

[Introducción a CSS](#)

[Sintaxis de las reglas en CSS](#)

[Añadir estilos a un documento](#)

[Estilos en línea](#)

[Hojas de estilo incrustadas](#)

[Hojas de estilo externas](#)

[Utilizar link](#)

[Importar el documento de estilos \(@import\)](#)

[Contenidos de las hojas de estilos CSS.](#)

[Tipos de elementos CSS. Elementos de bloque y en línea.](#)

[Especificar valores en CSS.](#)

[Unidades de medida](#)

[Especificar el color](#)

[Selectores](#)

[Selectores de elemento](#)

[Herencia](#)

[Selectores contextuales](#)

[Árbol del documento](#)

[Selector descendiente](#)

[Selector hijo \(>\)](#)

[Selector hermano \(+\) adyacente](#)

[Selector hermano \(~\) no adyacente](#)

[Selectores por ID \(#\)](#)

[Selectores de clase \(.\)](#)

[Pseudoselectores](#)

[Pseudoclases](#)

[Pseudoclases enlaces.](#)

[Otras Pseudoclases CSS.](#)

[Pseudoelementos](#)

[Validador de css](#)

Introducción a CSS

Las hojas de estilo en cascada (CSS - *Cascading Style Sheets*) son un estándar W3C para definir la presentación de los documentos Web. La *presentación* es el modo en el que un documento se muestra en pantalla. Lo más importante es darse cuenta que CSS mantiene las instrucciones de presentación separadas del contenido del documento HTML.

Sintaxis de las reglas en CSS

Las hojas de estilo consisten en una o más reglas que describen cómo debería mostrarse en pantalla un elemento.

El siguiente ejemplo contiene dos reglas:

```
h1 {color: #EEEEEE;}  
p {font-size: 12px;  
font-family: Verdana, sans-serif; }
```

La primera regla establece h1 en gris. El segundo especifica que los párrafos deben ir en Verdana 12 píxeles de alto o sans-serif:

Con estos ejemplos podemos ver que los componentes de una regla de hoja de estilo son:

```
selector { propiedad : valor; }
```

Las dos secciones principales de una hoja de estilo son el **selector** (que identifica el elemento al que se le va a aplicar el estilo) y la **declaración** (que es el conjunto de propiedad:valor). La declaración consta de una propiedad y su valor correspondiente. Las llaves permiten definir declaraciones múltiples, separadas por ';' que componen un bloque de declaración. Un ejemplo de bloque de declaración es el siguiente:

```
p {font-size: 11px; font-weight: bold; color: #CC0066; }
```

Si tenemos en cuenta que CSS ignora los espacios en blanco y saltos de línea la misma regla se puede escribir así:

```
p {  
    font-size: 11px;  
    font-weight: bold;  
    color: #CC0066;  
}
```

Añadir estilos a un documento

Se pueden añadir estilos a un documento de tres maneras:

- incluir estilos en línea,
- como elementos de estilo incrustados en la parte superior del documento
- y como archivos externos que podrán vincularse o importarse al documento.

Estilos en línea

Se puede añadir información de estilo a un elemento concreto mediante el atributo **style** en la etiqueta HTML. Consideremos algunos ejemplos:

```
<h1 style="color: red">Esta cabecera es de color rojo</h1>
```

```
<p style="font-size: 12px; font-family: 'Trebuchet MS', sans-serif">  
Este texto tiene un tamaño de 12px y de tipo de letra Trebuchet MS  
</p>
```

No es conveniente utilizarlos.

Hojas de estilo incrustadas

Incrustar un bloque de estilo en la parte superior del documento HTML usando la etiqueta **style**.

```
<html>  
  <head>  
    <style type="text/css">  
      h1 {color: #666666; }  
      p {font-size: 90%; font-family: Verdana, sans-serif; }  
    </style>  
    <title>Mi primera pagina web con estilo</title>  
  </head>  
  ...  
</html>
```

El elemento style debe colocarse en la cabecera del documento. Es obligatorio utilizar el atributo type en los documentos HTML y XHTML, además, de momento el único estilo viable es text/css.

Hojas de estilo externas

La manera más potente de usar CSS es reunir todas las reglas de estilo en un documento de texto independiente y crear vínculos a ese documento desde todas las páginas de un sitio.

Para hacer referencia a una hoja de estilos externa tenemos dos métodos:

- utilizando link
- con la directiva @import.

Utilizar link

```
<head>  
  <link rel="stylesheet" href="estilos/estilo1.css" type="text/css" />  
</head>
```

El atributo **href** indica la URL del documento de la hoja de estilo.

Se puede vincular más de una hoja de estilo en un documento, y es muy importante el orden en que están añadidas pues se aplican de arriba a abajo, por lo que la última hoja de estilos aplicada modifica los estilos que coinciden con los anteriores.

Importar el documento de estilos (@import)

```
<head>
<style type="text/css">
<!--
  @import url(estilos/estilo1.css);
  body { font-size: 13px }
  p {font-family: Verdana;}
-->
</style>
</head>
```

El método @import se puede utilizar simultáneamente con otros métodos.
Podemos importar un estilo y definir otros en la propia página.
El método link no se puede mezclar.

Contenidos de las hojas de estilos CSS.

Las hojas de estilo son documentos de texto con al menos una regla de hoja de estilo.

Se pueden incluir comentarios de la siguiente forma:

```
/* Este é un comentario */
```

Tipos de elementos CSS. Elementos de bloque y en línea.

En general, se puede decir que los elementos de bloque pueden contener elementos en línea y en bloque, mientras que los elementos en línea sólo pueden contener datos y otros elementos en línea.

Para HTML los párrafos (p), cabeceras (como h1), listas (ol, ul, dl) y divs son los elementos de bloque más comunes. El texto enfatizado (em) y los hiperenlaces (a) son ejemplos de elementos en línea comunes.

Un elemento de bloque de CSS (**display: block**) siempre genera saltos antes y después de él. Llena el largo disponible del elemento padre que lo contiene sea al largo del cuerpo del documento o un espacio menor definido, como un div.

Los elementos en línea CSS (**display: inline**) no generan saltos de línea. Aparecen en el flujo de línea y sólo pasarán a otra línea si no tienen espacio.

Especificar valores en CSS.

Unidades de medida

px Píxel 12px

pt Punto

pc Pica

em Em Unidad de medida relativa que equivale al largo de la letra "M" mayúscula

ex Ex Unidad de medida relativa que es la altura de la letra minúscula "x"

in Pulgadas

mm Milímetros

cm Centímetros Medida métrica

Especificar el color

Para especificar el color se hace como en HTML, hay varias maneras:

- Por nombre (h1 { color: olive;}).Colores: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, orange, purple, red, silver, teal, white, yellow.
- Por el valor RGB, que se puede especificar por cualquiera de estos métodos:

```
{color: #0000FF;}  
{color: #00F;}  
{color: rgb(0,0,255);}  
{color: rgb(0%, 0%, 100%);}
```

Selectores

El selector es la parte de la regla de estilo que identifica el elemento al que se le aplican las instrucciones de presentación.

```
selector { propiedad : valor; }
```

Selectores de elemento

Se aplica a elementos concretos, es el selector más sencillo. Son etiquetas de formato HTML (P, H1, BODY,...)

Ejemplos:

```
h1 {color: blue;}  
h2 {color: blue;}  
p {color: blue;}
```

El ejemplo siguiente tiene el mismo efecto que el anterior, los selectores se pueden agrupar separándolos por comas:

```
h1, h2, p {color: blue;}
```

También tenemos un selector de elementos "universal" (*),ejemplo:

```
* {color: grey;}
```

Esta regla de estilo pone en gris todos los elementos del documento que no tengan especificado otro color.

Ejemplo:

```
h1{color:orange;}  
* {color: grey;}
```

Sale todo en gris, excepto los h1 que salen en naranja.

Herencia

A través de la herencia, cada antecesor transmite sus propiedades a todos sus descendientes.

<pre>BODY { color: #FF0000 }</pre>	Entonces todos los descendientes de BODY tendrán implícita la propiedad color:#FF0000
<pre><!DOCTYPE html> <HTML> <HEAD> <TITLE>Tutorial</TITLE> <STYLE TYPE="text/css"> body {text-align: center} P { color: #FF0000 } </STYLE> </HEAD> <BODY> <P>Este es el tutorial de <u>CSS</u> </P> </BODY > </HTML></pre>	El texto se mostrará en color rojo y "CSS" aparecerá del mismo color pero subrayado. <u> es hijo de <p> y hereda su color. <u> es nieto de <body> y hereda su alineación. <p> es hijo de <body> y hereda su alineación.

Selectores contextuales

Los selectores de elemento se aplican a todos los casos en los que se encuentra el elemento en el documento. Por el contrario, los selectores contextuales permiten aplicar estilos a los elementos en función de su contexto, o que tengan cierta relación con otro elemento.

Hay varios tipos de selectores contextuales: descendiente, hijo y hermano.

Árbol del documento

Un documento HTML puede describirse como un árbol.

Árbol: Grafo no orientado conexo sin ciclo. Esta compuestos por nodos o módulos conectados por líneas llamadas ramas.

En todo árbol hay un nodo denominado raíz u origen y que carece de un antecesor. Cualquier otro nodo tiene un antecesor único en el nivel anterior y uno o más sucesores, incluso ninguno en el nivel más bajo. Ejemplo de árbol: un sistema de archivos.

Tomamos como referencia el siguiente documento html:

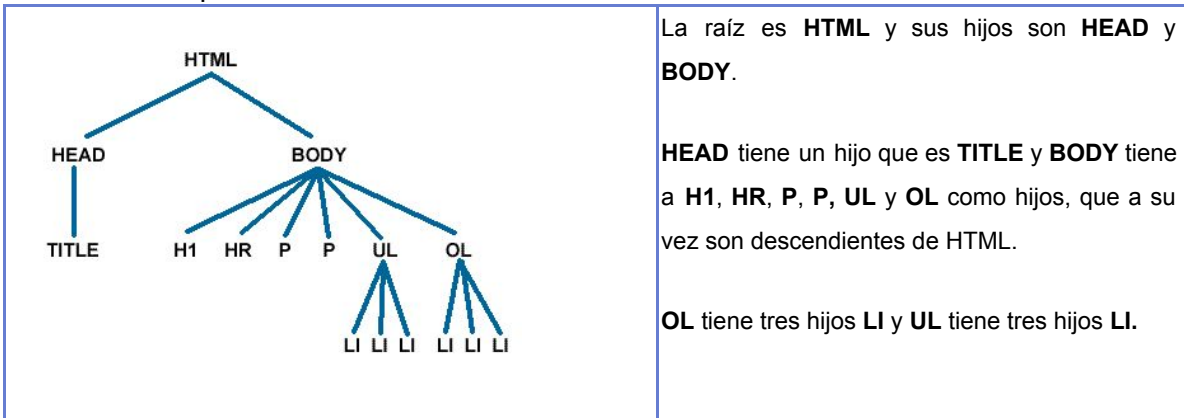
```
<HTML>
  <HEAD>
    <TITLE>Mi página</TITLE>
  </HEAD>
  <BODY>
    <H1>Bienvenidos todos a mi p&aacute;gina</H1>
    <HR>
    <P>Un p&aacute;rrafo cualquiera .
    <P>Otro p&aacute;rrafo cualquiera.
    <UL>
      <LI>Elemento 1 de la lista 1</LI>
```

```

        <LI>Elemento 2 de la lista 1</LI>
        <LI>Elemento 3 de la lista 1</LI>
    </UL>
    <OL>
        <LI>Elemento 1 de la lista 2</LI>
        <LI>Elemento 2 de la lista 2</LI>
        <LI>Elemento 3 de la lista 2</LI>
    </OL>
</BODY>
</HTML>

```

El árbol correspondiente sería:



Selector descendiente

Seleccionan los elementos contenidos en otro elemento. Se indican en una lista de elementos separados por un espacio, empezando por el elemento de más alto nivel.

Por ejemplo:

li em {color: blue;}	Especifica que los elementos em deben tener color azul, pero sólo si son descendientes de un elemento de lista (li). Todos los demás elementos em no se verán afectados.
-----------------------------	--

También se pueden agrupar: **h1 em, h2 em, h3 em {color: red;}**

Los selectores descendientes también se pueden "anidar" en varias capas de profundidad.:

ol a em {color: yellow;}	Pone en amarillo "sólo" texto enfatizado (em) de los hiperenlaces (a) que están en las listas ordenadas (ol).
<pre> Elemento 1 de la lista 2 Elemento 2 de la lista 2 Elemento 3 de la lista 2 </pre>	

Otro ejemplo:

p a { color: red; }	Muestra los dos enlaces de color rojo.
<pre> <p>Enlace</p> <p>Enlace</p> </pre>	

Pero si se utiliza el selector descendiente combinado con el selector universal (*), se puede restringir el alcance de un selector descendiente.

```
p * a { color: red; }
```

```
<p><a href="#">Enlace1</a></p>  
<p><span><a href="#">Enlace2</a></span></p>  
<p><b><a href="#">Enlace3</a></b></p>
```

Sólo el segundo y tercer enlace se muestra en rojo.

La razón es que el selector **p * a** se traduce como: todos los elementos de tipo **<a>** que se encuentran dentro de cualquier elemento, que a su vez se encuentra dentro de un **<p>**.

Como el primer elemento es **<a>** directamente dentro de **<p>** no se cumple la condición del selector **p * a**.

Selector hijo (>)

Un selector hijo, es similar a un selector descendiente pero sólo selecciona **hijos directos (descendientes directos)** de un elemento dado. Es decir, no deben existir "niveles intermedios".

- Los elementos hijos están separados por el símbolo (>).

Ejemplo:

```
p > em {background-color: gray;}
```

```
<p>O sistema operativo <em>Windows Vista</em>  
está sendo substituído, em moitos equipos, polo  
<strong><em>Windows XP</em></strong></p>
```

Sólo el primer *em* tendrá color de fondo gris, porque el segundo es hijo de un elemento *strong*.

Ejercicio: hijos de UL color rojo e hijos de OL color verde.

Selector hermano (+) adyacente

Este selector se utiliza para seleccionar un elemento que **sigue inmediatamente a otro con el mismo elemento padre**.

- Se utiliza el signo (+) para seleccionarlos.

```
h1 + p {color: blue;}
```

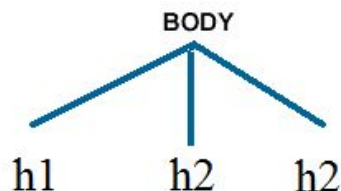
Primer párrafo que sigue a un encabezado de primer nivel.
El color se aplica al p que es hermano de h1.

Otro ejemplo:

```
h1 + h2 { color: red; }
```

```
<body>  
<h1>Titulo1</h1>  
<h2>Subtítulo</h2>  
...  
<h2>Outro subtítulo</h2>  
...
```

El color se aplica al h2 que es hermano adyacente de h1.
El estilo del selector **h1 + h2** se aplica al primer elemento **<h2>**, pero no al segundo **<h2>**



porque:

- El elemento padre de <h1> es <body>, el mismo padre de los dos <h2>. Por lo tanto, los dos elementos <h2> cumplen la condición de tener el mismo padre.
- El primer elemento <h2> aparece en el código HTML inmediatamente después de <h1>, por lo que este elemento <h2> también cumple la condición del selector adyacente.
- Por el contrario, el segundo elemento <h2> no aparece inmediatamente después de <h1> por lo que no cumple la condición del selector adyacente y por lo tanto no se aplica el estilo h1 + h2.

```
<h1>Titulo1</h1>
<h2>Subtítulo1</h2>
...
<h2>Otro subtítulo 2</h2>
...
<h1>Titulo3</h1>
<h2>Subtítulo 3</h2>
```

h1 + h2 { color: red; }

El primer h2 es hermano adyacente de h1.

El segundo h2 no es hermano adyacente.

El tercer h2 es hermano adyacente de h1.

Nota: el color se aplica al hermano de la derecha. (a h2 que es hermano de h1)

Resultado:

Titulo1

Subtítulo1

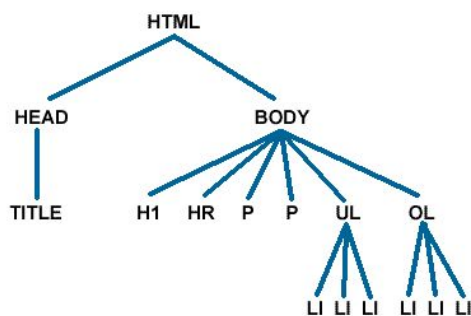
...

Otro subtítulo 2

...

Titulo3

Subtítulo 3



Ejercicio:

¿Qué elementos son hermanos adyacentes?

Aplicar un color diferente a cada hermano.

Selector hermano (~) no adyacente

Este selector es similar al anterior pero el elemento afectado no necesita estar precedido de inmediato al elemento de la izquierda.

```
h1 ~ h2 { color: red; }
<body>
<h1>Titulo1</h1>
<h2>Subtítulo</h2>
<h2>Otro subtítulo</h2>
```

Poner en rojo todos los <h2> que tengan un <h1> anterior. Los <h1> y <h2> han de ser hermanos. El color se aplica a los dos h2, porque ambos tienen un hermano <h1> por la izquierda. El <h2> tiene que tener un hermano mayor <h1>

```

h1 + h2 { color: red; }
h1 ~ h2 { background-color: gray; }
<body>
<h1>Titulo1</h1>
<h2>Subtítulo</h2>
<h2>Outro subtítulo</h2>

```

Titulo1

Subtítulo

Outro subtítulo

Selectores por ID (#)

Los elementos HTML disponen de un atributo llamado identificador (id).

Un **id** debe ser único en cada documento. Si tenemos varios elementos que precisen un tratamiento similar utilizaremos **class**.

Ejemplo: <p id="noticia">

Hoy en día, los atributos id se utilizan para identificar secciones principales de las páginas. Algunos valores frecuentes son: contenido, cabecera, pie,...

Ejemplo: queremos que el párrafo identificado con *noticia* tenga una determinada apariencia. Regla de estilo:

```
p#noticia {font-size: 14px;}
```

Podremos omitir el selector <p> porque el id es único en el documento.

```
#noticia {font-size: 14px;}
```

Selectores de clase (.)

Se usa el atributo *class* para identificar distintos elementos como parte de un grupo conceptual. Así, los elementos de una clase se pueden modificar con una sola regla de estilo.

Ejemplo:

```

<h1 class="especial">¡Atención!</h1>
<p class="especial">Hoxe temos grandes rebaixas.</p>

```

Dos elementos de la clase especial.

También se puede hacer que un elemento pertenezca a más de una clase.

```
<p class="novo especial">Hoxe temos grandes rebaixas.</p>
```

El elemento <p> pertenece a dos clases.

Para definir los estilos de una determinada clase se usa el punto (.)

Ejemplos:

```

h1.especial {color: red;}
p.especial {color: blue;}

```

Los h1 de la clase *especial* se ponen en rojo.

Los p de la clase *especial* se ponen en azul.

Para ampliar una propiedad a todos los elementos de la misma clase, se debe omitir el nombre de la etiqueta en el selector. Por ejemplo:

```
.especial {color: green;}
```

Todos los elementos de la clase *especial* en

	verde.
--	--------

Si queremos que todos los elementos de una clase tengan una propiedad menos alguno podemos hacer así:

```
.especial {color: green;}
h1.especial{color: red;}
```

NOTA: no se debe exceder en su uso, es fácil perderse, repetirlos,.....

Pseudoselectores

Si queremos aplicar reglas de estilo a elementos como los vínculos visitados, la primera línea de un párrafo, la primera letra de un elemento, usamos los pseudoselectores.

Pseudoclases

Las pseudoclases funcionan como si existiera una clase aplicada a un grupo de elementos, en la mayoría de los casos será el elemento <a>.

Pseudoclases enlaces.

```
a:link { color: red; } /* enlace no visitado */
a:visited { color: blue; } /* enlace que ya fue visitado */
a:hover {color: yellow; } /* cambia la apariencia cuando se pasa el ratón por encima
del enlace, sin pulsar. */
a:active { color: lime; } /* cambia la apariencia justo cuando se pulsa en el enlace */
```

NOTA: [ejercicio de links resuelto](#)

Estas pseudoclases deben aparecer en un determinado orden:

:link, :visited, :hover, :active. LVHA

Algo típico es quitar los subrayados de los hiperenlaces.

```
a:link {color: red; text-decoration: none;}
a:visited {color: blue; text-decoration: none;}
```

O esta otra, que hace que el subrayado aparezca cuando ponemos el puntero encima del hiperenlace:

```
a:link {color: red; text-decoration: none;}
a:visited {color: blue; text-decoration: none;}
a:hover {color: red; text-decoration: underline;}
```

Otras Pseudoclases CSS.

Existen otras pseudoclases menos utilizadas que las de los enlaces.

:focus - Selecciona elementos que tienen el foco, como un elemento de un formulario.

Un ejemplo:

Input:focus {background-color: yellow;}

:first-child - Selecciona el primero hijo de un elemento padre.

Ejemplo: seleccionar el primer *li* de una *ul*:

ul li:first-child {color: red;}

:last-child - Selecciona el último hijo de un elemento padre.

Ejemplo: seleccionar el último *li* de una *ul*:

ul li:last-child {background-color: gainsboro;}

:only-child - Afecta a un elemento siempre y cuando sea el único hijo disponible.

Ejemplo: seleccionar los *li* de una *ul* que sean hijos únicos:

ul li:only-child {text-decoration: underline;}

:nth-child(nº) - Afecta a un hijo en la posición *nº*, (el número entre paréntesis será el número de la posición del hijo). En vez de *nº*, podemos usar las palabras: **odd** (índice impar) o **even** (índice par).

Ejemplo: seleccionar los *li* de una *ul* que ocupen posiciones pares:

ul li:nth-child(even) {text-decoration: underline;}

Ejemplo:

<pre>ul li:first-child {color: red;} ul li:last-child {background-color: gainsboro;} ul li:only-child {text-decoration: underline;} ol li:nth-child(odd) {color: green;} ol li:nth-child(even) {color: blue;}</pre>	<ul style="list-style-type: none">• Elemento 1 de la lista• Elemento 2 de la lista• Elemento 3 de la lista• Elemento 4 de la lista• Elemento 5 de la lista <ul style="list-style-type: none">• <u>Elemento 1 de la lista</u> <ol style="list-style-type: none">1. Elemento 1 de la lista2. Elemento 2 de la lista3. Elemento 3 de la lista4. Elemento 4 de la lista5. Elemento 5 de la lista
<pre> Elemento 1 de la lista Elemento 2 de la lista Elemento 3 de la lista Elemento 4 de la lista Elemento 5 de la lista Elemento 1 de la lista Elemento 1 de la lista Elemento 2 de la lista Elemento 3 de la lista Elemento 4 de la lista Elemento 5 de la lista </pre>	

Pseudoelementos

:first-line - Aplica una regla de estilo a la primera línea del elemento especificado.

Ejemplo: Añadir espacio extra a la primera línea del texto de cada párrafo:

p.first-line {letter-spacing: 6pt;}

Otro ejemplo: **p:first-line {color: red;}**

:first-letter - Aplica un estilo a la primera letra de un elemento.

Ejemplo: modificar el estilo de la primera letra de los párrafos clasificados como "definicion":

p.definicion:first-letter {font-size: 300%; color: red; }

:before y **:after** – Permite añadir contenido y darle formato, antes y después de un elemento.

content:

Genera contenido de forma dinámica. Puede tomar, entre otros, los valores: cadena de texto entre comillas o una URL.

Ejemplo: quiero añadir un logo antes de la cabecera h2 de la clase logo:

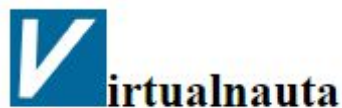
h2.logo:before {content:

url(http://www.virtualnauta.com/templates/virtualnauta_home/favicon.ico)}

Utilizamos una clase llamada "logo" para señalar en qué título vamos a anteponer el logo de Virtualnauta:

`<h2 class="logo">irtualnauta</h2>`

Nos quedaría:



Ejemplo: antes de un párrafo añadir una imagen y el texto HOLA

p:before{content: url(favicon.ico) "HOLA"}

Ejemplo: añadir un pie de página después del último párrafo identificado como último.

#ultimo:after { content: " Fin de los contenidos de la página"}

Siendo el código html: `<p id="ultimo"> TEXTO </p>`

Ejemplo: lo mismo pero añadiendo el texto en color rojo y fondo gris:

#ultimo:after {

content: " Fin de los contenidos de la página" ; color:red ; background:gray}

Validador de css

<https://jigsaw.w3.org/css-validator/>



CSS Validation Service

Verifica Hojas de Estilo en Cascada (CSS) y documentos (X)HTML con hojas de estilo

mediante URI

mediante Carga de Archivo

mediante Entrada directa

Validar mediante URI

Introduce la URI de un documento (HTML con CSS o sólo CSS) que desees validar:

Dirección:

► [Más opciones](#)

Check