初学 Python，相信大家遇到的一大难题就是编码问题，如下：

1. >>>
2. Traceback (most recent call last):
3. File "/Users/FishC/Documents/Python/test.py", line 2, in <module>
4. print(f1.read())
5. File "/Library/Frameworks/Python.framework/Versions/3.4/lib/python3.4/encodings/ascii.py", line 26, in decode
6. return codecs.ascii\_decode(input, self.errors)[0]
7. UnicodeDecodeError: 'ascii' codec can't decode byte 0xce in position 0: ordinal not in range(128)

复制代码

这样的报错信息着实令大家很头疼，因为在网上很难找到解决方案。  
  
这里小甲鱼给大家总结一下几种会导致编码问题的案例，并逐一解释......  
  
  
**案例一：中文输出是乱码？**

1. # Python 版本：2.7.6
2. >>> string1 = "我爱鱼C工作室"
3. >>> string1
4. '\xe6\x88\x91\xe7\x88\xb1\xe9\xb1\xbcC\xe5\xb7\xa5\xe4\xbd\x9c\xe5\xae\xa4'
5. >>> print string1
6. 我爱鱼C工作室
7. >>> string2 = "I love FishC"
8. >>> string2
9. 'I love FishC'

复制代码

**问：为何无法直接显示中文的字符串？**  
  
**分析：**  
  
因为 Python2.x 的版本默认的编码是 ASCII，ASCII 默认只用一个字节来存放数据。由于中国汉字博大精深，一个字节是不足以存放所有的汉字的。因此，string1 只能打印出中文字符串在内存中的数据，这并不是错误。  
  
**解决方案：**  
  
使用 Python3，因为 Python3 默认使用 UTF-8 编码。  
  
**延伸知识：**  
  
1. 可以用以下方法获得当前的默认编码：

1. >>> import sys
2. >>> sys.getdefaultencoding()
3. 'ascii'

复制代码

2. [字符集与字符集编码详解](http://bbs.fishc.com/thread-46797-1-1.html)  
  
  
**案例二：普通字符串和 Unicode 字符串进行拼接抛出 UnicodeDecodeError 异常**

1. >>> string = "我爱" + u"FishC"
2. Traceback (most recent call last):
3. File "<stdin>", line 1, in <module>
4. UnicodeDecodeError: 'ascii' codec can't decode byte 0xe6 in position 0: ordinal not in range(128)

复制代码

**分析：**  
  
使用 + 号进行字符串拼接，左边是普通字符串，右边是 Unicode 字符串。当两种类型的字符串拼接的时候，Python 会自动将左边的中文字符串转换为 Unicode 字符串，再进行拼接操作。但由于 "我爱" 的 ASCII 编码为 '\xe6\x88\x91\xe7\x88\xb1'，其中十六进制 '\xe6' 对应的值是 230。当编码值在 0 ~ 127 的时候，Unicode 和 ASCII 是兼容的，转换不会有什么问题。但当值大于 128 的时候，ASCII 编码便不能直接转换为 Unicode 了。因此，抛出 UnicodeDecodeError。  
  
**解决方案：**  
  
1. 使用 Python3  
  
2. 指定转换为 Unicode 的解码方式：

1. >>> string = "我爱".decode('utf-8') + u"FishC"
2. >>> print string
3. 我爱FishC

复制代码

3. 将 Unicode 字符串部分进行编码：

1. >>> string = "我爱" + u"FishC".encode("utf-8")
2. >>> print string
3. 我爱FishC

复制代码

**延伸知识：**  
  
Unicode 编码系统的发明是为了统一各国文字的编码，因此把它称为万国码。Unicode 为每种语言设置了唯一的二进制编码表示方式，也就是说无论哪个国家的语言，都可以在 Unicode 上找到对应的代码。因此，当不同的编码系统进行相互转换的时候，可以利用 Unicode 做一个“中介”。  
  
其他编码系统到 Unicode 的转换过程我们称为解码（decode），将 Unicode 转换为其他编码系统的过程称之为编码（encode）。例如 A 编码需要转换为 B 编码，过程如下：  
  
A编码 -> decode(A) -> Unicode -> encode(B) -> B 编码  
  
  
**案例三：文件编码与 Python 编码不同**  
  
test.txt 内容如下，并保存为 GB2312 编码：

1. 我爱鱼C工作室，真的！

复制代码

test.py 内容如下：

1. f1 = open("test.txt")
2. print(f1.read())
3. f1.close

复制代码

代码执行后会报错：

1. >>>
2. Traceback (most recent call last):
3. File "/Users/FishC/Documents/Python/test.py", line 4, in <module>
4. print(f1.read())
5. File "/Library/Frameworks/Python.framework/Versions/3.4/lib/python3.4/encodings/ascii.py", line 26, in decode
6. return codecs.ascii\_decode(input, self.errors)[0]
7. UnicodeDecodeError: 'ascii' codec can't decode byte 0xce in position 0: ordinal not in range(128)

复制代码

**分析：**  
  
如果前边的内容都可以理解了，那么解决这样的编码问题就不再难得住你啦~~~  
  
使用 open 打开文件的编码格式取决于系统（可以通过 locale.getpreferredencoding() 获得），认真看报错信息，这里系统使用 ASCII 对文件内容进行解码，遇到错误......因为我们知道文件的存放格式是 GB2312，因此我们只需要在打开文件的时候设置 encoding="gb2312" 即可解决问题：

1. f1 = open("test.txt", encoding="gb2312")
2. print(f1.read())
3. f1.close

复制代码