

Aquí tienes una tabla con ejemplos de **propiedades CSS** que pueden cambiar con **transiciones**, junto con diferentes valores de **duración**, **timing-function** y **delay**, y sus efectos visuales esperados.

Propiedad CSS	Descripción	Ejemplo de Cambio	Duración	Timing-Function	Delay	Efecto Esperado
background-color	Color de fondo	Azul → Rojo	1s	ease	0s	Cambio suave de color
Opacity	Transparencia	1 → 0.5	500ms	linear	0s	Se vuelve semi-transparente de forma uniforme
transform: scale()	Tamaño	scale(1) → scale(1.5)	1s	ease-in-out	0.3s	Crece con un efecto suave después de 0.3s
transform: rotate()	Rotación	0deg → 180deg	2s	ease-out	0s	Gira lentamente y desacelera al final
Width	Ancho	100px → 300px	1.5s	ease-in	0.5s	Se expande con una aceleración inicial, después de 0.5s
Height	Altura	50px → 200px	2s	ease	0s	Aumenta de tamaño de forma fluida
border-radius	Bordes redondeados	0px → 50px	700ms	ease-in-out	0s	Se transforma en un círculo suavemente
box-shadow	Sombra	sin sombra → 10px 10px 20px rgba(0,0,0,0.5)	1s	linear	0.2s	Aparece una sombra de forma uniforme tras 0.2s
Color	Color del texto	Negro → Rojo	800ms	ease-out	0s	Cambio progresivo del color del texto
letter-spacing	Espaciado entre letras	0px → 5px	1s	ease	0s	Las letras se separan progresivamente
text-shadow	Sombra de texto	Sin sombra → 2px 2px 5px black	600ms	ease-in-out	0s	La sombra del texto aparece suavemente
margin-left	Margen izquierdo	0px → 50px	1s	linear	0.3s	Se mueve a la derecha de forma uniforme tras 0.3s

Las animaciones en CSS permiten animar elementos sin necesidad de JavaScript. Se controlan con la regla @keyframes y se aplican mediante las siguientes propiedades:

Propiedad	Descripción	Ejemplo
animation	Define una animación en una sola línea.	animation: mover 2s ease-in-out infinite;
animation-name	Nombre de la animación, definido con @keyframes.	animation-name: mover;
animation-duration	Duración de la animación.	animation-duration: 2s;
animation-timing-function	Controla la aceleración (ease, linear, ease-in, ease-out, ease-in-out).	animation-timing-function: ease-in-out;
animation-delay	Retraso antes de que comience la animación.	animation-delay: 1s;
animation-iteration-count	Número de veces que se repite la animación (infinite para siempre).	animation-iteration-count: infinite;
animation-direction	Dirección de la animación (normal, reverse, alternate, alternate-reverse).	animation-direction: alternate;
animation-fill-mode	Controla el estado del elemento antes y después de la animación.	animation-fill-mode: forwards;
animation-play-state	Permite pausar o reanudar la animación (running, paused).	animation-play-state: paused;

2. Propiedades de Transformaciones (transform)

Las transformaciones en CSS permiten mover, rotar, escalar o sesgar elementos.

Propiedad	Descripción	Ejemplo
transform	Aplica una transformación (rotación, escala, etc.).	transform: rotate(45deg);
transform: translate(x, y)	Mueve el elemento en los ejes X e Y.	transform: translate(50px, 100px);
transform: scale(x, y)	Escala el tamaño del elemento.	transform: scale(1.5, 2);
transform: rotate(deg)	Rota el elemento.	transform: rotate(180deg);
transform: skew(x, y)	Inclina el elemento en los ejes X e Y.	transform: skew(10deg, 15deg);
transform: matrix(a, b, c, d, e, f)	Aplica una matriz de transformación avanzada.	transform: matrix(1, 0, 0, 1, 50, 100);

Transformaciones en 2D

Estas transformaciones afectan los elementos en un espacio **bidimensional** (X, Y).

Función	Descripción	Ejemplo
translate(x, y)	Mueve el elemento en el eje X e Y.	transform: translate(50px, 100px);
translateX(n)	Mueve solo en el eje X.	transform: translateX(80px);
translateY(n)	Mueve solo en el eje Y.	transform: translateY(50px);
scale(x, y)	Escala el tamaño del elemento.	transform: scale(1.5, 2);
scaleX(n)	Escala solo en el eje X.	transform: scaleX(2);
scaleY(n)	Escala solo en el eje Y.	transform: scaleY(1.5);
rotate(deg)	Rota el elemento en 2D.	transform: rotate(45deg);
skew(x-deg, y-deg)	Inclina el elemento en X y Y.	transform: skew(10deg, 15deg);
skewX(deg)	Inclina solo en el eje X.	transform: skewX(20deg);
skewY(deg)	Inclina solo en el eje Y.	transform: skewY(-10deg);
matrix(a, b, c, d, e, f)	Aplica una transformación avanzada combinada.	transform: matrix(1, 0.5, -0.5, 1, 30, 50);

Transformaciones en 3D

Estas transformaciones afectan los elementos en **tres dimensiones** (X, Y, Z).

Función	Descripción	Ejemplo
translate3d(x, y, z)	Mueve el elemento en los 3 ejes.	transform: translate3d(50px, 100px, 30px);
translateZ(n)	Mueve el elemento en el eje Z.	transform: translateZ(200px);
scale3d(x, y, z)	Escala el elemento en los 3 ejes.	transform: scale3d(1.5, 2, 0.8);
scaleZ(n)	Escala en el eje Z.	transform: scaleZ(1.2);
rotateX(deg)	Rota el elemento en el eje X.	transform: rotateX(180deg);
rotateY(deg)	Rota el elemento en el eje Y.	transform: rotateY(90deg);
rotateZ(deg)	Rota en el eje Z (igual que rotate()).	transform: rotateZ(30deg);
rotate3d(x, y, z, deg)	Rota en los 3 ejes al mismo tiempo.	transform: rotate3d(1, 1, 0, 45deg);
perspective(n)	Define profundidad en 3D.	perspective: 500px;