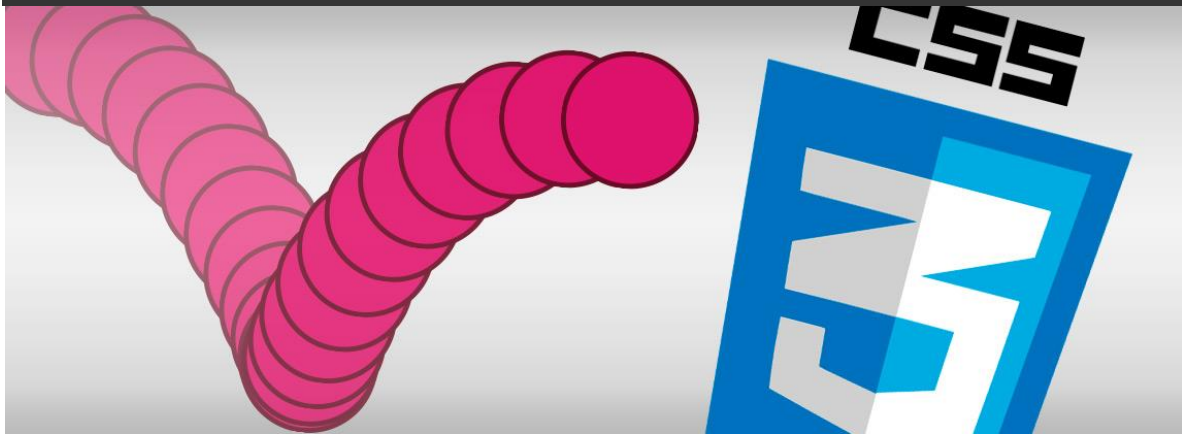


# Animaciones en CSS3



Las animaciones en CSS3 nos suponen un nuevo aspecto a tener en cuenta a la hora de crear nuestro diseño Web. Una animación es un elemento sujeto a una transformación, durante un tiempo especificado. Las transformaciones **ya las vimos**, y las podemos aplicar a un objeto para que se mueva todo el tiempo. A parte de animar cualquier otro atributo CSS, como los colores, bordes... Aunque recomiendo usarlo con cierto control, ya que si se mueven demasiados elementos en la página, pueden despistar a nuestro usuario visitante.

## La sintaxis de las animaciones

Las animaciones se crean a partir del lenguaje CSS, con la siguiente instrucción:

```
animation: nombre duración función-de-tiempo retraso contador dirección estado;
```

Describamos los diferentes atributos, antes de ver el ejemplo:

- *nombre*: podemos asignar lo que queramos. Este nombre se puede reutilizar para más objetos que tengan el mismo tipo de movimiento.

- *duración*: el tiempo que dura una repetición de la animación, en segundos. Podemos usar decimales, con el punto.
- *función-de-tiempo*: es la aceleración que tendrá la animación. También vimos los principales tipos en las **transiciones**.
- *retraso*: el tiempo que pasará desde que se termine de cargar el contenido de la página y se inicie la animación.
- *contador*: las veces que se repite. Podemos usar *infinite* para que no termine.
- *dirección*: sería si la animación se mueve hacia delante o hacia atrás (normal o *reverse*).
- *estado*: si la animación se está reproduciendo o está pausada.

Ahora ya conocemos cómo podemos invocar una animación. Solo nos falta especificar qué debe hacer y cómo cambiará el objeto animado. Para hacerlo, necesitamos otra instrucción CSS, los *@keyframes*:

```
@keyframes nombre
{
0%   {left:-10%; top:0px;}
100% {left:200%; top:80px;}
}
```

Lo que podemos ver, es que llamamos a la función *@keyframes*, con el nombre que queramos especificar (debe coincidir con el que hemos puesto al *animate*). Después podemos especificar cómo estará el objeto a los diferentes pasos de la animación. En el ejemplo, hemos animado el *left* y el *top*. Y podemos ver que al principio y al final, no tienen el mismo valor. La animación será una transición suavizada entre el 0% y el 100%. Este ejemplo quedaría de la siguiente forma:

Así quedaría el código del ejemplo:

```
<div class="example">
  <div class="object"></div>
</div>
```

```

/* CSS */
.example {
    background-color:#fefefe;
    border: 1px solid #ccc;
    padding:20px;
    position:relative;
    height: 100px;
}

.object {
    position:absolute;
    left:0;
    height: 8px;
    padding: 25px;
    width: 8px;
    background-color: #caebb7;
    border:2px solid #5b973b;
    animation: mover 4s ease 0 infinite normal running;
    -o-animation: mover 4s ease 0 infinite normal running; /* Opera */
    -ms-animation: mover 4s ease 0 infinite normal running; /* IE */
    -moz-animation: mover 4s ease 0 infinite normal running; /* Firefox */
    -webkit-animation: mover 4s ease 0 infinite normal running; /* Safari
y Chrome */
}

@keyframes mover
{
    0%   { left:0%; }
    50% { left:90%; transform:rotate(1080deg); }
    100% { left:0%; }
}

/* Para mayor compatibilidad entre navegadores, tendríamos
que repetir el proceso con los prefijos correspondiente. El siguiente,
sería para Safari y Chrome, pero tendríamos que añadir los demás. */
@-webkit-keyframes mover
{
    0%   {left:0%;}
    50% {left:90%; -webkit-transform:rotate(1080deg);}

```

```
100% {left:0%;}  
}
```

En este ejemplo hemos animado el atributo `left`, y a su vez la rotación. Y especificando que el principio (0%) y el fin (100%) de la animación, tengan los mismos valores, para que así parezca que la animación aparente más fluidez. Se pueden especificar todos los atributos que queramos. Pero cuanto más compleja sea la animación, más lento irá el navegador del usuario. Hay que tenerlo en cuenta si queremos que llegue a muchos usuarios visualizándose de forma correcta. Con pocas animaciones, prácticamente no se notará.