PCS6　Python 与中文

概述

中文编码总是个难题。不过 Python 可以完美地支持中文，不管你的字符编码是 GB2312，GBK 或 UTF-8。在这里，不准备讲述有关 ASCII/Unicode 编码和中文编码问题，这两部分内容可以分别参考字符编码介绍和汉字编码介绍：

字符编码介绍：*http://www.ruanyifeng.com/blog/2007/10/ascii_unicode_and_utf-8.html*

精巧地址：*http://bit.ly/2BEWnC*

汉字编码介绍：*http://www.css8.cn/css8_document/gb2312.htm*

精巧地址：*http://bit.ly/2Oxpp7*

下面介绍 Python 中的中文编码问题。

应用

Python 中文编码转换，主要有以下 4 个对象来处理字符串转换操作等事项。

```
class str(basestring)
 |  str(object) -> string
 |
 |  Return a nice string representation of the object.
 |  If the argument is a string, the return value is the same object.
......
class unicode(basestring)
```

```
 |  unicode(string [, encoding[, errors]]) -> object
 |
 |  Create a new Unicode object from the given encoded string.
 |  encoding defaults to the current default string encoding.
 |  errors can be 'strict', 'replace' or 'ignore' and defaults to 'strict'.
......
encode(...)
   S.encode([encoding[,errors]]) -> object

   Encodes S using the codec registered for encoding. encoding defaults to
the default encoding. errors may be given to set a different error handling
scheme. Default is 'strict' meaning that encoding errors raise a
UnicodeEncodeError. Other possible values are 'ignore', 'replace' and
'xmlcharrefreplace' as well as any other name registered with
codecs.register_error that is able to handle UnicodeEncodeErrors.
decode(...)
   S.decode([encoding[,errors]]) -> object

   Decodes S using the codec registered for encoding. encoding defaults to
the default encoding. errors may be given to set a different error handling
scheme. Default is 'strict' meaning that encoding errors raise a
UnicodeDecodeError. Other possible values are 'ignore' and 'replace' as well
as any other name registerd with codecs.register_error that is able to handle
UnicodeDecodeErrors.
```

str 表示将某对象转换成字符串表示。

unicode 表示将某个字符串按照某个编码方式转换为一个 unicode 对象，unicode 是内部编码，世界上所有字符的统一编码集，即每个字符都有唯一的二进制表示方法，可以在*http://www.unicode.org/*上查到某个字符的 unicode 码。

在 Python 中，如果要正确显示中文字符，简单做法就是把字符串先转化为 unicode，然后再由 unicode 对象转化成任意其他系统可以显示的编码。如果你的操作系统是 GNU/Linux 的，只能正确显示 UTF-8，而你要显示的文本是从一个 Windows 系统下过来的 GB2312 编码，那么你要做的就是将 GB2312 转化成 unicode，然后由 unicode 转化成 UTF-8，这样就可以正确显示了。

```
In [26]: f = open('test')

In [27]: line = f.readline()

In [28]: uline = unicode(line, 'gb2312')

In [29]: uline
```

```
Out[29]: u'lines: 0.838643\n'

In [30]: u8line = uline.encode('utf-8')

In [31]: u8line
Out[31]: 'lines: 0.838643\n'
```

encode 将对象按照指定编码进行编码。

decode 将对象按照指定编码进行解码。

```
In [6]: s = "你好"

In [7]: s.decode("gbk")
Out[7]: u'\u6d63\u72b2\u30bd'

In [8]: s.decode("gbk").encode("utf-8")
Out[8]: '\xe6\xb5\xa3\xe7\x8a\xb2\xe3\x82\xbd'
```

或者

```
In [12]: s = "你好"

In [13]: unicode(s, "gbk")
Out[13]: u'\u6d63\u72b2\u30bd'

In [14]: unicode(s, "gbk").encode("utf-8")
Out[14]: '\xe6\xb5\xa3\xe7\x8a\xb2\xe3\x82\xbd'
```

小结

字符编码是个比较困扰人的问题，因为世界上有太多不同的编码方式，如果全世界统一用一个字符集和一种编码实现方式，那么就不会有这么多问题了，但这是不可能的。有关字符编码的知识还可以在下面的链接中找到更多。

- 谈谈 Unicode 编码：*http://www.pconline.com.cn/pcedu/empolder/gj/other/0505/616631.html*
  精巧地址：*http://bit.ly/17VlhU*
- 字符编码笔记：ASCII，Unicode 和 UTF-8：
  http://www.ruanyifeng.com/blog/2007/10/ascii_unicode_and_utf-8.html
  精巧地址：*http://bit.ly/2BEWnC*

- 汉字编码问题：*http://www.css8.cn/css8_document/gb2312.htm*
  精巧地址：*http://bit.ly/2Oxpp7*
- 如何解决 Python 中文编码问题：*http://webclipping.com.cn/2007/12/18/如何解决Python-中文编码问题/*
  精巧地址：*http://bit.ly/vgeZ*