



Carmelo José Jaén Díaz



Nombres de clases: mejores prácticas y ejemplos

-
- C.F.G.S. DAW
 - 6 horas semanales
 - Mes aprox. de impartición: Nov - Dic
 - iPasen - cjaedia071@g.educaand.es

Índice



Objetivo

Glosario

Interacción persona - ordenador

Objetivos

Características. Usable.

Características. Visual.

Características. Educativo y actualizado.

OBJETIVO



- **Analizar y seleccionar los colores y las tipografías adecuados para la visualización en la pantalla.**
- **Utilizar marcos y tablas para presentar la información de manera ordenada.**
- **Identificar nuevos elementos y etiquetas en HTML5.**
- **Reconocer las posibilidades de modificar etiquetas HTML.**
- **Valorar la utilidad de las hojas de estilo para conseguir un diseño uniforme en todo el sitio web.**

GLOSARIO



Formularios. Documentos interactivos utilizados para recoger información en un sitio web. Esta información es enviada al servidor, donde es procesada. Cada formulario contiene uno o varios tipos de controles que permiten recolectar la información de varias formas diferentes.

Fuentes seguras. Fuentes tipográficas que los usuarios tenían instaladas por defecto en su dispositivo. En la actualidad, gracias a que la mayoría de los navegadores soportan la directiva @font-face, es posible utilizar casi cualquier tipografía a través de Google Fonts.

Guías de estilo. Documentos con directrices que permiten la normalización de estilos. En estas guías se recogen los criterios y normas que debe seguir un proyecto; de esta forma se ofrece una apariencia más uniforme y atractiva para el usuario.

HTML. Lenguaje de marcado de hipertexto utilizado en las páginas web. Este tipo de lenguaje presenta una forma estructurada y agradable, con hipervínculos que conducen a otros documentos y con inserciones multimedia (sonido, imágenes, vídeos...).

GLOSARIO



HTML5. Última versión del lenguaje para la programación de páginas web HTML. Los sitios implementados con este lenguaje solo pueden visualizarse correctamente en los navegadores más actuales.

Legibilidad. Cualidad deseable en una familia tipográfica. Se trata de la facilidad de la lectura de una letra. Esta cualidad puede venir determinada por varios parámetros como el interletrado, el interpalabrado o el interlineado.

Marcos. Son las ventanas independientes incorporadas dentro de la página general. Gracias a ellos, cada página quedará dividida en varias subpáginas, permitiendo realizar un diseño más organizado y limpio a la vista. Con HTML5 ha quedado obsoleto.

Tipografía. Se trata del tipo de letra que se escoge para un determinado diseño. Según la RAE, significa "modo o estilo en que está impreso un texto" o "clase de tipos de imprenta".

INTRODUCCIÓN



Cuando trabajas en proyectos de desarrollo web, una de las tareas más importantes es elegir los **nombres de clases en CSS** de manera consistente y significativa. Una buena convención de nombres de clases no solo hace que tu código sea más legible y mantenible, sino que también facilita la colaboración con otros miembros del equipo.

En este artículo, se estudian buenas prácticas a la hora de elegir los nombres de nuestras clases CSS y varias convenciones de nombres de clases, junto con ejemplos prácticos para cada una.



NOMBRES DE CLASES EN CSS: MEJORES PRÁCTICAS

Guiones (-) vs. guiones bajos (_)



Cuando se trata de nombrar clases en CSS, seguir las mejores prácticas es esencial para mantener un código claro, organizado y fácil de mantener. En esta sección, se exponen una serie de consideraciones clave para nombrar clases en CSS de manera efectiva.

La elección entre guiones y guiones bajos para separar palabras en nombres de clases es una decisión importante. Ambos son comunes en CSS, pero es importante elegir uno y mantener la consistencia en todo el código.

/ Ejemplo con guiones */*

```
.header-menu {}
```

/ Ejemplo con guiones bajos */*

```
.header_menu {}
```

NOMBRES DE CLASES EN CSS: MEJORES PRÁCTICAS

Mayúsculas o minúsculas



Hay dos enfoques principales cuando se trata de capitalizar palabras en nombres de clases:

```
/* Con CamelCase (primera letra  
de cada palabra en mayúscula) */  
.HeaderMenu {}
```

```
/* Con minúsculas */  
.headermenu {}
```


NOMBRES DE CLASES EN CSS: MEJORES PRÁCTICAS

Separación de palabras



Independientemente de si se usan guiones o guiones bajos, es importante separar palabras de manera clara y significativa para mejorar la legibilidad del código.

```
/* Clase bien separada */  
.header-menu {}
```

NOMBRES DE CLASES EN CSS: MEJORES PRÁCTICAS

Breve y descriptivo



Intenta mantener los nombres de clases cortos pero descriptivos. Evita abreviaturas confusas y opta por nombres que reflejen claramente el propósito de la clase.

```
/* Clase confusa y abreviada */  
.hm {}  
  
/* Clase descriptiva y clara */  
.header-menu {}
```

NOMBRES DE CLASES EN CSS: MEJORES PRÁCTICAS

Separar la estructura del estilo



Es importante separar la estructura del estilo al nombrar clases en CSS. Evita nombrar clases en función de la apariencia visual o del diseño específico, ya que estos pueden cambiar con el tiempo. En su lugar, nombra las clases basadas en la función o el propósito del elemento dentro del contexto de la aplicación.

```
/* Mal - Basado en la apariencia visual */  
.big-red-button {}
```

```
/* Bien - Basado en la función */  
.primary-button {}
```

NOMBRES DE CLASES EN CSS: MEJORES PRÁCTICAS

Mantener la escalabilidad



Al nombrar clases, piensa en la escalabilidad del proyecto. Es importante elegir nombres que sean lo suficientemente genéricos como para ser aplicables en diferentes contextos y situaciones.

```
/* Escalable y reutilizable */
```

```
.link {}
```

```
/* No específico y poco escalable:  
solo necesario si el link del footer  
es diferente a cualquier link del sitio*/
```

```
.footer-link {}
```

NOMBRES DE CLASES EN CSS: EJEMPLOS

BEM (Block Element Modifier)



Lo más importante en los nombres de clases en CSS es mantener la consistencia dentro de un proyecto y seguir los convenios establecidos por el equipo de desarrollo. Si el equipo ya está utilizando un enfoque específico, es mejor seguirlo para evitar confusiones y mantener un código uniforme. Veamos algunas convenciones que se pueden utilizar.

La convención BEM divide los nombres de clase en tres partes: Bloque, Elemento y Modificador. Esto proporciona una estructura clara y predecible para organizar los estilos CSS. A continuación te mostramos algunos ejemplos:

/ Bloque */*

`.header {}`

/ Elemento */*

`.header_logo {}`

/ Modificador */*

`.header_logo-large {}`

NOMBRES DE CLASES EN CSS: EJEMPLOS

OOCSS (Object-Oriented CSS)



OOCSS se centra en la reutilización de estilos mediante la creación de clases independientes y modulares. Veamos algunos ejemplos:

```
/* Clase base */  
.button {}
```

```
/* Modificador */  
.button-primary {}
```

```
/* Modificador */  
.button-large {}
```

NOMBRES DE CLASES EN CSS: EJEMPLOS

SMACSS



SMACSS (Scalable and Modular Architecture for CSS) proporciona pautas para organizar y estructurar las hojas de estilo de manera más escalable y modular. A continuación puedes ver algunos ejemplos:

```
/* Componente */
```

```
.menu {}
```

```
/* Elemento */
```

```
.menu_item {}
```

```
/* Estado */
```

```
.menu_item-active {}
```

NOMBRES DE CLASES EN CSS: EJEMPLOS

Atomic CSS



Atomic CSS se basa en la idea de crear clases para estilos atómicos, es decir, estilos pequeños y reutilizables que se aplican individualmente a los elementos. Veamos algunos ejemplos:

```
/* Alineación de texto */  
.text-center {}
```

```
/* Color de fondo */  
.bg-blue {}
```

```
/* Margen */  
.m-2 {}
```


NOMBRES DE CLASES EN CSS: EJEMPLOS

Nombres semánticos



Utilizar nombres semánticos ayuda a mantener un código más descriptivo y fácil de entender. Veamos algunos ejemplos:

```
/* Botón principal */  
.primary-button {}
```

```
/* Menú de navegación */  
.navigation-menu {}
```

```
/* Avatar de perfil */  
.profile-avatar {}
```

NOMBRES DE CLASES EN CSS: EJEMPLOS



Al elegir una convención de nombres de clases en CSS, es importante considerar las necesidades específicas de tu proyecto y equipo de desarrollo. Lo más importante es mantener la consistencia en todo el código para facilitar el mantenimiento y la colaboración.

Espero que esta lección te haya proporcionado una visión general de las convenciones de nombres de clases en CSS y te ayude a elegir la mejor opción para tus proyectos.