



UTN.BA

UNIVERSIDAD TECNOLÓGICA NACIONAL  
FACULTAD REGIONAL BUENOS AIRES

# **Diplomatura Profesional Front End Developer**

**Módulo 1:**  
**Desarrollo Web en HTML 5 y CSS 3**  
**HTML 5 y CSS3**

# Transformaciones

---

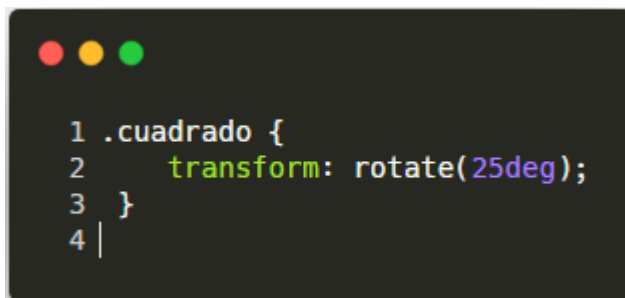
La propiedad CSS `transform` permite manipular el sistema de coordenadas de un elemento usando las funciones de transformación.

## rotate

La función CSS **`rotate()`** define una transformación que gira un elemento alrededor de un punto fijo en un plano 2D sin deformarlo.

El punto fijo alrededor del cual gira el elemento, mencionado anteriormente, también se conoce como el origen de transformación. Se establece de manera predeterminada en el centro del elemento.

```
transform: rotate(deg);
```



```
1 .cuadrado {  
2   transform: rotate(25deg);  
3 }  
4 |
```

## scale

La función CSS **`scale()`** define una transformación que modifica el tamaño de un elemento en el plano 2D. Debido a que la cantidad de escalado está definida por un vector, puede cambiar el tamaño de las dimensiones horizontal y vertical a diferentes escalas.

```
scale(sx) scale(sx, sy)
```

```
1 .circulo {  
2     transform: scale(2.5);  
3 }  
4  
5 .cuadrado {  
6     transform: scale(2.5,4); /*x,y*/  
7 }
```

## translate

La función de CSS **translate()** recoloca un elemento en el eje horizontal y/o vertical.

`transform: translate(tx, ty);`

```
1 .circulo {  
2     transform: translate(50px, 100px); /*x,y*/  
3 }  
4  
5 .cuadrado {  
6     transform: translateX(100px); /*x*/  
7 }
```

## Transiciones

---

Las transiciones CSS son pequeños cambios en propiedades de la hoja de estilos desencadenados por acontecimientos generados por interacciones del usuario, como por ejemplo cuando el mouse pasa por encima de algo (:hover) el cambio se ve tosco. En una transición, estos cambios en las propiedades se producen de manera progresiva durante un intervalo de tiempo.

## Estructura

Transition: [property] [duration] [timing-function] [delay];

Transition: all 0.3s ease 0.5s;

**transition-property:** Especifica el nombre o nombres de las propiedades CSS a las que deberían aplicarse las transiciones. Sólo las propiedades que se enumeran aquí son animadas durante las transiciones; los cambios en el resto de las propiedades suceden de manera instantánea como siempre. En caso que sean varias propiedades se puede usar el valor all.

**transition-duration:** Especifica la duración en la que sucederán las transiciones.

**transition-timing-function:** Especifica la curva cúbica bézier que se usa para definir cómo se computan los valores intermedios para las propiedades.

**transition-delay:** Define el tiempo de espera entre el momento en que se cambia una propiedad y el inicio de la transición.

Algunas propiedades que pueden ser animadas: background-color, background-image, background-position, background-size, border-color , border-radius, color, font-size, opacity y entre otras.

```
1 .border {
2     width:500px;
3     height:300px;
4     background:#676470;
5     color:#fff;
6     transition:all 0.3s ease;
7 }
8
9 .border:hover{
10     box-shadow: inset 0 0 0 25px #53a7ea;
11 }
```

## Propiedad transition-timing-function

Valor	Descripción
<b>linear</b>	La animación tiene la misma velocidad de principio a fin.
<b>ease</b>	La animación tiene un comienzo lento, luego rápido, antes de que termine lentamente.
<b>ease-in</b>	La animación tiene un comienzo lento.
<b>ease-out</b>	La animación tiene un final lento.
<b>ease-in-out</b>	La animación tiene un comienzo lento y un final lento.
<b>step-start</b>	Equivalente a pasos (1, inicio)
<b>step-end</b>	Equivalente a pasos (1, fin)
<b>steps(int,start   end)</b>	Especifica una función paso a paso, con dos parámetros. El primer parámetro especifica el número de intervalos en la función. Debe ser un número entero positivo (mayor que 0). El segundo parámetro, que es opcional, es el valor "inicio" o "final", y especifica el punto en el que se produce el cambio de valores dentro del intervalo. Si se omite el segundo parámetro, se le asigna el valor "fin"
<b>cubic-bezier(n,n,n,n)</b>	Defina sus propios valores en la función cubic-bezier Los valores posibles son valores numéricos de 0 a 1

