

# 第10期-理解编码与回车换行符

2020年1月4日 14:09

## vs中的换行符（不可见）

CR LF：回车符 + 换行符，Unicode 字符 000D + 000A

LF：换行符，Unicode 字符 000A

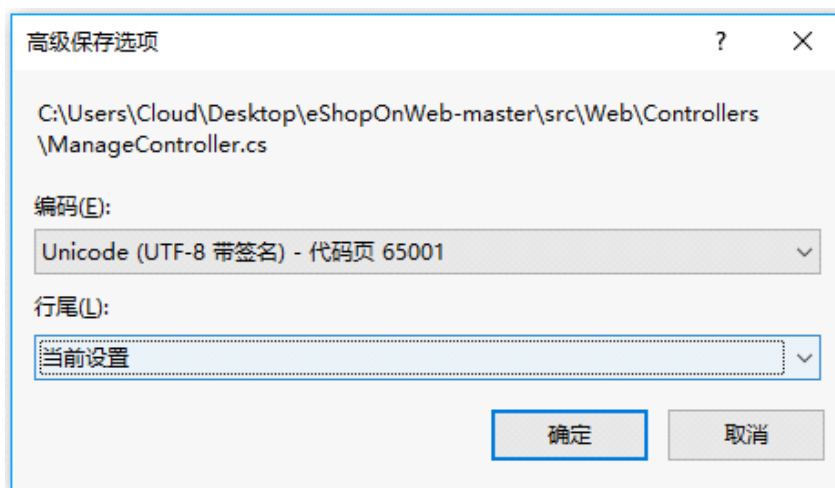
NEL：下一行，Unicode 字符 0085

LS：行分隔符，Unicode 字符 2028

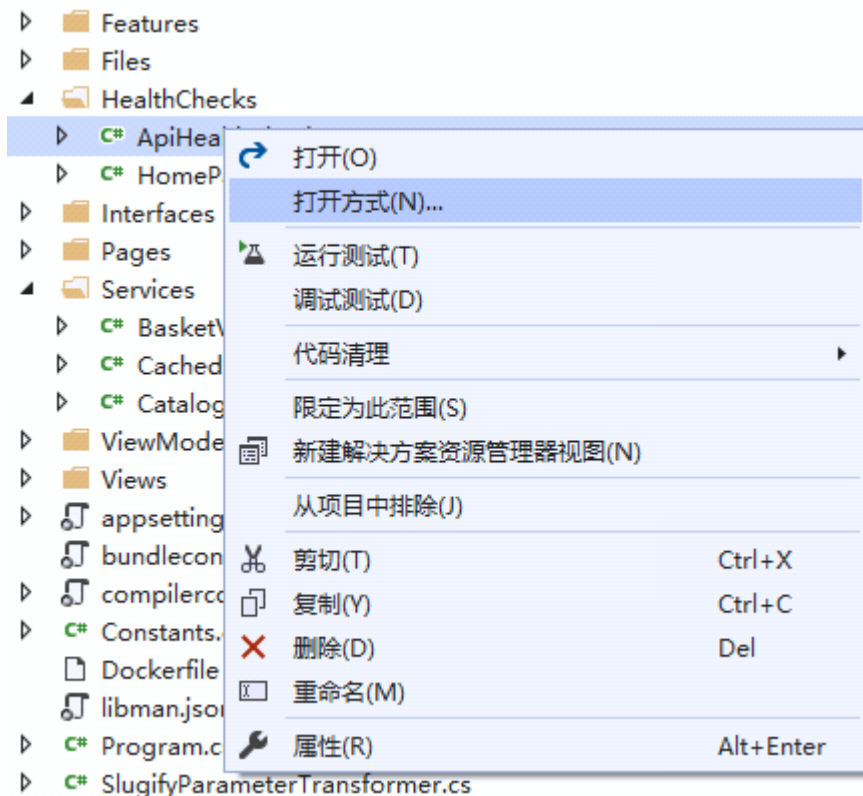
PS：段落分隔符，Unicode 字符 2029

## 高级保存选项

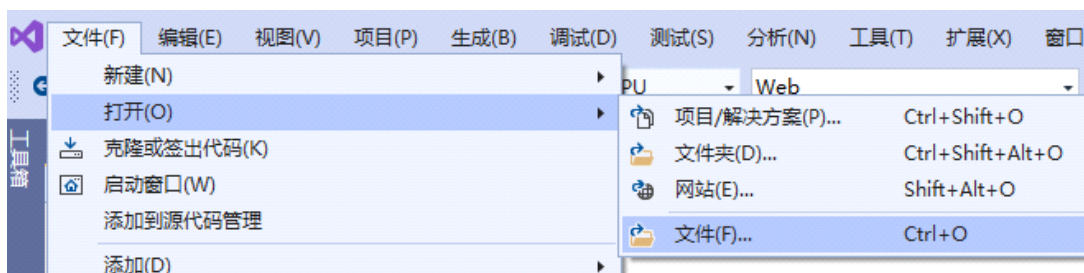
可以使用高级保存选项设置文件的编码格式和换行格式。



## 使用指定的编码打开项目中的文件



## 使用指定的编码打开其它非项目中的文件



## 回车与换行的区别

在计算机还没有出现之前，有一种叫做电传打字机（Teletype Model 33）的玩意，每秒钟可以打10个字符。但是它有一个问题，就是打完一行换行的时候，要用去0.2秒，正好可以打两个字符。要是在这0.2秒里面，又有新的字符传过来，那么这个字符将丢失。

于是，研制人员想了个办法解决这个问题，就是在每行后面加两个表示结束的字符。一个叫做"回车"，告诉打字机把打印头定位在左边界；另一个叫做"换行"，告诉打字机把纸向下移一行。

这就是"换行"和"回车"的来历，从它们的英语名字上也可以看出一二。

后来，计算机发明了，这两个概念也就被般到了计算机上。那时，存储器很贵，一些科学家认为在每行结尾加两个字符太浪费了，加一个就可以。于是，就出现了分歧。

Unix系统里，每行结尾只有"<换行>"，即"\n"；Windows系统里面，每行结尾是"<回车><换行>"，即"\r\n"；Mac系统里，每行结尾是"<回车>"。一个直接后果是，Unix/Mac系统下的文件在Windows里打开的话，所有文字会变成一行；而Windows里的文件在Unix/Mac下打开的话，在每行的结尾可能会多出一个^M符号。

[http://www.ruanyifeng.com/blog/2006/04/post\\_213.html](http://www.ruanyifeng.com/blog/2006/04/post_213.html)

## 文本编码

[细说 unicode 、 utf-8 、 utf-16 、 ascii 、 gbk 、 gb2312](#)

[字符编码的故事: ASCII, GB2312, Unicode, UTF-8, UTF-16](#)