**Hover, Focus y otros estados en Tailwind CSS**

* **Tabla de contenido**
* **Hover**
* **Focus**
* **Active**

**Hover**

En Tailwind CSS, la pseudo-clase hover se utiliza para aplicar estilos a un elemento cuando el puntero del mouse se encuentra sobre él. Es una forma sencilla y poderosa de agregar interactividad y mejorar la experiencia del usuario al resaltar elementos en respuesta a la interacción del usuario.

La sintaxis básica para aplicar estilos hover en Tailwind CSS es agregar el prefijo hover: antes de la clase que deseas aplicar cuando el puntero está sobre el elemento. A continuación, se muestra un ejemplo simple de cómo usar la pseudo-clase hover:

<button

class="bg-blue-500 hover:bg-blue-700 text-white font-bold py-2 px-4 rounded"

>

Hover me

</button>

En este ejemplo, el botón tendrá un fondo azul cuando el puntero del mouse no esté sobre él (bg-blue-500), pero cuando el puntero se coloque sobre el botón, cambiará a un fondo azul más oscuro (bg-blue-700) debido al uso de hover:bg-blue-700.

Puedes aplicar hover a cualquier clase de Tailwind CSS, lo que te permite agregar interactividad a diversos elementos como botones, enlaces, íconos, tarjetas y más. A continuación, te mostraré cómo puedes utilizar hover con diferentes propiedades de estilo en Tailwind CSS:

Cambiar el color de texto al pasar el puntero sobre un enlace:

<a href="#" class="text-blue-500 hover:text-blue-700">Enlace</a>

Aumentar el tamaño de un ícono cuando se pasa el puntero sobre él:

<i class="fas fa-heart text-red-500 hover:text-red-700 text-xl"></i>

Mostrar una transición de opacidad al pasar el puntero sobre una imagen:

<img src="imagen.jpg" class="hover:opacity-75" alt="Imagen" />

Aplicar una sombra al pasar el puntero sobre una tarjeta:

<div class="bg-white shadow-lg p-4 rounded-md hover:shadow-xl">

Contenido de la tarjeta

</div>

Cambiar el color de fondo de una celda de tabla cuando se pasa el puntero sobre ella:

<table class="border-collapse">

<tr>

<td class="border px-4 py-2 hover:bg-gray-100">Celda 1</td>

<td class="border px-4 py-2 hover:bg-gray-100">Celda 2</td>

</tr>

</table>

Focus#

En CSS, la pseudo-clase :focus se utiliza para aplicar estilos a un elemento cuando este tiene el enfoque. Esto sucede cuando un elemento (como un campo de entrada o un enlace) recibe el enfoque del usuario, generalmente mediante un clic o al navegar mediante teclado (usando la tecla Tab, por ejemplo). El uso de estilos :focus es fundamental para mejorar la accesibilidad y la experiencia del usuario, ya que indica claramente qué elemento está siendo seleccionado o activo.

En Tailwind CSS, la sintaxis para aplicar estilos :focus es similar a la de :hover. Simplemente agrega el prefijo focus: antes de la clase que deseas aplicar cuando el elemento tiene el enfoque.

Aquí tienes algunos ejemplos de cómo utilizar :focus en Tailwind CSS:

Cambiar el color de borde de un campo de entrada cuando está enfocado:

<input type="text" class="border focus:border-blue-500" />

Resaltar un enlace con un color de fondo cuando está enfocado:

<a href="#" class="bg-blue-500 text-white px-4 py-2 rounded focus:bg-blue-700">

Enlace

</a>

Aumentar el tamaño de un botón cuando está enfocado:

<button

class="bg-green-500 hover:bg-green-700 text-white font-bold py-2 px-4 rounded focus:scale-110"

>

Hacer clic

</button>

Cambiar el color de fondo de una tarjeta cuando está enfocada:

<div

class="bg-white shadow-lg p-4 rounded-md hover:shadow-xl focus:bg-gray-100"

>

Contenido de la tarjeta

</div>

Es importante tener en cuenta que para que los estilos :focus sean aplicados, el elemento debe ser capaz de recibir el enfoque. Por ejemplo, enlaces (<a>), campos de entrada (<input>), áreas de texto (<textarea>) y elementos interactivos como botones (<button>) son elementos por defecto que pueden recibir el enfoque. Si deseas aplicar estilos :focus a otros elementos, como divs o párrafos, es posible que necesites agregar el atributo tabindex para permitir que el elemento reciba el enfoque de manera accesible.

**Active**

En Tailwind CSS, manejar el estado active se refiere a aplicar estilos a un elemento cuando está activado o presionado por el usuario. Esto ocurre generalmente cuando un usuario hace clic o toca un elemento interactivo, como un botón o un enlace. Al aplicar estilos active, puedes proporcionar retroalimentación visual instantánea al usuario para indicar que su interacción ha sido reconocida.

La sintaxis básica para aplicar estilos active en Tailwind CSS es agregar el prefijo active: antes de la clase que deseas aplicar cuando el elemento está activo. Es similar a cómo se utiliza hover y focus en Tailwind CSS.

Aquí tienes algunos ejemplos de cómo manejar el estado active en Tailwind CSS:

Cambiar el color de fondo de un botón cuando está activo:

<button

class="bg-blue-500 hover:bg-blue-700 active:bg-blue-800 text-white font-bold py-2 px-4 rounded"

>

Clic aquí

</button>

En este ejemplo, el botón tendrá un fondo azul oscuro cuando está activo debido al uso de active:bg-blue-800. Esta clase se aplicará cuando el botón está presionado por el usuario.

Cambiar el color de texto de un enlace cuando está activo:

<a href="#" class="text-blue-500 hover:text-blue-700 active:text-blue-800">

Enlace

</a>

En este caso, el enlace tendrá un color de texto azul oscuro cuando está activo debido al uso de active:text-blue-800.

Aumentar el tamaño de un botón cuando está activo:

<button

class="bg-green-500 hover:bg-green-700 active:scale-110 text-white font-bold py-2 px-4 rounded"

>

Clic aquí

</button>

Este botón se escalará ligeramente cuando está activo debido al uso de active:scale-110.

Al igual que con hover y focus, puedes combinar active con otras clases de Tailwind CSS para crear efectos más complejos y personalizados en respuesta a las interacciones del usuario. Además, también puedes personalizar tus propias variantes active y agregarlas al archivo de configuración (tailwind.config.js) para definir estilos activos personalizados para tu proyecto específico.

**APRENDE a manejar de los ESTADOS de TAILWIND**

<https://www.youtube.com/watch?v=G_1W5UPFFWk>