

Mapping all those "strange" digits to "0" through "9"

 devblogs.microsoft.com/oldnewthing/20040416-00

April 16, 2004



Raymond Chen

In an earlier article, I discussed how the `Char.IsDigit()` method and its Win32 counterpart, `GetStringTypeEx` report things to be digits that aren't just "0" through "9". If you really care just about "0" through "9", then you can test for them explicitly. For example, as a regular expression, use `[0-9]` instead of `\d`. Alternatively, for a regular expression, you can enable ECMA mode via `RegexOptions.ECMAScript`. Note that this controls much more than just the interpretation of the `\d` character class, so make sure to read carefully to ensure that you really want all the ECMA behavior. It has been pointed out to me that there is a way to convert all those "strange" digits to the "0" through "9" range, namely by calling the `FoldString` function with the `MAP_FOLDDIGITS` flag. (I put the word "strange" in quotation marks because of course they aren't strange at all. Just different.) This converts digits but doesn't help with decimal points, so you still have to deal with correctly interpreting "1,500" as either "one thousand five hundred" (as it would be in the United States) or "one and a half" (as it would be in most of Europe). For that, you need to call `GetLocaleInfo` to get the `LOCAL_SDECIMAL` and `LOCAL_STHOUSAND` strings.

Raymond Chen

Follow

