## CSS3 3D变换

首先推荐一个大佬博客文章：

<https://www.cnblogs.com/shenzikun1314/p/6390181.html>

↑ 总结非常全面。

1. CSS3 3D具有的属性和方法
2. 3D的属性比2D多

2D属性只有2种：

Transform:

Transform-origin:

3D 属性有6种：

Transform:

Transform-origin:

Transiform-style:（规定如何让子元素在3D空间内显示）

Perspective:（规定3D透视效果，父元素上定义，单位px）

Perspective-origin:（规定透视的视角位置）

Backface-visibility:定义在元素背对屏幕的时候是否可见。

1. 3D的方法种类与2D的方法相似，都具备位移，放大缩小，旋转。只不过并没有倾斜（特指skewZ()）而已。
   1. 矩阵方法,matrix();参数较多，矩阵理解超级难，不多研究。
   2. 各大方法中都有只针对X,Y,Z轴变换的方法。



1. perspective是对透视的定义，默认为50% 50%，如果没有开启透视，那么也就没有视距的概念，所有的东西都是一个2D的图形。因为没有了视距的概念，即使设置了translateZ()也看不出效果。



不用考虑当perspective为0或者不设置的时候，视角是在距离元素多远的地方。没有透视，显示的东西永远都是一个投影。

比如一个div的高度为200，宽度为100。他父元素的perspective为200px,

那么就相当于是，你站在距离div 200px的地方看他，在这个位置测出的高度正是高度为200，宽度为100。而不是这个元素本身宽高200 100，然后你在200px外看他，他就会比其他同样比例的东西（没有设置透视的）要小。

1. translateZ(); 向Z轴位移，事实上，这个属性必须与perspective属性并存，且两者都不能为默认值或者有一方不设置才能看出效果。

假设perspective的为200px; 你所看到的宽度和高度为200 100.

那么当translateZ()向200靠近的时候他才会变大，超过了200，那么他就跑到你的后脑勺后面，你就看不到他了。

Eg:

<!--正常放置，无变化-->

<div><img src="images/tianyi.jpg"></div>

<!--加入透视和Z轴方向位移-->

<div style = "perspective: 200px;"><img src="images/tianyi.jpg" style = "transform:translateZ(50px)"></div>



1. rotate3d(x,y,z,deg);

这个方法有三个参数，x是x轴应该旋转的倍数，y是y轴应该旋转的倍数，z是z轴应该旋转的倍数，deg是这三个轴单位倍数应该旋转的角度。

注意：前三个参数只能是小数，而且三个数字的平方和加一起必须为1！否则会被非标准转化。

1. 