1.如何以16进制读取一个文件？

二进制读取‘rb’肯定没有错

With open(r’2.txt’,’rb’) as f:

a = f.read()

但是f.read()读取的是字节串，需要转换成什么类型，找到合适的转换方法就可以了。

如：

字节串转字符串：

字节码解码成字符串：bytes(b’\x31\x32\x61\x62’).decode(‘ascii’)🡺12ab

字节串转16进制表示，夹带ascii，str(bytes(b’\x01\x0212’))[2:-1]🡺\0x1\0x212

字节串转16进制表示，固定两个字符表示：str（binascii.b2a\_hex(b’\x01\x0212’))[2:-1] 🡺01023132

字节串转16进制数组：[hex(x) for x in bytes(b’\x01\x0212’)]🡺[‘0x1’,’0x2’,’0x31’,’0x32’]

2.Python中的字节串

在python3中，用bytes对象来处理字节字符串。

Bytes和可变的bytearray与str不同，只能用字节作为序列值，即0~256之间的整数表示。

print(bytes([101,100]))

--b'ed'

print(list(b'hello'))

--[104, 101, 108, 108, 111]

从python3.0开始，所有没有前缀的字符串都是unicode。字节字符串要加前缀b。

Unicode字符串是不可以用字节表示的文本，而将数据保存到文本或者通过网络发送必须是二进制数据，所以必须将字符串对象编码为字节序列。

字符串转字节序列：

str.encode(encoding,errors)方法，encoding为编解码器，默认值为‘utf-8’，

print('hello'.encode())

-- b'hello'

bytes（source,encoding,errors）构造函数，创建一个新的字节序列，source为字符串，encoding为编码器，必须指定encoding，没有默认值，

print(bytes(‘hello’,encoding=’utf-8’))

-- b'hello'

Bytes.decode(encoding,errors)方法，字节码解码成字符串，encoding必须与str.encode里的encoding相同

Print(b’hello’.decode())

-- ‘hello’

Str(source,encoding,errors)构造函数与bytes（）构造函数相似，source是字节序列，必须指定str函数的参数encoding，它没有默认值，

Str(b’hello’,’utf-8’)

-- ‘hello’

Python字符串是不可变的，bytes字节序列也是不可变的，有优点有缺点，优点是由于不变性，字符串可以做字典的键或set的元素，因为一旦初始化字符串的值就不会改变。缺点是当要修改字符串时，都需要创建一个全新的字符串实例。Bytearray时bytes可变版本，字节数组可以通过append，pop，insert等方法添加修改删除字节序列。

3.B’’的表示的注意

Python中用类似b’\x09\x89\x78’可以表示字节串，实际上b’’是类似字符串，但是可以用hex格式表示其中的每个字节的值，这样很直观。

B’\x’表示的时候，一定要注意\x只能是一个字节一个字节的表示，有个例子如下：

例1：手误造成的有问题的例子

a = b'\xFE006795'

a[0] --> 254 # 0xFE

a[1] --> 48 # 字符'0'的ASCII码

a[2] --> 48 # 字符'0'的ASCII码

a[3] --> 54 # 字符'6'的ASCII码

a[4] --> 55 # 字符'7'的ASCII码

a[5] --> 57 # 字符'9'的ASCII码

a[6] --> 53 # 字符'5'的ASCII码

a[7] --> index out of range

例2：正常使用的例子

b = b'\xFE\x00\x67\x95'

b[0] --> 254

b[1] --> 0

b[2] --> 103

b[3] --> 149

b[4] --> index out of range

分析：

例1：b’\Xfe006975’，只有加了’\x’前缀的两个字符‘FE’才被认为是HEX格式的字节数据，其后的数据都没有’\x’前缀，所以被认为是ASCII字符表示。

例2：b’\xFE\x00\x67\x95’其实表示的是一个字节数组，是一种字符串，只是字符串的内容并不是以ASCII码和Unicode码来显示，而是直接显示在内存中的内容。

既然是字符串，那么就可以使用其索引和切片等特性。