[**[转]python 模块 chardet下载及介绍**](http://www.cnblogs.com/themost/p/7210941.html)

来源：<http://blog.csdn.net/tianzhu123/article/details/8187470/>

在处理字符串时，常常会遇到不知道字符串是何种编码，如果不知道字符串的编码就不能将字符串转换成需要的编码。面对多种不同编码的输入方式，是否会有一种有效的编码方式？chardet是一个非常优秀的编码识别模块。

chardet 是python的第三方库，需要下载和安装。下载的地址有：

1.推荐地址： <http://download.csdn.net/download/aqwd2008/4256178>

2.官方地址： <http://pypi.python.org/pypi/chardet>

如果采用源代码安装方法，有可能会提示缺少setuptools这个模块。因此这里我们采用另外一种更为方便的安装方法，不论你选择哪种安装包，将它解压得到其中的文件夹【chardet】将这个文件夹复制到【python安装根目录\Lib\site-packages】下，确保这个位置可以被python引用到。如果不能被引用到请加入环境变量。

在安装完chardet模块，我就可以使用它了，来看一段示例代码。

**[python]** [view plaincopy](http://blog.csdn.net/aqwd2008/article/details/7506007)

1. **import** chardet
2. **import** urllib
4. #可根据需要，选择不同的数据
5. TestData = urllib.urlopen('http://www.baidu.com/').read()
6. **print** chardet.detect(TestData)
8. 运行结果：
9. {'confidence': 0.99, 'encoding': 'GB2312'}

运行结果表示有99%的概率认为这段代码是GB2312编码方式。

另外一个相对高级的应用。

**[python]** [view plaincopy](http://blog.csdn.net/aqwd2008/article/details/7506007)

1. **import** urllib
2. **from** chardet.universaldetector **import** UniversalDetector
3. usock = urllib.urlopen('http://www.baidu.com/')
4. #创建一个检测对象
5. detector = UniversalDetector()
6. **for** line **in** usock.readlines():
7. #分块进行测试，直到达到阈值
8. detector.feed(line)
9. **if** detector.done: **break**
10. #关闭检测对象
11. detector.close()
12. usock.close()
13. #输出检测结果
14. **print** detector.result
16. 运行结果：
17. {'confidence': 0.99, 'encoding': 'GB2312'}

应用背景，如果要对一个大文件进行编码识别，使用这种高级的方法，可以只读一部，去判别编码方式从而提高检测速