



CSS 基础

- 👉 标签
- 空
- + 新增属性

css用于设置html 标签的样式，以构建美观的web 页面。



html

标签内部：

```
1 <div style="color: #000000; "></div>
```

HTML

style 标签：

```
1 <style type="text/css">
2     div {
3         color: #000000;
4     }
5 </style>
```

HTML

引入 css 文件：

```
1 <head>
2     <link rel="stylesheet" type="text/css" href="./mystyle.css">
3 </head>
```

HTML

选择器

选择器即标签选择，用于指定要对哪一个/哪些标签进行样式设置。

基本选择器

假设有如下标签：

```
1 <div class="classValue" id="idValue"> 基本选择器 </div>
```

HTML

可用下面三种指定：

```
1 /* class选择器：推荐 */
```

CSS

```
2 .classValue { }
3
4 /* id选择器 (id值唯一, 且id属性只能1个值)*/
5 #idValue { }
6
7 /* 标签选择器*/
8 div { }
```

css中选择器会将所有符合条件的标签都进行样式设置, 如div{ } 会设置所有div标签的样式。将所有标签都设置单独的class/id是很麻烦的, 因此css中还有很多丰富的选择器来指定具体的标签。

属性选择器

标签的某个属性值符合条件, 则会被选中。

```
1 /* 定位所有含class属性的元素 */
2 *[class]{ }
3
4 /* 定位含class属性的div元素 */
5 div[class]{ }
6
7 /* 定位同时含class和title属性的div元素 */
8 div[class][title] { }
9
10 /* 根据属性值选择 */
11 div[value="1"] { }
12
13 /* 文本值搜索匹配: ^=开头、$=结尾 *= 包含文本 ~=包含文本词 */
14 div[title~="a"] { }
```

CSS

后代选择器

后代选择器, 即根据父子标签的关系来进行匹配。

```
1 <div class="classValue">
2   <div>
3     <p> 后代选择器 </p>
4   </div>
5 </div>
```

HTML

可通过如下方式选择:

```
1 /* 子选择器: 只能逐级定位 */
2 .classValue > div > p { }
3
4 /* 后代选择器: 允许跳级定位 */
5 .classValue p{ }
```

CSS

兄弟选择器

在同一个父标签下, 选择一个子标签进行样式设置。

```
1 <div class="classValue">
2   <div>子节点1</div>
3   <div>子节点2</div>
4   <div>子节点3</div>
5 </div>
```

HTML

首节点选择器: 选择第一个子div节点

```
1 /* 第一个指定标签节点 */
2 div:first-of-type { }
3
4 /* 第一个节点, 不区分标签 */
5 div:first-child { }
```

CSS

```
6
7
```

兄弟选择器：选择下一个子 div 节点

```
1  div + div { }
```

CSS

子选择器：选择指定节点

```
1  /* 获取指定位置节点，不区分标签：下面1表示第一个 */
2  div:nth-child(1) { }
3
4  /* 获取指定标签节点 */
5  div:nth-of-type(1) { }
```

CSS

深度选择器

在组件化开发时，使用 `scoped` 指定局部样式可防止组件之间样式产生影响。但父组件有时需要对子组件样式进行修改，此时就可通过深度选择器来选择子组件的标签进行设置。

深度选择器符号为 `>>>`，另外在一些场景下它还有两个别名 `::v-deep` 和 `/deep/`。

```
1  >>> div { }
```

CSS

伪元素选择器

```
1  div:before { }
2
3  div:after{ }
```

CSS

尺寸布局

元素尺寸

固定尺寸

```
1  div{
2      width: 100px; /* 宽 */
3      height: 100px; /* 高 */
4  }
```

CSS

动态尺寸

```
1  div{
2      max-width: 100px; /* 最大宽 */
3      max-height: 100px; /* 最大高 */
4
5      min-width: 100px; /* 最大宽 */
6      min-height: 100px; /* 最大高 */
7
8      display: inline-block; /* 随内容撑大 */
9  }
```

CSS

元素定位 | position

相对定位：相对于自身起点移动，占用空间保持，会覆盖其他元素。

```
1  div{
2      position: relative;
3  }
```

CSS

```
4 }
```

绝对定位：在文档流之外，不占用位置，会覆盖其他元素。

```
1  div{
2    position: absolute;
3    /* 靠右 */
4    right: 0px;
5  }
```

CSS

浮动：靠左或靠右，超出父元素边界会换行。

```
1  div{
2    /*left、right*/
3    float: right;
4  }
```

CSS

块、行级元素 | display

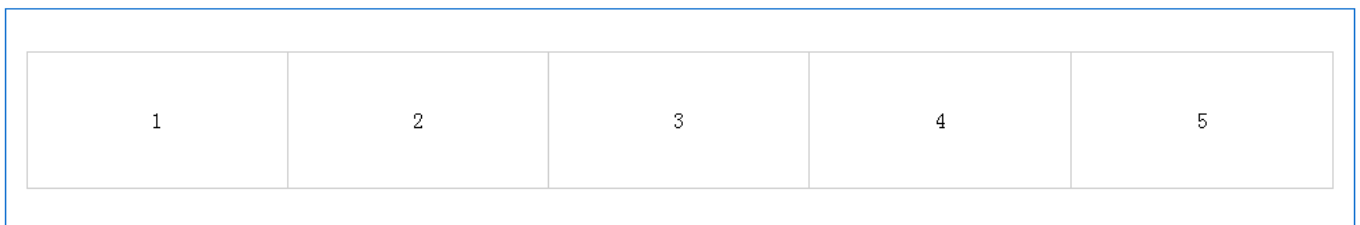
块级元素：占据一定矩形空间，可以通过设置高度、宽度、内外边距等属性，来调整的这个矩形。

行级元素：自己的独立空间，它是依附于其他块级元素存在的，因此，对行内元素设置高度、宽度、内外边距等属性，都是无效的。

每一个元素都有默认的display属性值（可以通过修改此值来强制改变元素块行级别），其值如下：

```
1  div {
2    display: block; /* div默认，块级元素 */
3
4    display: inline;      /* 行级元素，整体尺寸随内容尺寸变化，span默认 */
5    display: inline-block; /* 行内块元素，整体尺寸随内容尺寸变化，且可以设置宽高，总宽超出父元素会折行 */
6
7    display: table;       /* 表格元素 */
8    display: table-cell;  /* 单元格元素，可以使多个div，不会折行，并且每格高度会保持一致 */
9
10   display: none; /* 隐藏元素，不保留元素原本占用的空间 */
11 }
```

CSS



为上图父容器设置 table，小块1到5设置 table-cell 后，会有如下特点：

- 无论 table-cell 小块有几个，都不会产生折行。
- table-cell 小块高度会随其它小块高度变化而变化，从而高度对齐。
- 如果只有一个 table-cell 小块，小块会**垂直居中**于父容器。

弹性布局 Flex

参考地址：<https://www.runoob.com/w3cnote/flex-grammar.html>

元素设置

伪类

<https://www.runoob.com/css/css-pseudo-classes.html>

伪元素

伪元素相当于在指定元素后方添加了一个新元素，但实际未添加：

CSS

```
1  div:after {
2      width: 100px;
3      clear: both; /*清除浮动*/
4      content: ""; /*元素内容*/
5  }
6
```

案例：利用伪元素在文本前方添加圆点

CSS

```
.p {
    position: relative;
    padding-left: 20px;
    width: 100px;
}
.p:before {
    position: absolute;
    left: 20px;
    top: 2px;
    content: '';
    width: 10px;
    height: 10px;
    vertical-align: middle;
    background: #40aafa;
    border-radius: 20px;
}
```

文本溢出 Overflow

overflow 属性用于控制内容溢出元素框时显示的方式：

CSS

```
div {
    overflow: visible; /* 超出依然显示，默认值 */
    overflow: hidden; /* 超出部分隐藏 */
    overflow: scroll; /* 超出部分隐藏，但可以通过滚动条查看，和auto一致 */
    overflow: inherit; /* 继承父元素设置 */
}
```

样式值设置

css 变量

全局变量

定义：

CSS

```
1  :root{
2      --a: 20px;
3      --b: 40px;
4  }
```

使用：

CSS

```
1  div{
2      height: var(--a);
3      width: var(--b);
4  }
```

局部变量

CSS

```
1  div{
2    --w: 20px;
3    width: var(--w);
4    height: var(--w);
5  }
```

渐变颜色

```
.div {
  width:100%;
  height:600px;
  FILTER: progid:DXImageTransform.Microsoft.Gradient(gradientType=0,startColorStr=#b8c4cb,endColorStr=red); /* IE 6 7 8 */
  background: -ms-linear-gradient(top, #fff, #0000ff); /* IE 10 */
  background: -moz-linear-gradient(top, #b8c4cb, #f6f6f8); /* 火狐 */
  background: -webkit-gradient(linear, 0% 0%, 0% 100%, from(#b8c4cb), to(#f6f6f8)); /* 谷歌 */
  background: -webkit-gradient(linear, 0% 0%, 0% 100%, from(#fff), to(#0000ff)); /* Safari 4-5, Chrome 1-9 */
  background: -webkit-linear-gradient(top, #fff, #0000ff); /* Safari5.1 Chrome 10+ */
  background: -o-linear-gradient(top, #fff, #0000ff); /* Opera 11.10+ */
}
```

CSS

鼠标样式

悬浮光标

```
1  <div style="cursor:pointer;"></div>
```

HTML

cursor 取值

```
1  1、default 默认光标(通常是一个箭头)
2
3  2、auto 默认。浏览器设置的光标。
4
5  3、crosshair 光标呈现为十字线。
6
7  4、pointer 光标呈现为指示链接的指针(一只手)
8
9  5、move 此光标指示某对象可被移动。
10
11 6、e-resize 此光标指示矩形框的边缘可被向右(东)移动。
12
13 7、ne-resize 此光标指示矩形框的边缘可被向上及向右移动(北东)。
14
15 8、nw-resize 此光标指示矩形框的边缘可被向上及向左移动(北西)。
16
17 9、n-resize 此光标指示矩形框的边缘可被向上(北)移动。
18
19 10、se-resize 此光标指示矩形框的边缘可被向下及向右移动(南东)。
20
21 11、sw-resize 此光标指示矩形框的边缘可被向下及向左移动(南西)。
22
23 12、s-resize 此光标指示矩形框的边缘可被向下移动(北西)。
24
25 13、w-resize 此光标指示矩形框的边缘可被向左移动(西)。
26
27 14、text 此光标指示文本。
28
29 15、wait 此光标指示程序正忙(通常是一只表或沙漏)。
30
31 16、help 此光标指示可用的帮助(通常是一个问号或一个气球)。
```

.properties

部分样式

```
div {
  contenteditable: true; /* 使div变为输入框 */
}
```

CSS

语法介绍

单位介绍

- px: 绝对单位, 页面按精确像素展示
 - em: 相对单位, 基准点为父节点字体的大小, 如果自身定义了 font-size 按自身来计算; em 不是一个固定的值。可以设置 body {font-size: 62.5%}
 - rem: 相对单位, 可理解为”root em”, 相对根节点 html 的字体大小来计算, CSS3 新加属性, chrome/firefox/IE9+ 支持。
 - vw: viewpoint width, 视窗宽度, 1vw 等于视窗宽度的 1%。
 - vh: viewpoint height, 视窗高度, 1vh 等于视窗高度的 1%。
- (vmin: vw 和 vh 中较小的那个。vmax: vw 和 vh 中较大的那个。)

【注意】vw, vh, vmin, vmax: IE9+ 局部支持, chrome/firefox/safari/opera 支持, ios safari 8+ 支持, android browser4.4+ 支持, chrome for android39 支持

- in: 寸
- %: 百分比
- cm: 厘米
- mm: 毫米
- ex: 取当前作用效果的字体的 x 的高度, 在无法确定 x 高度的情况下以 0.5em 计算 (IE11 及以下均不支持, firefox/chrome/safari/opera/ios safari/android browser4.4+ 等均需类内属性前缀)
- ch: 以节点所使用字体中的”0”字符为基准, 找不到时为 0.5em (ie10+, chrome31+, safari7.1+, opera26+, ios safari 7.1+, android browser4.4+ 支持)