

# PCS4 常用自省

## 概述

---

自省，简单来说就是自我检查。Python 提供这个强大的功能，以方便程序员查看各个对象的信息。常用自省函数有 `help()`、`dir()`、`type()`、`id()` 等。

## 应用

---

### help()

用来查看很多 Python 自带的帮助文档信息，可以使用 `help()`。

```
~$ ipython
Python 2.5.2 (r252:60911, May 7 2008, 15:19:09)
Type "copyright", "credits" or "license" for more information.

IPython 0.8.2 -- An enhanced Interactive Python.
?          -> Introduction and overview of IPython's features.
%quickref  -> Quick reference.
help       -> Python's own help system.
object?    -> Details about 'object'. ?object also works, ?? prints more.

In [1]: help("str")

In [2]: import pygments
```

```
In [3]: help("pygments")
```

如上所示，在 Python 交互环境中，输入 `help("obj")` 后即可看到 `obj` 的帮助信息。就像 `bash` 中 `man` 随机帮助页一样。另外，也可以直接通过 `help()` 进入如下帮助实用程序：

```
In [4]: help()
```

```
Welcome to Python 2.5! This is the online help utility.
```

```
If this is your first time using Python, you should definitely check out
the tutorial on the Internet at http://www.python.org/doc/tut/.
```

```
Enter the name of any module, keyword, or topic to get help on writing
Python programs and using Python modules. To quit this help utility and
return to the interpreter, just type "quit".
```

```
To get a list of available modules, keywords, or topics, type "modules",
"keywords", or "topics". Each module also comes with a one-line summary
of what it does; to list the modules whose summaries contain a given word
such as "spam", type "modules spam".
```

```
help> keywords
```

```
Here is a list of the Python keywords. Enter any keyword to get more help.
```

and	elif	if	print
as	else	import	raise
assert	except	in	return
break	exec	is	try
class	finally	lambda	while
continue	for	not	with
def	from	or	yield
del	global	pass	

## dir()

`dir()` 可以列出对象的所有属性，如下：

```
In [6]: import os
```

```
In [7]: dir(os)
```

```
Out[7]:
['EX_CANTCREAT',
 'EX_CONFIG',
 'EX_DATAERR',
```

```
'EX_IOERR',  
'EX_NOHOST',  
'EX_NOINPUT',  
.....
```

## type()

type() 返回对象的类型，如下：

```
In [8]: type(os)  
Out[8]: <type 'module'>
```

## id()

id() 返回对象的“唯一序号”。其中，对于引用对象来说，返回的是被引用对象的 id()。

```
In [29]: a = 'abc'  
  
In [30]: b = a  
  
In [31]: id(a)  
Out[31]: 137472992  
  
In [32]: id(b)  
Out[32]: 137472992
```

## hasattr()和 getattr()

hasattr() 和 getattr() 分别判断对象是否有某个属性及获得某个属性值。

```
In [35]: hasattr(a, 'split')  
Out[35]: True  
  
In [36]: getattr(a, 'split')  
Out[36]: <built-in method split of str object at 0x831abe0>
```

## callable()

callable() 判断对象是否可以被调用。

```
In [37]: callable(a)  
Out[37]: False  
  
In [38]: callable(a.split)  
Out[38]: True
```

## isinstance()

isinstance() 可以确认某个变量是否有某种类型。

```
In [44]: isinstance(a, str)
```

```
Out[44]: True
```

```
In [45]: isinstance(a, int)
```

```
Out[45]: False
```

## 小结

---

以上这些都是常用的自省方法，还有其他的可参见相关资源：

- 自省的威力：

*[http://www.woodpecker.org.cn/diveintopython/power\\_of\\_introspection/index.html](http://www.woodpecker.org.cn/diveintopython/power_of_introspection/index.html)*

精巧地址：*<http://bit.ly/2cYq5p>*

- Python 自省指南：*<http://www.ibm.com/developerworks/cn/linux/l-pyint/index.html>*

精巧地址：*<http://bit.ly/32ZEgp>*

- Python 学习笔记- 2.自省：*<http://www.xwy2.com/article.asp?id=106>*

精巧地址：*<http://bit.ly/J4LZ5>*