

Fiche les animations en CSS3

La propriété **animation** permet d'animer des éléments du html. Une animation permet à un élément de passer graduellement d'un style à un autre (idée d'interpolation). Le nombre de propriétés css qu'il est possible de modifier n'est pas limité. Le style va changer progressivement en fonction des valeurs indiquées.

Pour faire fonctionner les animations en CSS3, il est nécessaire d'utiliser une **commande @keyframes** qui permettra de définir les différentes étapes de cette animation.

1. L'animation

Les animations se décomposent en plusieurs propriétés :

animation-name : le nom de la commande keyframes à utiliser.

Déclaration avec les prefixes propriétaires :

```
-webkit-animation-name: nomAnimation;  
-moz-animation-name: nomAnimation;  
-ms-animation-name: nomAnimation;  
-o-animation-name: nomAnimation;  
animation-name: nomAnimation;
```

animation-duration : le temps total de l'animation en secondes (s) ou en millisecondes (ms)

animation-timing-function : la courbe de vitesse (rythme avec accélération et décélération) de l'animation. Valeurs : linear, ease, ease-in, ease-out, ease-in-out...

animation-iteration-count : le nombre de répétition de l'animation (1,2,etc.).

La valeur infinie permet de jouer une animation en continu.

animation-direction : l'animation doit être inversée et/ou alternée. Permet de jouer une animation en sens inverse (en fonction du cycle).

- normal : l'animation va de l'étape 1 à n.
- reverse : l'animation va de l'étape n à 1. (sens contraire)
- alternate : l'animation va de l'étape 1 à n les cycles impairs (1,3,5...) et de l'étape n à 1 les cycles pairs (2,4,6...)
- alternate-reverse : l'animation va de l'étape 1 à n les cycles pairs et de l'étape n à 1 les cycles impairs.

animation-play-state : mise en pause de l'animation. En cours d'étude de suppression par le W3C – valeurs playing ou paused.

animation-delay : le délai avant que l'animation ne démarre en secondes (s) ou millisecondes (ms)

animation-fill-mode : permet de préciser si l'on veut, lorsque l'animation se termine, la maintenir dans son état de fin (à 100%) ou revenir à l'état de départ (0%).

- **backwards** : les éléments sont dans l'état de la première étape de l'animation avant le début de celle-ci.
- **forwards** : les éléments sont dans l'état de la dernière étape de l'animation après la fin de celle-ci.

Il existe aussi la propriété raccourcie : animation qui permet de déclarer en une seule fois toutes les propriétés exceptées `animation-play-state` et `animation-fill-mode`.

Syntaxe de l'animation :

```
div {  
    animation: nomAnimation 3s infinite linear alternate ;  
}
```

`nomAnimation` est le nom de la commande `@keyframes` qui permettra de spécifier les étapes de l'animation

2. La commande css @keyframes

La commande css `@keyframes` permet de définir

- Le déroulement de l'animation : les différentes étapes
- Ce qui va être modifié : les propriétés css qui évolueront durant ces étapes.

Les étapes de l'animation sont indiquées soit avec des pourcentages soit avec les mots clés "from" et "to" (équivalent à 0% et 100%). On passe ainsi d'un état initial qui est 0% à un état final qui est 100% avec, si l'on utilise les pourcentages, la possibilité de créer différentes étapes intermédiaires.

A chaque étape, on redéfinit les valeurs des styles css qui doivent être animés, la transition entre les valeurs étant réalisée automatiquement.

Syntaxe de la commande @keyframes :

```
@keyframes mvt {  
    0% {transform: scale(1)}  
    100% {transform: scale(3)}  
}
```

Exemple

Une bille orange a été construite en css à l'aide d'un carré de 50px de côtés aux bords arrondis. Un texte « := » a été ajouté en html.

Cette bille est animée. L'animation appelée rouleBille dure 5 secondes, se déroule de façon linéaire, sans accélération ni décélération. Lorsque l'animation arrive au bout des étapes, elle roule en sens inverse pour revenir au point de départ puis redémarre, sans interruption.

Code html :

```
<div id="bille">:=</div>
```

Code css :

```
div#bille {
    position:relative; /* pour faire fonctionner le déplacement left */
    width :50px;
    height :50px;
    line-height:50px;
    text-align:center;
    border-radius :25px;
    background-color: orange ;
    color:#fff;
    font-weight:bold;

    animation: rouleBille linear 2s infinite alternate;
}

@keyframes rouleBille {
    0% {transform: rotate(90deg); left:0px;}
    100% {transform: rotate(360deg); left:200px;}
}
```