**Python读 CSV**

**import csv**  添加csv模块

csvfile = file('csv\_test.csv', 'rb') 以二进制读的方式打开指定的csv文件

reader = csv.reader(csvfile) 将csv文件中内容读入reader中

for line in reader: 循环逐行访问内容

print line 一行的数据会以 逗号 拆分存储在list中

可以通过line[0] 的方式访问一行的第一列数据

csvfile.close() 使用完文件记得关闭文件

**Python 读写mongodb**

**from** pymongo **import** MongoClient 添加模块

client = MongoClient(**'localhost'**, 27017) 连接mongodb数据库  
db = client.local 连接到具体的local database

collection=db.test 选择collection test

collection.insert({'Date':line[0],'Time':line[1] }) 插入一条mongodb记录的方法

**python 编码问题**

任何字符串都是一串二进制字节的序列，而ASCII码是最经典的编码方式，它将序列中的每个字节理解为一个字符，可表示阿拉伯数字、字母在内的128个不同字符。很明显，汉字在ascii中是无法表示的。

中国制定了GBK（GB2312的扩展）编码

UTF-8与GBK类似，也是一种兼容ASCII码的不定长编码形式，它的长度变化更大，因此可以表示几乎所有世界文字。

Unicode是一种定长的编码方式（同ASCII），不过它是每2字节认为是一个字符，如ASCII中0x61表示'a'，在Unicode中用0x0061表示'a'，它可映射所有文字，而且对于多种写法的字，如強/强，它都可以唯一地区分它们。

由于Unicode编码的字符串体积很大，因此一般来说Unicode编码只是文字在内存中的内在形式，具体的存储（如文件、网页等）都需要靠外在的编码（UTF-8、GBK等）诠释。

Python中实际上有两种字符串，分别是str类型和unicode类型

str类型的本质就是一坨二进制串，源文件（或获取的网页）的编码是怎样，它就跟着是怎样。所以我们要在源文件前面加 # encoding: utf-8 指定代码编码方式是二进制utf-8

相比于str，unicode是真正的字符串。Python明确地知道它的编码

Python可以使用 encode ecode 显式转化两种编码方式：

encode(encoding): 将unicode转换为str，并使用encoding编码；

decode(encoding)：将str转换为unicode，其中str以encoding编码。

编码总结：

1.unicode是支持所有文字的统一编码，但一般只用作文字的内部表示，文件、网页（也是文件）、屏幕输入输出等处均需使用具体的外在编码，如GBK、UTF-8等；

2.encode和decode都是针对unicode进行“编码”和“解码”，所以encode是unicode->str的过程，decode是str->unicode的过程；

3.unicode和str是一对孪生兄弟，来自basestring，所以用isinstance(s, basestring)来判断s是否为字符串。

Windows系统下文件默认编码方式是 ANSI也就是二进制的GBK编码方式