



# HTML

## Transiciones

¿Queréis hacer animaciones web pero no sabéis por dónde comenzar? Pues antes de poner los manos a la obra hay que entender las diferencias entre una transición y una animación en CSS.

Las **transiciones** proporcionan un cambio de un estado A a un estado B. Mientras que las **animaciones** dentro de CSS permiten modificar la apariencia y comportamiento de un elemento en varios fotogramas clave conocidos como **Keyframes**, que pueden ir de un estado A, a un estado B, C, etc.

### Ejemplo de transición

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="./style.css" />
    <title>Transiciones</title>
  </head>
  <body>
    <div class="container">
      <div class="box"></div>
    </div>
  </body>
</html>
```

```
.box {
  width: 200px;
  height: 200px;
  background: red;
  transition: background 1s linear;
}

.box:hover {
  background: blue;
}
```

Al hacer hover en la caja, el color de fondo cambia a azul gracias a la transición definida en la propiedad **background**.

## Personalización de Transiciones

Para personalizar una transición, se pueden utilizar diferentes propiedades:

- **transition-property**: Indica qué propiedad va a cambiar (ejemplo: `background-color`).
- **transition-duration**: Define la duración de la transición (ejemplo: `1s` para 1 segundo).
- **transition-timing-function**: Permite definir funciones de tiempo para controlar la velocidad de la transición (ejemplo: `linear`, `ease-in-out`).

Al añadir una transición adecuada, se logrará un efecto fluido y controlado en las animaciones web.