



transform 2D

讲师：许井龙

微信：ngsteel

1. transform 2d 简述
2. 缩放函数 scale()
3. 变型中心点
4. 倾斜函数 skew()
5. 位移函数 translate()
6. 旋转函数 rotate()
7. 变形函数组合使用

- 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。**

- 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
 - ① $0 \leq (x, y, n) < 1$ 缩小
 - ② $(x, y, n) = 1$ 默认值
 - ③ $(x, y, n) > 1$ 放大

- 普通的元素放大，设置元素的宽度(width)与高度(height)，会有如下问题，
 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. 可以设置在元素内任一点进行缩放。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。

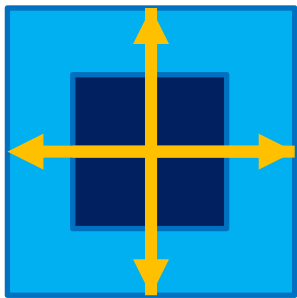
- 使用场景
 - 放大一个元素的同时，需要放大其内容及其子元素
 - 放大一个元素，不影响现有布局。
- 实际运用
 - 例如商品图片放大

- 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
- 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**

说明：

x-axis X轴
y-axis Y轴
z-axis Z轴

`transform-origin: center`
`transform-origin: 50%`
`transform-origin: 100px 100px`
默认值



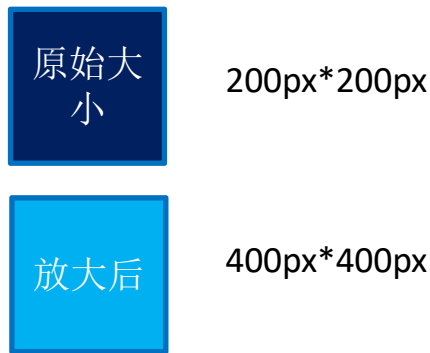
`transform-origin: top left`
`transform-origin: 0 0`



`transform-origin: 50% 100%`
`transform-origin: top`
`transform-origin: 100px 200px`

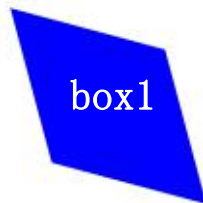


`transform-origin: 100% 50%`
`transform-origin: right`
`transform-origin: 200px 100px`



扩展知识：在2D变换中，变换中心点的 z-axis 值可以不写，默认为0

- 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。



```
.box1{ transform: skew(15deg, 15deg)}  
.box2{ transform: skew(-15deg, -15deg)}  
.box3{ transform: skewX(15deg)}  
.box4{ transform: skewX(-15deg)}  
.box5{ transform: skewY(15deg)}  
.box6{ transform: skewY(-15deg)}
```

1. 只能对块元素设置缩放
2. 可以设置在元素缩放的中心点。
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. 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。

- 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

可能需要的配色：

蔚蓝 #14B1E6

蓝黑 #131639

群青 #2957A5

- 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

- 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：

1. 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

- 当两个函数组合使用，例如
 - transform: translateX(200px) scale(2)
 - transform: scale(2) translateX(200px)
- 第一个函数会导致后面的函数“坐标”的变换，因此第一种写法，向左侧移动距离会看来更远。



translateX(200px)
scale(2)



scale(2)
translateX(200px)

200px的参照坐标轴



初始元素



变化后的元素



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，只有在块元素中才生效。
4. 元素变形，元素的文本及其后代元素都会变形。

