

transform, transition, animation, keyframes区别

链接: <https://segmentfault.com/a/1190000012698032>

这里不讲最基础的东西比如这些的含义, 站内很多人都有讲, 这里对比一下讲一讲区别, 加深一下理解。

看不明白的地方欢迎评论, 我更改一些措辞, 或者加一些示例。

其实要区别一些属性的最好方法就是看翻译, 翻译成中文很多就顾名思义了!

transform转变

`transform` 并不是一个动态属性, 他是一个静态的属性。专注于改变元素基于默认位置的以下值。注意, 只是改变, 并不是一个过程, 只是针对css的各种特殊改变, 完全可以看作是跟 `width, height, position` 等同类的属性。

值	描述
none	定义不进行转换。
matrix(n,n,n,n,n,n)	定义 2D 转换，使用六个值的矩阵。
matrix3d(n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n)	定义 3D 转换，使用 16 个值的 4x4 矩阵。
translate(x,y)	定义 2D 转换。
translate3d(x,y,z)	定义 3D 转换。
translateX(x)	定义转换，只是用 X 轴的值。
translateY(y)	定义转换，只是用 Y 轴的值。
translateZ(z)	定义 3D 转换，只是用 Z 轴的值。
scale(x,y)	定义 2D 缩放转换。
scale3d(x,y,z)	定义 3D 缩放转换。
scaleX(x)	通过设置 X 轴的值来定义缩放转换。
scaleY(y)	通过设置 Y 轴的值来定义缩放转换。
scaleZ(z)	通过设置 Z 轴的值来定义 3D 缩放转换。
rotate(angle)	定义 2D 旋转，在参数中规定角度。
rotate3d(x,y,z,angle)	定义 3D 旋转。
rotateX(angle)	定义沿着 X 轴的 3D 旋转。
rotateY(angle)	定义沿着 Y 轴的 3D 旋转。
rotateZ(angle)	定义沿着 Z 轴的 3D 旋转。
skew(x-angle,y-angle)	定义沿着 X 和 Y 轴的 2D 倾斜转换。
skewX(angle)	定义沿着 X 轴的 2D 倾斜转换。
skewY(angle)	定义沿着 Y 轴的 2D 倾斜转换。
perspective(n)	为 3D 转换元素定义透视视图。

transition过渡

`transition` 是指当它所绑定的属性（比如width）发生改变的时候，不再突变，而是根据速度曲线慢慢变化。这种改变不仅仅是定义好的hover改变，也包括后期js导致的改变，甚至不小心替换class名导致的css属性改变，也就是说无论在何种情况下发生的属性改变，都会导致过渡。这个时候就会知道 `transition` 并不是与 `transform` 绑定的属性，初学者（比如我>.<）在刚看的时候总会以为有 `transform` 必然有 `transition`，不然不可用，然而其实 `transition` 是完全可以不需要 `transform` 而独立使用的。`transform` 可以看作只是一个基础属性，与

width, height, position同类的属性，而 transition 只是规定了发生过渡的时候所需的自定义规则，两者独立使用也可。

值	描述
transition-property	规定设置过渡效果的 CSS 属性的名称。
transition-duration	规定完成过渡效果需要多少秒或毫秒。
transition-timing-function	规定速度效果的速度曲线。
transition-delay	定义过渡效果何时开始。

animation动画

先看表

值	描述
animation-name	规定需要绑定到选择器的 keyframe 名称。
animation-duration	规定完成动画所花费的时间，以秒或毫秒计。
animation-timing-function	规定动画的速度曲线。
animation-delay	规定在动画开始之前的延迟。
animation-iteration-count	规定动画应该播放的次数。
animation-direction	规定是否应该轮流反向播放动画。

它已然规定需要绑定 keyframe，这就说明 animation 才是与 keyframes 不可分割的，不定义 animation，keyframes 写了毫无意义，keyframes 不写，animation 怎么也动不起来。

由此可知，keyframes 是帮助 animation 的一个东西，那我们先看看 keyframes 再说 animation。

keyframes关键帧

熟悉动画制作的同学估计都知道关键帧，他也可以看作时间帧，这个时间在表述的时候以百分比的形式展示出来。

```
@keyframes move
{
  0%   {top:0px;}
  25%  {top:200px;}
  50%  {top:100px;}
  75%  {top:200px;}
  100% {top:0px;}
}
```

在每个百分比里，其实就是写好的css样式，里面可以规定width等基础属性，当然也可以写 `transform`！按照时间从小到大进行过渡

那最后 `keyframes` 就是已经写好的有过渡的动画动作，这才是动态的！`transform` 只是某刻的转变，加了时间，才叫动态，才叫动画！

那写好了动画，给谁用，就用到了 `animation`，将 `animation` 配置好写到css里，就可以给特定元素辅以动画了！每写出来的一个 `keyframes`，只是代表一个动画行为，变长变短，转圈圈，仅此而已，没有别的含义了。使用 `keyframes` 的就是 `animation` 了。

transition与animation的区别

所以其实要说区别，应该是要区分 `transition` 与 `animation` 的区别，说了这么多，顾名思义就可以看出来，一个是过渡，一个是动画！硬要说的话，过渡是只有两个帧的动画，动画是有很多次过渡的过渡。

`transition` 是由属性的变化而触发的，无论这个变化是不是我们期望的，只要属性改变，就会触发 `transition`。`animation` 的触发就是我们所设置的，根据她所规定的值指定我们想要的规则，可以在一开始就执行，可以执行一次再也不变。