



Carmelo José Jaén Díaz



# Scroll Snap CSS: Controla el desplazamiento de tu web

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

# 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

---



El **Scroll Snap**, desplazamiento con ajuste o con bloqueo en español, es una propiedad de CSS que nos permite controlar el desplazamiento o scroll en contenedores para que sea suave. Esta característica es particularmente útil en sitios web que necesitan una navegación fluida y organizada entre secciones de contenido.



# *Scroll Snap* EN CSS

---



Anteriormente, conseguir este tipo de comportamiento requería el uso de JavaScript. Sin embargo, gracias a Scroll Snap de CSS, ahora podemos implementarlo con solo unas pocas líneas de código gracias a la propiedad `scroll-snap-type`.

Esta propiedad nos permite **definir puntos de *anclaje*** a los que el desplazamiento se detendrá de forma automática al hacer scroll. Esto es muy interesante en sitios web que presentan una disposición en bloques horizontales o verticales, facilitando que los usuarios naveguen de manera fluida y directa entre sus secciones.

# ¿PARA QUÉ SIRVE EL *Scroll Snap* EN CSS?

---



El desplazamiento o scroll es la acción de mover el contenido de una página web hacia arriba o hacia abajo para ver áreas no visibles en la pantalla. En un desplazamiento convencional, el usuario puede deslizar el contenido de forma continua, y este se mueve en función de la dirección y la intensidad del deslizamiento.

Sin embargo, con el **Scroll Snap** de CSS, podemos modificar este comportamiento para que el desplazamiento se alinee con ciertos puntos de anclaje dentro del contenedor. Por ejemplo, al desplazar hacia abajo en una página web, en lugar de permitir un movimiento fluido, el desplazamiento puede detenerse automáticamente en ciertos puntos predefinidos, como el inicio de cada sección o bloque de contenido.



# IMPLEMENTACIÓN DE *Scroll Snap* EN CSS

---



Para implementar *Scroll Snap* en CSS, aplicaremos la propiedad `scroll-snap-type` al contenedor correspondiente con la siguiente sintaxis.

## SINTAXIS

```
.contenedor-scroll {  
    scroll-snap-type: [x | y | block | inline] [mandatory | proximity];  
}  
.seccion {  
    scroll-snap-align: [start | center | end | none];  
}
```

# IMPLEMENTACIÓN DE *Scroll Snap* EN CSS

---



- La propiedad `scroll-snap-type` especifica cómo se debe comportar el desplazamiento dentro del contenedor. Puede definir el tipo de desplazamiento en los ejes horizontal (`x`) o vertical (`y`), así como también en bloque (`block`) o en línea (`inline`). Además, determina si el desplazamiento debe detenerse obligatoriamente en los puntos de anclaje especificados (`mandatory`) o si puede detenerse en el punto más cercano (`proximity`).
- Un aspecto importante a tener en cuenta es la propiedad `scroll-snap-align`, que se aplica a los elementos hijos del contenedor. Esta propiedad determina cómo se alineará cada elemento con respecto al punto de anclaje. Los valores comunes para `scroll-snap-align` son `start`, `end`, `center`, `none`, etc. Por ejemplo, `scroll-snap-align: start;` alinea el inicio del elemento con el punto de anclaje.

# IMPLEMENTACIÓN DE *Scroll Snap* EN CSS

## Ejemplo



Supongamos que tenemos un contenedor que contiene varias secciones y queremos aplicar *Scroll Snap* para que el desplazamiento se detenga en cada sección. Para ello, creamos un contenedor con la clase *contenedor-scroll* que contiene varias secciones con la clase *seccion*. Aplicamos *Scroll Snap* al contenedor verticalmente (*scroll-snap-type: y mandatory*), lo que hace que el desplazamiento se detenga obligatoriamente en cada sección. Además, establecemos *scroll-snap-align: start* en cada sección para alinear el inicio de cada una con el punto de anclaje.

# IMPLEMENTACIÓN DE *Scroll Snap* EN CSS

## Ejemplo



Código HTML:

```
<div class="contenedor-scroll">  
  <div class="seccion seccion-1">Sección 1</div>  
  <div class="seccion seccion-2">Sección 2</div>  
  <div class="seccion seccion-3">Sección 3</div>  
  <div class="seccion seccion-4">Sección 4</div>  
</div>
```

# IMPLEMENTACIÓN DE *Scroll Snap* EN CSS

## Ejemplo



Código CSS:

```
body, html {  
    overflow: hidden;  
}  
  
.contenedor-scroll {  
    height: 300px; /* Altura del  
contenedor */  
    overflow-y: scroll; /* Habilitar el  
desplazamiento vertical */  
    scroll-snap-type: y mandatory; /*  
Desplazamiento vertical obligatorio */  
}
```

```
.seccion {  
    height: 300px; /* Altura de cada  
sección */  
    scroll-snap-align: start; /*  
Alinear el inicio de cada sección con el  
punto de anclaje */  
    font-size: 22px;  
    padding: 20px;  
    color: white;  
    font-weight: bold;  
}
```

# IMPLEMENTACIÓN DE *Scroll Snap* EN CSS

## Ejemplo



Código CSS:

```
.seccion-1 {  
    background-color: #FF6B6B;  
}  
.seccion-2 {  
    background-color: #59d6b2;  
}  
.seccion-3 {  
    background-color: #e8c168;  
}  
.seccion-4 {  
    background-color: #6B6BFF;  
}
```

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/JoPOmZo>



# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Secciones que ocupan el 100% del alto



Con **Scroll Snap**, podemos crear secciones de contenido que ocupen todo el alto de la pantalla y se desplacen de manera suave. Esto es útil para presentaciones de una sola página donde cada sección se alinearé perfectamente y los usuarios podrán navegar de manera intuitiva por el contenido.

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Secciones que ocupan el 100% del alto



Código HTML:

```
<div class="scroll-container">
  <div class="scroll-item item1">¡Hola!</div>
  <div class="scroll-item item2">¿Qué tal?</div>
  <div class="scroll-item item3">¿Practicando con Scroll Snap?</div>
  <div class="scroll-item item4">Me gusta el CSS</div>
  <div class="scroll-item item5">♥</div>
</div>
```



# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Secciones que ocupan el 100% del alto



Código CSS:

```
body, html {  
    margin: 0;  
    padding: 0;  
    height: 100%;  
    font-family: Arial;  
    font-size: 24px;  
}
```

```
.scroll-container {  
    width: 100%;  
    height: 100%;  
    overflow-y: auto;  
    scroll-snap-type: y mandatory;  
    /* Activamos el desplazamiento por  
    defecto */  
}
```

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Secciones que ocupan el 100% del alto



Código CSS:

```
.scroll-item {  
    height: 100vh; /* Ocupar toda la  
altura de la ventana */  
    scroll-snap-align: start; /*  
Alineamos los elementos al inicio de la  
ventana de visualización */  
    display: flex;  
    justify-content: center;  
    align-items: center;  
    font-size: 2rem;  
}
```

```
.item1 {  
    background-color: #ff9999;  
}  
.item2 {  
    background-color: #99ff99;  
}  
.item3 {  
    background-color: #9999ff;  
}  
.item4 {  
    background-color: #ffff99;  
}  
.item5 {  
    background-color: #ff99ff;  
}
```

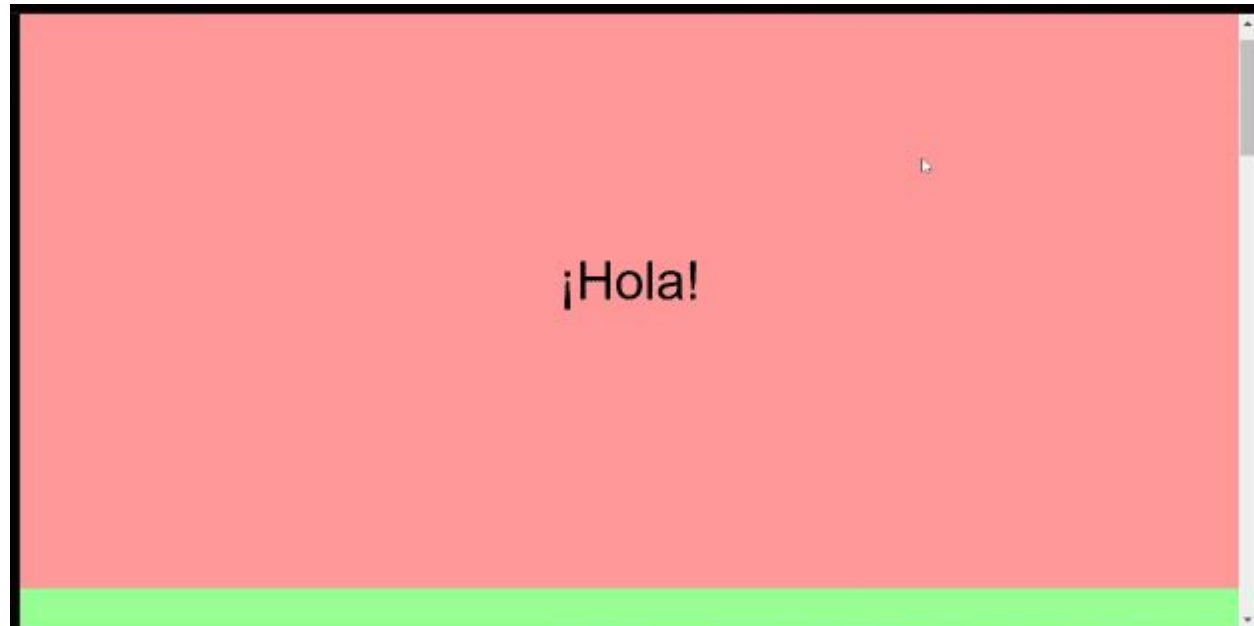
# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Secciones que ocupan el 100% del alto



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/EaYbddY>



# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Productos en contenedores horizontales



En tiendas online, las galerías de productos pueden contar con **Scroll Snap** para permitir a los usuarios explorar los diferentes productos de forma horizontal. Cada producto se muestra en su propia tarjeta o *card* y el **Scroll Snap** asegura que cada tarjeta esté alineada perfectamente cuando se hace *scroll* horizontal, facilitando la comparación entre productos.

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Productos en contenedores horizontales



Código HTML:

```
<div class="product-gallery">
  <div class="product-card">
    
    <h3>Producto 1</h3>
    <p>Descripción del Producto
1</p>
    <button>Añadir al
carrito</button>
  </div>
```

```
<div class="product-card">
  
  <h3>Producto 2</h3>
  <p>Descripción del Producto
2</p>
  <button>Añadir al
carrito</button>
</div>
```

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Productos en contenedores horizontales

---



Código HTML:

```
<div class="product-card">
  
  <h3>Producto 3</h3>
  <p>Descripción del Producto
3</p>
  <button>Añadir al
carrito</button>
</div>
```

```
<div class="product-card">
  
  <h3>Producto 4</h3>
  <p>Descripción del Producto
3</p>
  <button>Añadir al
carrito</button>
</div>
```

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Productos en contenedores horizontales



Código HTML:

```
<div class="product-card">
  
  <h3>Producto 5</h3>
  <p>Descripción del Producto
1</p>
  <button>Añadir al
carrito</button>
</div>
```

```
<div class="product-card">
  
  <h3>Producto 6</h3>
  <p>Descripción del Producto
2</p>
  <button>Añadir al
carrito</button>
</div>
```

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Productos en contenedores horizontales



Código HTML:

```
<div class="product-card">
  
  <h3>Producto 7</h3>
  <p>Descripción del Producto
3</p>
  <button>Añadir al
carrito</button>
</div>
```

```
<div class="product-card">
  
  <h3>Producto 8</h3>
  <p>Descripción del Producto
3</p>
  <button>Añadir al
carrito</button>
</div>
</div>
```



# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Productos en contenedores horizontales



### Código CSS:

```
.product-gallery {  
  display: flex;  
  overflow-x: scroll;  
  scroll-snap-type: x mandatory;  
}  
  
.product-card {  
  flex: 0 0 auto;  
  width: 200px;  
  margin-right: 10px;  
  scroll-snap-align: start;  
  text-align: center;  
}
```

```
.product-card img {  
  width: 100px;  
  height: auto;  
}  
  
.product-card h3, .product-card p {  
  margin: 10px 0;  
}  
  
.product-card button {  
  background-color: #007bff;  
  color: #fff;  
  border: none;  
  padding: 10px 20px;  
  cursor: pointer;  
  transition: background-color 0.3s;  
}  
  
.product-card button:hover {  
  background-color: #0056b3;  
}
```

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Productos en contenedores horizontales



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/raBYRZL>



# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Galería vertical con texto descriptivo



En esta ocasión, vamos a crear una galería de imágenes con texto dispuesto de forma vertical. Esto puede ser útil cuando queremos mostrar imágenes con descripciones o detalles adicionales debajo de cada imagen. Cada imagen está alineada al inicio del contenedor gracias a la propiedad `scroll-snap-align`. Al hacer *scroll*, nos detenemos en cada elemento. Esto puede ser útil para presentaciones de productos, portafolios o cualquier otro tipo de contenido visual que queramos destacar.

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Galería vertical con texto descriptivo



Código HTML:

```
<div class="gallery">
  <div class="image"></div>
  <p>Lorem ipsum dolor sit amet,
consectetur adipiscing elit.</p>
  <p>Lorem ipsum dolor sit amet,
consectetur adipiscing elit.</p>
```

```
  <div class="image"></div>
  <p>Lorem ipsum dolor sit amet,
consectetur adipiscing elit.</p>
  <p>Lorem ipsum dolor sit amet,
consectetur adipiscing elit.</p>
```

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Galería vertical con texto descriptivo



Código HTML:

```
<div class="image"></div>
```

```
<p>Lorem ipsum dolor sit amet,  
consectetur adipiscing elit.</p>
```

```
<p>Lorem ipsum dolor sit amet,  
consectetur adipiscing elit.</p>
```

```
<div class="image"></div>
```

```
<p>Lorem ipsum dolor sit amet,  
consectetur adipiscing elit.</p>
```

```
<p>Lorem ipsum dolor sit amet,  
consectetur adipiscing elit.</p>
```

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Galería vertical con texto descriptivo



### Código HTML:

```
<div class="image"></div> <p>Lorem ipsum dolor sit  
amet, consectetur adipiscing elit.</p>  
  <p>Lorem ipsum dolor sit amet,  
consectetur adipiscing elit.</p>  
</div>
```

### Código CSS:

```
body, html {  
  margin: 0;  
  padding: 0;  
  font-family: Arial, sans-serif;  
}
```

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Galería vertical con texto descriptivo



### Código CSS:

```
.gallery {  
  width: 100%;  
  max-width: 600px;  
  margin: 0 auto;  
  height: 100vh;  
  overflow-y: scroll;  
  scroll-snap-type: y mandatory;  
  display: flex;  
  flex-direction: column;  
}
```

```
.image {  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  font-size: 2em;  
  scroll-snap-align: start; /* Hace que  
  cada imagen se alinee al inicio del  
  contenedor */  
}  
p {  
  padding: 20px;  
}
```

# EJEMPLOS PRÁCTICOS DE *Scroll Snap*

## Galería vertical con texto descriptivo



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/raBYRZL>





# COMPATIBILIDAD DE *Scroll snap* CON LOS NAVEGADORES WEB

---



Antes de implementar **Scroll Snap** en tu proyecto, es importante verificar su compatibilidad con los navegadores web más utilizados. Para ello, podemos consultar la compatibilidad de **Scroll Snap** con los navegadores utilizando herramientas como Can I Use. En general, la propiedad cuenta con un amplio soporte en la mayoría de los navegadores modernos, lo que nos ofrece la confianza de implementarla sin problemas en la mayoría de los casos.

Como has visto, el **Scroll Snap** en CSS te permite controlar el desplazamiento, o *scroll* horizontal y vertical, de tu sitio web. Con esta funcionalidad, puedes crear experiencias de desplazamiento suaves para mejorar la navegación entre diferentes tipos de contenido.