CSS Markers

I've always been curious about how people create customized list marker items, the bullets or numbers that appear in lists for webpages.

Firefox shipped an easier way to customize these lists using ::marker pseudo elements.

The following example replaces all the markers for an unordered list with ✅. It also changes the text of the default markers so that if the emoji cannot be displayed for some reason, the default markers will be customized without the emoji.

We add a second selector for the `li` element without the marker so we can introduce spacing between the marker and the content to the item.

```css
@supports selector(::marker) {
  ul li::marker {
    color: rebeccapurple;
    font-family: cursive;
    font-size: 2em;
    font-weight: 900;
    content: '✅';
  }

  '✅'ul li {
    padding-left: 1em;
  }
}
```

The example is wrapped in a selector feature query to make sure that markers are supported without making a mess off the existing design.

Jason Pamental's Responsive Web Typography Newsletter issue 33 proposes an alternative that uses counters and ::before pseudo elements to accomplish a similar effect.

```css
ol {
  list-style: none;
  counter-reset: list-counter;
}

ol li {
  font-family: Merriweather, serif;
  counter-increment: list-counter;
}

ol li::before {
  color: #c05522;
  display: block;
  float: left;
  font-family: Montserrat, sans-serif;
  margin-left: -1.25em;
  content: counter(list-counter) ". ";
}
". "
```

The main differences are:

The second example removes the number marker before inserting content with the ::before pseudo element.

The example without markers uses counters to increase their value. This would make continuation lists where we start a list from values other than 1 problematic.

Make sure that you test the solution with accessibility tools to make sure it doesn't cause any problems.