说明:过渡是从一种状态到另外一种状态,过渡需要触发条件,通常过渡都是写在开始 状态 规定哪些属性需要过渡 transition-property all 默认值 全部属性过渡, 多个属性过渡, 用逗号隔开 transition-duration 🖯 过渡的时间 默认值 0s 1.讨渡效果transition 过渡的运动曲线 transition-timing-function ease 默认值 逐渐慢下来 linear 均速 transition-delay 🖯 过渡的延迟 连写: transition: all 1s linear 0s 优点:对自盒子不用计算向上和向左移动多少 1.位移 transform: translate(-50%,-50%) 特点: 若仅仅只是识位移, 盒子不会脱标, 盒子原有的位置还在标准流中 正=>顺时针 2.旋转 transform: rotate(45deg) 2.2D转换 负=>逆时针 3.旋转点 transform-origin: 水平 垂直 🎅 可以设置方位名称,也可以设置具体数值 1.大于0小于1=>缩放 3.缩放/扩大 transform: scale (number, number) 2.大于1=>扩大 1.定义动画 @keyframs 名称 { 0%{ 100%{ 2.指定动画名称 animation-name 3.完成一个周期所需要的时间 animation-duration 3.动画 4.动画的运动曲线 linear匀速 ease缓冲 steps步长 animation-timing-function: linear 5.次数 infinite无线循环 animation-iteration-count:infinite 6.是否你想播放,如果想逆向播放的前提是次数需要大于2次 animationdirection:alternate 7.结束之后的状态 在哪里 animation-fill-mode: forwards 8.延迟几秒开始 animation-delay: 3s 9.暂停 animation-play-state: paused 1.x轴 正值向右, 负值向左 2.y轴 正值向下, 负值向上 1.位移: transform: translate(x,y,z) 3.z轴 正值向前, 负值向后 注意:如果要设置多个轴时,不要分开设置,要写在同一个transform中 transform: - transform: rotateX(角度); 2.旋转 😑 - transform: rotateY(角度); 4.3D转换 - transform: rotateZ(角度); 1.视距越大,物体越小,*视距越小,物体越大 2.translateZ值越大,物体越大,translateZ越小,物体越小 3.translateZ一定要与perspective 一起使用,近大远小 注意: 1.X正值=>右 4.左手法则 2.Y正值=>下

Css2D-3D