### **1.基础的选择器**

|  |  |  |
| --- | --- | --- |
| 选择器 | 含义 | 示例 |
| \* | 通用元素选择器，匹配任何元素 | \* { margin:0; padding:0; } |
| E | 标签选择器，匹配所有使用E标签的元素 | p { font-size:2em; } |
| .info和E.info | class选择器，匹配所有class属性中包含info的元素 | .info { background:#ff0; }  p.info { background:#ff0; } |
| #info和E#info | id选择器，匹配所有id属性等于footer的元素 | #info { background:#ff0; }  p#info { background:#ff0; } |

### **2.组合选择器**

|  |  |  |
| --- | --- | --- |
| 选择器 | 含义 | 示例 |
| E,F | 多元素选择器，同时匹配所有E元素或F元素，E和F之间用逗号分隔 | Div,p { color:#f00; } |
| E F | 后代元素选择器，匹配所有属于E元素后代的F元素，E和F之间用空格分隔 | #nav li { display:inline; }  li a { font-weight:bold; } |
| E > F | 子元素选择器，匹配所有E元素的子元素F | div > strong { color:#f00; } |
| E + F | 毗邻元素选择器，匹配所有紧随E元素之后的同级元素F | p + p { color:#f00; } |

### **3.CSS 2.1 属性选择器**

|  |  |  |
| --- | --- | --- |
| 选择器 | 含义 | 示例 |
| E[att] | 匹配所有具有att属性的E元素，不考虑它的值。（注意：E在此处可以省略，比如“[cheacked]”。以下同。） | p[title] { color:#f00; } |
| E[att=val] | 匹配所有att属性等于“val”的E元素 | div[class=”error”] { color:#f00; } |
| E[att~=val] | 匹配所有att属性具有多个空格分隔的值、其中一个值等于“val”的E元素 | td[class~=”name”] { color:#f00; } |
| E[att|=val] | 匹配所有att属性具有多个连字号分隔（hyphen-separated）的值、其中一个值以“val”开头的E元素，主要用于lang属性，比如“en”、“en-us”、“en-gb”等等 | p[lang|=en] { color:#f00; } |

**注：CSS 2.1 属性选择器还有一个特点就是使用多个选择器，同事满足这多个选择器：**blockquote[class=quote][cite] { color:#f00; }

### **4.CSS 2.1 中的伪类**

|  |  |  |
| --- | --- | --- |
| 选择器 | 含义 | 示例 |
| E:first-child | 匹配父元素的第一个子元素 | p:first-child { font-style:italic; }  input[type=text]:focus { color:#000; background:#ffe; }  input[type=text]:focus:hover { background:#fff; }  q:lang(sv) { quotes: “\201D” “\201D” “\2019″ “\2019″; } |
| E:link | 匹配所有未被点击的链接 |
| E:visited | 匹配所有已被点击的链接 |
| E:active | 匹配鼠标已经其上按下、还没有释放的E元素 |
| E:hover | 匹配鼠标悬停其上的E元素 |
| E:focus | 匹配获得当前焦点的E元素 |
|  |  |

### **5.CSS 2.1中的伪元素**

|  |  |  |
| --- | --- | --- |
| 选择器 | 含义 | 示例 |
| E:first-line | 匹配E元素的第一行 | p:first-line { font-weight:bold; color;#600; }  .preamble:first-letter { font-size:1.5em; font-weight:bold; }  .cbb:before { content:”"; display:block; height:17px; width:18px; background:url(top.png) no-repeat 0 0; margin:0 0 0 -18px; }  a:link:after { content: ” (” attr(href) “) “; } |
| E:first-letter | 匹配E元素的第一个字母 |
| E:before | 在E元素之前插入生成的内容 |
| E:after | 在E元素之后插入生成的内容 |

### **6.CSS 3的同级元素通用选择器**

|  |  |  |
| --- | --- | --- |
| 选择器 | 含义 | 示例 |
| E ~ F | 匹配任何在E元素之后的同级F元素 | p ~ ul { background:#ff0; } |

### **7．CSS 3 属性选择器**

|  |  |  |
| --- | --- | --- |
| 选择器 | 含义 | 示例 |
| E[att^=”val”] | 属性att的值以”val”开头的元素 | div[id^="nav"] { background:#ff0; } |
| E[att$=”val”] | 属性att的值以”val”结尾的元素 |
| E[att\*=”val”] | 属性att的值包含”val”字符串的元素 |

### **8. CSS 3中与用户界面有关的伪类**

|  |  |  |
| --- | --- | --- |
| 选择器 | 含义 | 示例 |
| E:enabled | 匹配表单中激活的元素 | input[type="text"]:disabled { background:#ddd;} |
| E:disabled | 匹配表单中禁用的元素 |
| E:checked | 匹配表单中被选中的radio（单选框）或checkbox（复选框）元素 |
| E::selection | 匹配用户当前选中的元素 |

### **9. CSS 3中的结构性伪类**

|  |  |  |
| --- | --- | --- |
| 选择器 | 含义 | 示例 |
| E:root | 匹配文档的根元素，对于HTML文档，就是HTML元素 | p:nth-child(3) { color:#f00; }  p:nth-child(odd) { color:#f00; }  p:nth-child(even) { color:#f00; }  p:nth-child(3n+0) { color:#f00; }  p:nth-child(3n) { color:#f00; }  tr:nth-child(2n+11) { background:#ff0; }  tr:nth-last-child(2) { background:#ff0; }  p:last-child { background:#ff0; }  p:only-child { background:#ff0; }  p:empty { background:#ff0; } |
| E:nth-child(n) | 匹配其父元素的第n个子元素，第一个编号为1 |
| E:nth-last-child(n) | 匹配其父元素的倒数第n个子元素，第一个编号为1 |
| E:nth-of-type(n) | 与:nth-child()作用类似，但是仅匹配使用同种标签的元素 |
| E:nth-last-of-type(n) | 与:nth-last-child() 作用类似，但是仅匹配使用同种标签的元素 |
| E:last-child | 匹配父元素的最后一个子元素，等同于:nth-last-child(1) |
| E:first-of-type | 匹配父元素下使用同种标签的第一个子元素，等同于:nth-of-type(1) |
| E:last-of-type | 匹配父元素下使用同种标签的最后一个子元素，等同于:nth-last-of-type(1) |
| E:only-child | 匹配父元素下仅有的一个子元素，等同于:first-child:last-child或 :nth-child(1):nth-last-child(1) |
| E:only-of-type | 匹配父元素下使用同种标签的唯一一个子元素，等同于:first-of-type:last-of-type或 :nth-of-type(1):nth-last-of-type(1) |
| E:empty | 匹配一个不包含任何子元素的元素，注意，文本节点也被看作子元素 |