



链滴

使用 unittest 和 HTMLTestRunner, 编写 简单百度搜索自动化测试用例

作者: [abcma](#)

原文链接: <https://ld246.com/article/1566210397373>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

百度页面搜索自动化测试用例：

```
# coding=utf-8
import unittest, time
from selenium import webdriver

# 定义测试类，父类为unittest.TestCase
class TestBaidu(unittest.TestCase):

# 定义setUp()方法用于测试用例执行前的初始化工作。
    def setUp(self):
        self.driver = webdriver.Chrome()
        self.driver.implicitly_wait(30)
        self.base_url = 'http://www.baidu.com'
        self.driver.maximize_window()

# 定义测试用例，以“test_”开头命名的方法
#注意，方法的入参为self
#可使用unittest.TestCase类下面的各种断言方法用于对测试结果的判断
    def test_baidu(self):
        driver = self.driver
        driver.get(self.base_url + '/')
        driver.find_element_by_id('kw').clear()
        driver.find_element_by_id('kw').send_keys('unittest')
        driver.find_element_by_id('su').click()
        time.sleep(3)
        title = driver.title
        self.assertEqual(title, 'unittest_百度搜索')

# 定义tearDown()方法用于测试用例执行之后的善后工作，例如：关闭浏览器等
    def tearDown(self):
        self.driver.quit()
        print('百度自动化测试通过')

# 如果直接运行该文件(__name__值为__main__)，则执行以下语句，常用于测试脚本是否能够正常运行
if __name__ == "__main__":
    unittest.main()
```

使用HTMLTestRunner模块，导出自动化测试报告

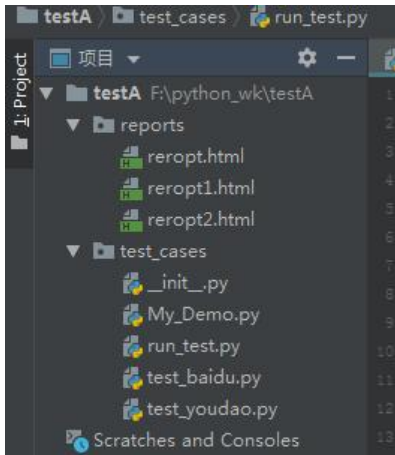
```
from test_cases import test_baidu
import unittest
import HTMLTestRunnerCN

# 确定生成报告的路径
filepath = "../reports/reropt2.html"
fp = open(filepath, 'wb')

# 定义一个单元测试容器
suite = unittest.TestSuite()
#将测试用例加入到容器
suite.addTest(test_baidu.TestBaidu('test_baidu'))
```

```
#生成报告的Title,描述
runner = HTMLTestRunnerCN.HTMLTestReportCN(stream=fp, title='自动化测试报告')
#运行测试用例
runner.run(suite)
```

文件目录:



测试报告浏览:

自动化测试报告

测试人员: QA
 开始时间: 2019-08-19 08:54:19
 合计耗时: 0:00:33.456000
 测试结果: 共 2, 通过 2, 通过率= 100.00%

概要(100.00%) | 通过(2) | 失败(0) | 错误(0) | 所有(2)

用例集/测试用例	总计	通过	失败	错误	详细
test_cases.test_baidu.TestBaidu	1	1	0	0	收起
test_baidu					通过 pt1_1: 未发现unittest_百度搜索 百度自动化测试通过
test_cases.test_youdao.TestYoudao	1	1	0	0	收起
总计	2	2	0	0	通过率: 100.00%

HTMLTestRunner中文版下载安装, 请留言。