

Proposal to improve the explanation of D143

Asmus Freytag

February 1, 2011

Proposal

I propose a modification of the explanatory text around D143 s along the lines I describe below, to make this definition easier to comprehend. (This does not affect the definition nor the technical content of the modified text and is therefore entirely editorial in nature).

Background

As worded, the existing text is difficult to follow. This is partly because the set of properties is unintuitive as defined (intuitively one expects a fourth value "uncased"). Because of that, it is easy to get turned around and the way the text is worded doesn't help. Once you "get it", the text is fine, but it is of little help to correct anyone who's mentally on the wrong track.

Details

Here's the text with proposed modifications as indicated:

Uncased characters do not affect the results of casing detection operations such as the string function `isLowercase(X)`. Thus a space or a number added to a string does not affect the results.

So far so good

There is a degenerate case, such as "123", where the string contains no cased letters and thus `isLowercase("123")` evaluates as true. In many situations it may be appropriate for implementations to also test whether there are any cased characters in the strings. This is accomplished by testing for (`isLowercase(X)` AND `isCased(X)`), using the following definition:

The paragraph above can be made much simpler and more straightforward by first moving the final note up, and then revising the text leading into definition 143. The following replace the above paragraph:

The examples in Table 3-16 show that these conditions are not mutually exclusive. "A2" is both uppercase and titlecase; "3" is uncased, so it is *simultaneously* lowercase, uppercase, and titlecase.

Only when a string, such as "123", contains no cased letters will all three conditions, `isLowerCase`, `isUpperCase` and `isTitleCase` evaluate as true. This can be used to check for the presence of cased letters, using the following definition:

Then we are ready for the definition D143, text unchanged:

D143 isCased(X): isCased(X) is true when isLowercase(X) is false, or isUppercase(X) is false, or isTitlecase(X) is false.

- Any string X for which isCased(X) is true contains at least one character that has a case mapping other than to itself.
- For example, isCased("123") is false because all the characters in "123" have case mappings to themselves, while isCased("abc") and isCased("A12") are both true.
- The derived binary property Changes_When_Casemapped is listed in the data file DerivedCoreProperties.txt in the Unicode Character Database.

Finally, the reader is ready to follow the need for the dual test, so this can come at the end (not as a bullet, so it stands out).

To find out whether a string contains only lowercase letters, implementations need to test for (isLowercase(X) AND isCased(X)),

To view the original text see <http://www.unicode.org/versions/Unicode6.0.0/ch03.pdf> or <http://www.unicode.org/versions/Unicode5.2.0/ch03.pdf> (which is the same).