

Diplomatura en



# FullStack Developer

---

CLASE 13– Pseudoclases,  
pseudoelementos y animaciones

# Selectores

## Selectores básicos

- **Selector de tipo:** para todos los elementos que coinciden con el nombre del elemento especificado→ Por ejemplo si queremos modificar los input **en css**

```
input{  
} #Todos los elementos del tipo input serán seleccionados
```

- **Selector de clase:** tiene en cuenta el nombre de la clase, selecciona los elementos teniendo en cuenta su nombre

En html→ `<p class="subTitle">Contenido del párrafo</p>`

En css→ `.subTitle{  
} #se escriben de esta manera`

# Selectores básicos

- **Selector ID:** solo puede aplicarse a un elemento y no a varios de ellos

`<p class="subTitle1">Contenido del párrafo 1</p>`

`<p class="subTitle2">Contenido del párrafo 2</p>`

→ en css

```
#subTitle{  
}
```

- **Selector Universal:** selecciona todos los elementos, y se puede restringir a un espacio de nombre específico o a todos los espacios de nombre

→ en css

```
*{ /*Propiedades a incluir en el selector  
}
```

# Selectores básicos

## Combinadores:

- **Combinador general de hermanos:** el segundo elemento sigue al primero y ambos comparten el mismo elemento como padre

En css → `p ~ span {`  
                  `font-size: 20px;`  
                  `}/*` el elemento afectado será el `<span>` que esté después de un elemento `<p>` y que ambos sean hijos del elemento padre.

- **Combinador de hijo:** `">"` selecciona los elementos que son hijos directos del primer elemento

En css → `.contenedor > p {`  
                  `}/*` selecciona a todos los hijos de elementos con la clase `contenedor`.

# Selectores básicos

## Pseudo Classes en Css:

Es una **palabra clave** que se añade a los selectores y que especifica un estado especial del elemento seleccionado

```
En css→ selector: pseudoclase{  
    propiedad: valor;  
}/* primero va el selector y luego la pseudo clase separada por :
```

Por ejemplo→ **:hover**→ aplicará un estilo cuando el usuario haga hover sobre el elemento especificado por el selector.

# Sintaxis

- **Selectores**

Se utilizan para seleccionar los elementos HTML a los que se aplicarán los estilos. Pueden ser elementos específicos (por ejemplo, ejemplo, “ para párrafos), clases (por ejemplo, clase” para elementos con una clase específica) o identificadores (por ejemplo, ejemplo, “# para elementos con un ID específico).

- **Propiedades**

Son los atributos que definen cómo se verá un elemento seleccionado. Pueden incluir propiedades como “font size background color”, entre muchas otras. Cada propiedad tiene un valor asociado que determina cómo se aplicará la propiedad al elemento seleccionado.

- **Valores**

Son los posibles ajustes que se pueden aplicar a una propiedad. Por ejemplo, para la propiedad “color”, los valores pueden ser nombres de colores como red o blue, códigos hexadecimales como #ff0000 o valores de color RGB como rgb(255,0,0)

# Sintaxis

- **Reglas**

Está compuesta por un selector y un conjunto de declaraciones rodeadas por llaves. El selector selecciona los elementos HTML a los que se aplicarán las declaraciones. A su vez, las declaraciones están formadas por propiedades y valores.

- **Comentarios**

Los comentarios se escriben precedidos de `/*` y `*/`, permiten añadir notas o explicaciones al código.

No se muestran en la página web sino que sirven a los desarrolladores de guía sobre el trabajo.

- **Anidamiento**

Se puede anidar selectores dentro de otros selectores para aplicar estilos a elementos específicos dentro de otros elementos más generales.

# Sintaxis Ejemplo

```
.contenedor{  
  Background-color : red; /* el color elegido es el indicado */  
}
```

Entonces seleccionamos el elemento a través de su clase  
Abrimos {

Dentro de las llaves colocamos primero la propiedad, luego  
: y luego el color, expresado en cualquiera de las formas  
antes indicada.

El ; indica que cerramos la propiedad  
Cerramos }



# Propiedades

Propiedad	Descripción
color	Establece el color del texto.
font-size	Define el tamaño de la fuente del texto.
font-family	Especifica la fuente utilizada para el texto.
background-color	Establece el color de fondo de un elemento.
width	Define el ancho de un elemento.
height	Establece la altura de un elemento.
margin	Controla los márgenes externos de un elemento.

# Propiedades

Propiedad	Descripción
padding	Define el relleno interno de un elemento.
border	Establece el estilo, el ancho y el color del borde de un elemento.
display	Define cómo se muestra un elemento en el flujo del documento.
position	Controla la posición de un elemento en relación con su contenedor.
float	Permite que un elemento flote hacia la izquierda o la derecha.
text-align	Alinea el contenido de texto dentro de un elemento.
text-decoration	Agrega decoración al texto, como subrayado o tachado.
box-shadow	Agrega una sombra alrededor de un elemento.

# Animaciones

Las animaciones en CSS permiten a los desarrolladores web **agregar efectos dinámicos** a sus páginas sin necesidad de utilizar JavaScript. Estas animaciones pueden **mejorar la experiencia del usuario** y hacer que una página web sea más **atractiva visualmente**.



# Animaciones Conceptos básicos

Las propiedades que se utilizan son principalmente dos:

## 1. @keyframes

La regla @keyframes permite definir los cambios en las propiedades CSS en diferentes etapas de la animación. Por ejemplo:

css→

```
@keyframes example {  
  0% { opacity: 0; }  
  100% { opacity: 1; }  
}
```

# Animaciones Conceptos básicos

## 2. Propiedad animation

La propiedad animation se utiliza para aplicar la animación definida con @keyframes a un elemento. Aquí están las subpropiedades más comunes:

**animation-name:** Nombre de la animación definida con @keyframes.

**animation-duration:** Duración de la animación (por ejemplo, 2s)

**animation-timing-function:** Define cómo se distribuye la animación a lo largo del tiempo (por ejemplo, ease, linear, ease-in, ease-out).

**animation-delay:** Tiempo antes de que comience la animación.

**animation-iteration-count:** Número de veces que la animación se repite (por ejemplo, infinite para repetir indefinidamente).

**animation-direction:** Dirección de la animación (por ejemplo, normal, reverse, alternate).

# Animaciones Conceptos básicos

CSS→

```
@keyframes slidein {  
  from {  
    transform: translateX(0%);  
  }  
  to {  
    transform: translateX(100%);  
  }  
}  
.element {  
  animation-name: slidein;  
  animation-duration: 3s;  
  animation-timing-function: ease-in-out;  
}
```

# Propiedades adicionales de Animación

## animation-fill-mode

Esta propiedad especifica cómo se aplican los estilos de la animación antes y después de que la animación se ejecute.

**none:** No aplica estilos fuera de la animación.

**forwards:** Mantiene los estilos finales después de la animación.

**backwards:** Aplica los estilos iniciales antes de que comience la animación.

**both:** Aplica ambos estilos, iniciales y finales.

## animation-play-state

**Animation-play-state:** permite pausar y reanudar la animación;

**running** se esté ejecutando; **paused:** está pausada



# Propiedades adicionales de Animación

## animation-fill-mode

Esta propiedad especifica cómo se aplican los estilos de la animación antes y después de que la animación se ejecute.

**none:** No aplica estilos fuera de la animación.

**forwards:** Mantiene los estilos finales después de la animación.

**backwards:** Aplica los estilos iniciales antes de que comience la animación.

**both:** Aplica ambos estilos, iniciales y finales.

## animation-play-state

**Animation-play-state:** permite pausar y reanudar la animación;

**running** se esté ejecutando; **paused:** está pausada



# Ejemplo completo

## HTML

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Ejemplo de Animación CSS</title>
  <style>
    @keyframes colorChange {
      0% { background-color: red; }
      50% { background-color: yellow; }
      100% { background-color: green; }
    }

    .animated-box {
      width: 100px;
      height: 100px;
      background-color: red;
      animation: colorChange 5s infinite;
    }
  </style>
</head>
<body>
  <div class="animated-box"></div>
</body>
</html>
```

Para profundizar en las animaciones en CSS, puedes consultar las siguientes fuentes:

MDN Web Docs sobre Animaciones CSS

W3Schools: CSS Animations

Las animaciones CSS son una herramienta poderosa para agregar dinamismo y atractivo visual a las páginas web. Con una comprensión sólida de sus propiedades y cómo utilizarlas, puedes crear efectos sorprendentes y mejorar la experiencia del usuario.