**COUNTER CSS**

* **COUNTER-RESET:** crea o reinicia un contador.
* **COUNTER-INCREMENT:** Incrementa un valor del contador.
* **CONTENT**: Inserta el contenido generado (debe usarse con un pseudoelemento).
* **COUNTER():** Función que agrega el valor de un contador a un elemento.

Básicamente el uso de la directiva **@KEYFRAMES** te permite definir el comportamiento de tu animación punto por punto, y cualquier elemento puede usar esta animación por medio de la regla **animation-name.**

Dentro de **@KEYFRAMES** especificamos cada punto de nuestra animación por medio de porcentajes.

* Darle nombre a nuestro keyframe (con **ANIMATION-NAME**: blink;)
* Darle tiempo a nuestro keyframe (con **ANIMATION-DURATION**: 2s;)
* Decirle al keyframe cuantas veces se tiene que repetir (con **ANIMATION-ITERATION-COUNT**: infinite;)
* **ANIMATION TIMING FUNCTION:** Básicamente es la aceleración con la cual correrá nuestra animación.
* **ANIMATION DELAY:** Es el tiempo que nuestra animación tardará en empezar.

**PROPIEDAD ANIMATION-DIRECTION**

Mediante esta propiedad indicamos cómo debe ejecutarse una animación: hacia delante (del principio al final), hacia atrás (del final al principio), una vez hacia delante y otra hacia detrás, etc. La animación habrá sido definida en una regla @keyframes y va a ser aplicada a un elemento seleccionado mediante un selector CSS. El valor por defecto para esta propiedad es **normal** (implica que la animación se ejecutará hacia delante). Sus valores posibles son cualquiera de estas palabras clave:

* **normal:** la animación se ejecutará hacia delante. Si se repite, cuando vuelve a empezar parte de la situación inicial.
* **reverse:** la animación se ejecutará hacia detrás. Si se repite, cuando vuelve a empezar parte de la situación inicial.
* **alternate:** la animación se ejecutará una vez en un sentido y otra vez en otro, comenzando hacia delante, luego hacia detrás y así sucesivamente el número de repeticiones especificado.
* **alternate-reverse:** la animación se ejecutará una vez en un sentido y otra vez en otro, comenzando hacia detrás, luego hacia delante y así sucesivamente el número de repeticiones especificado.

Si se desean aplicar varias animaciones, se especificarán sus direcciones separadas por comas, siendo cada valor correspondiente a la animación cuyo nombre figura en el mismo orden en la propiedad animation-name.

Esta propiedad puede no ser reconocida por los navegadores antiguos o requerir del uso de prefijos específicos o no ser reconocida por algunos navegadores actuales. Consulta en Mozilla Developer Network para conocer si debes aplicar prefijos o si la propiedad será reconocida por un navegador concreto.

**PROPIEDAD ANIMATION-PLAY-STATE**

Esta propiedad permite detener una animación que se está ejecutando (ponerla en pausa). Su valor por defecto es running y sus dos valores posibles son:

* **running:** la animación está en ejecución
* **paused:** la animación está en pausa

Esta propiedad puede no ser reconocida por los navegadores antiguos o requerir del uso de prefijos específicos para algunos navegadores actuales. Consulta en Mozilla Developer Network para conocer si debes aplicar prefijos.

Ejemplo: .pubMov:hover{ animation-play-state:paused; } con esta regla damos lugar a que la animación se detenga si el usuario pone el puntero del ratón sobre el elemento animado.

**PROPIEDAD ANIMATION-FILL-MODE**

Esta propiedad permite detener definir cómo debe comenzar y cómo debe quedar un elemento que tiene una animación. Su valor por defecto es none (implica que el elemento comenzará y quedará en el estado que tenía antes de que comenzara la animación) y sus valores posibles son:

* **none:** el elemento comenzará y quedará en el estado previo a la animación.
* **forwards:** el elemento quedará en el estado final de la animación.
* **backwards:** el elemento se pondrá en el estado indicado para el comienzo de la animación inmediatamente y esperará en ese estado hasta que se cumpla el tiempo indicado en animation-delay. Una vez se cumpla ese tiempo la animación continuará la ejecución desde ese estado inicial.
* **both:** combinación de las dos opciones anteriores.

Esta propiedad puede no ser reconocida por los navegadores antiguos o requerir del uso de prefijos específicos para algunos navegadores actuales.

* **Composite**: Ordena las partes de la página. Propiedades como opacity y transform.
* **Paint**: Rellena píxeles. Implica colores, imágenes, textos, sombras…
* **Layout**: Diseño de la página. Ancho, margin, padding, border…