestsFromTestCase方法获取RomanNumeralConverterTest的所有test方法
然后通过TextTestRunner来运行

四、运行部分测试方法



```
1 if __name__ == "__main__":
2     import sys
3     suite = unittest.TestSuite()
4     if len(sys.argv) == 1:
5         suite = unittest.TestLoader().loadTestsFromTestCase(\
6             RomanNumeralConverterTest)
7     else:
8         for test_name in sys.argv[1:]:
9             suite.addTest(\
10                 RomanNumeralConverterTest(test_name))
11     unittest.TextTestRunner(verbosity=2).run(suite)
```



五、一次运行多个测试用例



```
1 if __name__ == "__main__":
2     import unittest
3     from recipe5 import *
4     suite1 = unittest.TestLoader().loadTestsFromTestCase( \
```

2012年11月 (1)

2012年8月 (1)

2012年3月 (4)

2011年11月 (5)

2011年10月 (1)

2011年9月 (1)

2011年8月 (4)

2011年7月 (2)

2011年6月 (5)

2011年4月 (4)

2010年12月 (1)

2010年9月 (2)

最新评论

1. Re:使用python的unittest做测试

非常详细，好。

--朋友博客集

2. Re:nginx摘录

--summarize可以查看哪些文件更新
svn diff -r {2012-3-9}:HEAD --summarize

--蛇小狼

阅读排行榜

1. 使用python的unittest做测试(4108)

2. python包管理工具pip(1907)

3. Facebook接入（上）(857)

4. Facebook接入（下）(844)

5. yahooM充值接入(417)

评论排行榜

1. 使用python的unittest做测试(1)

2. nginx摘录(1)

推荐排行榜

1. 使用python的unittest做测试(2)

2. 数学基础的算法分析(1)

3. 我只是个程序员(1)

```
5         RomanNumeralConverterTest)
6 suite2 = unittest.TestLoader().loadTestsFromTestCase( \
7         RomanNumeralComboTest)
8 suite = unittest.TestSuite([suite1, suite2])
9 unittest.TextTestRunner(verbosity=2).run(suite)
```



六、在test model 中 使用test suites



```
1 def high_and_low():
2     suite = unittest.TestSuite()
3     suite.addTest(\
4         RomanNumeralConverterTest("test_parsing_millenia"))
5     suite.addTest(\
6         RomanNumeralConverterTest("test_parsing_one"))
7     return suite
8
9 def combos():
10    return unittest.TestSuite(map(RomanNumeralConverterTest,\
11        ["test_combo1", "test_combo2", "test_combo3"]))
12
13 def all():
14    return unittest.TestLoader().loadTestsFromTestCase(\
15        RomanNumeralConverterTest)
16
17 if __name__ == "__main__":
18     for suite_func in [high_and_low, combos, all]:
19         print "Running test suite '%s'" % suite_func.func_name
20         suite = suite_func()
21         unittest.TextTestRunner(verbosity=2).run(suite)
```



七、将已有的测试代码迁移到unittest

不使用unittest



```
1 from recipe7 import *
2
3 class RomanNumeralTester(object):
4     def __init__(self):
5         self.cvt = RomanNumeralConverter()
6
7     def simple_test(self):
8         print "+++ Converting M to 1000"
9         assert self.cvt.convert_to_decimal("M") == 1000
10
11    def combo_test1(self):
12        print "+++ Converting MMX to 2010"
13        assert self.cvt.convert_to_decimal("MMXX") == 2010
14
15    def combo_test2(self):
16        print "+++ Converting MMMMDCLXVIII to 4668"
17        val = self.cvt.convert_to_decimal("MMMMDCLXVII")
18        self.check(val, 4668)
19
20    def other_test(self):
21        print "+++ Converting MMMM to 4000"
22        val = self.cvt.convert_to_decimal("MMMM")
23        self.check(val, 4000)
24
```

```
25     def check(self, actual, expected):
26         if (actual != expected):
27             raise AssertionError("%s doesn't equal %s" % \
28                                 (actual, expected))
29
30     def test_the_system(self):
31         self.simple_test()
32         self.combo_test1()
33         self.combo_test2()
34         self.other_test()
35
36 if __name__ == "__main__":
37     tester = RomanNumeralTester()
38     tester.test_the_system()
```



迁移到unittest



```
1 from recipe7 import *
2 from recipe7_legacy import *
3 import unittest
4
5 if __name__ == "__main__":
6     tester = RomanNumeralTester()
7
8     suite = unittest.TestSuite()
9     for test in [tester.simple_test, tester.combo_test1, \
10                 tester.combo_test2, tester.other_test]:
11         testcase = unittest.FunctionTestCase(test)
12         suite.addTest(testcase)
13
14     unittest.TextTestRunner(verbosity=2).run(suite)
```



八、简单化

一个测试函数里面写一个assert***,便于查看bug

九、测试边缘

比如测试一个到4000为止的整数,需要测试

1,4000, 0 4001, -1, None, 1.2

错误的输入也是必要的

通过迭代测试边缘

[View Code](#)

分类: [python](#)

[好文要顶](#)[关注我](#)[收藏该文](#)

蛇小狼 +加关注

关注 - 8

粉丝 - 9

2

0

(请你对文章做出评价)

« 上一篇: [使用cmd模块,让脚本更舒服些](#)

» 下一篇: [svn手册摘录](#)

Feedback

1 楼

2012-03-29 11:02 by 朋友博客集

非常详细，好。

支持(0) 反对(0)

[刷新评论](#) [刷新页面](#) [返回顶部](#)注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#)网站首页。

【推荐】50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库

【推荐】融云即时通讯云—豆果美食、Faceu等亿级APP都在用

【推荐】Excel 报表开发18 招式，人人都能做报表

【推荐】阿里云万网域名：.xin .com将推出重磅优惠

【活动】高性能+大数据的HTML5 App开发交流会（深圳站）



最新IT新闻：

- 途牛网第一季度净亏损8370万美元 同比亏损扩大
 - 安全公司称40%的机器人都在为非作歹
 - 音乐和YouTube：相爱容易，相处太难
 - 爱奇艺拟2017年A股借壳上市？管理层需先解决亏损问题
 - 李开复：守护一个快乐摄影师
- » 更多新闻...



最新知识库文章：

- 一个前端的自我修养
 - 架构漫谈（九）：理清技术、业务和架构的关系
 - 架构漫谈（八）：从架构的角度看如何写好代码
 - 架构漫谈（七）：不要空设架构师这个职位，给他实权
 - 架构漫谈（六）：软件架构到底是要解决什么问题？
- » 更多知识库文章...

Copyright Copyright 蛇小狼