

Cosas que debes saber sobre Modern CSS para el 2025

...

Raúl Ruiz Ramírez
3raul.com

Estilo



Agenda

- ¿Quién soy?
- ¿Qué es Modern CSS?
- Principios clave
- Adiciones a CSS

¿Quien soy?



- Cofundador de WeBorder
- Senior Engineer en Care.com
- Egresado de la UACJ en Sistemas Digitales y Comunicaciones.

¿Qué es Modern CSS?

- No se refiere a una versión específica de CSS
- Conjunto de principios y prácticas que buscan optimizar el desarrollo web
- Características como CSS Grid, Cascade Layers, selectores condicionales y nuevos espacios de color

Principios clave

Modularidad

Dividir código en componentes reutilizables, puedes utilizar diferentes técnicas como nesting o BEM.

```
<div class="card">
  <div class="card__title">Titulo 1</div>
</div>

<div class="card card--blue">
  <div class="card__title">Titulo 2</div>
</div>
```

```
.card {
  background-color: black;
  width: 500px;
  padding: 2rem;
  margin-bottom: 1rem;
}

.card__title {
  color: white;
}

.card--blue {
  background-color: blue;
}
```

Rendimiento

Escribir CSS eficiente para mejorar la velocidad de carga de las páginas web.

1. Especificidad de selectores.
 - a. El navegador tarda más en procesar selectores complejos y anidados.
2. Evitar el uso excesivo de @import.
 - a. Cada @import genera una nueva petición al servidor, lo que puede retrasar la carga.
3. Reducir el número de reglas
 - a. Mientras más reglas CSS tenga el navegador que procesar, más tiempo tardará en cargar la página.
4. Optimizar las propiedades font-family
 - a. Cargar muchas fuentes puede afectar significativamente el rendimiento.
5. Shorthand properties
 - a. Las propiedades abreviadas (shorthand) permiten definir varios valores en una sola línea.

```
/* No */
header nav ul li a.active {
|   background-color: red;
}

/* Si */
.link--active {
|   background-color: red;
}
```


Otros principios clave

```
@import "perro.css";

:root {
  --default-bg-color: #253179;
  --default-hover-color: #002060;
}

.card {
  background-color: var(--default-bg-color);
  width: 500px;
  padding: 2rem;
  margin-bottom: 1rem;
  cursor: pointer;

  &:hover {
    background-color: var(--default-hover-color);
  }
}

header {
  display: flex;
  flex-direction: column;
}

nav {
  container-name: navContainer;
  container-type: inline-size;
  flex-basis: 50%;
}
```

1. Mantenibilidad

- a. Utilizar una estructura de código clara y organizada para facilitar la lectura y la modificación del CSS.

2. Escalabilidad

- a. Escribir CSS que pueda adaptarse fácilmente a diferentes tamaños de pantalla y dispositivos.

3. Accesibilidad

- a. Asegurar que el CSS sea accesible para todos los usuarios, incluyendo aquellos con discapacidades.

Adiciones CSS

Anidamiento de reglas (nesting)

Permite agrupar selectores CSS de forma más organizada y legible, mejorando la mantenibilidad del código

```
.card {  
  background-color: var(--default-bg-color);  
  width: 500px;  
  padding: 2rem;  
  margin-bottom: 1rem;  
  
  &:hover {  
    background-color: var(--default-hover-color);  
  }  
}
```

```
.component {  
  &__child-element {  
  }  
}  
  
/* In Sass this becomes */  
.component__child-element {  
}
```

Container queries

Permiten aplicar estilos a los elementos en función del tamaño de su contenedor padre

```
<body>
  <header>
    <h1>Bienvenido a mi página</h1>
    <nav>
      <ul>
        <li>Menu 1</li>
        <li>Menu 2</li>
        <li>Menu 3</li>
      </ul>
    </nav>
  </header>
  <main>
    <section class="section-one">Section one</section>
    <div class="card">
      <div class="card__title">Titulo 1</div>
    </div>

    <div class="card card--blue">
      <div class="card__title">Titulo 2</div>
    </div>
  </main>
</body>
```

```
17
18 header {
19   display: flex;
20 }
21
22 nav {
23   container-name: navContainer;
24   container-type: inline-size;
25   width: 50%;
26 }
27
28 @container navContainer (max-width: 500px) {
29   /* no aplica porque el elemento esta fuera del con
30   .section-one {
31     border: 2px solid black;
32   }
33   /* Este si aplica */
34   li {
35     border: 2px solid black;
36   }
37 }
```

Rienvenuto a mi página

nav 620 × 86

- Menu 1
- Menu 2
- Menu 3

Section one

Rienvenuto a mi página

nav 488 × 98

- Menu 1
- Menu 2
- Menu 3

Section one

El pseudo selector :has()

Permite seleccionar elementos padre que contienen elementos específicos

```
.card {  
  background-color: var(--default-bg-color);  
  width: 500px;  
  padding: 2rem;  
  margin-bottom: 1rem;  
  cursor: pointer;  
  
  &:hover {  
    background-color: var(--default-hover-color);  
  }  
  
  &:has(picture) {  
    display: flex;  
  }  
}  
  
.card__title {  
  color: white;  
  
  &:has(+ picture) {  
    flex-grow: 1;  
    border: 2px solid black;  
    padding: 1rem;  
    text-align: center;  
  }  
}
```

```
<div class="card">  
  <div class="card__title">Titulo 1</div>  
</div>  
  
<div class="card card--blue">  
  <div class="card__title">Titulo 2</div>  
  <picture>  
    <img  
      src="https://upload.wikimedia.org/wikipedia/commons/t  
    />  
  </picture>  
</div>
```

Titulo 1

Titulo 2



Otras adiciones a CSS

- Subgrid
- Colores P3
- view transitions
- Scroll-Driven Animations
- @property
- @scope
- zoom
- mod() - operador modulo
- round()
- text-wrap

¡Muchas gracias!

Raúl Ruiz Ramírez

3raul.com