

Writing CSS in 100 Years

Ire Aderinokun @ dotCSS

So, how are we writing
CSS in 100 years?

Lots of new features!

JS in **CSS!**

**this is totally fake*

```
main {  
    width: document.body.clientWidth > 800 ? "700px" : "100%";  
}
```

But still, we **don't break**
the web

Use the **cascade**

```
main {  
    width: 100%;  
    width: document.body.clientWidth > 800 ? "700px" : "100%";  
}
```

Use the **cascade**

```
main {
```

```
    width: 100%;
```

```
    width: document.body.clientWidth > 800 ? "700px" : "100%";
```

```
}
```

```
body {  
    display: beyonce🔥;  
    beyonce🔥-template-areas: "header header header"  
                               "navigation main ads"  
                               "footer footer footer";  
    beyonce🔥-template-columns: 150px 1fr 150px;  
    beyonce🔥-template-rows: 100px 1fr 30px;  
}
```

Use **feature queries**

```
body { /* fallback rules here */ }
```

```
@supports (display: beyonce🔥) {  
  body { /* awesome rules here */ }  
}
```

CSS Feature Queries [↗](#)

CSS Feature Queries allow authors to condition rules based on whether particular property declarations are supported in CSS using the @supports at rule.

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	16	68	76	12	12.4			4.4	8.2
10	17	69	77	12.1	13.1			4.4.4	9.2
11	18	70	78	13	13.2	all	78	76	10.1
	76	71	79	TP	13.3				
		72	80						

✓ ✗ Partial Support

Global: 95.73% + 0% = 95.73%

Use **positive** feature queries

```
@supports (display: beyonce🔥) {  
  body { /* awesome rules here */ }  
}
```

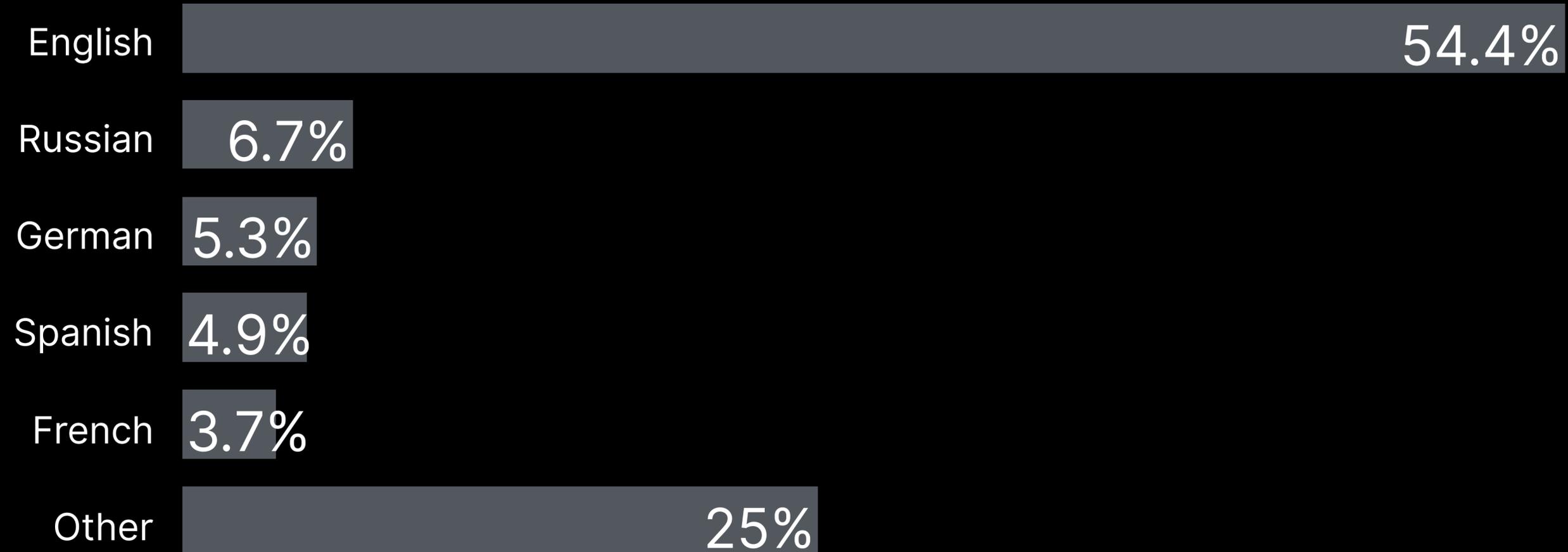
Avoid **negative** feature queries

```
@supports not (display: beyonce🔥) {  
  body { /* fallback rules here */ }  
}
```

Everyone is online



In 2019, 54% of the web is **English**



In 2019, only 41% of the “**developing world**” is online

■ Online ■ Offline

Developing World

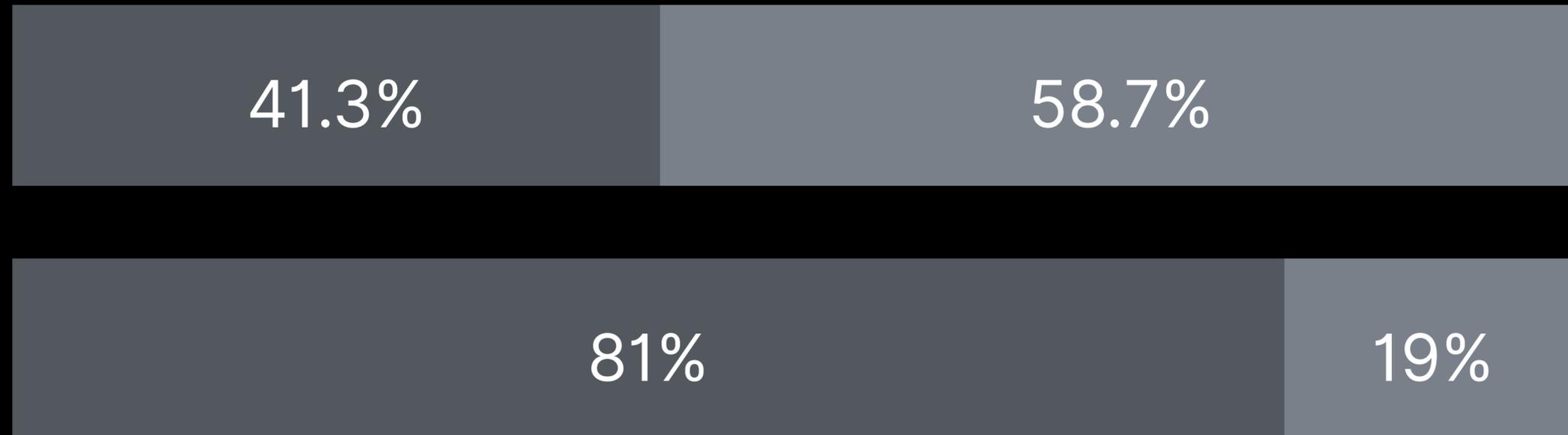
41.3%

58.7%

Developed World

81%

19%



In 100 years,
100% of humans are online

3 key considerations

 Internationalisation

 Accessibility

 Personalisation

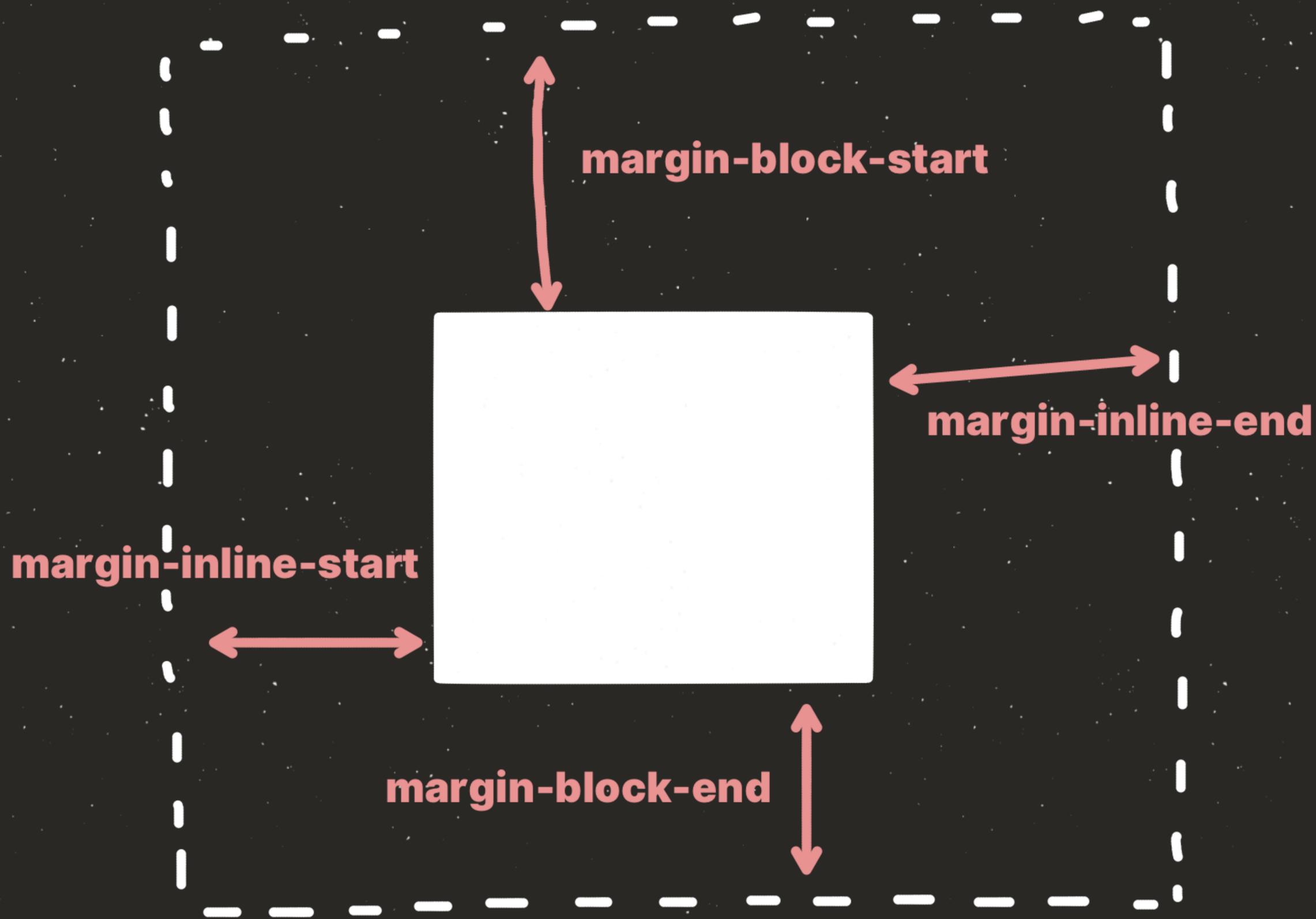
 We can't assume LtR-TtB

Left usually doesn't mean left

```
p {  
    margin-left: 10px;  
}
```

Left usually doesn't mean left

```
html[lang="ar"] p {  
    margin-right: 10px;  
}
```



Use **logical properties**

```
p {  
    margin-inline-start: 10px;  
}
```



This is a test of CSS logical properties

10px

هذا اختبار للخصائص المنطقية لـ CSS



10px

 Don't break the a11y of HTML

“HTML is **by default accessible**.

As developers, it's our job to

not fck that up”

Don't use CSS for **content**

```
input[type="password"]::before {  
  content: "Please enter your password";  
}
```



Don't **undo** accessible styles

```
*:focus {  
    outline: none; 🤔  
}
```

Style **responsibly**

- Contrast
- Colours
- Text and element sizes

 Adapt styling to user preferences

Adapt to **animation** preferences

```
.element { animation: 3s infinite alternate slideIn; }
```

```
@media (prefers-reduced-motion: reduce) {
```

```
  .element { animation: none; }
```

```
}
```

Adapt to **colour** preferences

```
body { background-color: white; color: black; }
```

```
@media (prefers-color-scheme: dark) {
```

```
    body { background-color: black; color: white; }
```

```
}
```

Adapt to **data** preferences

**not yet available*

```
.element { background-image: url("image-1800w.png"); }
```

```
@media (prefers-reduced-data: reduce) {
```

```
  .element { background-image: url("image-600w.jpg"); }
```

```
}
```

A dark, top-down view of a bed with a white quilted blanket. A laptop is open on the right side, and a tablet is lying flat on the left side. A mouse is visible in the bottom right corner. The text is overlaid in the center of the image.

Everyone is online,
but **not everyone** is on the
iPhone 100 ProMaxPlusXL

 Performant CSS **matters**

Be careful of **expensive animations**

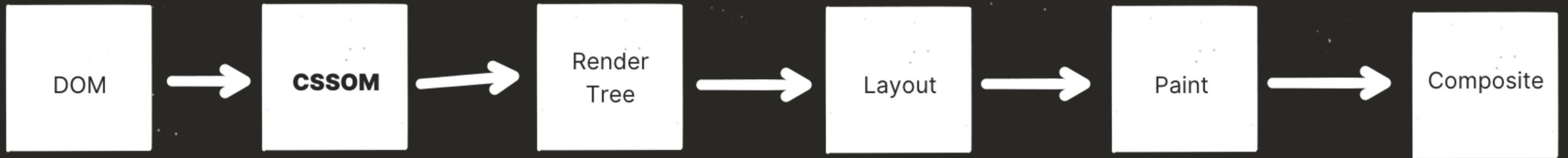
```
button {  
    top: 0;  
    transition: top 1s; 🤔  
}  
  
button:hover { top: 10px; }
```

Be careful of **expensive animations**

```
button {  
    transform: none;  
    transition: transform 1s;  
}  
  
button:hover { transform: translateY(10px); }
```

CSS is *still* **render-blocking**

**Blocking the
page render**



Inline **critical CSS**

```
<head>
```

```
  <style id="critical"> #hero { ... } </style>
```

```
  <script> loadCSS("full.css"); </script>
```

```
</head>
```

Split stylesheets based on media type

```
<link rel="stylesheet"  
      media="(max-width: 600px)" src="mobile.css">  
  
<link rel="stylesheet"  
      media="(min-width: 600px)" src="desktop.css">
```

Split stylesheets based on media type

```
<link rel="stylesheet"
```

```
  media="(prefers-color-scheme: light)" src="light.css">
```

```
<link rel="stylesheet"
```

```
  media="(prefers-color-scheme: dark)" src="dark.css">
```

Preload stylesheets

```
<head>
```

```
  <link rel="preload" as="style" href="main.css">
```

```
  ...
```

```
  <link rel="stylesheet" href="main.css">
```

```
</head>
```

HTTP/2 Push

```
<FilesMatch ".html$">
```

```
    Header set Link "</main.css>; rel=preload; as=style"
```

```
</FilesMatch>
```

Writing CSS in ~~2119~~ **2019**

- ✓ Progressively enhance the experience with new features
- ✓ Use logical properties that are adaptable to different languages
- ✓ Don't break the accessibility of HTML
- ✓ Adapt styling to user preferences
- ✓ Beware of expensive CSS animations
- ✓ Serve smaller, modular stylesheets wherever possible
- ✓ Pre-emptively load stylesheets

Thank you!

Where to find me

- ireaderinokun.com
- bitsofco.de
- [@ireaderinokun](https://twitter.com/ireaderinokun)

Further reading & sources

- [CSS Logical Properties \(Working Draft\)](#)
- [Proposal: prefers-reduced-data](#)
- [loadCSS\(\)](#)
- [Preload \(Candidate Recommendation\)](#)