# transform 2D

**变形的本质是通过中心点变形坐标轴，进而影响元素的变形**

* 变形(transform)是**CSS3新增的一个样式**，该样式配合4个变形函数一起使用，可以让元素发生如下变化
  1. 缩放 scale()
  2. 倾斜 skew()
  3. 位移 translate()
  4. 旋转 rotate()
* 当然我们也可以组合使用上述函数
* **配合过渡(transition)可以轻松实现很多“有规律的”酷炫特效**。

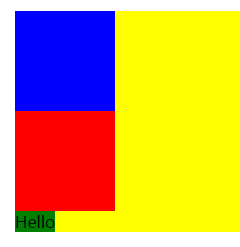
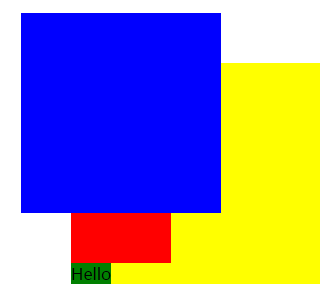
## scale()缩放函数

**缩放的本质是通过中心点缩放坐标轴，进而影响元素的缩放**

* 语法介绍
  + scale(x, y) 定义 2D 缩放转换，改变元素的宽度和高度。
  + scaleX(n) 定义 2D 缩放转换，改变元素的宽度。
  + scaleY(n) 定义 2D 缩放转换，改变元素的高度。
* 如何使用
  + transform: scale(x, y)
  + transform: scaleX(n)
  + transform: scaleY(n)
  + 参数说明
  + 0 <= (x, y, n) < 1 缩小
  + (x, y, n) = 1 默认值
  + (x, y, n) > 1 放大

普通的缩放 VS 2D缩放

* 普通的元素放大，设置元素的宽度(width)与高度(height),会有如下特点，
  1. 默认基于元素左上角缩放。
  2. 会影响兄弟元素的位置。
  3. 如果父元素未设置大小，会影响父元素的大小。
* 缩放函数缩放元素的特点，
  1. 只能对块元素或行内块元素设置缩放（确切的说所有的变换函数都只能设置在块元素中）
  2. 可以设置在任一一点进行缩放。也被称作变形的中心点
  3. 不会影响兄弟元素的位置。
  4. 如果父元素未设置大小，不会影响父元素的大小。
  5. 使用缩放函数，其内容及子元素也随之缩放

缩放如果配合过渡一起使用，缩放方向不同

1.正常文档流元素（改变宽高像素） --- 缩放的基准点：元素的左上角，过渡时都是基于左上角缩放

2.定位元素（改变宽高像素） ---缩放的基准点基于是否设置了 top, right, bottom, left，当不设置时，基准点是元素左上角；当设置某一位置时，另一个位置使用默认位置，如设置bottom后，就相当于设置了left bottom，基于这个基准点缩放；如设置right，就相当于设置了right top，基于这个基准点缩放；而设置left或top时，基准点还是左上角；当设置两个位置后，基准点就是这两个位置

3.scale() 基准点是变换中心点(transform-origin)。

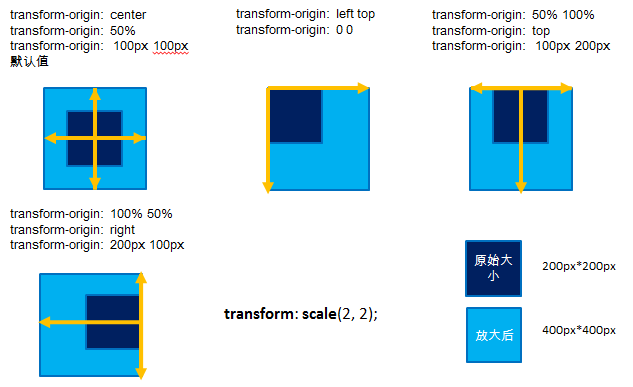
* 使用场景
  + 放大一个元素的同时，需要放大其内容及其子元素
  + 放大一个元素，不影响现有布局。
* 实际运用
  + 例如商品图片放大

## 变换中心点

* 语法介绍
  + transform-origin: x-axis y-axis z-axis;
* 参数说明
  + x-axis 定义视图被置于 X 轴的何处。可能的值：
    - left
    - center
    - right
    - length
    - %
  + y-axis 定义视图被置于 Y 轴的何处。可能的值：
    - top
    - center
    - bottom
    - length
    - %
  + z-axis 定义视图被置于 Z 轴的何处。可能的值：
    - length

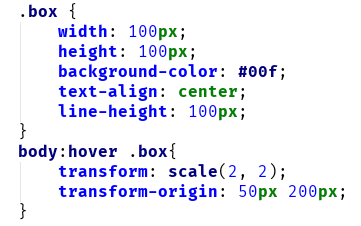
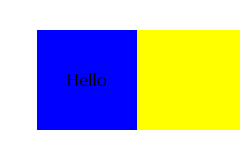
基于元素的左上角(0, 0)计算得到，右侧，下方都是正值

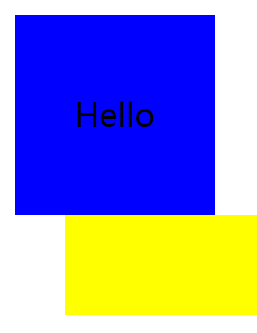
* **扩展知识：**
  + **z-axis 的值不能使用%和关键字**
  + **2D变换不需要考虑 z-axis，可以不写，默认0**



**缩放的本质是通过中心点缩放坐标轴，进而影响元素的缩放**

当变换中心点不在元素上时，要理解缩放的本质就可以了

 缩放后

如上例，变换中心在元素下面100px处，当鼠标移入时执行缩放，此时缩放倍数是2，坐标轴要放大2倍，那么元素相对于变换中心就应该是200px，所以元素看起来在父元素上面，由于坐标轴放到2倍，元素相应也要放大2倍

## skew()倾斜函数

**倾斜的本质是通过中心点倾斜坐标轴，进而影响元素的倾斜**

* 语法介绍
  + skew(x-angle,y-angle) 定义 2D 倾斜转换，沿着 X 和 Y 轴。
  + skewX(angle) 定义 2D 倾斜转换，沿着 X 轴。
  + skewY(angle) 定义 2D 倾斜转换，沿着 Y 轴。
* 如何使用
  + transform: skew(x-angle,y-angle)
  + transform: skewX(angle)
  + transform: skewY(angle)
* 参数说明
  + angle 表示旋转的角度，单位是deg。

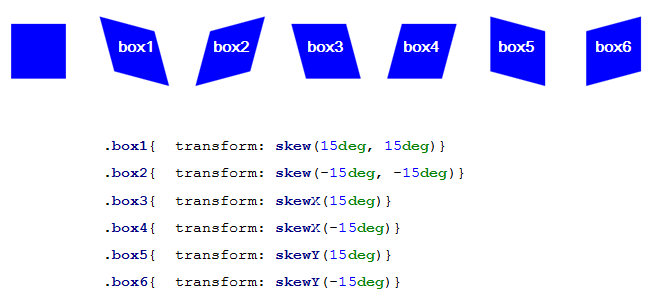
**当定义沿某一轴倾斜时，如果是正度数，另一轴的正方向向该轴*正方向*倾斜该度数，如果是负度数，另一轴的负方向向该轴*正方向*倾斜该度数**

**倾斜的本质是通过中心点倾斜坐标轴，进而影响元素的倾斜**

倾斜函数特点

1. 只能对块元素或行内块元素设置倾斜
2. 可以设置变换中心点。
3. 不会影响兄弟元素的位置。
4. 如果父元素未设置大小，不会影响父元素的大小。
5. 元素倾斜后，其内容及子元素的都会发生倾斜，可以对内容或子元素设置“反向”倾斜解决。

当设置两个轴都倾斜时，实际上两个轴的倾斜时同时进行的，但为了理解，可以认为先定义沿着X轴执行倾斜，由于是倾斜坐标轴，此时X轴未变化，而Y轴倾斜了一定角度，进而影响元素倾斜；执行后，要定义沿着Y轴进行倾斜，此时Y轴未变化，而X轴倾斜一定角度，进而影响元素倾斜



关于body:hover .box选择器和.box:hover选择器的区别：

如果让.box倾斜，.box:hover选择器当鼠标移入.box时（鼠标移入后不要动），执行倾斜，此时如果.box倾斜后鼠标不在.box上（如X、Y都倾斜45度，倾斜后.box消失，当然不一定非要倾斜45度，只要保证倾斜后鼠标不在.box上就行），那么就不是hover状态，此时稍微移动鼠标，就取消hover，.box就不再倾斜，恢复原状，恢复后，鼠标又处于hover状态，继续倾斜，这样就会看到.box不断的倾斜/恢复；而body:hover .box选择器，hover是给body的，现在只倾斜.box，只要鼠标在body上，就一直是hover状态，.box一直处于倾斜状态，.box就不会不断的倾斜/恢复。本质上就是 是不是处于hover状态

* 使用场景
  + 需要使用菱形效果
* 实际运用
  + 个性化菜单
  + 特殊的样式

## translate()位移函数

* 语法介绍
  + translate(x,y) 定义 2D 转换，沿着 X 和 Y 轴移动元素。
  + translateX(n) 定义 2D 转换，沿着 X 轴移动元素。
  + translateY(n) 定义 2D 转换，沿着 Y 轴移动元素。
* 如何使用
  + transform: translate(x,y)
  + transform: translateX(n)
  + transform: translateY(n)
* 参数说明
  + x, y, n 表示移动的距离 (length) 。例如 200px

扩展知识：

* 1. **变换中心点对位移函数 translate()不其起作用**
  2. **位移过大会导致父元素出现滚动条，此时需对父元素设置over-flow: hidden**
* 使用场景

改变一个元素的位置，但是不影响父元素的的大小及兄弟元素的位置。

* 实际使用

动画特效

## rotate()旋转函数

**旋转的本质是通过中心点旋转坐标轴，进而影响元素的旋转**

* 语法介绍
  + rotate(*angle*) 定义 2D 旋转，在参数中规定角度。
* 如何使用
  + transform: rotate(*angle)*
* 参数说明
  + angle 表示旋转的角度。例如 45deg（顺时针旋转45度）, -45deg（逆时针旋转45deg）

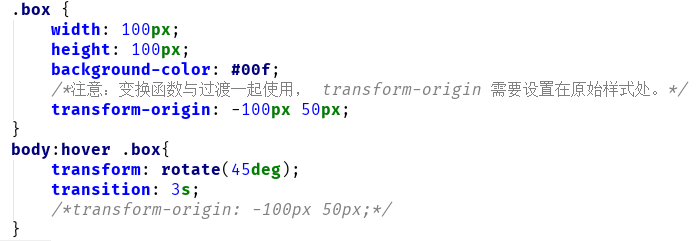
扩展知识：

改变变换中心点的位置，可以实现不同的旋转效果

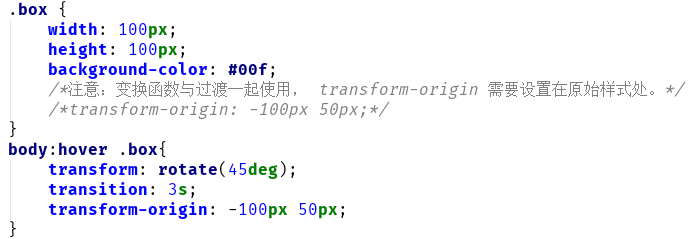
* 使用场景
  + 让一个元素旋转时
* 实际使用
  + 表盘
  + 照片墙

## 过渡函数对变换中心点设置位置的影响

变换函数与过渡一起使用时，如果中心点、变形函数、过渡函数写在一起，当执行时，每过渡1s，变形一次，需要重新计算中心点，中心点就不是固定的，此时变形轨迹和中心点固定的变形轨迹是不同的，但变形结果是相同的



上图中心点是固定的，过渡执行时，中心点依然不变



上图中心点是不固定的，过渡每执行1s，坐标轴就旋转一个角度，.box就要跟着旋转一个角度，中心点就需要跟着重新计算，但和上面最终旋转结果是一样的，都是旋转45度，只不过是旋转轨迹不同

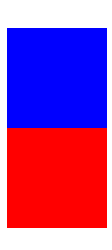
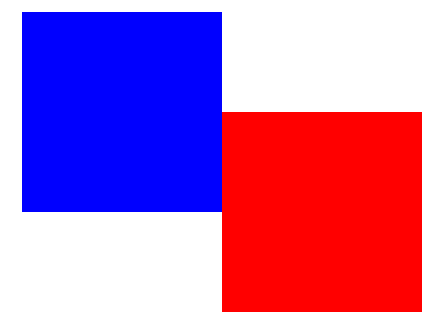
## 变换函数综合使用及位置对变形结果的影响

**记住本质是变形坐标轴就可以了**



多个变换函数综合使用，变形是同时进行的，但为了好理解，可以从左向右依次变形

.box先向右位移200px，中心点对位移不起作用，然后放大2倍，坐标轴放大2倍，元素跟着放大2倍

.box2先放大2倍，坐标轴放大2倍，元素放大2倍，此时坐标轴放大2倍后，本来设置的向右位移200px就变成400px了，所以看起来.box2比.box向右多位移200px变形前变形后

扩展知识：所有变形函数

1. 元素变形，不会影响相邻兄弟元素的位置。
2. 元素变形，如果父元素没有指定高度，不会导致父元素高度的变化。
3. 元素变形，只有在块元素或行内块元素中才生效。
4. 元素变形，元素的文本及其后代元素都会变形。