The Paged Web

No. 12 · November 2026
A newsletter about print CSS

Columns, floats and pages: a multi-column layout that keeps flowing from one page to the next, typeset from HTML by bdhtml2pdf.



Fig. 1. Pages are where print CSS starts.

Multi-column layout is the oldest trick in print and one of the hardest for HTML-to-PDF engines. The columns have to be balanced, headings have to span them, and when the content is longer than a page the columns have to continue on the next page instead of being cut off. Everything on this page was typeset from one HTML file with modern CSS: custom properties, oklch() colors, @page margin boxes for the footer, and the multi-column properties. The paragraphs themselves were inserted by a short script, which ran before the layout.

Columns that keep going

Each column here is filled from the top of the page to the bottom before the next one starts. Only on the last page are the columns balanced, so that the text ends level. Paragraphs break between lines, never inside a line, and margins at the top of a column are dropped. Fonts are embedded and subset, so the file stays small. The figures are inline SVG and stay vector graphics in the PDF, sharp at any zoom level. Nothing was loaded from the network while this document was produced.

Try it. This newsletter is a single HTML file. Its columns are filled page by page and the last page is balanced, as in browsers that print multi-column layouts.

A floated figure sits in the column of the text that flows beside it. When a column break would separate the figure from that text, the break moves before the text instead, so the figure never hangs off the end of a column. The headline above spans all three columns. Everything before it is balanced as one set of columns, and everything after it starts a new set, as the CSS Multi-column Layout specification describes for column-span: all.

Everything on this page was typeset from one HTML file with modern CSS: custom properties, oklch() colors, @page margin boxes for the footer, and the multi-column properties. The paragraphs themselves were inserted by a short script, which ran before the layout. Floats used to be a problem for column layout: a figure inside the flow made the whole block unbreakable, so it could overflow its column. Now the text wraps around the figure and the figure moves with the lines next to it.

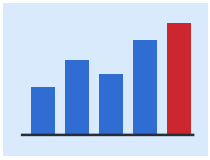


Fig. 2. SVG stays vector in the PDF.

> *“Text stays text: you can search it, select it and copy it.”*

Fonts are embedded and subset, so the file stays small. The figures are inline SVG and stay vector graphics in the PDF, sharp at any zoom level. Nothing was loaded from the network while this document was produced. Vertical margins collapse as they do in normal flow, both between paragraphs and through boxes without borders or padding. At a column break, the margin before the break is truncated, so every column starts flush with the top.

Floats in columns: the figure goes where its text goes

The headline above spans all three columns. Everything before it is balanced as one set of columns, and everything after it starts a new set, as the CSS Multi-column Layout specification describes for column-span: all. Long documents are the real test. A newsletter, a catalog or a report with several pages of columns must not lose a single line between pages. This one has enough text to need a second page.

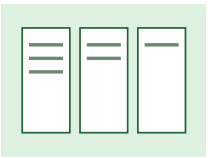


Fig. 3. Three columns, one flow.

Floats used to be a problem for column layout: a figure inside the flow made the whole block unbreakable, so it could overflow its column. Now the text wraps around the figure and the figure moves with the lines next to it. When a column set continues on the next page, it starts again at the top of the page area, below the running header if there is one, and the column rule is drawn on every page.

Margins, edges and breaks

Vertical margins collapse as they do in normal flow, both between paragraphs and through boxes without borders or padding. At a column break, the margin before the break is truncated, so every column starts flush with the top. That is what this sample shows. Download the PDF and select the text: every word is there, in reading order, column by column and page by page.

Long documents are the real test. A newsletter, a catalog or a report with several pages of columns must not lose a single line between pages. This one has enough text to need a second page. bdhtml2pdf is a free, open-source successor to wkhtmltopdf, written in Rust. It runs the page's JavaScript, styles the result with Firefox's CSS engine, and writes a vector PDF in a single small process.

Boxes break too. A box with a border, padding or a background that does not fit at the end of a column continues in the next one and is drawn in slices, like `box-decoration-break: slice`. This one is short, so it stays whole.

When a column set continues on the next page, it starts again at the top of the page area, below the running header if there is one, and the column rule is drawn on every page. Multi-column layout is the oldest trick in print and one of the hardest for HTML-to-PDF engines. The columns have to be balanced, headings have to span them, and when the content is longer than a page the columns have to continue on the next page instead of being cut off.

That is what this sample shows. Download the PDF and select the text: every word is there, in reading order, column by column and page by page. Each column here is filled from the top of the page to the bottom before the next one starts. Only on the last page are the columns balanced, so that the text ends level. Paragraphs break between lines, never inside a line, and margins at the top of a column are dropped.

bdhtml2pdf is a free, open-source successor to wkhtmltopdf, written in Rust. It runs the page's JavaScript, styles the result with Firefox's CSS engine, and writes a vector PDF in a single small process. A floated figure sits in the column of the text that flows beside it. When a column break would separate the figure from that text, the break moves before the text instead, so the figure never hangs off the end of a column.