

使用spyder3调试python程序的简明教程



燃烧杯

关注

赞赏支持

使用spyder3调试python程序的简明教程



燃烧杯

关注



0.278

2018.01.21 00:24:35 字数 1,973 阅读 42,751

说是简明教程，其实是我自己尝试用spyder调试python过程的一个记录，因为spyder的调试功能是基于pdb，而我又没有pdb的基础，所以刚开始上手时感觉很不习惯，而且那时我又很懒，没去找官方文档，仅仅在百度和csdn上找了找，没找到比较好的资料，于是放弃了，过了一段时间之后，突然又心血来潮去找了官方文档，外加自己的一些尝试，总算入门了spyder的调试功能，特地记录下来与大家共享,我使用的spyder版本是3.1.4(使用pip list命令查看spyder版本)

Spyder官方文档地址

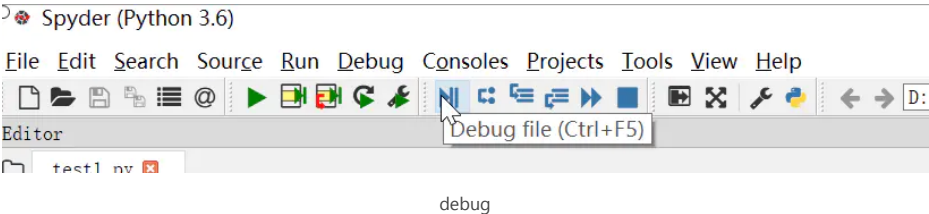
<http://pythonhosted.org/spyder/>

开始调试

先写一个简单的小程序用于调试：

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Mon Aug 28 23:59:40 2017
4
5 @author: 燃烧杯
6 """
7
8 a = 'a'
9 b = 'b'
10 c = 'c'
11 e = 'e'
12 f = 'f'
13 g = 'g'
14 h = 'h'
15 print(a)
```

我们暂时先不打断点，用debug的方式运行该代码试试



点击spyder工具栏上的Debug file按钮，或者使用快捷键Ctrl+F5开始调试。

在ipython界面会输出如图所示的内容：



写下你的评论...

评论0

赞19

...

推荐阅读

Python Matplotlib 设置坐标轴刻度以10为底

阅读 399

玩转 PyCharm，这篇文章就够了

阅读 1,516

第四章 可解释的模型

阅读 69

Pandas从入门到精通（5）- 变形

阅读 25

Cython系列教程：二. 构建Cython代码

阅读 70

19个Python库来帮你写Python脚本

阅读 301

使用spyder3调试python程序的简明教程



燃烧杯

关注

赞赏支持

```
4
5 @author: 燃烧杯
---> 6 ""
7
```

first debug

出现了ipdb提示符，说明我们已经进入了调试模式，上面输出的内容可以看出是代码的第一行，接着在提示符中输入c(continue的缩写，表示程序继续向下执行到下一个断点)，会输出如下内容：

```
In [6]: debugfile('D:/IDE/pyProject/pdbTest/test1.py', wdir='D:/IDE/
pyProject/pdbTest')
> d:\ide\pyproject\pdbtest\test1.py(6)<module>()
4
5 @author: 燃烧杯
---> 6 ""
7
8 a = 'a'

ipdb> c
a

In [7]:
```

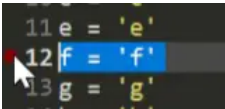
first_debug_end

程序执行结束，可见即使我们没有打断点，仍然会在第一句执行之前中断（经测试，中断的时候第一句还没有执行），这个和我用过的其他编译器不太一样（比如eclipse和IntelliJ，在不打断点的情况下会正常执行到底），一开始还让我困惑了一下，后来就适应了。如果你仔细看刚才的工具栏截图的话，会发现在debug按钮组的第五个按钮和刚刚的'c'命令是一样的功能，但是不知道为什么，在我这个版本的spyder里有这个按钮一些bug（具体来说就是在程序执行结束之后不会自动退出pdb，而且之后再想使用'q'命令退出也退出了不了，换言之，就是卡死在了pdb里面），如果你使用的是更高版本的spyder的话，这个bug可能已经修复了，可以尝试一下。

打断点的两种姿势

普通的breakpoint

用spyder打断点的方法非常简单，只要在想打断点的那一行行首双击鼠标即可，如图所示，我们尝试建立一个断点：



break_point

在每次开始debug之前，先在spyder的ipython界面中输入 %reset 把工作空间的所有变量清除，以免影响到我们接下来的测试。
按下Ctrl+F5开始debug，进行如图所示的操作：



推荐阅读

Python Matplotlib 设置坐标轴刻度以10为底
阅读 399

玩转 PyCharm ， 这篇文章就够了
阅读 1,516

第四章 可解释的模型
阅读 69

Pandas从入门到精通（5）- 变形
阅读 25

Cython系列教程：二. 构建Cython代码
阅读 70

使用spyder3调试python程序的简明教程



燃烧杯

关注

赞赏支持

```
4
5 @author: 燃烧杯
6 """
7
8 a = 'a'

ipdb> c
> d:\ide\pyproject\pdbtest\test1.py(12)<module>()
10 c = 'c'
11 e = 'e'
3--> 12 f = 'f'
13 a = 'a'
```

to_breakpoint

然后我们就到达了断点处，从箭头(-->)以及 `d:\ide\pyproject\pdbtest\test1.py(12)<module>()` 中的数字12可以看出程序刚刚执行到了第12行（也就是我们打断点的这一行），第12行到底有没有执行呢？只要测试一下f变量是否存在就可以了，尝试在ipdb中进行如下输入：

```
ipdb> !f
*** NameError: name 'f' is not defined

ipdb> !e
'e'

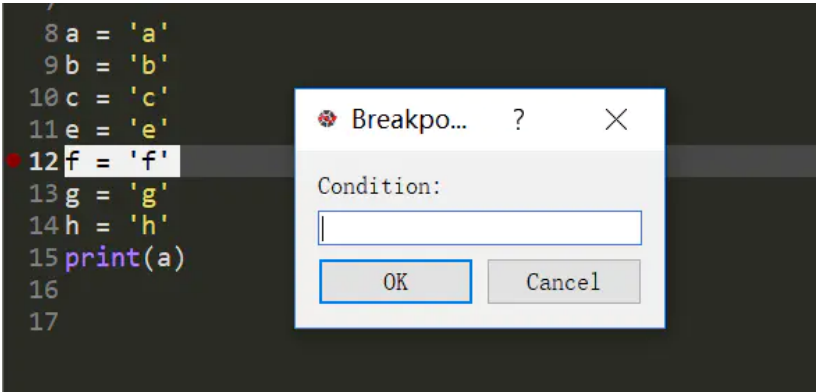
ipdb> |
```

ipdb

`!(python语句)` 的意思就是在当前状态下执行该python语句，我刚刚的用法的意思显然是查看变量内容，从 `!f` 的错误信息可以看出f尚未定义，即第12行代码(`f='f'`)还没有执行，查看e变量发现e变量已经被定义了，这说明第11行已经执行结束了。通过以上实验可以看出，spyder会在断点语句的执行之前中断

带条件的breakpoint

双击刚刚在第12行代码开头创建的“小红点”即可取消断点。
按住Ctrl+Shift，然后像刚才一样双击第12行行首，会弹出一个小框：



condition

在这个小框内可以输入断点的条件，可以是任意返回True或False的python语句，比如我输入

写下你的评论...

评论0

赞19



使用spyder3调试python程序的简明教程



燃烧杯

关注

赞赏支持

然后点击调试按钮，发现断点多了几个，这个表示条件断点(conditional breakpoint)，开始debug试一下。

```
In [31]: debugfile('D:/IDE/pyProject/pdbTest/test1.py', wdir='D:/IDE/
pyProject/pdbTest')
> d:\ide\pyproject\pdbtest\test1.py(6)<module>()
4
5 @author: 燃烧杯
----> 6 """
7
8 a = 'a'
```

```
ipdb> c
a
```

```
In [32]:
```

debug

发现程序只在第一句中中断一下，断点根本就没有起作用，因为在断点的时候，a变量为'a'，b变量为'b'，不符合条件当然不会中断。

现在重新开始debug，然后连续按三遍Ctrl+F10，然后发现程序执行到了第十行：

```
In [33]: debugfile('D:/IDE/pyProject/pdbTest/test1.py', wdir='D:/IDE/
pyProject/pdbTest')
> d:\ide\pyproject\pdbtest\test1.py(6)<module>()
4
5 @author: 燃烧杯
----> 6 """
7
8 a = 'a'

ipdb> > d:\ide\pyproject\pdbtest\test1.py(8)<module>()
6 """
7
----> 8 a = 'a'
9 b = 'b'
10 c = 'c'

> d:\ide\pyproject\pdbtest\test1.py(9)<module>()
7
8 a = 'a'
----> 9 b = 'b'
10 c = 'c'
11 e = 'e'

ipdb> > d:\ide\pyproject\pdbtest\test1.py(10)<module>()
8 a = 'a'
9 b = 'b'
----> 10 c = 'c'
11 e = 'e'
3 12 f = 'f'

ipdb> |
```

ctrl_f10

其实Ctrl+F10是单行执行的意思，每按一次执行一行，相当于点击了工具栏上如下图所示的按钮。

写下你的评论...

评论0

赞19

推荐阅读

Python Matplotlib 设置坐标轴刻度以10为底

阅读 399

玩转 PyCharm，这篇文章就够了

阅读 1,516

第四章 可解释的模型

阅读 69

Pandas从入门到精通 (5) - 变形

阅读 25

Cython系列教程：二. 构建Cython代码

阅读 70



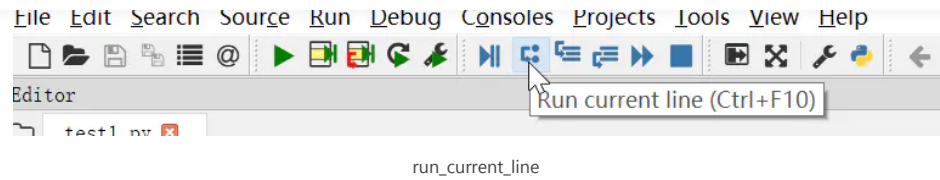
使用spyder3调试python程序的简明教程



燃烧杯

关注

赞赏支持



这个时候我可以使用刚刚讲过的 `!(python语句)` 来给a,b临时指定一个值，在ipdb的提示符中输入 `!a=4;b=5`，然后使用c命令继续执行，发现在条件断点处中断了，因为此时满足了我们刚刚给条件断点指定的条件：

```
ipdb> !a=4;b=5

ipdb> c
> d:\ide\pyproject\pdbtest\test1.py(12)<module>()
   10 c = 'c'
   11 e = 'e'
3--> 12 f = 'f'
   13 g = 'g'
   14 h = 'h'

ipdb> |
```

如果忘记了条件断点的条件是什么的话，可以按住Ctrl+Shift键双击“带问号的小红点”，然后就能看见条件是什么了，而且还可以修改条件，如果要取消断点的话，直接双击就可以了。

总结一下刚刚所讲的

- `Ctrl+F5` 以Debug模式运行文件
- 在debug之前记得用 `%reset` 指令清空一下ipython工作空间中的变量，以免影响debug中变量值的查看
- 无论你是否打断点，都会在第一行语句执行之前中断一次
- `!(python语句)`可以在pdb提示符下执行python语句，可以用来查看变量值或者给变量临时指定值
- c命令或者 `Ctrl+F12` 可以让程序执行到下一个断点
- q命令退出调试
- `Ctrl+F10` 单行执行
- 双击行首设置断点，按住 `Ctrl+Shift` 双击行首可以设置条件断点

剩下的一些细节

上面的例子已经包括了大多数常用的功能，如果曾经用过别的编译器的调试功能的话（如eclipse和IntelliJ等），看到这里就可以了，对于有调试经验的人来说，我下面要讲的两个功能只要看到按钮的名称就大概知道它是做什么的了。
如下：



写下你的评论...

评论0

赞19

推荐阅读

- Python Matplotlib 设置坐标轴刻度以10为底
阅读 399
- 玩转 PyCharm，这篇文章就够了
阅读 1,516
- 第四章 可解释的模型
阅读 69
- Pandas从入门到精通（5）- 变形
阅读 25
- Cython系列教程：二. 构建Cython代码
阅读 70

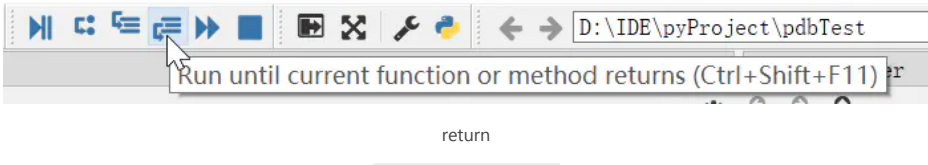
使用spyder3调试python程序的简明教程



燃烧杯

关注

赞赏支持



Step Into

上面一张图的按钮的功能我们称之为Step Into(下面一张图的按钮的功能我们称之为return)，用于进入一个函数体内部，为了更清楚的说明它的功能，我们给出一个例子，在spyder中创建如下程序：

```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Tue Aug 29 14:22:46 2017
4
5  @author: 燃烧杯
6  """
7
8  def myTest():
9      c = 'a'
10     d = 'b'
11     e = 'c'
12     return c
13
14 a = 'a'
15 b = 'b'
16 c = myTest()
17 f = 'f'
18 print(a)
```

我们开始debug，不断地按 **Ctrl+F10** 单行执行这个程序，当运行到 `c = myTest()` 这句时注意一下：



不管myTest()中有多少代码都直接当做一行跳了过去，用q命令退出调试。
重新debug该文件，单行执行到 `c = myTest()` 这行时按 **Ctrl+F11** 使用Step Into功能，发现我们进入了函数内部的代码段：



推荐阅读

- Python Matplotlib 设置坐标轴刻度以10为底
阅读 399
- 玩转 PyCharm ， 这篇文章就够了
阅读 1,516
- 第四章 可解释的模型
阅读 69
- Pandas从入门到精通（5）- 变形
阅读 25
- Cython系列教程：二. 构建Cython代码
阅读 70

使用spyder3调试python程序的简明教程



燃烧杯

关注

赞赏支持

推荐阅读

Python Matplotlib 设置坐标轴刻度以10为底

阅读 399

玩转 PyCharm , 这篇文章就够了

阅读 1,516

第四章 可解释的模型

阅读 69

Pandas从入门到精通 (5) - 变形

阅读 25

Cython系列教程: 二. 构建Cython代码

阅读 70

step_into

这就Step Into的主要功能.

Return

Return的功能与Step Into的功能刚好相反, 当使用Step Into进入函数之后, 按 **Ctrl+Shift+F11** 后会直接跳到该函数的执行的最后一行, 此时在按一遍 **Ctrl+Shift+F11** 或者 **Ctrl+F10** (单行执行) 就可以跳出函数了, 想要尝试的话可以自行在我上面给出的例子中尝试.

End

感谢阅读, 希望世界上的bug越来越少 (手动滑稽)

个人博客地址

19人点赞>

聊聊技术

"小礼物走一走, 来简书关注我"

赞赏支持

还没有人赞赏, 支持一下



燃烧杯 烫烫烫

总资产8 (约0.40元) 共写了7.3W字 获得256个赞 共127个粉丝

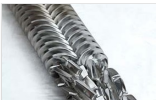
关注



python程序



python 培训



穿墙对拉螺杆



大电流连接器



CAD教程



python入门

被以下专题收入, 发现更多相似内容



Python



程序员

推荐阅读

更多精彩内容

写下你的评论...

评论0

赞19

使用spyder3调试python程序的简明教程



燃烧杯

关注

赞赏支持

myeclipse转idea开发备注

1.输入psv就会看到一个psvm的提示，此时点击tab键一个main方法就写好了。psvm 也就是public ...



兜兜又喊阿兵哥 阅读 592 评论 0 赞 0

VC调试以及TRACE()函数的用法

1.VC调试入门 1设置 为了调试一个程序，首先必须使程序中包含调试信息。一般情况下，一个从AppWizard 创建...



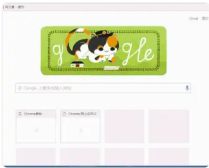
沈小浪 阅读 1,120 评论 0 赞 1

领略浏览器开发者工具

前言 相信无论是对于身居一线的coder,还是退居多年的老司机,managers来说,对于调试程序是不陌生的,对于w...



itclanCoder 阅读 1,813 评论 0 赞 7



常用的前端调试工具

在做前端开发时，我们需要用到一些调试工具用来调试我们的HTML、CSS或者JS 代码，俗话说预先善其事必先利其器，这...



Rella7 阅读 16,116 评论 0 赞 14



推荐阅读

Python Matplotlib 设置坐标轴刻度以10为底

阅读 399

玩转 PyCharm ， 这篇文章就够了

阅读 1,516

第四章 可解释的模型

阅读 69

Pandas从入门到精通（5）- 变形

阅读 25

Cython系列教程：二. 构建Cython代码

阅读 70



写下你的评论...

评论0

赞19