



Web Statique Transformation

Les transformations CSS

A partir de la version 3 de CSS il est devenu possible de transformer les objets intégrés en 2D (et même en 3D) sans vraiment modifier leurs propriétés (largeur, hauteur...). Cette technique a permis aux designers de réussir des effets époustouflants avec un minimum de code.

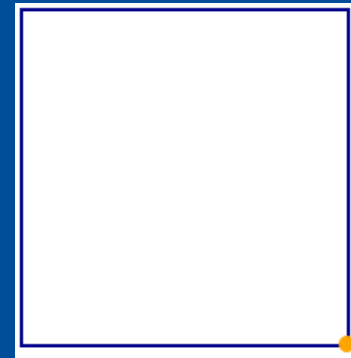
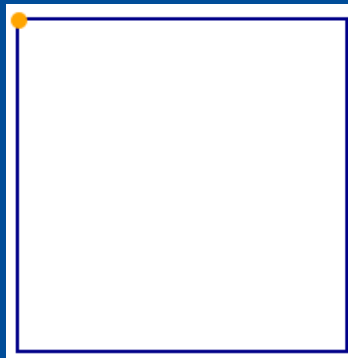
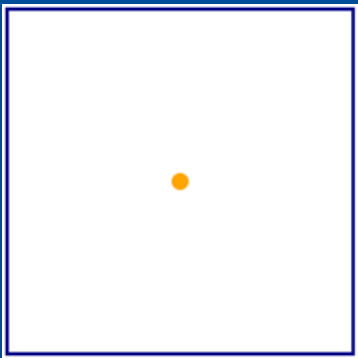
Propriété transform-origin

La propriété *transform-origin* permet de modifier le point d'origine d'un objet HTML. Il accepte comme valeur le X et le Y de l'origine souhaité séparés par un espace. Ces deux informations peuvent être spécifiées en pourcentage, en pixel, ou par les mots clé left, right, top, bottom et center.

point d'origine

transform-origin:0 0;

transform-origin:100% 100%;

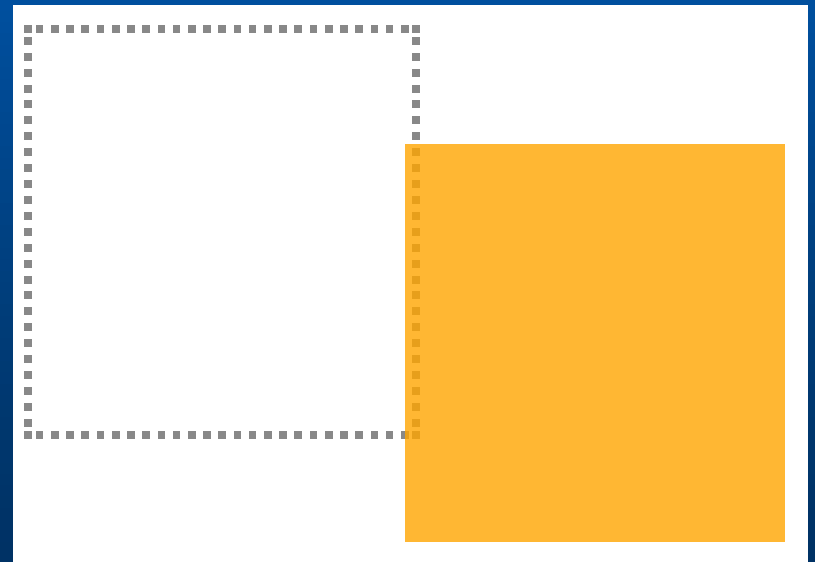


Fonction translate()

La fonction translate() ,fonction de transformation 2D, permet la translation de l'objet HTML d'un emplacement initial (où il est normalement défini) à un autre emplacement. Les coordonnées du nouvel emplacement sont renseignées entre les parenthèses de translate() et ils sont exprimés en pixel. L'origine des coordonnées, comme on l'a vu précédemment, est le point haut à gauche.

Exemple:

```
div{  
  width:100px;  
  height:100px;  
  background-color:orange;  
  transform:translate(100px,30px);  
}
```



Voir exécution Ex1

Fonction translate()

Le premier paramètre de la fonction translate() représente la translation horizontale, et le deuxième la translation verticale. Notez que les sens positifs sont vers la gauche et vers le bas. Si vous renseignez des valeurs négatives à la fonctions alors la translation sera appliquée dans le sens opposé.

Il existe aussi les fonctions translateX() qui réalise une translation horizontale et translateY() qui réalise une translation verticale. Dans ce cas, nous spécifions une seule valeur dans les parenthèses.

Fonction `scale()`

La fonction `scale()` permet de redimensionner l'objet HTML (avec son contenu). Les parenthèses de la fonction peuvent accueillir une seule valeur ou deux valeurs séparées par une virgule. Ces valeurs désignent le facteur d'agrandissement. Ils n'ont pas d'unité et peuvent contenir de virgule. S'ils sont supérieur à 1 alors l'objet est agrandi, sinon alors il est rétréci.

Si une seule valeur est renseignée dans les parenthèses, alors le même facteur d'agrandissement agit sur la largeur et la hauteur de l'objet.

Exemple:

```
div{  
  width:100px;  
  height:100px;  
  background-color:orange;  
  transform-origin:0 0;  
  transform:scale(1.5);  
}
```



Voir
Exécution
Ex2

```
div{  
  width:100px;  
  height:100px;  
  background-color:orange;  
  transform-origin:0 0;  
  /* transform:scale(1.5); */  
  transform:scale(2,0.8);  
}
```



Fonction scale()

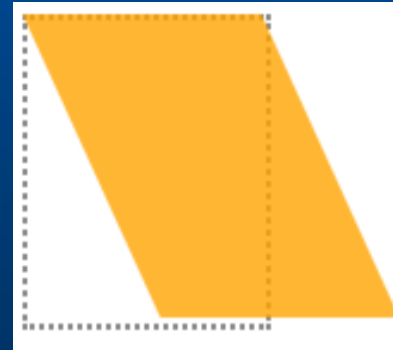
Il existe également les fonctions scaleX() et scaleY() qui appliquent respectivement un facteur d'agrandissement uniquement sur la largeur ou la hauteur.

Fonctions *skewX()* et *skewY()*

Les fonctions *skewX()* et *skewY()* permettent d'incliner un objet respectivement horizontalement et verticalement par rapport à son origine. La valeur renseignée dans les parenthèses désigne l'angle d'inclinaison et elle est exprimée en degré (deg).

Exemple d'inclinaison horizontale:

```
div{  
  width:100px;  
  height:100px;  
  background-color:orange;  
  transform-origin:0 0;  
  transform:skewX(30deg);  
}
```

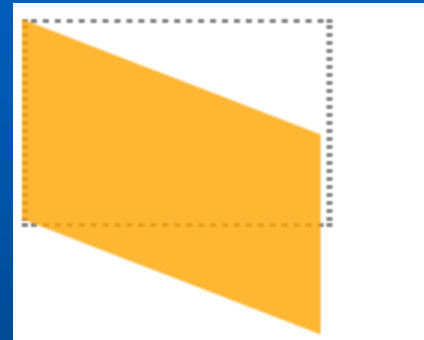


Voir exécution Ex3

Fonctions *skewX()* et *skewY()*

Exemple d'inclinaison verticale:

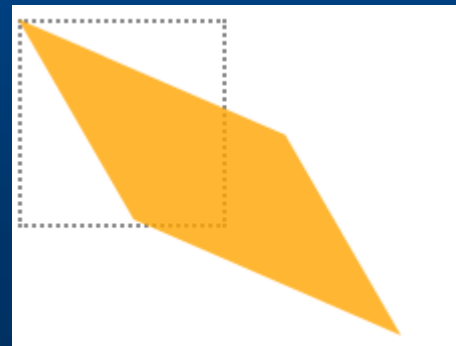
```
div{  
  width:100px;  
  height:100px;  
  background-color:orange;  
  transform-origin:0 0;  
  transform:skewY(30deg);  
}
```



Voir exécution Ex4

Exemple d'inclinaison horizontale et verticale:

```
div{  
  width:100px;  
  height:100px;  
  background-color:orange;  
  transform-origin:0 0;  
  transform:skewX(30deg) skewY(30deg);  
}
```



*Voir
exécution
Ex5*

Fonction rotate()

Comme son nom l'indique, la fonction rotate() applique une rotation à l'objet autour de l'axe traversant son origine. Les parenthèses contiennent l'angle de la rotation exprimée en degré.

Selon l'axe de rotation on distingue trois variantes de la fonction rotate():

rotate() ou rotateZ(): la rotation est appliquée par rapport à l'axe virtuel perpendiculaire à l'écran (le fameux axe des Z de z-index) qui traverse le point d'origine de l'objet. La rotation s'applique donc sur le même plan que l'écran.

rotateX(): la rotation est appliquée par rapport à l'axe horizontale traversant l'origine de l'objet (l'axe des X). La rotation aura donc un effet 3D

rotateY(): la rotation est appliquée par rapport à l'axe vertical traversant l'origine de l'objet (l'axe des Y). La rotation aura aussi un effet 3D

Fonction rotate()

Exemple d'une rotation sur l'axe des Z:

```
div{  
  width:100px;  
  height:100px;  
  background-color:orange;  
  transform-origin:50% 50%;  
  transform:rotate(30deg);  
}
```



*Voir
exécution
Ex6*

Exemple d'une rotation sur l'axe des X:

```
div{  
  width:100px;  
  height:100px;  
  background-color:orange;  
  transform-origin:50% 50%;  
  transform:rotateX(50deg);  
}
```



*Voir
exécution
Ex7*

Fonction perspective()

La fonction `perspective()`, fonction 3D, permet de prendre en compte les changements des dimensions en fonction de la profondeur (ou perspectives). Par exemple, si la rotation est appliquée par rapport à l'axe des X, le côté le plus loin (celui qui, soi-disant, entre dans l'écran) sera plus petit que celui qui en sort.

La fonction `perspective()` reçoit une valeur exprimée en pixel qui désigne la distance (virtuelle) qui sépare l'observateur de l'axe de rotation (qui traverse l'origine de l'objet). Plus cette distance est petite, plus l'effet de perspective est accentué et inversement.

Reprenons l'exemple dernier, celui de la rotation par rapport à l'axe des X et appliquons lui les perspectives:

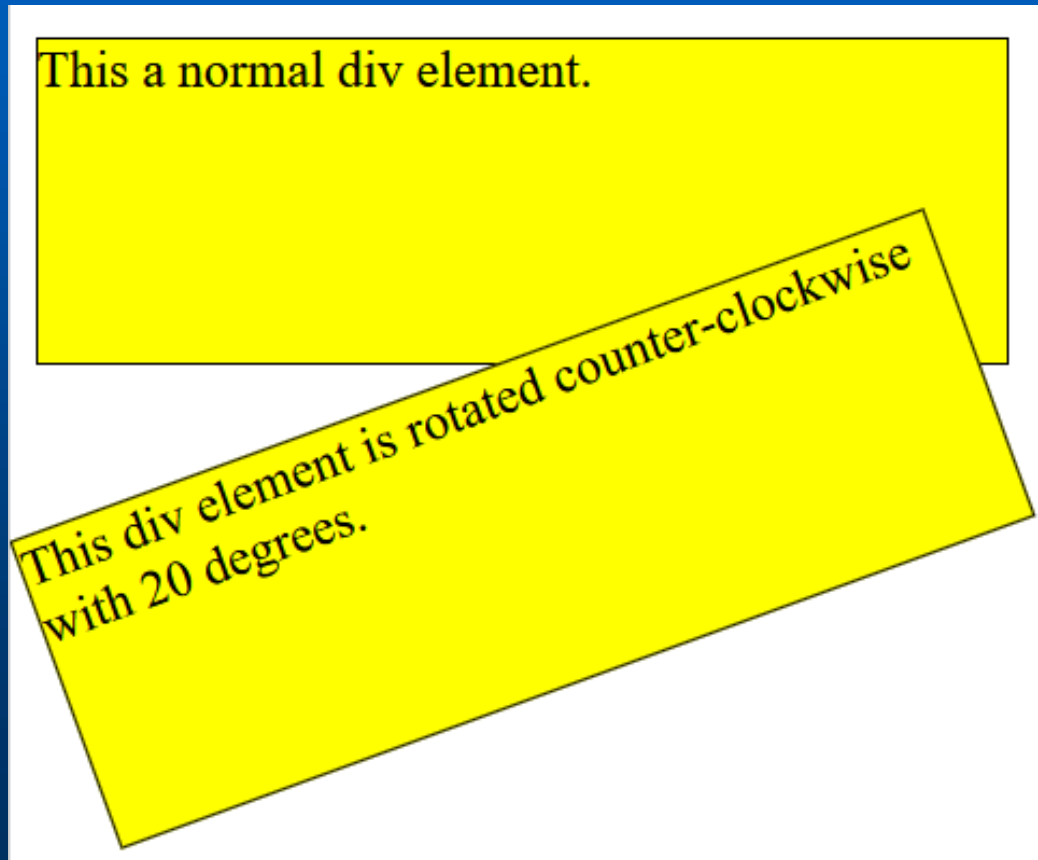
```
div{  
  width:100px;  
  height:100px;  
  background-color:orange;  
  transform-origin:50% 50%;  
  transform:perspective(150px) rotateX(50deg);  
}
```



Voir exécution Ex 8

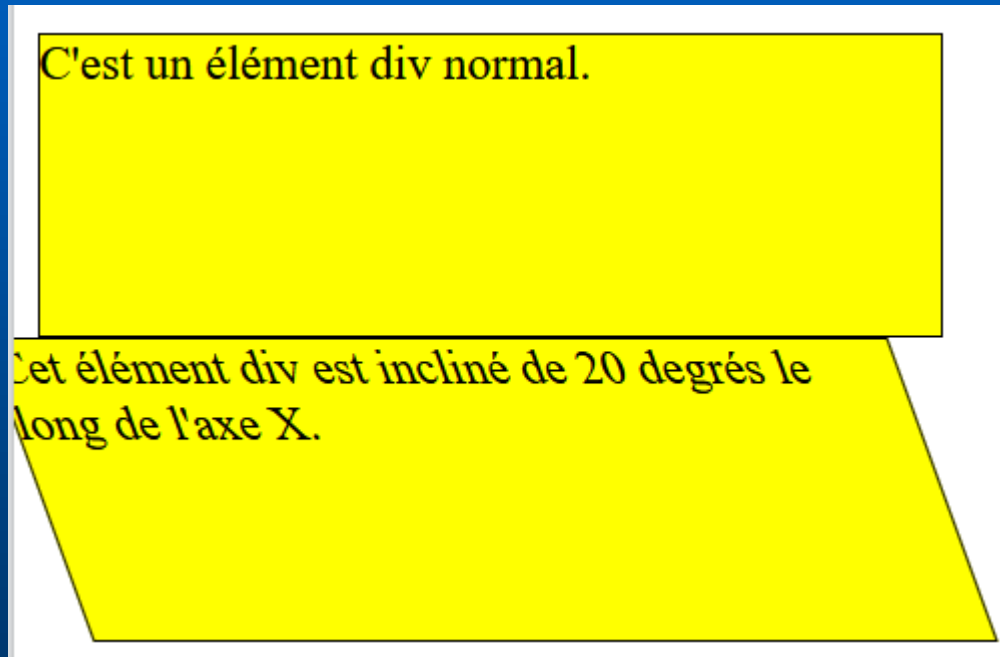
Exercice

Ecrire le code HTML utilisant une feuille de styles pour créer la page suivante :



Exercice

Ecrire le code HTML utilisant une feuille de styles pour créer la page suivante :



Exercice

Ecrire le code HTML utilisant une feuille de styles pour créer la page suivante :

