



TEXT CLASS REVIEW

TEMAS A TRATAR EN LA CUE

- Transiciones en CSS.
- Crear animaciones en CSS.

TRANSICIONES EN CSS

Corresponden a un tipo de animación que permite cambiar los valores de propiedades durante un tiempo determinado. Para poder utilizarlo, solo se necesita especificar la propiedad CSS que se desea afectar, y la duración del efecto. Nótese que se puede aplicar más de un efecto de transición a la vez.

Existen algunos componentes que permiten controlarlas:

- **transition-property:** esta propiedad especifica el nombre de la propiedad CSS, sobre la que tendrá efecto la transición.
- **transition-duration:** determina cuánto tiempo tardará la transición en completarse, en segundos (s) o milisegundos (ms).
- **transition-timing-function:** especifica la curva cúbica Bézier de velocidad del efecto de transición.
- **transition-delay:** define cuándo comenzará el efecto de transición. Su valor está definido en segundos (s), o milisegundos (ms).

1. Curva de Bézier

Se refiere a un sistema desarrollado en los años 60 para el trazado de dibujos técnicos, diseño aeronáutico y de automóviles. Se utilizan ampliamente para el modelado de curvas suaves, como, por ejemplo, las que se utilizan para definir la velocidad de una transición CSS; en este caso, solo basta con agregar cuatro parámetros a la función *cubic-bezier*: x1, x2, y1 e y2. Existen sitios web dedicados al modelado de este tipo de curvas, como [cubic-bezier](#) y [easings](#).

2. Transition

Permite agregar todas las propiedades vistas anteriormente en una sola línea de código, siguiendo el siguiente orden:

```
1 transition: transition-property transition-duration transition-  
2 timing-function transition-delay
```

CREAR ANIMACIONES EN CSS

CSS permite la animación del contenido HTML de nuestras páginas web, sin la necesidad de utilizar JavaScript.

- **@Keyframes**

Las animaciones en CSS permiten cambiar un elemento gradualmente desde un estilo a otro. Se pueden realizar cuantos cambios se quieran, y las veces que se quieran. Para utilizarlas, se deben definir **@keyframes**, que consisten en c; éstos necesitan tener asociados un nombre, el cual se debe referenciar dentro de la regla de estilo a la que aplicaremos la animación.

```
1 @keyframes opacidad {  
2     0% {  
3         /* primer estilo */  
4     }  
5     50 % {  
6         /* segundo estilo */  
7     }  
8     100 % {  
9         /* tercer estilo */  
10    }  
11 }
```

En el interior de las declaraciones de **@keyframes**, se deben definir las acciones o cambios que ocurrirán en el elemento afectado por tal animación.



- **ANIMATION**

Nos permite definir características de la animación que realizaremos, tales como: el nombre del `@keyframes` asociado, la duración de la animación, entre otras.

Propiedad
animation-name
animation-duration
animation-delay
animation-iteration-count
animation-direction
animation-timing-function
animation-fill-mode
keyframes

- **animation-name:** en esta propiedad se debe especificar el nombre de la animación `@keyframes` relacionada.
- **animation-duration:** establece el tiempo que tomará la animación en completar un ciclo. Se pueden utilizar segundos (s), y milisegundos (ms).
- **animation-timing-function:** esta propiedad define la curva de velocidad de la animación. Permite utilizar una función `cubic-bezier()`, o keywords como: `ease linear`, `ease-in`, `ease-out` y `ease-in-out`.
- **animation-delay:** especifica el retraso del inicio de la animación.
- **animation-iteration-count:** determina la cantidad de veces que se reproducirá la animación. Acepta números y la keyword *infinite*, que permite repetir una animación infinitamente.



- **animation-direction:** indica al navegador si una animación se debe reproducir hacia adelante, hacia atrás, o en ciclos alternativos. Aquí se puede definir que una animación ocurra de izquierda a derecha (**normal**), de derecha a izquierda (**reverse**), de izquierda a derecha y luego de derecha a izquierda (**alternate**), o primero de derecha a izquierda y luego de izquierda a derecha (**alternate-reverse**).
- **animation-fill-mode:** por defecto, las animaciones CSS no afectan un elemento antes de que el primer **@keyframes** es reproducido, o después de que el último lo sea, este comportamiento se puede modificar con **animation-fill-mode**. Esta propiedad especifica el estilo de un elemento determinado cuando no se están reproduciendo animaciones. Puede indicarse que una animación mantenga los valores de estilo especificados por el último **keyframe (forwards)**, que tome los valores de estilo especificados por el primer **keyframe (backwards)**, o ambos (**both**).

Todas las propiedades definidas anteriormente, se pueden colapsar en una sola: *animation*, la cual permite establecer todos los valores necesarios en una línea de código.

```
1 animation: animation-name animation-duration animation-timing-  
2 function animation-delay animation-iteration-count animation-  
3 direction
```