



# Hyphens CSS: Separar palabras con guiones al final de línea

- 
- C.F.G.S. DAW
  - 6 horas semanales
  - Mes aprox. de impartición: Ene - Feb
  - iPasen - [cjaedia071@g.educaand.es](mailto:cjaedia071@g.educaand.es)

Carmelo José Jaén Díaz

# ———— 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

---



¿Te ha pasado que al diseñar una página web te encuentras con el problema de que las palabras no caben correctamente en las líneas de texto y necesitas usar el guion para dividir las de forma automática? Este inconveniente puede hacer que tus diseños se vean desordenados y poco profesionales. Pero ¡no te preocupes! Existe una solución fácil de implementar con CSS: la propiedad **hyphens** CSS.

En esta lección, se estudia cómo esta propiedad te ayuda a separar las palabras con guiones al final de línea automáticamente.



# ENTENDIENDO *Hyphens* CSS: SOLUCIÓN PARA TEXTOS QUE SE DESBORDAN



Si no nos gusta cómo quedan nuestros textos porque desbordan los contenedores o quedan líneas de texto demasiado vacías, la solución está en dividir las palabras para que quepan en el espacio que queda, y para ello, se utiliza **hyphens** de CSS. Esta propiedad nos permite controlar la división de palabras al final de una línea de texto, mejorando la legibilidad y estética del contenido. Echemos un vistazo a un ejemplo para comprender mejor cómo funciona esta herramienta y cómo puede ayudarte en tus diseños.

# EJEMPLO DE USO DE *Hyphens* CSS

---



Este ejemplo muestra cómo la propiedad `hyphens` en CSS puede influir en la forma en que se muestran las palabras al final de una línea de texto:

1. **Texto sin `hyphens`:** En este primer contenedor, el texto se presenta sin el uso de la propiedad `hyphens`. Esto significa que el navegador no tiene instrucciones específicas sobre cómo manejar la división de palabras al final de la línea. Como resultado, cuando el texto es demasiado largo para caber en una línea, se desborda el contenedor. Esto afecta a la legibilidad del texto y hace que el diseño general parezca desordenado.
2. **Texto con `hyphens`:** En el segundo contenedor, se aplica la propiedad `hyphens` con el valor `auto`. Esto le indica al navegador que determine automáticamente dónde dividir las palabras al final de la línea para evitar el desbordamiento del texto. Al especificar `auto`, el navegador utilizará las reglas de separación de sílabas del idioma predeterminado del documento para dividir las palabras de manera adecuada. Esto mejora la legibilidad del texto y ayuda a mantener un diseño limpio y ordenado.



# EJEMPLO DE USO DE *Hyphens* CSS

---



Código HTML:

```
<div class="container">
  <p class="texto-sin-hyphens">
<b>Texto sin hyphens:</b> Este es
un texto largo que puede desbordar
su contenedor si no se maneja
correctamente. Cuando las palabras
no se separan adecuadamente al
final de la línea, la legibilidad
del texto se ve afectada y el
diseño general puede parecer
desordenado.</p>
</div>
```

```
<div class="container">
  <p class="texto-con-hyphens"
lang="es"><b>Texto con hyphens:</b>
Este es un texto largo que puede
desbordar su contenedor si no se
maneja correctamente. Cuando las
palabras no se separan
adecuadamente al final de la línea,
la legibilidad del texto se ve
afectada y el diseño general puede
parecer desordenado.</p>
</div>
```

# EJEMPLO DE USO DE *Hyphens* CSS



## Código CSS:

```
.container {  
  border: 1px solid #ccc;  
  padding: 10px;  
  margin-right: 40px;  
  width: 230px;  
  font-size: 42px;  
  display: inline-flex;  
}  
.texto-con-hyphens {  
  hyphens: auto;  
}
```

no se maneja correctamente. Cuando las palabras no se separan adecuadamente al final de la línea, la legibilidad del texto se ve afectada y el diseño general puede parecer

no se maneja correctamente. Cuando las palabras no se separan adecuadamente al final de la línea, la legibilidad del texto se ve afectada y el diseño general

La salida del código anterior puede verse en la imagen adjunta o en el siguiente link:

<https://codepen.io/Carmelo-Jos-Ja-n-D-az/pen/dPbZxPK>

# PROPIEDAD *hyphens* CSS:

## Sintaxis y valores

---



La propiedad **hyphens** en CSS controla la división de palabras al final de una línea cuando el texto se desborda.

### SINTAXIS

```
.hyphenation {  
    hyphens: auto | manual | none;  
}
```

### VALORES

- **none**: Desactiva la división automática y manual de palabras. El texto puede desbordarse si no cabe en el contenedor.

### VALORES

- **auto**: El navegador decide dónde dividir las palabras para evitar el desbordamiento del texto. Utiliza las reglas de separación de sílabas del idioma predeterminado del documento o del sistema operativo.
- **manual**: Las palabras no se dividirán automáticamente. Debes utilizar el carácter de guión (-) para indicar dónde se pueden dividir las palabras manualmente.

# PROPIEDAD *hyphens* CSS:

## Ejemplo



Por ejemplo, para permitir la división automática de palabras, puedes usar:

```
.hyphenation {  
  hyphens: auto;  
}
```

Esto permitirá que el navegador decida dónde dividir las palabras para evitar el desbordamiento del texto.

# DEFINIR EL IDIOMA PARA UNA CORRECTA SEPARACIÓN DE SÍLABAS CON *hyphens* CSS

---



Además de la sintaxis y los valores de la propiedad `hyphens`, es importante definir el idioma del contenido para que la división de palabras funcione correctamente. Esto se logra utilizando la propiedad `lang` en HTML para especificar el idioma del texto y proporcionar al navegador la información necesaria para aplicar las reglas de separación de sílabas adecuadas.

Como sabemos, según la ortografía de la lengua española, se puede emplear el guion para dividir aquellas palabras situadas al final de una línea que, por falta de espacio, deben escribirse en parte en la siguiente línea. La división de estas palabras no se puede realizar en cualquier lugar porque existen unas reglas específicas para hacer correctamente las separaciones. Por ello, en CSS es necesario especificar el idioma del texto que se está formateando. Por ejemplo, podríamos hacer estas divisiones `te-/léfono`, `telé-/fono` o `teléfo-/no`, pero no estas `tel/éfono`, `teléf/ono` o `teléfon/o`.

# DEFINIR EL IDIOMA PARA UNA CORRECTA SEPARACIÓN DE SÍLABAS CON *hyphens* CSS

---

Por ejemplo, si el texto está en español, debes incluir la siguiente declaración en el elemento `<html>`:

```
<html lang="es">
```

Esto indica que el contenido del documento está en español, lo que permite al navegador utilizar las reglas de separación de sílabas específicas del idioma español al dividir las palabras al final de una línea.

# FRAGMENTOS DE TEXTO EN OTRO IDIOMA AL PREDEFINIDO EN LA ETIQUETA HTML

---



Es importante destacar que si tienes fragmentos de código o texto en otro idioma dentro de tu documento HTML, puedes especificar el idioma correspondiente utilizando la etiqueta **lang** en esos elementos específicos. Por ejemplo, si tienes una cita o un fragmento de texto en inglés dentro de un documento en español, puedes agregar **lang="en"** al elemento que contiene ese texto en inglés.

```
<p lang="en">This is an example sentence in English within a paragraph.</p>
```

De esta manera, el navegador puede aplicar las reglas de separación de palabras adecuadas para ese idioma específico, asegurando una división de palabras correcta y coherente en todo el contenido del sitio web, independientemente del idioma en que esté escrito.

# EJEMPLOS DE USO

## Usando hyphens en cards



El siguiente código crea una card con una imagen, un título (`h3`) y un párrafo (`p`). Ambos elementos de texto (`h3` y `p`) tienen la propiedad `hyphens` establecida en `auto`, lo que permite que el navegador divida automáticamente las palabras al final de una línea si es necesario para evitar el desbordamiento del contenedor. Esto mejora la legibilidad del texto, especialmente en dispositivos con pantallas más pequeñas donde el espacio es limitado.



# EJEMPLOS DE USO

## Usando hyphens en cards



Código HTML:

```
<div class="card">
  
  <h3>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</h3>
  <p>Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris
nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in
reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
pariatur. </p>
</div>
```

# EJEMPLOS DE USO

## Usando hyphens en cards



### Código CSS:

```
body {  
  font-family: Arial, sans-serif;  
  margin: 0;  
  padding: 0;  
  background-color: #f4f4f4;  
}  
  
.card {  
  max-width: 300px;  
  background-color: #fff;  
  border-radius: 8px;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
  margin: 20px auto;  
  padding: 20px; }
```

```
.card img {  
  width: 100%;  
  border-radius: 8px;  
  margin-bottom: 10px;  
}  
  
.card h3 {  
  font-size: 22px;  
  margin-bottom: 10px;  
  hyphens: auto;  
}  
  
.card p {  
  font-size: 18px;  
  line-height: 1.5;  
  hyphens: auto; }
```

# EJEMPLOS DE USO

## Usando hyphens en cards



La salida del código anterior puede verse en la imagen adjunta o en el siguiente link:

<https://codepen.io/Carmelo-Jos-Ja-n-D-az/pen/XJrVKxK>



# EJEMPLOS DE USO

## Usando *hyphens* con *line-clamp*



La combinación de las propiedades `hyphens` y `line-clamp` puede ser especialmente útil para controlar la legibilidad del texto en diseños responsivos. `Hyphens` se encarga de dividir las palabras al final de una línea, mientras que `line-clamp` trunca el texto que excede un número específico de líneas y muestra puntos suspensivos al final del texto truncado.

En el siguiente ejemplo mostramos un extracto de texto en una card con un máximo de tres líneas, y conseguimos que las palabras se dividan automáticamente al final de una línea si es necesario.

# EJEMPLOS DE USO

## Usando *hyphens* con *line-clamp*



Código HTML:

```
<div class="card">
  
  <div class="texto">
    <h3>Título de la card</h3>
    <p>Este es un texto largo que puede desbordar su contenedor si no se
utiliza hyphens. Cuando las palabras no se separan adecuadamente al final
de la línea, la legibilidad del texto se ve afectada y el diseño general
puede parecer desordenado.</p>
  </div>
</div>
```

# EJEMPLOS DE USO

## Usando *hyphens* con *line-clamp*



### Código CSS:

```
.card {  
  border: 1px solid #ccc;  
  border-radius: 8px;  
  overflow: hidden;  
  width: 200px;  
}  
  
.texto {  
  padding: 10px;  
}
```

```
.texto h3 {  
  margin-bottom: 10px;  
}  
  
.texto p {  
  display: -webkit-box;  
  -webkit-line-clamp: 3;  
  /* Mostrar solo 3 líneas de texto */  
  -webkit-box-orient: vertical;  
  overflow: hidden;  
  hyphens: auto;  
  /* Dividir palabras al final de línea */  
}
```

# EJEMPLOS DE USO

## Usando *hyphens* con *line-clamp*



La salida del código anterior puede verse en la imagen adjunta o en el siguiente link:

<https://codepen.io/Carmelo-Jos-Ja-n-D-az/pen/gbYoM>

EB



# COMPATIBILIDAD DE *hyphens* CSS CON LOS NAVEGADORES ACTUALES

---



Según Can I Use, la propiedad CSS *hyphens* tiene una buena compatibilidad con los navegadores modernos, incluidos Chrome, Firefox, Safari, Edge y Opera. Sin embargo, es posible que algunos navegadores más antiguos no la admitan completamente. Es importante verificar la compatibilidad específica según las necesidades del proyecto y considerar soluciones alternativas para garantizar una experiencia consistente para todos los usuarios.

Ahora que has aprendido sobre la propiedad *hyphens* en CSS y cómo puede ayudarte a mejorar la apariencia y legibilidad del texto te queda aplicar este conocimiento en tus proyectos. Recuerda siempre definir el idioma adecuado en tu documento HTML y utilizar la propiedad *hyphens* de manera estratégica para evitar desbordamientos y líneas de texto vacías.