098 模拟医院分诊排号系统

* 核心技术

在Python中，字符串编码操作有3种数据状态，首先是明文字符串，也就是程序员所定义的字符，然后经过转换，在内存中将以Unicode形式存储数据，最后，保存到磁盘上时会相应的转换成UTF-8或GBK等编码格式。转换过程如图1所示。

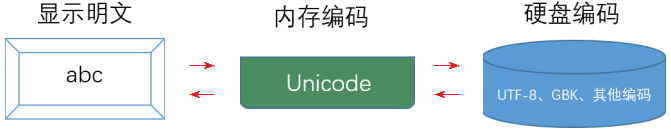


图1 当前代码文件的路径

图1中，上面箭头表示为编码过程，在Python中使用字符串对象的encode方法来实现编码。下面箭头表示为解码过程，那么解码字符串则需要使用bytes对象的decode方法来实现。

encode方法是字符串对象内置的一个实现方法,用于实现编码操作。语法格式如下：

str.encode([encoding[,errors]])

参数encoding，表示要进行编码的字符格式，如果不指定该参数，则默认编码格式为UTF-8。参数errors，用于指定错误的处理方式，它的值可以是strict(遇到非法字符抛异常)、ignore(忽略非法字符)、replace(用”?”替换非法字符)或xmlcharrefreplace(使用XML的字符引用)等，默认值为strict。

decode方法是bytes对象内置的一个实现方法,用于实现解码操作。语法格式如下：

bytes.decode([encoding[,errors]])

参数encoding，表示要进行解码的字符格式，如果不指定该参数，则默认编码格式为UTF-8。参数errors，用于指定错误的处理方式，它的值可以是strict(遇到非法字符抛异常)、ignore(忽略非法字符)、replace(用”?”替换非法字符)或xmlcharrefreplace(使用XML的字符引用)等，默认值为strict。

下面，我们将已经编码后的bytes内容进行解码。代码如下：

1. #定义字节编码
2. Bytes1=bytes(b'\xe5\xbc\x84\xe6\xb8\x85\xe6\xa5\x9aPython\xe5\xad\x97\xe7\xac\xa6\xe7\xbc\x96\xe7\xa0\x81\xef\xbc\x8c\xe4\xbb\xa5\xe5\x85\x8d\xe8\xa2\xab\xe7\xbc\x96\xe7\xa0\x81\xe9\x97\xae\xe9\xa2\x98\xe6\x89\x80\xe5\x9b\xb0\xe6\x89\xb0!')
3. #定义字节编码
4. Bytes2=bytes(b'\xc5\xaa\xc7\xe5\xb3\xfePython\xd7\xd6\xb7\xfb\xb1\xe0\xc2\xeb\xa3\xac\xd2\xd4\xc3\xe2\xb1\xbb\xb1\xe0\xc2\xeb\xce\xca\xcc\xe2\xcb\xf9\xc0\xa7\xc8\xc5!')
5. str1=Bytes1.decode("utf-8")#进行UTF-8解码
6. str2=Bytes2.decode("GBK") #进行GBK解码
7. print(str1) #输出UTF-8解码后的内容
8. print(str2) #输出GBK解码后的内容

运行程序，得到如下结果：

弄清楚Python字符编码，以免被编码问题所困扰!

弄清楚Python字符编码，以免被编码问题所困扰!

zip() 函数用于将可迭代的对象作为参数，将对象中对应的元素打包成一个个元组，然后返回由这 些元组组成的列表。如果各个迭代器的元素个数不一致，则返回列表长度与最短的对象相同，利用 \* 号操作符，可以将元组解压为列表。

语法：

zip([iterable, ...])

参数说明：iterabl -- 一个或多个迭代器 ;

返回值：元组列表。