day14

三.CSS hack

|  |
| --- |
| -webkit-  -o-  -ms-  -moz-  由于不同浏览器对css的解析认识不同，会导致一份css在不同浏览器生成的页面效果不同。  我们可以在样式前，添加前缀，浏览器会自动解析符合自己内核前缀代码。编写这种代码的过程，就叫做编写css hack |

四.css reset和normalize.css

1. css reset/css重置 \*{margin:0;padding:0}

|  |
| --- |
| 把一些各个浏览器解析不同的样式属性值，  非常暴力直接的，清空或者改变成其他值。  不管默认样式是否需要，都清除了。  如果我们需要一些元素的默认样式，那我们得自己手动添加  项目中如果使用css reset，1.是别人写好的 css reset  2.自己手动编写css reset |

2.normalize.css

|  |
| --- |
| 相对于传统的css reset,normalize.css是一种现代的，为html5准备的，优质的替代方案  normalize.css有几个特点  1.保护有用的默认样式，而不是完全去掉  2.一般化的样式，为大部分html元素提供  3.修复浏览器自身的bug  4.解释代码，用注释和详细的文档来解释每一句命令  <https://github.com/necolas/normalize.css>  自己做项目的时候，必须要引入css reset或者normalize.css。二选一 |

五.转换（重点\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*）

1.什么是转换

|  |
| --- |
| 改变元素在页面中的位置，大小，角度，以及形状  2D转换，只在x轴和y轴发生转换效果  3D转换，添加Z轴 |

2.转换属性

|  |
| --- |
| transform:function(转换函数)  学一个转换属性 transform  学4个转换函数 |

3. transform-origin

|  |
| --- |
| 转换原点  取值：以px为单位的数字  %  关键字 x:left/center/right y:top/center/bottom  取值个数  1.两个值，表示原点在x轴上和y轴上的位置  2.三个值，x,y,z上 位置 |

4.2d转换

①位移

|  |
| --- |
| transform:translate()  取值 1.translate(x,y)指定元素在x轴和y轴上的位移  2.translate(x)和translateX(x)指定元素在x轴上的位移  3.translateY(y) 指定元素在y轴上的位移 |

②缩放

|  |
| --- |
| transform:scale()  取值：1.取一个值 scale(v1)  v1>1 x轴和y轴都放大  v1=1 x轴和y轴都是原尺寸  0<v1<1 x轴和y轴都缩小  -1<v1<0 x轴和y轴都缩小,并反转  v1<-1 x轴和y轴都放大,并反转  2.取两个值scale(x,y); 分别设置x轴和y轴的缩放  3.scaleX(x) 单独设置x轴缩放  4.scaleY(y) 单独设置y轴缩放 |

③旋转

|  |
| --- |
| transform:rotate(ndeg)  取值 n:+ 顺时针 - 逆时针  注意：  1.转换原点会影响旋转效果  2.旋转是连同坐标轴一起旋转的，会影响旋转之后的位移方向 |

④倾斜

|  |
| --- |
| transform:skew()  取值 1.skew(x)等同于skewX(x)  让元素向着x轴发生倾斜，实际上改变的是y轴的角度  x:+ y轴逆时针倾斜  x:- y轴顺时针倾斜  2.skewY(y)  让元素向着y轴发生倾斜，实际上改变的是x轴的角度  y:+ x轴顺时针倾斜  y:- x轴逆时针倾斜  3.skew(x,y) |

练习

|  |
| --- |
| 02\_ex  创建div 200px\*200px。背景随意，鼠标悬停时，  该元素向右偏移200px，向下偏移100px  旋转135deg,向x轴倾斜45deg，放大到1.5倍 |

面试题

|  |
| --- |
| 使用css，让一个div(200px\*200px)在任何时候，都处于屏幕正中心 |

5.3D转换

①透视距离

|  |
| --- |
| 模拟人的眼睛到3D转换的元素之间的距离，  距离不同，看到效果不同  perspective:以px为单位的数字  透视距离，写在转换元素的父元素上 |

②3D旋转

|  |
| --- |
| transform：  取值  1.rotateX(xdeg)  以x轴为中心轴，旋转元素(老式爆米花机，烤羊腿)  2.rotateY(ydeg)  以y轴为中心轴，旋转元素（棉花糖，旋转木马，钢管舞，陀螺）  3.rotateZ(zdeg)  以z轴为中心轴，旋转元素(电风扇，摩天轮，风车)  4.rotate3D(1,1,1,100deg)  x,y,z取值0，代表不参与旋转  取值>0，代表该轴参与旋转 |

六.过渡

|  |
| --- |
| 让css的值，在一段时间内平缓变化的效果  使用过渡，要设置两个css的状态  1.开始的时候的值  2.结束的时候的值 |

1.设置过渡的属性

|  |
| --- |
| transition-property  取值，直接写css属性名称，多个属性名称之间用空格分开  all 所有支持过渡的属性，都参与此次过渡效果  支持过渡的属性  1.颜色属性  2.多数取值为具体数字的属性  3.阴影  4.转换transform  5.visibility |

2.设置过渡的时长

|  |
| --- |
| transition-duration:  取值：s/ms为单位的数字 |

3.设置过渡时间曲线函数

|  |
| --- |
| transition-timing-function  取值  1.ease 默认值 慢开，中快，慢关  2.linear 全程匀速  3.ease-in 慢开，快关  4.ease-out 快开，慢关  5.ease-in-out 慢速开始，慢速结束，中间先变快后变慢 |

4.过渡的延迟

|  |
| --- |
| transition-delay:  取值，以s/ms为单位的数字 |

5.过渡代码的编写位置

|  |
| --- |
| 写在原来选择器中，过渡效果，有去有回  写在hover中，过渡效果，有去无回 |

6.过渡的简写

|  |
| --- |
| transition:property duration timing-function delay;  最简方式 transition：duration; |

练习：

|  |
| --- |
| 翻滚吧，亮亮！  04\_ex 一个div，向右移动800px，并且旋转108deg  使用转换，使用过渡  爱的魔力转圈圈！  先旋转108deg，再向右移动800px 10分钟 等深圳同学上线 |

七.动画

1.什么是动画

|  |
| --- |
| 多个过渡放到一起 |

2.使用 关键帧 来控制动画的每一个状态

|  |
| --- |
| 关键帧  1.动画执行的时间点  2.这个时间点上的样式 |

3.动画的使用步骤

①声明动画

|  |
| --- |
| @keyframes 动画名称{  0%{动画样式}  ........  100%{动画样式}  } |

②使用动画

|  |
| --- |
| 1.选中要使用的动画名称  animation-name:动画名称;  2.设置动画执行时间  animation-duration: s/ms;  3.设置动画的时间曲线函数  animation-timing-function:ease-out;  4.设置动画开始的延迟时间  animation-delay: s/ms; |

4.动画的其它属性

|  |
| --- |
| 5.设置动画的播放次数  animation-iteration-count:  取值：整数，具体的播放次数  infinite 无限  6.动画的播放顺序  animation-direction:  取值：1.normal 默认值 顺序播放  2.reverse 翻转播放  3.alternate 轮流播放，第一遍正，第二遍反，第三遍正......  奇数次正着放，偶数次反着放 |

5.动画的简写方式

|  |
| --- |
| animation:name duration timing-function delay count direction  最简方式 animation：name duration; |

6.动画的其它样式

|  |
| --- |
| 设置动画在播放之前，和播放之后的状态  animation-fill-mode:  取值：1.none 默认，不填充  2.backwards 动画播放之前的延迟时间内，  将动画的第一帧填充在元素中  3.forwards 动画播放完成之后，  将动画的最后一帧填充在元素中  4.both 同时设置backwards和forwards |
| 动画的播放状态  animation-play-state:  取值 paused 暂停  running 播放 |

7.动画的兼容性 csshack

|  |
| --- |
| 如果要兼容低版本浏览器，需要在动画声明的时候添加前缀  @keyframes 动画名称{ }  @-webkit-keyframes 动画名称{ }  @-ms-keyframes 动画名称{ }  @-o-keyframes 动画名称{ }  @-moz-keyframes 动画名称{ } 17:03~17:18休息 |

练习，坤坤打亮亮，10分钟

8.animate.css

|  |
| --- |
| <https://daneden.github.io/animate.css/> |

作业

1.阅读normalize.css文件

2.尝试使用animate.css