python 编码转换

主要介绍了python的编码机制，unicode, utf-8, utf-16, GBK, GB2312,ISO-8859-1 等编码之间的转换。

常见的编码转换分为以下几种情况：

自动识别 字符串编码

可以使用 chardet 模块自动识别 字符创编码

chardet 使用方法

unicode 转换为其它编码(GBK, GB2312等)

例如：a为unicode编码 要转为gb2312。a.encode('gb2312')

```
# -*- coding=gb2312 -*-
a = u"中文"
a_gb2312 = a.encode('gb2312')
print a_gb2312
```

GBK 与 GB2312的区别

GB 码，全称是GB2312-80《信息交换用汉字编码字符集基本集》，1980年发布，是中文信息处理的国家标准，在大陆及海外使用简体中文的地区（如新加坡等）是强制使用的唯一中文编码。P- Windows3.2和苹果OS就是以GB2312为基本汉字编码， Windows 95/98则以GBK为基本汉字编码、但兼容支持GB2312。GB码共收录6763个简体汉字、682个符号，其中汉字部分：一级字3755，以拼音排序，二级字3008，以偏旁排序。该标准的制定和应用为规范、推动中文信息化进程起了很大作用。

GBK编码是中国大陆制订的、等同于UCS的新的中文编码扩展国家标准。GBK工作小组于1995年10月，同年12月完成GBK规范。该编码标准兼容GB2312，共收录汉字21003个、符号883个，并提供1894个造字码位，简、繁体字融于一库。

GBK 包括 GB2312的所有编码，有些字GB2312没有，需要用GBK来编码。

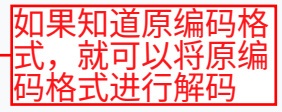
转: gbk, gb2312,big5,unicode,utf-8,utf-16的区别

其它编码(utf-8,GBK)转换为unicode

例如：a为gb2312编码，要转为unicode. unicode(a, 'gb2312')或a.decode('gb2312')

```
# -*- coding=gb2312 -*-
a = u"中文"
a_gb2312 = a.encode('gb2312')
print a_gb2312

a_unicode = a_gb2312.decode('gb2312')
assert(a_unicode == a)
a_utf_8 = a_unicode.encode('utf-8')
```



```
print a_utf_8
```

非unicode编码之间的转换

编码1(GBK,GB2312) 转换为 编码2(utf-8,utf-16,ISO-8859-1)

可以先转为unicode再转为编码2

如gb2312转utf-8

```
# -*- coding=gb2312 -*-
a = u"中文"
a_gb2312 = a.encode('gb2312')
print a_gb2312

a_unicode = a_gb2312.decode('gb2312')
assert(a_unicode == a)
a_utf_8 = a_unicode.encode('utf-8')
print a_utf_8
```

判断字符串的编码

isinstance(s, str) 用来判断是否为一般字符串
isinstance(s, unicode) 用来判断是否为unicode
如果一个字符串已经是unicode了，再执行unicode转换有时会出错(并不都出错)

下面代码为将任意字符串转换为unicode

```
def u(s, encoding):
    if isinstance(s, unicode):
        return s
    else:
        return unicode(s, encoding)
```

unicode 与其它编码之间的区别

为什么不所有的文件都使用unicode，还要用GBK，utf-8等编码呢？

unicode可以称为抽象编码，也就是它只是一种内部表示，一般不能直接保存。
保存到磁盘上时，需要把它转换为对应的编码，如utf-8和utf-16。

其它方法

除上以上的编码方法，在读写文件时还可以使用codecs的open方法在读写时进行转换。

命令行默认编码检测和设置

可以用python自带的模块locale来检测命令行默认编码和设置命令行编码。

```
import locale

#get
locale.getdefaultlocale()
#('zh_CN', 'cp936')

#set
locale.setlocale(...)
```

汉字转Unicode编码

```
    pd_name = pd_name.decode('utf-8')
    print pd_name
    nname = ""
    for c in pd_name:
      c = "%%u%04X" % ord(c);
      nname += c
```

更多编码相关

- 全角半角转换的Python实现