爱创课堂前端培训

# css3

第2天课堂笔记（本课程共2天）

班级：北京前端训练营17期

讲师：冯楠娜

日期：2018年9月1日

爱创课堂官网 ：[www.icketang.com](http://www.icketang.com)

# 复习

css3伪元素：

::before 表示在原有内容之前插入

::after 表示在原有内容之后插入

content: 必须书写不能省略

行内元素

::first-letter 首字母（设置文字属性）

::first-line 首行文字(文字属性)

::selection 鼠标圈选（背景颜色）

|  |
| --- |
| 1. .box::first-letter { 2. font-size: 50px; 3. color: red; 4. } 5. .box::first-line { 6. color: red; 7. } 8. .box::selection { 9. background-color: lightblue; 10. color: #000; 11. } |

border-radius: 可以使用像素，百分数

box-shadow: (inset) 右偏移量 下偏移量 模糊半径 延伸量 颜色;

text-shadow: 右偏移量 下偏移量 模糊半径 颜色;

background-origin: 背景起源（content-box）

background-clip: 背景裁切(content-box)

background-size: 背景图宽度 背景图高度;

cover(覆盖)

contain(容纳)

结合精灵图使用：精灵图减半

盒子宽高减半

背景图减半

background-position减半

渐变颜色： background-image: -webkit-linear-gradient(渐变开始方向top, 渐变颜色 百分数, 渐变颜色……);

多背景：一个元素可以添加多个背景，用逗号隔开

# 过渡

## 过渡的基本形式

元素 → 元素:hover

都是干嘣的效果，瞬间从开始完成到结束状态。

过渡效果：将开始状态到结束状态可以将中间过程也可以查看到。

第一个参数: 参与过渡的属性（一般我们习惯书写all，所有变化的属性都参与过渡）

第二个参数： 完成时间，单位s

第三个参数： 缓冲描述（linear匀速 ease变速 贝塞尔曲线）

第四个参数： 延迟时间，单位也是s,如果没有延迟0s。

|  |
| --- |
| 1. transition: width 1s cubic-bezier(0.59, 0.41, 0.17, 1.21) 10s; |

可以拆分为四个小属性

|  |
| --- |
| 1. transition-property 2. transition-duration 3. transition-timing-function 4. transition-delay |

## 过渡参与的属性

数值型的属性可以参与过渡。比如width height font-size 等

可以参与过渡

|  |
| --- |
| 1. background-color 2. background-position |

# 变形

## 2D变形

transform: 变形

旋转：rotate()

单位是deg

正方向：顺时针



|  |
| --- |
| 1. .box img { 2. /\*img是行内元素\*/ 3. display: block; 4. width: 100%; 5. } |

|  |
| --- |
| 1. transform: rotate(530deg); |

* 放缩

scale(数字)

没有单位

0-1表示缩小

1-无穷 表示放大

|  |
| --- |
| 1. transform: scale(0.5); |

* skew(水平斜切，垂直斜切)

单位是deg

|  |
| --- |
| 1. transform: skew(10deg,40deg); |

同一个元素可以有多个变形，用空格隔开。

元素没有变形none

## 2D空间移动

空间移动：translateX() 表示水平方向移动（表示相对于自身当前位置进行偏移）

translateY() 表示垂直方向移动，单位px

|  |
| --- |
| 1. transform: translate(100px, 100px); |

绝对定位元素的水平居中：

left: 50%;

mgl: -自身宽度/2;(可以替换成空间移动)

|  |
| --- |
| 1. left: 50%; 2. transform: translateX(-50%); |

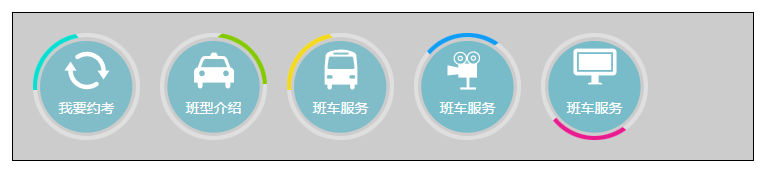
变形练习：

结构：

|  |
| --- |
| 1. <div class="box"> 2. <ul> 3. <li><a href="">我要约考</a></li> 4. <li class="no2"><a href="">班型介绍</a></li> 5. <li class="no3"><a href="">班车服务</a></li> 6. <li class="no4"><a href="">班车服务</a></li> 7. <li class="no5"><a href="">班车服务</a></li> 8. </ul> 9. </div> |

样式：

|  |
| --- |
| 1. .box { 2. width: 700px; 3. padding: 20px; 4. border: 1px solid #000; 5. background-color: rgba(0,0,0,.2); 6. margin: 50px auto; 7. overflow: hidden; 8. } 9. .box ul li { 10. float: left; 11. width: 107px; 12. height: 107px; 13. /\*同级元素有距离使用mg\*/ 14. margin-right: 20px; 15. position: relative; 16. } 17. .box ul li::after { 18. content: ""; 19. /\*行内元素\*/ 20. position: absolute; 21. left: 0px; 22. top: 0px; 23. width: 107px; 24. height: 107px; 25. background-image: url(images/a\_1.png); 26. /\*过渡\*/ 27. transition: all 1s linear 0s; 28. } 29. .box ul li.no2::after { 30. background-image: url(images/a\_2.png); 31. } 32. .box ul li.no3::after { 33. background-image: url(images/a\_3.png); 34. } 35. .box ul li.no4::after { 36. background-image: url(images/a\_4.png); 37. } 38. .box ul li.no5::after { 39. background-image: url(images/a\_5.png); 40. } 41. .box ul li::before { 42. content: ""; 43. position: absolute; 44. width: 46px; 45. height: 42px; 46. background-image: url(images/tb\_1.png); 47. top: 15px; 48. /\*绝对定位水平居中\*/ 49. left: 50%; 50. transform: translateX(-50%); 51. /\*自定义压盖效果\*/ 52. z-index: 4; 53. } 54. .box ul li.no2::before { 55. background-image: url(images/tb\_2.png); 56. width: 40px; 57. } 58. .box ul li.no3::before { 59. background-image: url(images/tb\_3.png); 60. width: 34px; 61. } 62. .box ul li.no4::before { 63. background-image: url(images/tb\_4.png); 64. width: 41px; 65. } 66. .box ul li.no5::before { 67. background-image: url(images/tb\_5.png); 68. width: 44px; 69. } 70. .box ul li a { 71. display: block; 72. width: 100%; 73. height: 100%; 74. /\*lh数值大于盒子的高度 偏下显示\*/ 75. line-height: 150px; 76. text-align: center; 77. /\*有定位的元素压盖没有定位的元素 78. 有定位的元素才可以使用z-index 79. \*/ 80. position: relative; 81. z-index: 10; 82. } 83. .box ul li:hover::after { 84. background-image: url(images/a\_1\_h1.png); 85. transform: rotate(360deg); 86. } 87. .box ul li.no2:hover::after { 88. background-image: url(images/a\_2\_h1.png); 89. } 90. .box ul li.no3:hover::after { 91. background-image: url(images/a\_3\_h1.png); 92. } 93. .box ul li.no4:hover::after { 94. background-image: url(images/a\_4\_h1.png); 95. } 96. .box ul li.no5:hover::after { 97. background-image: url(images/a\_5\_h1.png); 98. } |



# 动画

## 3.1 定义动画

定义动画：

需要添加前缀

@-浏览器前缀-keyframes 动画名称 {

from {} 开始帧

to {} 结束帧

}

key: 关键

frame帧

每一帧书写的属性必须一一对应

|  |
| --- |
| 1. /\*定义动画\*/ 2. @-webkit-keyframes donghua { 3. /\*每一帧状态书写\*/ 4. /\*开始帧\*/ 5. from { 6. width: 200px; 7. background-color: lightblue; 8. } 9. /\*结束帧\*/ 10. to { 11. width: 600px; 12. background-color: green; 13. } 14. } |

## 3.2 调用动画

animation: 调用动画

第一个参数：动画名称

第二个参数：动画完成时间，单位是s(表示从动画开始到结束一次动画完成时间)

第三个参数： 缓冲描述 linear ease

第四个参数： 动画延迟时间，单位s，延时时间体现在第一次动画开始。

第五个参数： 动画次数（infinite无限）

第六个参数：自动补全反方向动画（alternate）反方向也算动画次数

最后一个参数：保持最后一帧的状态（forwards）

## 3.3 百分数定义动画

表示在整个动画中该时间动画完成的状态

|  |
| --- |
| 1. /\*定义动画\*/ 2. @-webkit-keyframes donghua { 3. /\*每一帧状态书写\*/ 4. /\*开始帧\*/ 5. 0% { 6. width: 200px; 7. background-color: lightblue; 8. } 9. 10% { 10. width: 400px; 11. background-color: red; 12. } 13. 30% { 14. width: 600px; 15. background-color: yellow; 16. } 17. 90% { 18. width: 100px; 19. background-color: pink; 20. } 21. /\*结束帧\*/ 22. 100% { 23. width: 600px; 24. background-color: green; 25. } 26. } |

## 3.4 动画属性

animation-play-state: paused

play: 播放

state: 状态

paused: 暂停

|  |
| --- |
| 1. /\*暂停\*/ 2. .box:hover img { 3. -webkit-animation-play-state: paused; 4. } |

# 视口

分辨率：就是屏幕像素的数量，比如1600\*900，指的是横向有1600的像素点，纵向有900个像素点。所谓的像素点是物理元件最小的发光单位。

手机的分辨率是多少呢？非常的大。1920×1080

做手机页面的时候，一定要加上

<meta name="viewport" content="width=device-width" />

此时如果你写一个p的字号为font-size:14px;不用担心用户能不能看清！用户一定能看的真真的！此时厂商已经计算好了一个所谓视口约束宽度。14px和14px的感觉完全一样！

一般来说，手机视口约束之后都是320~420之间。而分辨率基本上都是1000~1200，也就是说，约束视口之后所谓的1px，实际上手机用了2点多个像素来渲染。手机中的图片，必须使用放大两倍的图片缩小使用，这样能精细。

约束视口完整写法：

<meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0, maximum-scale=1.0,user-scalable=no" id="viewport" />不用背诵

width=device-width 约束视口

initial-scale=1.0 初始视口倍数是1倍

minimum-scale=1.0 最小允许视口宽度是1倍

maximum-scale=1.0 最大允许视口宽度是1倍

user-scalable=no 不允许用户缩放视口

|  |
| --- |
|  |

# 五、百分比布局

1 pc有版心效果，移动没有版心因为移动端约束后视口宽度是300-400之间，不设置版心，显示更多内容

2 为了移动端效果和app效果也是没有版心

移动端布局宽度一般使用百分比布局。

标准文档流（浮动效果和标准文档流一样）

width: 表示相对于父盒子内容宽度的比值

height: 表示相对于父盒子内容高度的比值

padding,margin: 表示相对于父盒子内容宽度的比值

border:不能使用百分数

绝对定位：

width: 表示相对于定位参考盒子内容宽+padding的比值

height: 表示相对于定位参考盒子内容高+padding的比值

padding,margin: 表示相对于定位参考盒子内容宽+padding的比值

border:不能使用百分数

# 六、双飞翼模型

浮动元素的mgl负数，表示相对于上一个元素位置偏移。

中间自适应宽度，两边固定宽度的布局。

总结：

1 中间自适应宽度是根据margin设定，没有显示设置宽度，来达到自适应宽度的属性。

2 为了让left，right，container放在一行内我们用了浮动布局

3 为了使left元素固定在左边，我们通过marginleft实现，实际上移动的距离就是container的宽度，为了让其固定宽度，我们显性定义了宽度。

4 为了让right元素固定右边，我们通过marginleft实现，实际上移动的距离，是其自身的宽度。



|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |