String splitting handling of empty fields is incorrect or insufficiently documented (SOLVED)

07/27/2022 06:11 PM - scub8040 (Saverio M.)

Status: Closed
Priority: Normal
Assignee:

Description
Hello!

The string splitting needs to deal with some edge cases when it comes to empty strings/fields, for example, an empty string always returns an empty array.

There are other cases though, which I think are either incorrectly handled, or at least, they should documented.

The main case is a string exclusively composed of separators, e.g.:

"|||".split "|" # => []

Semantically speaking, such splitting does make sense, as an empty field is still a field. As the above example shows though, this returns an empty array (following the explained logic, it should return 4 empty strings).

IMO, this is incorrect. If for any reason this isn't, this should be documented though, as it's not obvious behavior (I've referred to this page: https://ruby-doc.org/core-3.0.0/String.html#method-i-split).

This result is definitely inconsistent with both logics explained above:

- if empty fields should be treated as effective fields, the function should return ["", "", "a", ""]
- if empty fields should be ignored, it should return ["a"]

Considering this second case, I think that the function is buggy; there's no reason to treat differently the empty fields on the left of a non-empty field, from the ones on the right.

Even if this behavior is considered correct, I think it's very valuable to document such cases, as they're not intuitive, especially the second.

History

#1 - 07/27/2022 06:25 PM - austin (Austin Ziegler)

scub8040 (Saverio M.) wrote:

There are other cases though, which I think are either incorrectly handled, or at least, they should documented.

The main case is a string exclusively composed of separators, e.g.:

"|||".split "|" # => []

Semantically speaking, such splitting does make sense, as an empty field is still a field. As the above example shows though, this returns an empty array (following the explained logic, it should return 4 empty strings).

IMO, this is incorrect. If for any reason this isn't, this should be documented though, as it's not obvious behavior (I've referred to this page: https://ruby-doc.org/core-3.0.0/String.html#method-i-split).

This is neither a behaviour bug nor a documentation bug.

From ri String#split:

If the limit parameter is omitted, trailing null fields are suppressed. If limit is a positive number, at most that number of split substrings will be returned (captured groups will be returned as well, but are not counted towards the limit). If limit is 1, the entire string is returned as the only entry in an array. If negative, there is no limit to the number of fields returned, and trailing null fields are not suppressed.
Emphasis added.

You get the behaviour you expect if you do:

```
"|||".split "|", -1 # => ["", ",", ",", ""]
```

#2 - 07/27/2022 06:42 PM - scub8040 (Saverio M.)
austin (Austin Ziegler) wrote in #note-1:

    scub8040 (Saverio M.) wrote:

    There are other cases though, which I think are either incorrectly handled, or at least, they should documented.

    This is neither a behaviour bug nor a documentation bug.

Uh, ok! Thanks.

EDIT: I've tried to close the issue, but couldn't.

#3 - 07/27/2022 06:44 PM - scub8040 (Saverio M.)

- Subject changed from String splitting handling of empty fields is incorrect or insufficiently documented to String splitting handling of empty fields is incorrect or insufficiently documented (SOLVED)

#4 - 07/28/2022 12:22 AM - znz (Kazuhiro NISHIYAMA)

- Status changed from Open to Closed