Allow a regexp as an argument to "count", to count more interesting things than single characters

09/11/2017 05:21 AM - duerst (Martin Dürst)

Currently, String#count only accepts strings, and counts all the characters in the string.

However, I have repeatedly met the situation where I wanted to count more interesting things in strings. These 'interesting things' can easily be expressed with regular expressions.

Here is a quick-and-dirty Ruby-level implementation:

```ruby
class String
  alias old_count count

  def count (what)
    case what
    when String
      old_count what
    when Regexp
      pos = -1
      count = 0
      count += 1 while pos = index(what, pos+1)
      count
    end
  end
end
```

Please note that the implementation counts overlapping occurrences; maybe there is room for an option like overlap: :no.

Related issues:
Related to Ruby master - Feature #12698: Method to delete a substring by regex match added

History

#1 - 09/11/2017 09:17 AM - Eregon (Benoit Daloze)
Should it behave the same as str.scan(regexp).size ?

I think the default should be no overlap, and increment the position by the length of the match.

#2 - 09/11/2017 09:35 AM - duerst (Martin Dürst)

Eregon (Benoit Daloze) wrote:

I think the default should be no overlap, and increment the position by the length of the match.

That would be fine by me, too.

#3 - 12/30/2017 04:37 AM - duerst (Martin Dürst)

Python allows to count strings, as follows:

```ruby
str.count(sub[, start[, end]])
```

Return the number of non-overlapping occurrences of substring sub in the range [start, end]. Optional arguments start and end are interpreted as in slice notation.

#4 - 11/19/2018 04:50 AM - duerst (Martin Dürst)

- Related to Feature #12698: Method to delete a substring by regex match added

08/25/2022