* **CSS规则判定**

1. **各浏览器的CSS前缀**

一个css规则正式成为w3c草案之前，如果该浏览器支持该规则，则会在该规则前面加入表示该厂商浏览器的”前缀标识”如：

firefox:-moz- Opera:-o- Safari/Google Chrome:-webkit- Konqueror:-khtml- IE: -ms-

**【JS中的写法】**

如果一个css规则存在”前缀标识”，则该规则在javascript中的写法是不同的：

* Firefox：Moz + 词法意义上首字母大写的原规则。如: MozBoxSizing
* Opera：O + 词法意义上首字母大写的原规则。如：OBoxSizing
* Chrome/Safari：webkit + 词法意义上首字母大写的原规则。如：webkitBoxSizing

1. **用JS判断浏览器是否支持指定的css规则**

|  |
| --- |
| //原boxsizing规则, 全部小写  console.log('boxsizing' in document.documentElement.style);  //false in Firefox 4.0.1  //带前缀标识的boxsizing规则正确写法 'M', 'B'和'S'大写  console.log('MozBoxSizing' in document.documentElement.style);  //true in Firefox 4.0.1 |

1. **表单**

**水平滑条**

<input type="range" style="-webkit-appearance:slider-horizontal;">

**渐变按钮**

<button style="-webkit-appearance:button;width:200px;height:30px;">gradient button</button>

**椭圆按钮**

<button style="-webkit-appearance:push-button;width:200px;">aqua button</button>

**搜索框**

<input type="text" style="-webkit-appearance:searchfield;" value="kitten"></input>

**CSS3**

1. **Attribute selectors**

[att^=val]：属性值前缀为”val”

[att$=val]：属性值后缀为”val”

[att\*=val]：属性值中包含字符”val”

CSS

|  |
| --- |
| p[title^="ho"] {background: green;} |

HTML

|  |
| --- |
| <p title="home"> This paragraph should have a green background. </p> <p title="contact"> The attribute selector doesn't match this paragraph. </p>  <p title="hot"> This paragraph should have a green background. </p> |

1. **multi-column layout**

两种方式：

按列宽column-width布局

按列数column-count布局

列之间的间距指定：column-gap

列之间的分隔线：column-rule:1px solid #000000;

合并列：column-span: all/values

目前此属性主流浏览器都不支持。

**浏览器支持**

firefox Safari 3

|  |
| --- |
| #columns {  -webkit-column-count: 3;  -webkit-column-gap: 25px;  -moz-column-count: 3;  -moz-column-gap: 25px;  column-count: 3;  column-gap: 25px; }  <div id="columns">  <p>Column A</p>  <p>Column B</p>  <p>Column C</p> </div> |

1. **box-shadow图片阴影效果**

**box-shadow:10px 10px 5px #888;**

1. **属性值**

box-shadow: 水平阴影-垂直阴影-模糊距离－阴影尺寸-阴影颜色－内部阴影/外部阴影

box-shadow: h-shadow v-shadow blur spread color inset/outset;

水平阴影：正值向右，负值向左；

垂直阴影：正值向下，负值向上；

模糊距离：数值越大，模糊度越高，0为无模糊

1. **省略写法**

**box-shadow:10px 10px 5px #888;**

像这样，如果只写三个值，第三个表示‘模糊距离’

1. **必填**

除了 水平阴影 和 垂直阴影 是必填之外，其它项可选；

1. **多个阴影**

box-shadow 可以向框添加一个或多个阴影，多个阴影之间用逗号分隔，如：

box-shadow: 0px 0px 3px #ccc, 0 10px 15px #eee inset;

1. **user-select**

target\_element{-webkit-user-select:none;}

定义是否选中元素中的内容

1. **appearance**

定义元素的外观样式

|  |  |
| --- | --- |
| **值** | **描述** |
| none |  |
| normal | 常规显示 |
| button | 显示为按钮 |
| push-button | 显示为按钮 |
| hyperlink | 显示为超链接 |
| radio | 显示为单选按钮 |
| checkbox | 显示为复选框 |
| pop-up-menu | 显示为单选下拉列表 |
| list-menu | 显示多选列表 |
| radio-group | 显示为单选按钮组 |
| checkbox-group | 显示为复选框组 |
| field | 显示多行文本框 |
| 显示为密码输入框 |  |

当appearance的值为none时，可以通过定义背景图片的方式，为元素定义各种外观，以单选按钮为例：

|  |
| --- |
| input[type=radio]{  -webkit-appearance: none;  appearance: none;  width: 18px;  height: 18px;  margin: 0;  cursor: pointer;  vertical-align: bottom;  background:url(/user/images/radio.png) no-repeat;  border:0;  -webkit-border-radius: 9px;  -webkit-box-sizing: border-box;  box-sizing: border-box;  }  input[type=radio]:checked{  background:url(/user/images/radio\_checked.png) no-repeat;  } |

**浏览器支持：**

safari　chrome Firefox

1. **color**
2. **HSL Colors**

background-color:(Hue,Saturation,Lightness);

* Hue色值:

范围：0-360； 0/360-red 120-green 240-blue

* Saturation饱和度:

100% is the full colour

* Lightness亮度:

0% is dark (black), 100% is light (white), and 50% is the average.

* 浏览器支持

Opera 9.5, Safari 3, Konqueror and Mozilla

例子：

background-color:hsl(120,100%, 50%);

1. **HSLA Colors**

background-color:(Hue,Saturation,Lightness,opacity);

opacity=0为完全透明,opacity=1为完全不透明

**浏览器支持：**

safari 3 and the Firefox 3 beta

1. **Colors**

color: hsla( 112, 72%, 33%, 0.68);

color: rgba(0, 0, 0, 0.75);

background: rgba(0, 0, 255, 0.75);

1. **box-sizing**
2. **box-sizing: border-box;**

border-box可以强制改变CSS2的盒模型的构成。如果指定了box-sizing,那么CSS中所设置的width height实际上是box所占的实际范围，即padding border被包含在width/height中了：

width/height = content width/height + padding + border

也就是padding border所占空间变，但是content区域的空间被缩小了。

1. **box-sizing: content-box;**

content-box不会改变CSS2的盒模型的构成

1. **用途**

平时布局时，如果设置好了，要想再改变下padding border等值时，就不得不修改width/height以免布局被破坏，但是如果设置成box-sizing: border-box;就不用考虑这种情况了，还有就是设置百分比的情况下，如果设置了padding/border，那么width/height的百分比值就得低于100%，这样很不精确，同样如果设置了box-sizing: border-box;这个问题也迎刃而解了。

**浏览器支持：**

Safari FF Opera

1. **resize**

resize:both/horizontal/vertical

both:宽高均可调节

horizontal：只可调节宽度

vertical：只可调节高度

为了防止破坏布局，可以为其设置min/max-width/height.

|  |
| --- |
| div.resize { width: 100px; height: 100px; border: 1px solid #000000; resize: both; overflow: auto; } |

**浏览器支持：**

safari chrome 5.0

1. **outline**

outline,outline-offset都不会占据空间；

outline-offset:偏移border的距离

outline:1px solid #cccccc;

outline-offset:10px;

当outline-offset的值为0时，outline会紧贴border.

**浏览器支持：**

Opera, Safari and Firefox

1. **Media Queries**

|  |
| --- |
| @media all and (min-width: 640px) {  #media-queries-1 { background-color: #0f0; }  }  @media screen and (max-width: 2000px) {  #media-queries-2 { background-color: #0f0; }  } |

<http://www.css3.info/preview/>