# html5插入视频和音频

## html4视频

之前借助flash

|  |
| --- |
| 1. <embed src="//player.video.iqiyi.com/a147354accc0191887458cfddaf5ceac/0/0/v\_19rqvdm9eo.swf-albumId=2133665100-tvId=2133665100-isPurchase=0-cnId=undefined" allowFullScreen="true" quality="high" width="480" height="350" align="middle" allowScriptAccess="always" type="application/x-shockwave-flash"></embed> |

## html5插入视频

不需要借助flash插件 直接使用标签video

video具有属性

src:插入视频路径

controls:控制条

autoplay:自动播放

loop:循环播放

还可以通过source标签引入路径

|  |
| --- |
| 1. <video controls> 2. <source src="video/2.mp4"> 3. </video> |

## html5插入音频

audio标签具有属性

src:路径

controls:控制条

autoplay:自动播放

loop:循环播放

同样可以使用source引入路径

|  |
| --- |
| 1. audio src="audio/a0.ogg" autoplay loop></audio> |

# css3

css3使用不需要任何声明。对于不认识属性静默不报错

## 2.1 属性选择器

通过标签属性值进行选中

[]：属性选择器

开头匹配：^=

|  |
| --- |
| 1. a[href $= "page2"] { 2. color: green; 3. } |

结尾匹配：$=

|  |
| --- |
| 1. /\*结尾匹配\*/ 2. a[href $= "page2"] { 3. color: green; 4. } |

任意匹配： \*=

|  |
| --- |
| 1. /\*任意匹配\*/ 2. a[href \*= "page2"] { 3. color: blue; 4. font-weight: bold; 5. } |

有匹配：不需要书写属性值

|  |
| --- |
| 1. /\*有匹配 存在这个属性名\*/ 2. img[alt] { 3. border-color: orange; 4. } |

属性选择器可以不书写标签名

属性选择器也可以连续交集

|  |
| --- |
| 1. [alt][src="images/3.jpg"] { 2. border-color: blue; 3. } |

html5可以自定义属性，必须用data-开头

短横匹配： |=

|  |
| --- |
| 1. div[data-xi |= "box"] { 2. /\*1box必须在开头 2如果还有其他属性值必须以短横连接\*/ 3. color: orange; 4. } |

分组匹配： ~=

|  |
| --- |
| 1. div[data-xi ~= "box"] { 2. /\*1具有box 2如果有其他属性值必须用空格分隔表示独立成\*/ 3. color: pink; 4. font-weight: bold; 5. } |

## 2.2 儿子序选择器

选中某个元素所有儿子中的老几。关注的是所有儿子中的整体排名

:first-child 第一个儿子

:last-child 最后一个儿子

:nth-child(n) 第n儿子

:nth-child(an + b) 连续多个

:nth-last-child(n) 表示倒数n个

|  |
| --- |
| 1. <div class="box"> 2. <p>p</p> 3. <p>p</p> 4. <p>p</p> 5. <p>p</p> 6. <h3>h3</h3> 7. <h3>h3</h3> 8. <h3>h3</h3> 9. </div> 10. <div class="box"> 11. <h3>h3</h3> 12. <h3>h3</h3> 13. <h3>h3</h3> 14. <p>p</p> 15. <p>p</p> 16. <p>p</p> 17. <p>p</p> 18. </div> |

## 2.3儿子类型序选择器

选中某个元素所有同种标签类型儿子中的老几。关注的是同种标签中儿子的排名情况

:first-of-type 同种标签中的第一个儿子

:last-of-type 同种标签中的最后一个儿子

:nth-of-type(n) 同种标签中的第n儿子

:nth-of-type(an + b) 同种标签中的连续多个

:nth-last-of-type(n) 同种标签中的表示倒数n个

|  |
| --- |
| 1. .box p:first-of-type { 2. color: red; 3. } 4. h3:last-of-type { 5. color: blue; 6. } 7. p:nth-of-type(3) { 8. color: orange; 9. font-weight: bold; 10. } 11. p:nth-last-of-type(2) { 12. color: green; 13. font-size: 30px; 14. } 15. .teshu:nth-of-type(2n) { 16. font-size: 50px; 17. color: yellow; 18. } |

## 2.4 关系节点选择器

div p {}

> :子级选择器 只能选中儿子节点不能选中其他后代元素

+： 后面第一个亲兄弟

~： 后面所有的亲兄弟

# 伪类

## hover任何都具有

:hover 鼠标悬停伪类

## 表单伪类

:checked 被选中伪类

:focus 获取焦点伪类

:disabled 不可用表单伪类

:enabled 可用表单伪类

## 节点伪类

:only-child 选中元素必须是某一个元素唯一的子节点

:empty 空节点伪类

# 伪元素

## before

::before 在原有内容之前增加

必须设置content：属性值书写要增加的内容

行内元素

## after

在原有内容之后增加

必须设置content属性，行内元素

## 应用

先导符号

|  |
| --- |
| 1. .box ul li a::before { 2. /\*必须设置content\*/ 3. content: ""; 4. /\*行内元素 不能设置宽高 5. 1小圆点设置宽高 6. 2位置便于调整 7. 定位 8. \*/ 9. position: absolute; 10. left: 10px; 11. top: 20px; 12. width: 10px; 13. height: 10px; 14. border-radius: 50%; 15. background-color: #eee; 16. } 17. .box ul li a:hover { 18. color: orange; 19. } 20. .box ul li a:hover::before { 21. background-color: orange; 22. } |

清除浮动的影响：

|  |
| --- |
| 1. div::after { 2. content: ""; 3. /\*行内元素\*/ 4. display: block; 5. /\*清除\*/ 6. clear: both; 7. } |

模拟交互效果

|  |
| --- |
| 1. .box::after { 2. content: "查看更多"; 3. /\*压盖效果\*/ 4. position: absolute; 5. width: 200px; 6. height: 200px; 7. background-color: rgba(255,255,255,.5); 8. left: 0; 9. top: 200px; 10. text-align: center; 11. line-height: 200px; 12. color: #000; 13. } 14. .box:hover::after { 15. top: 0; 16. } |

## 4.4 其他伪元素

::first-letter 首个汉字

::first-line 首行文字

::selection 圈选

|  |
| --- |
| 1. p::first-letter { 2. color: red; 3. } 4. p::first-line { 5. color: red; 6. } 7. p::selection { 8. background-color: orange; 9. } |

# 圆角

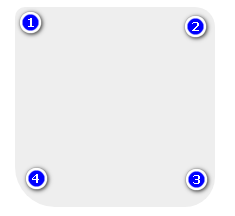
border-radius

## 像素表示法



|  |
| --- |
| 1. /\*半圆：有些边不设置圆角极限可以是短边全部\*/ 2. p:nth-child(3) { 3. width: 400px; 4. height: 200px; 5. border-radius: 200px 200px 0 0; 6. } |

## 5.2 百分数表示



还可以按照方向拆分：

|  |
| --- |
| 1. border-top-left-radius: 2. border-top-right-radius: 3. border-bottom-left-radius: 4. border-bottom-right-radius: |

# 盒子阴影

右偏移量

下偏移量

模糊半径

延伸量（省略）

颜色

|  |
| --- |
| 1. box-shadow: 5px 10px 10px 10px rgba(0,0,0,.5); |

一个元素可以同时设置多个阴影，用逗号隔开

|  |
| --- |
| 1. .no2 { 2. /\*多个阴影使用逗号隔开\*/ 3. box-shadow: 0 0 5px rgba(255,0,0,.9), 4. 5px 5px 5px rgba(0,255,0,.5); 5. } |

内阴影：inset

|  |
| --- |
| 1. .no3 { 2. box-shadow: inset 5px 5px 5px rgba(255,255,0,.5); 3. } |

文字阴影：类似于盒子阴影，只是没有内阴影和延伸量

|  |
| --- |
| 1. text-shadow: 0 0 5px rgba(255,0,0,.5); |

# 背景

## 背景起源

background-origin: content-box; 从内容盒左上角开始渲染

背景位置：表示相对于背景起源位置进行偏移

|  |
| --- |
| 1. background-origin: content-box; |

## 背景裁切

|  |
| --- |
| 1. background-clip: content-box; |

表示将其他区域背景图裁切掉，只剩下内容盒区域的背景

## 背景尺寸

background-size:宽 高；

* 像素表示法：

|  |
| --- |
| 1. background-size: 400px 500px; |

* 百分数表示法

第一个参数：背景图宽相对于大背景区域宽比值

第二个参数：背景图高相对于大背景区域高比值

|  |
| --- |
| 1. background-size: 50% 50%; |

* 单词表示法

cover(覆盖，尽可能大)

1背景图不变形 2大背景区域一定不会留白 3背景图有可能显示不完整

|  |
| --- |
| 1. background-size: cover; |

contain(容纳，显示全部)

1背景图不变形 2背景图一定显示完整 3大背景区域可能会有留白

* 背景尺寸和精灵图配合使用（减半）

1盒子宽高减半

2background-position减半

3background-size减半

|  |
| --- |
|  |



## 渐变背景

线性渐变

background-image: -webkit-linear-gradient(top, red,green)

css3有些属性或者属性值处于实验阶段，需要浏览器前缀

chrome safari -webkit-

firefox -moz-

IE -ms-

opera -o-

|  |
| --- |
| 1. .box1 { 2. width: 200px; 3. height: 200px; 4. border: 1px solid #000; 5. background-image: -webkit-linear-gradient(top,red,green); 6. background-image: -moz-linear-gradient(top,red,green); 7. background-image: -ms-linear-gradient(top,red,green); 8. background-image: -o-linear-gradient(top,red,green); 9. background-image: linear-gradient(top,red,green); 10. } |

第一个参数：渐变开始方向

其他参数：渐变颜色，可以书写多个，用逗号隔开

颜色后面可以用空格隔开一个百分数，表示该颜色在整体渐变中出现的位置

|  |
| --- |
| 1. background-image: -webkit-linear-gradient(left,red,green 80%,blue); |

## 多背景

一个元素可以渲染多个背景用逗号隔开，其他单一属性也是用逗号隔开，一一对应背景图

|  |
| --- |
|  |

# 过渡

## 8.1概述

transition: 过渡属性 过渡完成时间 缓冲描述 延迟时间;

|  |
| --- |
| 1. .box { 2. width: 300px; 3. height: 100px; 4. padding: 50px; 5. border: 1px solid #000; 6. background-color: lightblue; 7. /\*过渡 8. 第一个参数：参与过渡属性 all 9. 二：时间，单位是s 10. 三：缓冲描述 linear匀速 ease非匀速 贝塞尔曲线 11. 四：延迟时间，单位是s,如果是0s也必须设置单位 12. \*/ 13. /\*transition: width 1s linear 0s;\*/ 14. transition: all 1s linear 0s; 15. transition: all 1s cubic-bezier(0.14, 1.1, 0.82, -0.04) 0s; 16. } 17. /\*干嘣效果\*/ 18. .box:hover { 19. width: 700px; 20. height: 500px; 21. } |

transition-property:过渡属性

transition-duration:过渡时间

transition-timing-function:缓冲描述

transition-delay:延迟时间

## 8.2 参与过渡属性

一般数值型的都可以参与过渡

|  |
| --- |
| 1. background-color: //可以 2. background-position: //可以 |



|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |