一、字符编码问题

1. 文件编码与内存编码(或者说显示器编码)

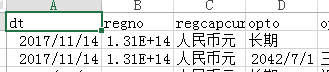
计算机内部会按我们制定的设置，认为文件是什么编码，并按这种编码查找字符集，在显示器上显示出来。

1. txt文件

txt文件是我比较喜欢的一种文件，因为它比较单纯。你把一个excel文件转成.csv，转换前是这样的：



转换后就成了这样：

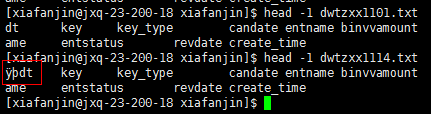


它变了，变了。

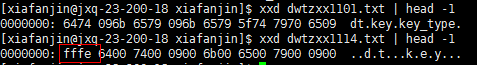
但是txt文件会比较单纯，但是这里也有一个小坑。txt文件在windows中不止一种，据我所知就有两种，一种叫文本文件，另一种叫Unicode文本。下面是excel转成txt时文件类型的选项：



下面的两个文件，dwtzxx1101.txt是文本文件，dwtzxx1101.txt是Unicode文本。两种文件用notepad打开在观感上没什么区别，但是在linux环境下看：



用xxd或者在windows下用ultraedit(ctrl+h)看：



竟然出现了fffe，这是个啥玩意，查了一下，结果如下：

在一个字节流或者文件中，如果最前面的字节分别是 FE和FF，就表明这个字节流或文件是Unicode Big-Endian 的；如果是FF和FE，就表明这个字节流或文件是Unicode Little-Endian 的。这两个字节是Unicode编码文本文件的BOM头。

这会有什么影响呢，当你想转码时，它会报错。



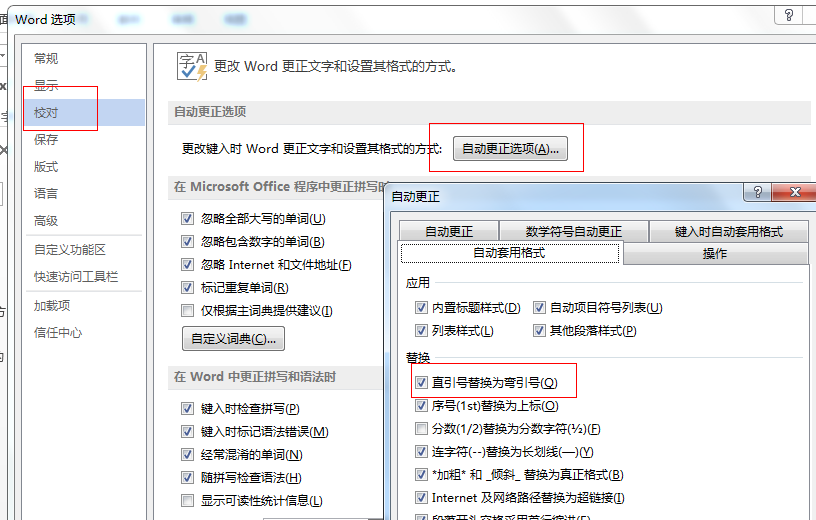
3.转码

linux

iconv -f GB18030 -t UTF-8 qtgsrz1101.txt -o qtgsrz1101\_utf8.txt

二、windows的引号

一个是直引号，一个是弯引号。

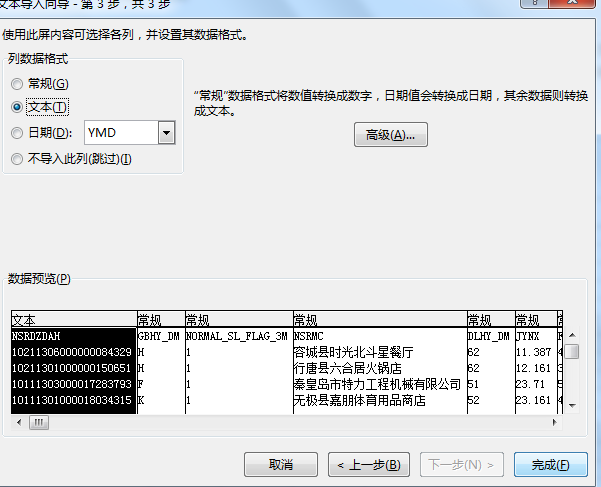


三、percent类型的50%，转换成数值类型，还是显示50%。excel这贱，遇到这种情况时，你采用数据->分裂，然后就变回来了。

四、Excel打开csv文件，身份证之类的东西，变成数字，处理办法。

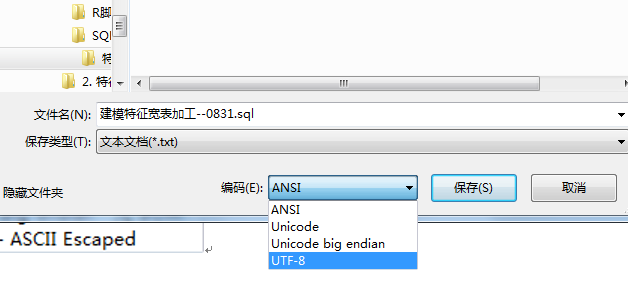
1.创建一个空的excel文件，数据->自文本

2.选中那一列，选文本格式，如下



五、再次吐槽Windows的编码

Excel保存为csv文件，你无法指定编码方式，网上说web选项里指定编码格式，这个不靠谱，试过之后，就知道然并卵。网传一种替代的方式是：Excel保存为csv文件后，你用notepad打开，然后另存为，选择编码格式，如下图：



但这个有些坑爹，须知utf-8格式的文本，有两种，一种带BOM头的，一种不带，而你这样保存的utf-8文本是带BOM头的，在ue下看，文件的前三个字节是EF BB BF。

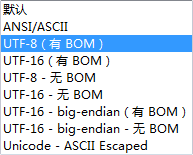
所以，靠谱的方式是：

Excel保存为csv文件后，用UE打开，这时候你会看到ue猜测的文件的编码，如下：



看，虽然在Excel的web选项里设置了编码为UTF-8，但Excel仍然把它保存成936格式了。

然后另存为，选择没有BOM头的UTF-8格式保存，如下图。



顺便说一下那个ansi

【编码方式讲解】

（1）ANSI编码

不同的国家和地区制定了不同的标准，由此产生了 GB2312, [BIG5](https://www.baidu.com/s?wd=BIG5&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3n101m19-ujm1PjT1P1Fb0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPjbYPWb1PWnk), JIS 等各自的编码标准。这些使用 2 个字节来代表一个字符的各种汉字延伸编码方式，称为 ANSI 编码。在简体中文系统下，ANSI 编码代表 GB2312 编码，在日文操作系统下，ANSI 编码代表 JIS 编码。 不同 ANSI 编码之间互不兼容，当信息在国际间交流时，无法将属于两种语言的文字，存储在同一段 ANSI 编码的文本中。 当然对于ANSI编码而言，0x00~0x7F之间的字符，依旧是1个字节代表1个字符。这一点是ASNI编码与Unicode编码之间最大也最明显的区别。例如上面演示的文件中英文字母和数字并没有出现乱码的情况。

（2）Unicode编码

Unicode（统一码、万国码、单一码）是一种在计算机上使用的字符编码。它为每种语言中的每个字符设定了统一并且唯一的二进制编码，以满足跨语言、跨平台进行文本转换、处理的要求。1990年开始研发，1994年正式公布。随着计算机工作能力的增强，Unicode也在面世以来的十多年里得到普及。

Unicode是[国际组织](https://www.baidu.com/s?wd=%E5%9B%BD%E9%99%85%E7%BB%84%E7%BB%87&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3n101m19-ujm1PjT1P1Fb0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPjbYPWb1PWnk)制定的可以容纳世界上所有文字和符号的字符编码方案。Unicode用数字0-0x10FFFF来映射这些字符，最多可以容纳1114112个字符，或者说有1114112个码位。码位就是可以分配给字符的数字。UTF-8、[UTF-16](https://www.baidu.com/s?wd=UTF-16&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3n101m19-ujm1PjT1P1Fb0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPjbYPWb1PWnk)、[UTF-32](https://www.baidu.com/s?wd=UTF-32&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3n101m19-ujm1PjT1P1Fb0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPjbYPWb1PWnk)都是将数字转换到程序数据的编码方案。

（3）UTF-8编码

UTF-8是UNICODE的一种变长字符编码又称万国码，由Ken Thompson于1992年创建。现在已经标准化为RFC 3629。UTF-8用1到6个字节编码UNICODE字符。用在网页上可以同一页面显示中文简体繁体及其它语言(如日文，韩文)。

Unicode 编码系统，可分为[编码方式](https://baike.baidu.com/item/%E7%BC%96%E7%A0%81%E6%96%B9%E5%BC%8F)和实现方式两个层次。UTF-8是Unicode的一种实现方式，他们的对应关系如下

|  |  |
| --- | --- |
| **Unicode编码(十六进制)** | **UTF-8 字节流(二进制)** |
| 000000-00007F | 0xxxxxxx |
| 000080-0007FF | 110xxxxx 10xxxxxx |
| 000800-00FFFF | 1110xxxx 10xxxxxx 10xxxxxx |
| 010000-10FFFF | 11110xxx10xxxxxx10xxxxxx10xxxxxx |

六、windows与linux的换行符

windows下编辑过的sql，没有语法问题，到了linux执行出错，这可能是换行符导致的。

下面是windows下编辑过的脚本里的一句话，在linux服务器上执行报错，把它手动黏贴在linux里，错误是这样的。

insert overwrite table dmt.dmt\_risk\_pop\_shop\_sale\_predict\_s\_m partition (dt='$THIS\_MONTH\_20')

select pop\_vender\_id,pre\_sale\_amt from datamining.dmt\_risk\_pop\_shop\_sale\_predict\_a where pre\_sale\_amt is not null;

--------------------------------------------------------------------我是分割线--------------------------------------------------------------------

hive> insert overwrite table dmt.dmt\_risk\_pop\_shop\_sale\_predict\_s\_m partition (dt='2018-02-20')

>

Display all 477 possibilities? (y or n)

> der\_id,pre\_sale\_amt from datamining.dmt\_risk\_pop\_shop\_sale\_predict\_a where pre\_sale\_amt is not null;

NoViableAltException(26@[])

在linux的hive里面，(dt='$THIS\_MONTH\_20')之后断掉了。用ue查看这个文件，发现(dt='$THIS\_MONTH\_20')后面的ascii码是0d0a09，09是横向制表符，这个不论。0d和0a两个便是windows的换行符。



0d ：\r：回车

0a ：\n：换行

在windows下：\r\n代表换行，拆分两个代码是：回到行首+换到下一行。下面引一篇csdn，建议去看原文：https://blog.csdn.net/stpeace/article/details/45767245

多数程序员都会在不经意间遇到下面这两个问题：

        1.  linux上用vim写的文件test.txt拷贝到windows上， 结果所有的内容都显示在一行中。（当然， 如果你Windows上的编辑显示器够智能， 那就是另外一回事了）

        2. Windows上建立的test.txt拷贝到linux上， 结果linux程序运行异常。（当然， 如果你的linux程序足够健壮， 那就没有问题了）

        ps. 当你在Windows上写好linux shell script, 拷贝到linux下运行， 就会出问题。

        为什么会出现上面的问题呢？ 这就需要我们了解\r\n和\n了。

        我们先以Windows为例来说明。 我们来新建一个dos.txt文件， 然后在第一行敲一下enter键， 然后保存文件。 此时， 我们查一下这个文件的大小， 结果发现是2个字节（而不是1个）， 用UltraEdit查阅一下， 就知道文件中的两个字节是0x0d和0x0a, 也就是\r和\n.  但是， 如果用C程序来读取dos.txt文件， 发现又根本没有读取到\r, 这是怎么回事呢？ 原来： 在Windows上的dos.txt文件中， 按一下enter键， 实际上就相当于C程序写\n字符，而由于历史原因， 写文件的时候， Windows系统会自动在\n前面加上\r, 这样， dos.txt文件中就有了\r\n了， 在读取的时候， Windows系统会自动去掉\r, 所以你的C程序根本读不读\r， 还是\n.

        实话实说， 我发现很多地方讲\r\n和\n, 都说得不太清楚， 下面我来说一下， 希望是更清楚， 而不是更模糊微笑：

        Windows系统中有如下等价关系：

        用enter换行 <====> 程序写\n  <====> 真正朝文件中写\r\n(0x0d0x0a) <====>程序真正读取的是\n

        linux系统中的等价关系：

        用enter换行 <====> 程序写\n  <====> 真正朝文件中写\n(0x0a)  <====> 程序真正读取的是\n

        现在， 我们看看本文开头部分的问题。 假设有一个linux下的unix.txt文件,  那么， 它在文件中的换行标志是：\n, 现在把unix.txt拷贝靠Windows上， 那好啊， Windows那双犀利的眼神仿佛是在对unix.txt文件说： 别跟我整什么\n, 我只认识文件中的\r\n, 如果你这个unix.txt文件里面有\r\n, 那我就认为是换行符， 否则， 我不认你。 如此一来， Windows压根就找不到unix.txt中的\r\n, 所以， 对于Windows而言， 压根就没有发现unix.txt有任何换行， 所以， 我们从Windows上看到的unix.txt文件显示在一行里面。

       同理， 假设Windows上有一个dos.txt文件， 那么， 它在文件中的换行标志是\r\n, 现在拷贝到linux下， 那好啊， 正如linus一样， linux这个倔强的家伙貌似在说： 别的我不管， 我遇到文件中的\n, 我就认为是换行， 至于其他的， 我只认为是正常的字符。 如此一来， \r就被当成了文件的正常部分， 也就是说， linux下的C程序不仅仅会读取到\n, 也会读取到它前面的\r.

       实际上Windows上的文件是dos格式的， 而linux上的文件是unix格式的， 我们可以通过linux上的dos2unix和unix2dos来实现转化， 当然， 有的linux环境下， 需要用busybox dos2unix和busybox unix2dos.

 下面， 我们用程序来简要验证一下上面的部分叙述。 现在Windows上创建一个dos.txt文件， 第一行写入12， 然后按enter, 然后在第二行写入3, 不按enter, 然后保存。 我们先来看看Windows上的程序：

如果文件需要在windows和linux上经常互相编辑，需要注意一下喔。

