

ÍNDICE

Introducción a CSS y selectores



Características básicas	4
Incluir CSS en un documento XHTML	8
Glosario básico	14
Comentarios en CSS	18
Selectores	22

INTRODUCCIÓN A CSS Y SELECTORES

Podemos definir el CSS como un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas.

 Este manual no es una referencia de propiedades CSS. Para una consulta exhaustiva y actualizada de propiedades y valores CSS por favor consultar:

En inglés:

<https://www.w3schools.com/cssref/default.asp>

En castellano:

https://developer.mozilla.org/es/docs/Web/CSS/Referencia_CSS

01

CARACTERÍSTICAS BÁSICAS

La utilización del lenguaje CSS para definir el aspecto de cada elemento de una página web

Al crear una página web, se utiliza en primer lugar el lenguaje HTML/XHTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos, etc.

Una vez creados dichos contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada uno de los elementos: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, etc.

FUNCIONAMIENTO BÁSICO DE CSS

Antes de la adopción de CSS, los diseñadores de páginas web debían definir el aspecto de cada elemento dentro de las etiquetas HTML de la página. Por ejemplo, utilizando la etiqueta `` con sus atributos `color`, `face` y `size` para definir el color, la tipografía y el tamaño del texto de cada elemento del documento.

El principal problema de esta forma de definir el aspecto de los elementos es que habría que insertar tantas etiquetas `` como elementos diferentes en toda la web. Por ejemplo: si la página tuviera 50 elementos diferentes, habría que insertar 50 etiquetas ``. Si el sitio web entero se compone de 10.000 páginas diferentes, habría que definir 500.000 etiquetas ``. Como cada etiqueta `` tiene 3 atributos, habría que definir 1.5 millones de atributos.

Por otra parte, el diseño de las páginas web están en constante evolución y es habitual modificar cada cierto tiempo su aspecto. Si se emplea la etiqueta ``, habría que modificar el valor de 1.5 millones de atributos para modificar el diseño general del sitio web.



La solución que propone CSS es mucho mejor: **CSS permite separar los contenidos de la página y su aspecto o presentación**. En el ejemplo anterior, dentro de la propia página HTML se reserva una zona en la que se incluye toda la información relacionada con los estilos de la página.

Utilizando CSS, en esa zona reservada se indica que todas las etiquetas `<h1>` de la página se deben ver de color rojo, con un tipo de letra Arial y con un tamaño de letra grande. Además, las etiquetas `<p>` de la página se deben ver de color gris, con un tipo de letra Verdana y con un tamaño de letra medio.

Definiendo los estilos de esta forma, no importa el número de elementos `<p>` que existan en la página, ya que todos tendrán el mismo aspecto establecido mediante CSS. Como se verá a continuación, esta forma de trabajar con CSS no es ideal, ya que si el sitio web dispone de 10.000 páginas, habría que definir 10.000 veces el mismo estilo CSS.



02

INCLUIR CSS EN UN DOCUMENTO XHTML

CSS aporta flexibilidad y ofrece diferentes opciones para realizar una misma tarea

Introducción a CSS y selectores

Una de las principales características del CSS es que permite tres opciones distintas para incluir los estilos en un documento HTML:

- 1 INCLUIR CSS EN EL MISMO DOCUMENTO HTML
- 2 DEFINIR CSS EN UN ARCHIVO EXTERNO
- 3 INCLUIR CSS EN LOS ELEMENTOS HTML

1. INCLUIR CSS EN EL MISMO DOCUMENTO HTML

Los estilos se definen en una zona específica del propio documento HTML. **Se emplea la etiqueta `style` de HTML y solamente se pueden incluir en la cabecera del documento** (sólo dentro de la sección `<head>`).

Este método se emplea cuando se define un número pequeño de estilos o cuando se quieren incluir estilos específicos en una determinada página HTML que completen los estilos que se incluyen por defecto en todas las páginas del sitio web.

El principal inconveniente es que si se quiere hacer una modificación en los estilos definidos, es necesario modificar todas las páginas que incluyen el estilo que se va a modificar.



2. DEFINIR CSS EN UN ARCHIVO EXTERNO

En este caso, **todos los estilos CSS se incluyen en un archivo simple de texto con la extensión `.css` que las páginas HTML enlazan mediante la etiqueta `<link>`**. Se pueden crear todos los archivos CSS que sean necesarios y cada página HTML puede enlazar tantos archivos CSS como necesite.

Si se quieren incluir los estilos en un archivo CSS externo, se deben seguir los siguientes pasos:

- 1. Se crea un archivo de texto y se le añade solamente el contenido definiendo los estilos.
- 2. Se guarda el archivo de texto con el nombre **estilos.css**
Se debe poner especial atención a que el archivo tenga extensión **.css** y no **.txt**.
- 3. En la página HTML se enlaza el archivo CSS externo mediante la etiqueta **<link>**.

Cuando el navegador carga la página HTML anterior, antes de mostrar sus contenidos también descarga los archivos CSS externos enlazados mediante la etiqueta `<link>` y aplica los estilos a los contenidos de la página.

Normalmente, la etiqueta `<link>` incluye cuatro atributos cuando se enlaza un archivo CSS:

- **rel:** indica el tipo de relación que tiene el recurso enlazado (en este caso, el archivo CSS) y la página HTML. Para los archivos CSS, siempre se utiliza el valor **stylesheet**.
- **type:** indica el tipo de recurso enlazado. Sus valores están estandarizados y para los archivos CSS su valor siempre es **text/css**.

- **href:** indica la URL del archivo CSS que contiene los estilos. La URL indicada puede ser relativa o absoluta y puede apuntar a un recurso interno o externo al sitio web.
- **media:** indica el medio en el que se van a aplicar los estilos del archivo CSS.

De todas las formas de incluir CSS en las páginas HTML, esta es la más utilizada con mucha diferencia. La principal ventaja es que se puede incluir un mismo archivo CSS en multitud de páginas HTML, por lo que se garantiza la aplicación homogénea de los mismos estilos a todas las páginas que forman un sitio web.

Con este método, el mantenimiento del sitio web se simplifica al máximo, ya que un solo cambio en un solo archivo CSS permite variar de forma instantánea los estilos de todas las páginas HTML que enlazan ese archivo.

Aunque generalmente se emplea la etiqueta `<link>` para enlazar los archivos CSS externos, también se puede utilizar la etiqueta `<style>`. En este caso, para incluir en la página HTML los estilos definidos en archivos CSS externos se utiliza una regla es-



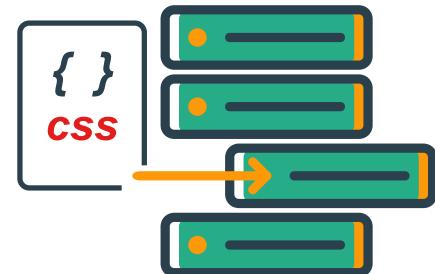
pecial de tipo `@import`. Las reglas de tipo `@import` siempre preceden a cualquier otra regla CSS (con la única excepción de la regla `@charset`). La URL del archivo CSS externo se indica mediante una cadena de texto encerrada con comillas simples o dobles o mediante la palabra reservada `url()`. De esta forma, las siguientes reglas `@import` son equivalentes:

```
@import '/css/estilos.css';  
@import "/css/estilos.css";  
@import url('/css/estilos.css');  
@import url("/css/estilos.css");
```

3. INCLUIR CSS EN LOS ELEMENTOS HTML

El último método para incluir estilos CSS en documentos HTML es el peor y el menos utilizado, ya que tiene los mismos problemas que la utilización de las etiquetas ``.

Esta forma de incluir CSS directamente en los elementos HTML solamente se utiliza en determinadas situaciones en las que se debe incluir un estilo muy específico para un solo elemento concreto.



03

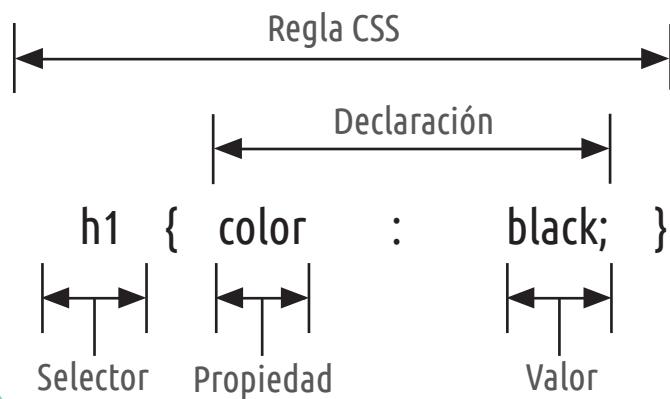
GLOSARIO BÁSICO

Definición de los términos fundamentales de cada una de las partes que componen los estilos CSS

Al definir un estilo CSS se utilizan reglas de texto simples para describir el aspecto de cada uno elementos que componen el estilo de la página. Los componentes que forman un estilo básico de CSS son: **regla, selector, declaración, propiedad y valor**.

Un archivo CSS puede contener en su sintaxis infinitas reglas CSS, cada regla puede contener infinitos selectores y cada declaración puede estar formada por un número infinito de pares propiedad/valor.

COMPONENTES DE UN ESTILO CSS BÁSICO



1

REGLA

Cada uno de los estilos que componen una hoja de estilos CSS. Cada regla está compuesta de una parte de "selectores", un símbolo de "{", otra parte denominada "declaración" y por último, un símbolo de "}".

2

SELECTOR

Indica el elemento o elementos HTML a los que se aplica la regla CSS.

3

DECLARACIÓN

Especifica los estilos que se aplican a los elementos. Está compuesta por una o más propiedades CSS.

4

PROPIEDAD

Permite modificar el aspecto de una característica del elemento.

5

VALOR

Indica el nuevo valor de la característica modificada en el elemento.

04

COMENTARIOS EN CSS

CSS permite incluir comentarios entre sus reglas y estilos para ayudar a entender su funcionamiento

Al igual que en HTML, se aconseja incluir comentarios en el CSS para ayudar a entender el funcionamiento del código al tiempo de haberlo programado, además de ayudar a otros a entenderlo. Los comentarios son contenidos de texto que el diseñador incluye en el archivo CSS para su propia información y utilidad.

Los navegadores ignoran por completo cualquier comentario de los archivos CSS, por lo que es común utilizarlos para estructurar de forma clara los archivos CSS complejos.

El comienzo de un comentario se indica mediante los caracteres `/*` y el final del comentario se indica mediante `*/`, tal y como se muestra en el siguiente ejemplo:

```
/* Este es un comentario en CSS */
```

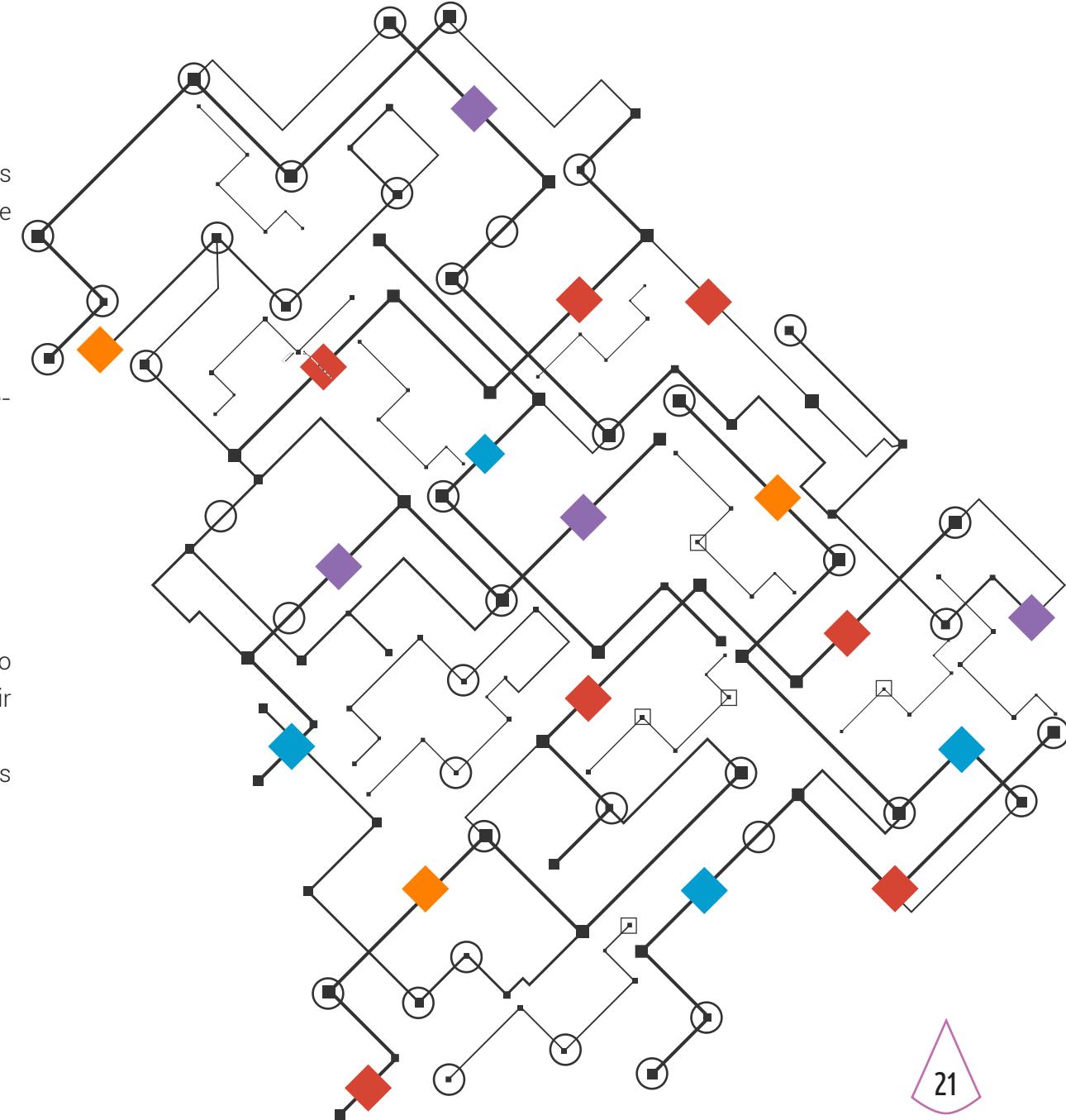
Los comentarios pueden ocupar tantas líneas como sea necesario, pero no se puede incluir un comentario dentro de otro:

```
/* Este es un  
comentario CSS de varias  
líneas */
```

Aunque los navegadores ignoran los comentarios, su contenido se envía junto con el resto de estilos, por lo que no se debe incluir en ellos ninguna información sensible o confidencial.

La sintaxis de los comentarios CSS es muy diferente a la de los comentarios HTML, por lo que no deben confundirse:

```
<!-- Este es un comentario en HTML -->  
<!-- Este es un  
comentario HTML de varias  
líneas -->
```



05

SELECTORES

Para crear diseños web profesionales es esencial conocer y dominar los selectores de CSS

Una regla de CSS, por muy básica que sea, está formada por una parte llamada "**selector**" y otra parte llamada "**declaración**". La declaración indica "qué hay que hacer" y el selector indica "a quién hay que hacérselo".

Por lo tanto, **los selectores son imprescindibles para aplicar de forma correcta los estilos CSS en una página.**

A un mismo elemento HTML se le pueden asignar infinitas reglas CSS y cada regla CSS puede aplicarse a un número infinito de elementos. Dicho en otras palabras, una misma regla puede aplicarse sobre varios selectores y un mismo selector se puede utilizar en varias reglas.

SELECTORES BÁSICOS

Existen diferentes tipos de selectores que permiten seleccionar de forma muy precisa elementos individuales o conjuntos de elementos dentro de una página web. No obstante, la mayoría de páginas de los sitios web se pueden diseñar utilizando solamente los cinco selectores básicos.

1. Selector universal

Se utiliza para seleccionar todos los elementos de la página y se indica mediante un asterisco (*). A pesar de su sencillez, no se utiliza habitualmente, ya que es difícil que un mismo estilo se pueda aplicar a todos los elementos de una página.

2. Selector de tipo o etiqueta

Selecciona todos los elementos de la página cuya etiqueta HTML coincide con el valor del selector. Para utilizar este selector, solamente es necesario indicar el nombre de una etiqueta HTML (sin los caracteres < y >) correspondiente a los elementos que se quieren seleccionar. Si se quiere aplicar los mismos estilos a dos etiquetas diferentes, se pueden encadenar los selectores.

3. Selector descendente

Utilizando este tipo de selectores es posible aplicar diferentes estilos a los elementos del mismo tipo. Un elemento es descendiente de otro cuando se encuentra entre las etiquetas de apertura y de cierre del otro elemento. Estos selectores siempre están formados por dos o más partes separadas entre sí por espacios en blanco. La última parte indica el elemento sobre el que se aplican los estilos y todas las partes anteriores indican el lugar en el que se tiene que encontrar ese elemento para que los estilos se apliquen.

4. Selector de clase

La principal característica de este selector es que en una misma página HTML varios elementos diferentes pueden utilizar el mismo valor en el atributo class. Son imprescindibles para diseñar páginas web complejas, ya que permiten disponer de una precisión total al seleccionar los elementos.

5. Selectores de ID

Permite seleccionar un elemento de la página a través del valor de su atributo id, porque el valor del atributo id no se puede repetir en dos elementos diferentes de una misma página. La sintaxis que se utiliza es el símbolo de la almohadilla (#) en vez del punto (.) como prefijo del nombre de la regla CSS.

TABLA DE SELECTORES CSS

Selector	Example	Example description
.class	.intro	Selects all elements with class="intro"
#id	#firstname	Selects the element with id="firstname"
*	*	Selects all elements
element	p	Selects all <p> elements
element,element	div, p	Selects all <div> elements and all <p> elements
element element	div p	Selects all <p> elements inside <div> elements
element>element	div > p	Selects all <p> elements where the parent is a <div> element
element+element	div + p	Selects all <p> elements that are placed immediately after <div> elements

Selector	Example	Example description
element1~element2	p ~ ul	Selects every element that are preceded by a <p> element
[attribute]	[target]	Selects all elements with a target attribute
[attribute=value]	[target=_blank]	Selects all elements with target="_blank"
[attribute~=value]	[title~=flower]	Selects all elements with a title attribute containing the word "flower"
[attribute =value]	[lang =en]	Selects all elements with a lang attribute value starting with "en"
[attribute^=value]	a[href^="https"]	Selects every <a> element whose href attribute value begins with "https"
[attribute\$=value]	a[href\$=".pdf"]	Selects every <a> element whose href attribute value ends with ".pdf"
[attribute*=value]	a[href*="w3schools"]	Selects every <a> element whose href attribute value contains the substring "w3schools"
:active	a:active	Selects the active link
::after	p::after	Insert something after the content of each <p> element

Fuente: https://www.w3schools.com/cssref/css_selectors.asp

Selector	Example	Example description
::before	p::before	Insert something before the content of each <p> element
:checked	input:checked	Selects every checked <input> element
:disabled	input:disabled	Selects every disabled <input> element
:empty	p:empty	Selects every <p> element that has no children (including text nodes)
:enabled	input:enabled	Selects every enabled <input> element
:first-child	p:first-child	Selects every <p> element that is the first child of its parent
::first-letter	p::first-letter	Selects the first letter of every <p> element
::first-line	p::first-line	Selects the first line of every <p> element

Fuente: https://www.w3schools.com/cssref/css_selectors.asp

Selector	Example	Example description
:first-of-type	p:first-of-type	Selects every <p> element that is the first <p> element of its parent
:focus	input:focus	Selects the input element which has focus
:hover	a:hover	Selects links on mouse over
:in-range	input:in-range	Selects input elements with a value within a specified range
:invalid	input:invalid	Selects all input elements with an invalid value
:lang(language)	p:lang(it)	Selects every <p> element with a lang attribute equal to "it" (Italian)
:last-child	p:last-child	Selects every <p> element that is the last child of its parent
:last-of-type	p:last-of-type	Selects every <p> element that is the last <p> element of its parent
:link	a:link	Selects all unvisited links
:not(selector)	:not(p)	Selects every element that is not a <p> element

Selector	Example	Example description
:nth-child(n)	p:nth-child(2)	Selects every <p> element that is the second child of its parent
:nth-last-child(n)	p:nth-last-child(2)	Selects every <p> element that is the second child of its parent, counting from the last child
:nth-last-of-type(n)	p:nth-last-of-type(2)	Selects every <p> element that is the second <p> element of its parent, counting from the last child
:nth-of-type(n)	p:nth-of-type(2)	Selects every <p> element that is the second <p> element of its parent
:only-of-type	p:only-of-type	Selects every <p> element that is the only <p> element of its parent
:only-child	p:only-child	Selects every <p> element that is the only child of its parent
:optional	input:optional	Selects input elements with no "required" attribute
:out-of-range	input:out-of-range	Selects input elements with a value outside a specified range

Fuente: https://www.w3schools.com/cssref/css_selectors.asp

Selector	Example	Example description
:read-only	input:read-only	Selects input elements with the "readonly" attribute specified
:read-write	input:read-write	Selects input elements with the "readonly" attribute NOT specified
:required	input:required	Selects input elements with the "required" attribute specified
:root	:root	Selects the document's root element
::selection	::selection	Selects the portion of an element that is selected by a user
:target	#news:target	Selects the current active #news element (clicked on a URL containing that anchor name)
:valid	input:valid	Selects all input elements with a valid value
:visited	a:visited	Selects all visited links

CSS permite la combinación de tipos de selectores para restringir el alcance de las reglas CSS. Utilizando solamente los selectores básicos es posible diseñar prácticamente cualquier página web. No obstante, CSS consta de selectores más avanzados que permiten simplificar las hojas de estilos.

The logo consists of the word "nowe" in a black sans-serif font. The letter "o" features a blue wavy line through it.

Espacio libre para tus ideas:
