CO^{W}TCH$_{O}$OX 's page system

This document will provide examples on how to use CO^{W}TCH$_{O}$OX 's page system. Look a the source code (the `.cow` file) to better understand how it work.

Basic usage

Automatically, Cowtchoox will create page elements, and fill them with the content of the document. If an element is too big to fit in the page, it will cut it. Here you can see the paragraph element with a red border is automatically cut across pages (the big blank space is there to make sure it overflows):

Example

This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut! This paragraph is cut!

This paragraph is cut! This paragraph is cut!

Nonbreaking

You may also want to prevent this behavior on a specific element, for instance a title shouldn't be cut across pages. To do that, add the `nonbreaking` attribute to the element. The element will not be cut and will go entirely on the next page if it overflows.

```
<div nonbreaking>
    This div can't be cut!
</div>
```

Pagebreak

Finally, you can also use page breaks, with `<!pagebreak/>`. It will immediately start a new page. It can be used inside other elements. Look at this beautiful page break:

More things

You can also use the `stickafter` and `stickbefore` attributes.
An element with `stickafter` will always be on the same page as the next element in the document. (Useful for titles, which you don't want to be alone at the end of a page)
An element with `stickbefore` will always be on the same page as the previous element in the document. (Useful for captions of images, the caption must be on the same page than the image)

Headers and footers

You can insert a header and a footer on each page automatically. To do so, add the following tags in the document's head:

```
<head>
    /* Other things */

    <footer>path/to/the/footer.cowx</footer>
    <header>path/to/the/header.cowx</header>
</head>
```

Now you have to write the header and footer files. The header and footer files should contain one HTML element that represent the footer (or the header). Here is an example of a footer file:

```
<footer>
    This is an example footer with a page number: <!page-number/>
</footer>
```

A default footer (the one used in this document) is available in the default folder, you can use it with:

```
<head>
    /* Other things */
    <footer relative-to="default-dir">default/footer.cowx</footer>
</head>
```

Tip

To prevent the header from appearing on the first page, add this css rule somewhere:

```
#page-1 doc-header {
    display: none !important;
}
```

Thing you need to know for CSS

This paragraph describes what HTML elements Cowtchoox will create, so you can write accurate CSS selectors. (You can open `out.html` with your browser and open the devtools to see how it is arranged)

- The body will contain `page` elements. Their id is `page-n` where `n` is the number of the page (Starting from 1).
- Each `page` element contains:
  - A `page-inside` element, that contains the page's content

- If present, a `doc-footer` element, that contains the footer
- If present, a `doc-header` element, that contains the header