

## CSS Animations



# ADRAR DIGIT@L ACADEMY

PÔLE NUMERIQUE DU CENTRE DE FORMATION ADRAR

- > SUPPORT, ADMINISTRATION SYSTEMES & RESEAUX
- > DEVELOPPEMENT D'APPLICATIONS WEB & MOBILES
- > TRANSFORMATION NUMERIQUE DES ENTREPRISES

<http://www.adrar-numerique.com>

## CSS Animations: Introduction

Avant l'arrivée de CSS 3, nous utilisions en premier, les « GIF », les images animées. Puis est arrivé le « Flash » qui a permis de faire de nombreuses animations bien plus poussées. Mais l'arrivée des smartphones a poussée cette technologie sur le banc de touche, elle est maintenant dépréciée.

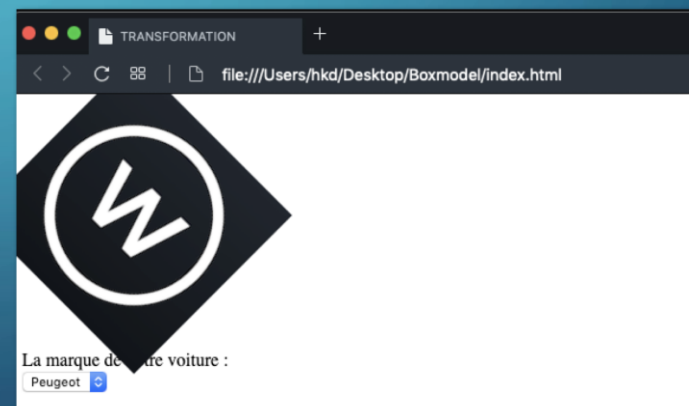
Aujourd'hui, il y a beaucoup plus de choix pour faire de l'animation WEB (JavaScript, SVG...), mais il est aussi possible de le faire nativement avec seulement du HTML/CSS !

## CSS Animations: Rotation 2D

Les animations peuvent dans leur ensemble s'appliquer sur n'importe quel élément HTML de notre page.

Voyons ensemble l'exemple de la rotation en 2D sur une simple image. Il suffit d'utiliser la propriété « transform » avec la valeur « rotate » et d'ajouter un paramètre pour l'angle de rotation:

```
#photo
{
  -webkit-transform: rotate(45deg); /* Safari */
  transform: rotate(45deg); /* Tous les autres navigateurs */
}
```



Nous voyons ici qu'il faut une ligne supplémentaire pour certains navigateurs.

## CSS Animations: Rotation 2D

Il est possible d'utiliser deux unités de mesure différentes pour gérer la rotation, à savoir:

- Les degrés => 45deg, -75deg,...
- Les tours => 0.25turn, -1turn,...

Par exemple,  $0.25\text{turn} = \frac{1}{4} \text{ de tour} = 90\text{deg}$

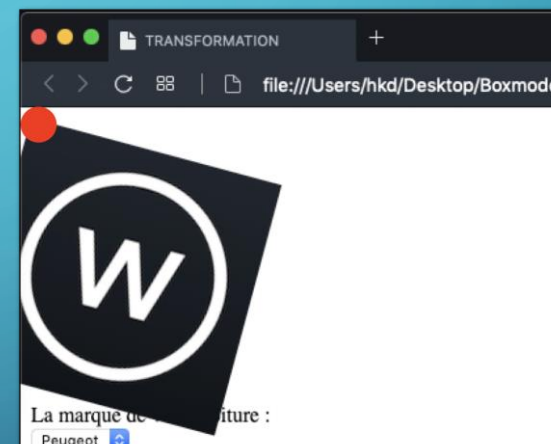
Pour info, par défaut la rotation s'effectue dans le sens des aiguilles d'une montre.  
Pour inverser ce sens, il faut mettre la valeur en négatif.



## CSS Animations: Rotation 2D

Lorsqu'on applique une transformation à un élément HTML, elle s'applique par défaut à partir du centre de l'élément. Pour changer cette origine, on utilise la propriété « transform-origin » comme suit:

```
#photo  
{  
  transform-origin: 0 0; /* transformation coin haut gauche */  
  transform: rotate(15deg);  
}
```

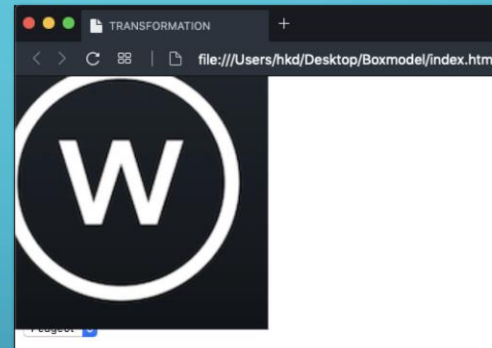


La propriété prend deux valeurs, celui des abscisses (x) et l'axe des ordonnées (y). Cette valeur peut être exprimée en pixel (px) ou en pourcentage (%). Nous pouvons aussi utiliser des mots-clés.

## CSS Animations: Agrandissement/Rétrécissement 2D

Avec la propriété « transform », nous pouvons aussi utiliser la valeur « scale » qui permet un agrandissement ou un rétrécissement d'un élément HTML:

```
#photo  
{  
  transform: scale(1.5);  
}
```

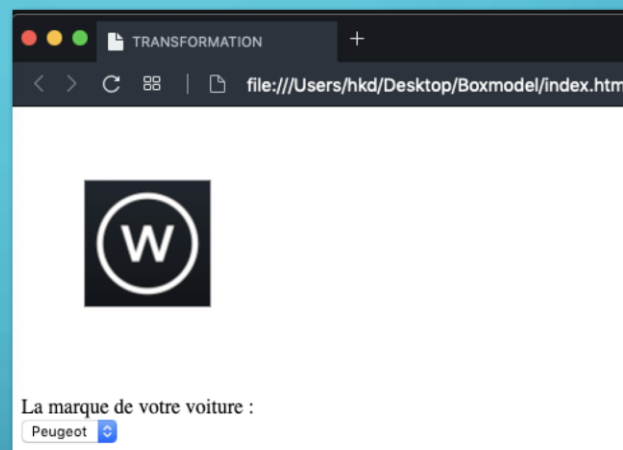


Le paramètre « 1.5 » signifie une taille relative de 1 fois et demi plus grande que la valeur de base. Une valeur de « 1 » signifiera, par exemple, que la taille de changera pas.

## CSS Animations: Agrandissement/Rétrécissement 2D

Si la valeur est inférieure à « 1 », l'élément se retrouvera rétréci :

```
#photo  
{  
  transform: scale(0.5);  
}
```

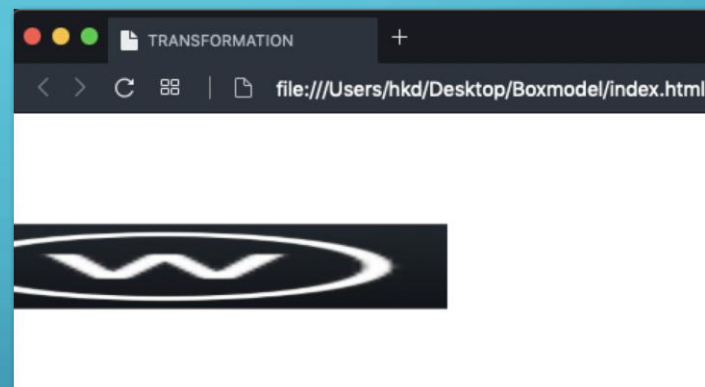


Ici, notre image a été réduite de moitié (1/2 fois la taille par défaut)

## CSS Animations: Agrandissement/Rétrécissement 2D

Nous pouvons aussi passer deux valeurs en paramètre de notre « scale », lui permettant de modifier une image sur deux axes différemment (x, y):

```
#photo  
{  
  transform: scale(2, 0.3);  
}
```



Ici, notre image a été agrandi 2 fois sur l'axe horizontal (x) et réduite à 0,3 sur l'axe vertical (y)



## CSS Animations: Agrandissement/Rétrécissement 2D

Nous pouvons enfin utiliser les valeurs « `scaleX` » et « `scaleY` » pour transformer notre élément que sur un seul plan:

```
#photo  
{  
  transform: scaleX(6);  
}
```

```
#photo  
{  
  transform: scaleY(6);  
}
```

## CSS Animations: Mise en pratique (20 min)

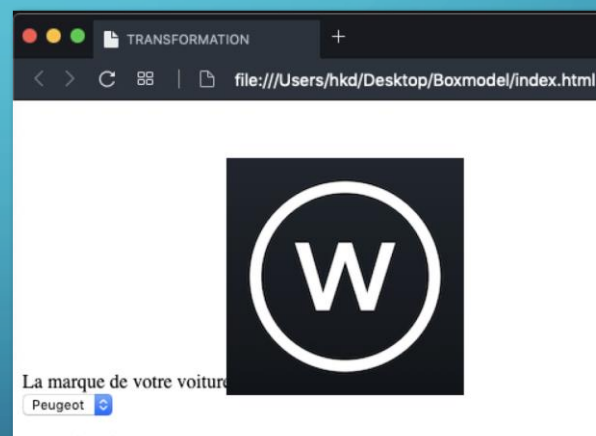
- Dans un nouveau dossier, créez un fichier HTML avec le HTML minimum et un fichier CSS. Liez-les.
- Ajoutez dans ce dossier, une image de votre choix (si pas d'idée, lorem ipsum)
- Modifiez votre CSS pour obtenir:
  - Un agrandissement de l'image lors du passage de la souris (x2)
  - Une rotation 1/2 tour lors du clic sur l'image, dans le sens horaire

Note: Vous aurez besoin d'utiliser des pseudo-classes

## CSS Animations: Translation 2D

En utilisant la valeur « translate » de la propriété « transform », nous pouvons déplacer un élément HTML dans notre page. Le premier paramètre sera à nouveau pour l'axe X et le second pour l'axe Y:

```
#photo  
{  
  transform: translate(170px, 40px);  
}
```



Nous pouvons aussi utiliser des valeurs négatives.

## CSS Animations: Translation 2D

Comme pour le « scale », nous pouvons utiliser les valeurs « translateX » et « translateY » pour transformer notre élément que sur un seul plan:

```
#photo  
{  
  transform: translateX(170px);  
}
```

```
#photo  
{  
  transform: translateY(170px);  
}
```



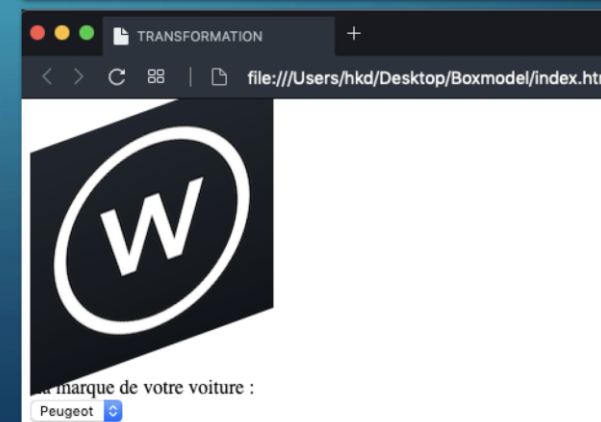
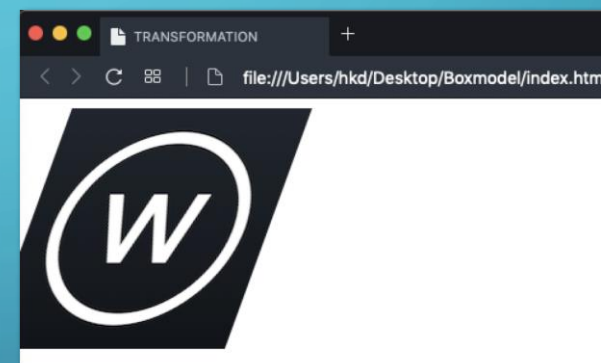
## CSS Animations: Transformation oblique 2D

Nous restons sur un fonctionnement identique avec les valeurs « skewX » et « skewY » de notre propriété « transform », nous pouvons étirer les côtés d'un élément HTML:

```
#photo  
{  
  transform: skewX(-20deg);  
}
```

Note: Nous pouvons aussi utiliser la valeur « skew » seule en lui spécifiant deux paramètres

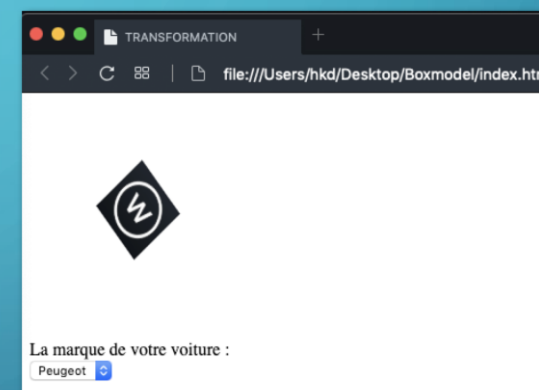
```
#photo  
{  
  transform: skew(-20deg);  
}
```



## CSS Animations: Super propriété 2D

Enfin, nous pouvons combiner les transformations dans une seule super-propriété, comme avec « background », pour cumuler les effets:

```
#photo  
{  
  transform: scale(0.3) rotate(45deg) skewY(10deg);  
}
```



Attention: L'ordre est important ! Les transformations seront effectuées les unes après les autres.

## CSS Animations: Mise en pratique (30 min)

- Réutilisez les fichiers de l'exercice précédent.
- Dans votre HTML, ajoutez une seconde image ainsi qu'un paragraphe contenant du texte
- Modifier votre CSS pour obtenir sur la seconde image:
  - Une transformation oblique de 40 degrés sur l'axe X, lors du passage de la souris
  - Une translation de 200px vers la droite, sur l'axe X, lors du clic de la souris
- Modifier votre CSS pour obtenir sur le paragraphe:
  - Un rétrécissement de moitié
  - Une translation de 20 px vers la droite
  - Une rotation de 35 degrés dans le sens horaire
  - Une transformation oblique de 10 degrés sur l'axe Y