

URLs, Footnotes, and Links

Code Samples

```
helo world
<<(url>>https://link.com/2<<>>
<<(url>>https://example.com/a/uniform/resource<<>>
your main text^[the annotation]
```

URLs

https://link.com/2

https://example.com/a/uniform/resource

Yadda yadda https: / /example.com /a /uniform /resource /locator?x=42&y=108 yadda yadda yadda yadda yadda yadda.

An example https://example.com/a/uniform/resource for a URL. And yet http://x.com another one.

When URLs get longer, they can quickly interfere with line breaking. In order to mitigate unsightly effects, two measures have been taken. For one thing, breakpoints are inserted¹

https: / /example.com /a /uniform /resource /locator /commonly /informally /termed /a /web /address /is /a /reference /to /a /web /resource /that /specifies /its /location/on/a/computer/network/and/a/mechanism/for /retrieving/it

Footnotes

Two ways to markup footnotes

get realized as footnotes or endnotes; in the latter case, use <<!footnotes>> command to place them

your main text^[the annotation]

your main text²

Links

When MD link syntax is used [like here](https://example.com/#like) like here³, a footnote is generated; a footnote mark (a superscript number) is placed after the linked text, and the URL will appear in the footnotes wherever they will be generated.

Since the anchor text is in no way marked, it is possible to achieve the same effect by leaving the anchor text empty; however, if you intend to use the same source for other output formats, it would probably be a good idea to always use a non-empty anchor text.⁴

Notes

1) — breakpoints in URLs follow the rules of the Unicode Line Breaking Algorithm (<http://www.unicode.org/reports/tr14/#SampleCode>) as implemented by the NodeJS `linebreak` module (<https://github.com/devongovett/linebreak>).

2) — the annotation

3) — <https://example.com/#like>

4) — ... and not to use 'click here' and similar wordings

that only make sense in an interactive setting.