



APRENDERAPROGRAMAR.COM

TEXT-ALIGN, COLOR, TEXT-
DECORATION, TEXT-
INDENT, WHITE-SPACE
NOWRAP, PRE, PRE-
WRAP, PRE-LINE O
NORMAL (CU01040D)

Sección: Cursos

Categoría: Tutorial básico del programador web: CSS desde cero

Fecha revisión: 2029

Resumen: Entrega nº40 del Tutorial básico: "CSS desde cero".

Autor: César Krall

PROPIEDADES PARA CONTROL DEL TEXTO CSS

CSS dispone de una serie de propiedades que permiten controlar la forma en que se muestra el texto. Entre ellas vamos a estudiar algunas como text-align, color, text-decoration, text-indent, white-space. Algunas de ellas son muy usadas como text-align. Otras como white-space, aún siendo menos usadas tienen un gran potencial.



PROPIEDAD TEXT-ALIGN

PROPIEDAD CSS text-align	
Función de la propiedad	Permite definir cómo se alinea una unidad de texto (párrafo o similar).
Valor por defecto	left
Aplicable a	Contenido de elementos tipo block y contenido de tablas
Valores posibles para esta propiedad	left, right, center, justify (el texto se alinea según se indique)
	inherit (se heredan las características del elemento padre).
Ejemplos aprenderaprogramar.com	#content1 {text-align: center;} .elementoJukeBox {text-align: justify;}

Los elementos inline dentro del texto como imágenes o spans quedan también afectados por esta propiedad. La propiedad actúa siempre en relación a la alineación horizontal del texto. La alineación vertical habrá que realizarla usando otras propiedades.

Alineación del texto left	Alineación del texto right	Alineación del texto center	Alineación del texto justify el texto queda alineado tanto por la derecha como por la izquierda mediante el ajuste del espacio.
---------------------------	----------------------------	-----------------------------	---

Esta propiedad se hereda directamente por parte de los elementos hijos.

PROPIEDAD COLOR

Para establecer el color de un texto se usa la propiedad color. En realidad sería más razonable que se llamara text-color, pero por motivos históricos se ha mantenido la denominación color. Ya hemos hablado de la forma de expresar colores y grados de transparencia para colores usando CSS en anteriores apartados del curso. Recordar que es posible usar notación RGB decimal o porcentual, notación hexadecimal o notaciones RGBA, HSL y HSLA. Ejemplo: contentJBox {color: # FF0000;} Esta propiedad se hereda directamente por parte de los elementos hijos con la única excepción de los enlaces <a> ... , cuyo color tiene que establecerse directamente si quiere ser cambiado.

PROPIEDAD TEXT-DECORATION

PROPIEDAD CSS text-decoration	
Función de la propiedad	Permite definir algunos elementos decorativos para texto como el subrayado o el efecto de tachado.
Valor por defecto	none
Aplicable a	Textos
Valores posibles para esta propiedad	none (no hay decoración), underline (subrayado normal), overline (subrayado por encima), line-through (efecto de tachado)
	inherit (se heredan las características del elemento padre).
Ejemplos aprenderaprogramar.com	#content1 {text-decoration: none;} .elementoJukeBox {text-decoration: line-through;}

Esta propiedad sirve para evitar el uso de las antiguas etiquetas HTML de subrayado `<u> ... </u>` y tachado `<strike> ... </strike>`. ¿Por qué no deben usarse estas etiquetas HTML? Porque como ya hemos comentado con anterioridad en los desarrollos web actuales HTML debe usarse exclusivamente para definir la estructura y contenedores del documento, mientras que los estilos deben estar definidos por separado mediante CSS.

Un valor adicional para esta propiedad era `blink`, que servía para indicar que el texto debía parpadear. No obstante, este valor se ha eliminado del estándar por lo que no debe usarse.

El valor `none` puede usarse para anular el subrayado que por defecto introducen los navegadores para los links (elementos `a`) si establecemos `{text-decoration: none;}`.

Aunque `text-decoration` es una propiedad sencilla, tiene un aspecto que puede resultar bastante problemático. Supongamos que queremos establecer que todo el texto dentro de un `div` vaya subrayado, pero que en el elemento con `id="contentMbox"` el texto no lleve subrayado. Podríamos pensar en usar este código:

```
div {text-decoration: underline;}
#contentMbox {text-decoration: none;}
```

El resultado es que aún dentro del elemento para el que hemos especificado `text-decoration:none;` el texto sigue estando subrayado. ¿Por qué? Porque `text-decoration` es una propiedad un tanto especial que se dibuja acumulativamente transmitiéndose de padres a hijos sin ser anulable por los hijos. En este caso lo que ocurre es que a todos los `div` se le aplica el subrayado y luego, acumulativamente, al elemento `contentMbox` se le aplica `text-decoration: none;`. El resultado es `underline + none = underline`, el texto aparece subrayado. Supongamos ahora que escribimos esto:

```
div {text-decoration: underline;}
#contentMbox {text-decoration: overline;}
```

El texto dentro del elemento `contentMbox` aparecerá con una línea por abajo (`underline`) y una línea por arriba (`overline`) debido al carácter acumulativo de esta propiedad.

PROPIEDAD TEXT-INDENT

PROPIEDAD CSS text-indent	
Función de la propiedad	Permite definir el indentado (desplazamiento hacia dentro) de la primera línea de un párrafo de texto.
Valor por defecto	0
Aplicable a	Textos
Valores posibles para esta propiedad	Una unidad de medida absoluta o relativa (incluido %) permitida en CSS inherit (se heredan las características del elemento padre).
Ejemplos aprenderaprogramar.com	#content1 {text-indent: 20px;} .elementoJukeBox {text-indent: 0.6em;}

En el caso de utilizarse un valor en %, el porcentaje se calcula como porcentaje del valor width del elemento padre.

PROPIEDAD WHITE-SPACE

PROPIEDAD CSS white-space	
Función de la propiedad	Determina el tratamiento que le da el navegador a los espacios en blanco.
Valor por defecto	normal
Aplicable a	Todos los elementos
Valores posibles para esta propiedad	<p>normal (si aparece una secuencia de varios espacios en blanco dentro de un texto en el documento HTML, el navegador los colapsa dejando un solo espacio; el espacio en blanco se usa para generar saltos de línea si se excede el tamaño del contenedor. Los saltos de línea en el documento HTML se ignoran)</p> <p>nowrap (el comportamiento es como el normal, pero el espacio en blanco no se usa para generar saltos de línea, por lo que todo el texto queda en una sola línea)</p> <p>pre (el texto se conserva tal y como esté en el documento HTML, conservando las secuencias de espacio y saltos de línea como estén allí definidos)</p> <p>pre-wrap (el comportamiento es como con wrap pero los espacios en blanco se usan para generar nuevas líneas si no hay espacio suficiente disponible)</p> <p>pre-line (se mantienen los saltos de línea definidos en el documento HTML, se colapsan secuencias de espacios en blanco y se usan espacios en blanco para generar nuevas líneas si no hay espacio suficiente disponible)</p> <p>inherit (se heredan las características del elemento padre).</p>
Ejemplos aprenderaprogramar.com	#content1 {white-space: pre;} .elementoJukeBox {white-space: nowrap;}

Esta propiedad evita el uso de la etiqueta HTML <pre> cuyo uso es no recomendado.

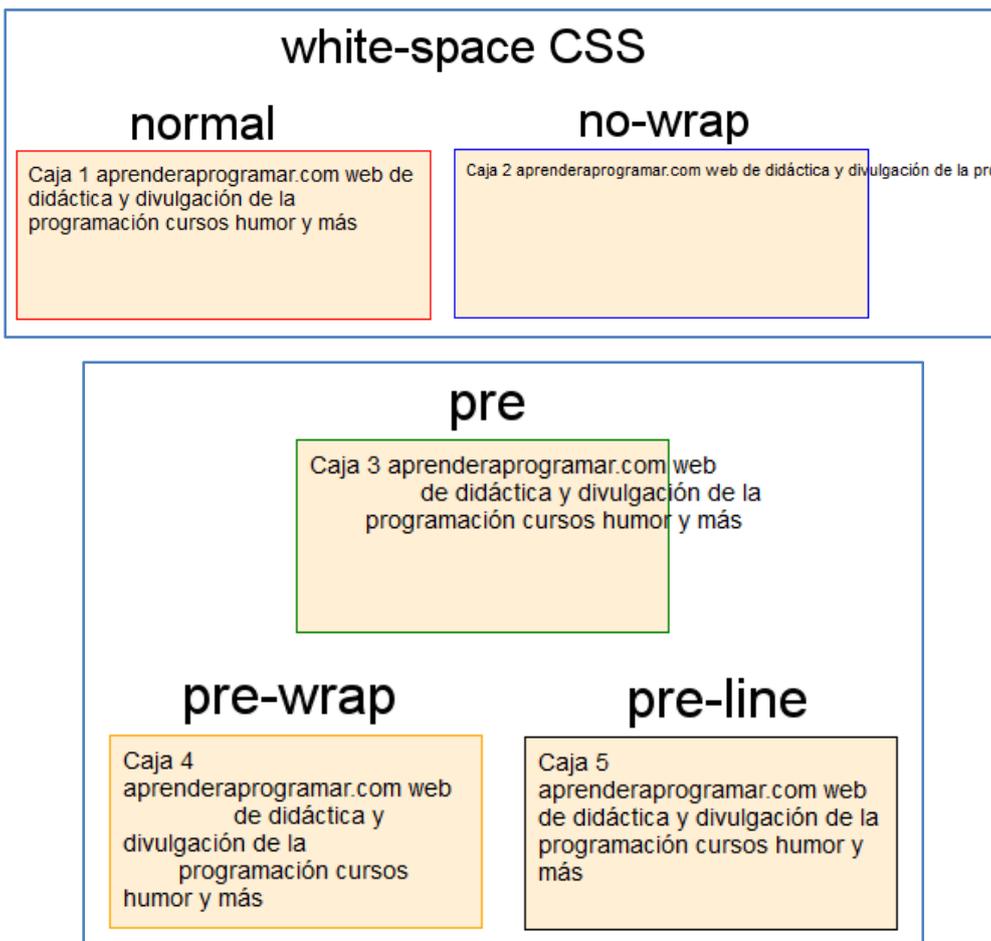
Puede resultar de gran interés para realizar ciertas maquetaciones de ciertas partes de páginas webs donde queremos que el texto se comporte de una manera específica en relación a los espacios en blanco, tabuladores y saltos de línea.

Crea un documento HTML con este contenido (respetando los espacios y saltos de línea):

```
<html>
<head>
<title>Portal web - aprenderaprogramar.com</title> <meta charset="utf-8">
<link rel="stylesheet" type="text/css" href="estilosCU01040DA.css">
</head>
<body>
<div id="caja1">Caja 1 aprenderaprogramar.com web
    de didáctica y divulgación de la
    programación cursos humor y más
</div>
<div id="caja2">Caja 2 aprenderaprogramar.com web
    de didáctica y divulgación de la
    programación cursos humor y más</div>
<div id="caja3">Caja 3 aprenderaprogramar.com web
    de didáctica y divulgación de la
    programación cursos humor y más</div>
<div id="caja4">Caja 4 aprenderaprogramar.com web
    de didáctica y divulgación de la
    programación cursos humor y más</div>
<div id="caja5">Caja 5 aprenderaprogramar.com web
    de didáctica y divulgación de la
    programación cursos humor y más</div>
<div class="limpiador"></div>
</body>
</html>
```

Y un archivo de hoja de estilos con estas reglas (pon el nombre de archivo adecuado) que nos permitirán ver el resultado del uso de white-space:

```
/* Curso CSS estilos aprenderaprogramar.com*/
*{font-family: arial;}
body {width:650px; border-style: dotted; border-width: 2px;}
div {border-style: solid; border-width:2px; margin:56px 0px 7px 7px;
padding:7px; background-color: #FFefd5; width:290px; height:110px; }
#caja1{ border-color:red; white-space:normal;}
#caja2{ border-color:blue; white-space: nowrap; font-size:12px;}
#caja3{ width: 225px; border-color:green; white-space: pre;}
#caja4{ width: 225px; border-color:orange; white-space: pre-wrap;}
#caja5{ width: 225px; border-color:black; white-space: pre-line}
.limpiador {clear:both; padding:0; width:0; height:0; border-style:none; }
```



En la caja 1 vemos el comportamiento normal, según el cual varios espacios en blanco seguidos son ignorados y el texto se va desarrollando sobre la caja y aprovechando un espacio en blanco para crear el salto de línea cuando es necesario debido a que el tamaño del contenedor es insuficiente.

En la caja 2 vemos el comportamiento no-wrap. El texto no salta de línea y se mantiene en una única línea, excediendo el tamaño del contenedor sin adaptarse a él. Si quisiéramos evitar que el texto salga fuera del contenedor podríamos usar la propiedad overflow.

En la caja 3 vemos el comportamiento con pre. El texto se mantiene tal cual ha sido definido en el documento HTML, incluyendo secuencias de espacios, tabuladores, saltos de línea, etc.

En la caja 4 vemos el comportamiento con pre-wrap. En este caso se mantiene el texto tal cual ha sido definido en el documento HTML, pero allí donde no hay espacio suficiente el texto crea nuevas líneas aprovechando espacios en blanco para hacerlo.

En la caja 5 vemos el comportamiento con pre-line. En este caso se mantienen los saltos de línea del documento HTML, pero los espacios en blanco se colapsan y se generan saltos de línea automáticos cuando el contenido excede el tamaño del contenedor.

EJERCICIO

Crea un documento HTML y un archivo con la hoja de estilos CSS que cumpla con estos requisitos:

- a) Deben existir tres contenedores (div1, div2 y div3) situados en horizontal, cada uno con margin 20px en todas direcciones, relleno de 10 píxeles en todas direcciones, ancho de 200 píxeles, altura de 400 píxeles y borde sólido de 4 píxeles de anchura.
- b) El div 1 debe tener alineación del texto centrada, color de texto #FF6347. Debe contener un texto suficientemente largo, con algunas palabras subrayadas y el primer párrafo indentado un 10%.
- c) El div 2 debe tener alineación del texto a la derecha, color de texto #008080. Debe contener un texto suficientemente largo, con algunas palabras subrayadas por encima y el primer párrafo indentado en 50 píxeles. Este div estará definido en el HTML como un texto con saltos de línea definidos en ciertos puntos. Al mostrarse por pantalla, deben usarse propiedades CSS para se mantengan los saltos de línea definidos en el documento HTML, se colapsen secuencias de espacios en blanco y se usen espacios en blanco para generar nuevas líneas si es necesario
- d) El div 3 debe tener alineación del texto justificada, color de texto #8B4513. Debe contener un texto suficientemente largo, con algunas palabras tachadas, y el primer párrafo indentado en un 20%. Además el texto contendrá una frase donde existan 15 espacios en blanco seguidos, y dichos espacios deben mostrarse cuando se visualice el texto (si coincide con un borde del contenedor deberán continuar en una nueva línea).

Para comprobar si tu código y respuestas son correctas puedes consultar en los foros aprenderaprogramar.com.

Próxima entrega: CU01041D

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