



APRENDERAPROGRAMAR.COM

SELECTORES CSS: ID.  
APLICAR CSS A PARTES DE  
UNA PÁGINA WEB.  
EJEMPLOS. DIFERENCIAR  
POR ETIQUETAS HTML EN  
IDS. (CU01011D)

Sección: Cursos

Categoría: Tutorial básico del programador web: CSS desde cero

Fecha revisión: 2029

**Resumen:** Entrega nº11 del Tutorial básico: "CSS desde cero".

Autor: César Krall

## SELECTORES CSS

Hemos estudiado cómo aplicar estilos CSS a todas las etiquetas HTML de un tipo, por ejemplo a todos los párrafos, todas las listas, todos los links, etc. Pero en numerosas ocasiones no queremos que todas las etiquetas de un tipo tengan un mismo estilo, sino aplicar distintos estilos a una etiqueta o a una parte de una página web. Para ello además del selector de etiqueta o palabra clave que ya conocemos existen otros selectores, entre los más usados tenemos los selectores por id y los selectores de clase.



Recordemos primero los selectores de etiqueta o palabra clave tipo etiqueta {propiedad:valor;}:

ul {color:red;} suponía aplicar color rojo al texto de todas las listas de tipo ul del documento HTML, es decir, a todas las etiquetas <ul> ... </ul>.

a {color:green; text-decoration: none;} suponía aplicar color verde y sin subrayado a todos los links del documento HTML, es decir, a todas las etiquetas <a> ... </a>.

De la misma forma se pueden usar como selector otras etiquetas HTML como div { }, span { }, p { }, etc. pero suponiendo que se apliquen a todos los elementos de ese tipo.

## SELECTOR POR ID

Trataremos ahora de aplicar estilos a sólo una parte de una página web o documento HTML. Esto implica dos cosas:

- a) En el documento HTML, habrá que identificar la parte del documento a la que queremos aplicar esos estilos incluyendo un atributo para la etiqueta que delimita el fragmento de código donde queremos aplicar estilos. Por ejemplo si queremos aplicar estilos dentro de un contenedor div la sintaxis sería <div id="nombreIdentificativoElegido"> ... </div>. Si quisiéramos aplicar estilos sólo a un párrafo la sintaxis sería <p id="nombreIdentificativoElegido"> ... </p>
- b) En la declaración de estilos tendremos que declarar los estilos a aplicar a ese fragmento de código realizando la declaración usando esta sintaxis.

```
#nombreIdentificativoElegido {
propiedad1ParaEseTipoDeElementos:valor;
propiedad2ParaEseTipoDeElementos:valor;
propiedad3ParaEseTipoDeElementos:valor;
...
propiedadnParaEseTipoDeElementos:valor;
}
```

Las propiedades se pueden definir en una sola línea si se prefiere así, el único requisito es separarlas con punto y coma.

Con un ejemplo lo veremos más claro. Partimos del código base HTML que estamos usando para el curso, en el cual tenemos un menú. Vamos a elegir un nombre para identificar ese menú. Podríamos elegir como nombre <<menu>> (usaremos nombres sin tildes). Ahora mismo solo tenemos un menú, pero en el futuro es posible que existan otros, por eso vamos a preferir elegir como nombre <<menu1>>. En el documento HTML vamos a identificar el menú con el atributo id aplicado al contenedor div dentro del cual está el menú. El código es este:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<!--Código ejemplo para el curso -->
<html>

<head>
<title>Portal web - aprenderaprogramar.com</title>
<meta charset="utf-8">
<meta name="description" content="Portal web aprenderaprogramar.com">
<meta name="keywords" content="aprender, programar, cursos, libros">
<link rel="stylesheet" type="text/css" href="rutaDelArchivo.css">
</head>

<!-- Contenido de la página web -->
<body>

<!-- Cabecera de la página web -->
<div>
<h1>Portal web aprenderaprogramar.com</h1>
<h2>Didáctica y divulgación de la programación</h2>
</div>
<!-- Fin de la cabecera de la página web -->
<br />
<!-- Contenedor para la parte central -->
<div>
<!-- menu -->
<div id="menu1">
<div>Menú</div>
<hr/>
<ul>
<li><a href="#">Inicio</a></li>
<li><a href="libros.html">Libros de programación</a> </li>
<li><a href="cursos.html">Cursos de programación</a> </li>
<li><a href="humor.html">Humor informático</a> </li>
</ul>
</div>
<!-- fin menu -->
<!--Aquí el resto del código del ejemplo -->

</html>
    
```

Hemos incluido <div id="menu1"> en el contenedor div que delimita nuestro menú.

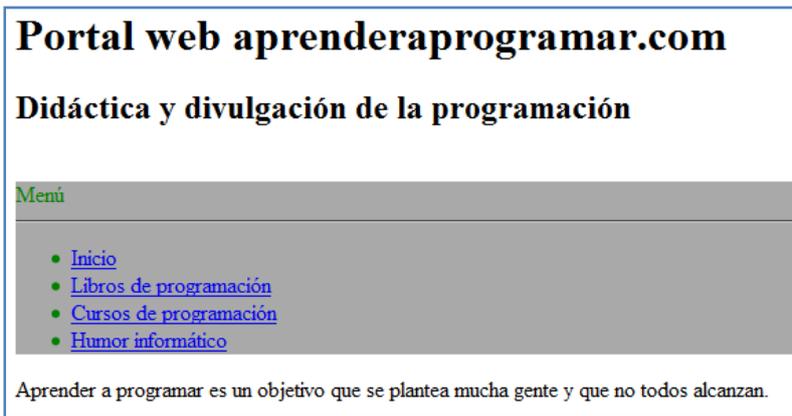
En el archivo css externo, en nuestro caso denominado estilos.css, vamos a definir que ese fragmento de página web (el menú) tenga el texto verde y fondo gris. Para ello escribiremos lo siguiente:

```
/* Curso CSS estilos aprenderaprogramar.com*/

#menu1 {
color:green;
background-color: DarkGray;
}
```

Si lo preferimos podemos escribir la declaración en una sola línea, obteniendo el mismo resultado. Es decir, podemos escribir #menu1 {color:green; background-color: DarkGray; }. Esto resulta más compacto y ahorra espacio, pero también puede ser menos claro a la hora de leer. Algunos programadores o diseñadores prefieren el estilo compacto y otros diferenciando líneas.

El resultado obtenido será similar a este:



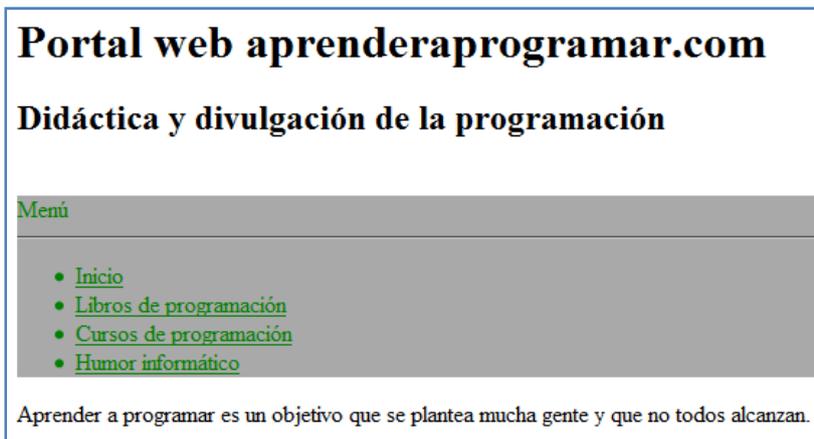
Analicemos ahora el resultado obtenido: se ha aplicado el fondo gris a la caja que define el contenedor div. Se ha aplicado color verde al texto y a las viñetas de la lista existente en el menú. Sin embargo, los links del menú mantienen su color azul y subrayado, a pesar de que hemos indicado que el color verde debería de aplicarse a todo el contenedor. ¿Por qué ocurre esto? CSS no aplica la misma importancia a todas las reglas o estilos que se especifican. En este caso la regla o estilo que hemos definido no supera en la escala de importancia CSS a la regla por defecto según la cual los links son de color azul y subrayado.

Para resolver este conflicto hemos de especificar que queremos que los elementos <a> ... </a> (links) dentro del bloque de código con identificador menu1 se muestren aplicando la regla de estilo definida para el menú. Modifica el archivo css escribiendo el siguiente código y visualiza la página web:

```
/* Curso CSS estilos aprenderaprogramar.com*/

#menu1, #menu1 a {color:green; background-color: DarkGray; }
```

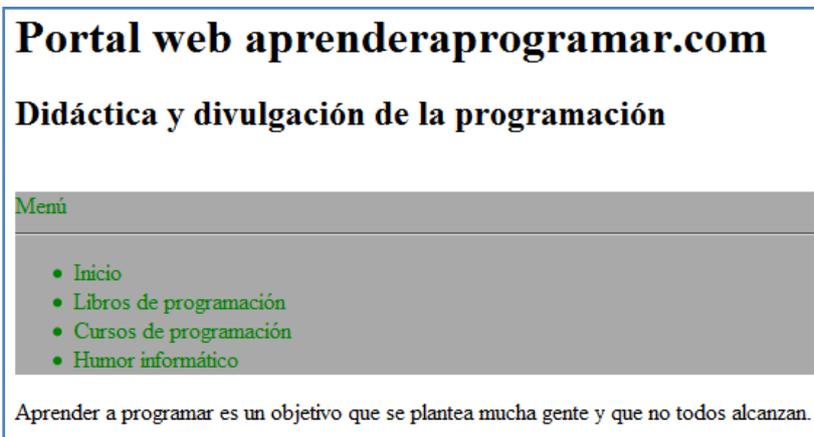
Mediante la sintaxis `#nombreIdentificativoElegido elementoAfectadoPorElEstilo {...}` estamos especificando “explícitamente” que deseamos aplicar esa regla de estilo al elemento indicado (en nuestro caso los links, `<a ... </a>`) dentro de un bloque de código definido por un identificador. De esta forma el navegador le concede preferencia o mayor importancia al estilo definido que al estilo por defecto de los links. El resultado en este caso es que se muestra el texto de los links en color verde (pero se sigue manteniendo el subrayado porque no hemos sobrescrito esta propiedad de los links):



Vamos a especificar ahora que los links del menú no aparezcan subrayados. Para ello modificaremos nuestro archivo css dejándolo con esta línea:

```
#menu1, #menu1 a { color:green; background-color:DarkGray; text-decoration:none;}
```

Ahora sí tenemos el menú en color verde y sin subrayado, ya que hemos sobrescrito la propiedad `text-decoration` que estaba dando lugar a la aparición del subrayado:



Es interesante de lo visto hasta ahora razonar cómo CSS trabaja con unos órdenes de precedencia o jerarquía de reglas CSS. No te preocupes de conocerlos con exactitud, ya que a lo largo del curso y a medida que vayas cogiendo experiencia irás aprendiendo más al respecto. También ten en cuenta que los órdenes de precedencia pueden cambiar según los navegadores o según las versiones CSS. Por tanto más que saber con exactitud los órdenes de precedencia nos interesa el ser capaces de interpretar por qué ocurren las cosas y la forma de trabajar de CSS y su filosofía.

## APLICAR ESTILOS DIFERENCIADOS DENTRO DE UN SELECTOR ID

El código que hemos visto anteriormente se puede escribir también de esta manera:

```
/* Curso CSS estilos aprenderaprogramar.com */

#menu1 {color:green; background-color: DarkGray; text-decoration:none; }

#menu1 a {color:green; background-color: DarkGray; text-decoration:none; }
```

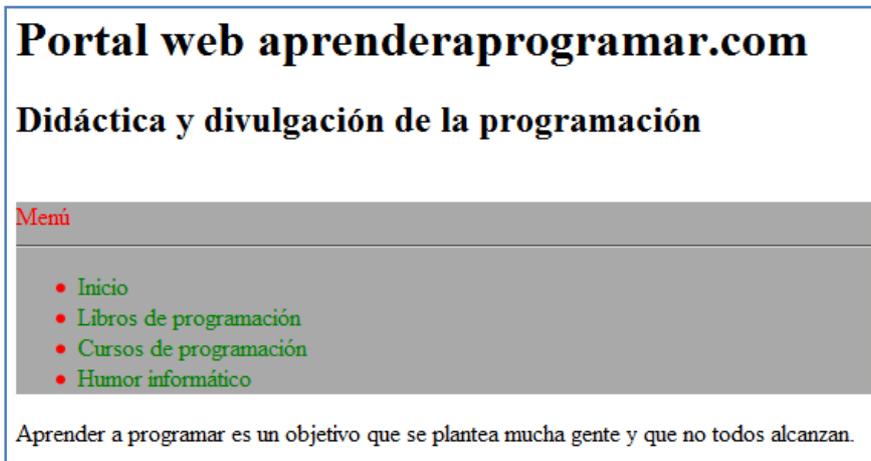
El efecto con este código sería el mismo que con el anterior. Realmente en este caso no tendría interés escribir la especificación para menu1 {...} y para menu1 a {...} por separado ya que resulta más compacto escribirlo en una línea. Sin embargo sí tiene interés en el caso de que deseemos aplicar un estilo general al menú y sobrescribir o especificar estilos diferenciados para los elementos <a> ... </a> dentro del menú.

En nuestro caso vamos a hacer lo siguiente: estableceremos como color general para el menú rojo y como fondo gris. Como fondo para los links (elementos a) dentro del menú no especificaremos ninguno, el color lo estableceremos en verde y eliminaremos el subrayado. Escribe este código y visualiza el resultado:

```
/* Curso CSS estilos aprenderaprogramar.com */

#menu1 {color:red; background-color: DarkGray; text-decoration:underline; }

#menu1 a {color:green; text-decoration:none; }
```



En este caso tenemos como color de los elementos del menú rojo, pero los links se muestran en verde porque para los elementos a el color ha sido sobrescrito y el navegador elige el estilo más interno.

Prueba a establecer distintas propiedades para #menu1 y para #menu1 a. Posiblemente te encuentres que en algunos casos los resultados no son tal y como esperas. Ten en cuenta que el navegador tiene que resolver sobre el orden de importancia de las reglas, las sobreescrituras, estilos por defecto, etc. lo cual hace que el proceso adquiera cierta complejidad. En ocasiones se producen conflictos en la definición de estilos que el navegador tratará de resolver, aunque en algunos casos quizás no pueda resolver el problema y simplemente ignore los estilos que entran en conflicto. Iremos hablando sobre estas situaciones a medida que vayamos viendo ejemplos.

## ¿CUÁNDO UTILIZAR SELECTORES POR ID?

El atributo id nos permite darle un nombre específico a una parte de un documento HTML definida por una etiqueta HTML, diferenciándola de las demás. En general este atributo se utiliza para poner nombres a distintas partes de un documento HTML, por ejemplo menu, articulo, formulario, footer, etc. de modo que cada parte de la web está identificada con un nombre único. No es recomendable aplicar el mismo nombre id a distintas partes de una web, ni siquiera aunque sean del mismo tipo. Un id debe ser único, por lo tanto ese nombre debe aparecer solo una vez en el documento HTML. Si queremos aplicar un mismo estilo en distintas partes de la web utilizaremos el atributo class, que explicaremos más adelante.

No es obligatorio identificar todas las partes de una web con ids. Puede haber partes del documento HTML con identificador y otras partes que carezcan de él. Incluso la totalidad de la web puede carecer de identificadores id.

El uso del símbolo # combinado con el atributo id permite aplicar CSS a partes muy concretas de una página web y se usa sobre todo cuando se quieren diferenciar partes estructurales del documento HTML como el menú o el footer, y en general aplicado a elementos <div> ... </div> que actúan como contenedores. Aunque se podría aplicar a etiquetas más específicas como <p> esto no es habitual ya que supondría estarle poniendo un nombre único a un párrafo y un párrafo en general no tiene tanta importancia dentro de una web como para ponerle un nombre específico. No obstante, ten en cuenta que existe la posibilidad de usar este atributo sobre cualquier etiqueta HTML.

## EJERCICIO

Analiza el siguiente código HTML. En él encontrarás tres etiquetas div. Haz lo siguiente:

- Establece atributos id para cada una de ellas con valores <<novedades>> para la primera, <<xFashion>> para la segunda y <<tMotion>> para la tercera.
- Usando CSS establece como color de texto el rojo (red) para los elementos h1 que se encuentren dentro del elemento con id <<novedades>>.

c) Usando CSS establece como color de texto el verde para los elementos h3 que se encuentren dentro de los elementos con id <<xFashion>> y <<tMotion>>.

d) Usando CSS establece como color de fondo para el elemento con id <<novedades>> el amarillo (yellow).

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Portal web - aprenderaprogramar.com</title>
<meta name="description" content="Portal web aprenderaprogramar.com">
<meta name="keywords" content="aprender, programar, cursos, libros">
<meta charset="utf-8">
</head>
<body>
<div>
<h1>Novedades</h1>
<p>Aquí presentamos las novedades del sitio.</p>
</div>
<div>
<h3>Lanzamos el producto X-FASHION</h3>
<p>Este producto permite estirar la piel hasta dejarla como la de un bebé.</p>
<p></p>
</div>
<div>
<h3>Mejoramos el producto T-MOTION</h3>
<p>Hemos lanzado una nueva versión del producto T-MOTION</p>
<p></p>
</div>
</body>
</html>
    
```

Para comprobar si tus respuestas son correctas puedes consultar en los foros aprenderaprogramar.com.

**Próxima entrega:** CU01012D

**Acceso al curso completo** en [aprenderaprogramar.com](http://aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente: [http://aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=75&Itemid=203](http://aprenderaprogramar.com/index.php?option=com_content&view=category&id=75&Itemid=203)