Clamping card

In this demo, you are setting the width using `clamp()` like so: `width: clamp(<min>, <actual>, <max>)`.

This sets an absolute min and max size, and an actual size. With values, that can look like:

```
.parent {
  width: clamp(23ch, 60%, 46ch);
}
```

The minimum size here is `23ch` or 23 character units, and the maximum size is `46ch`, 46 characters. Character width units are based on the font size of the element (specifically the width of the `0` glyph). The 'actual' size is 50%, which represents 50% of this element's parent width.

What the `clamp()` function is doing here is enabling this element to retain a 50% width *until* 50% is either greater than `46ch` (on wider viewports), or smaller than `23ch` (on smaller viewports). In the video, watch how the width of this card increases to its clamped maximum point and decreases to its clamped minimum as the parent stretches and shrinks. It then stays centered in the parent using additional properties to center it. This enables more legible layouts, as the text won't be too wide (above `46ch`) or too squished and narrow (less than `23ch`).

You can use this technique to implement responsive typography. For example: `font-size: clamp(1.5rem, 20vw, 3rem)`. In this case, the font-size of a headline would always stay clamped between `1.5rem` and `3rem` but would grow and shrink based on the `20vw` actual value to fit the width of of the viewport.

This is a great technique to ensure legibility with a minimum and maximum size value, but remember it is not supported in all modern browsers so make sure you have fallbacks and do your testing.

□□□ #2
□ □□□ □□□ 2026-10-31 23:03:07 UTC
□ □□□ □□□ 2026-10-31 23:03:08 UTC