CSS 专业技巧

2018-02-19 css-protips [JavaScript之禅](http://mp.weixin.qq.com/s?__biz=MzU0OTE3MjE1Mw==&mid=2247483924&idx=1&sn=a6382cabf1f0147f99a91ea3a1a14840&chksm=fbb2a4c2ccc52dd4752c37c57a04b895130f672bdb5bff44838d07a61aa11a26944ebbd8aeb1&mpshare=1&scene=23&srcid=0311U0q4mJFEVZiK4BRNWA3t##)

ps：清晨先来一首好听的音乐 点击上方关注我们☝

早去早回陈洁仪 - 最好http://res.wx.qq.com/mmbizwap/zh_CN/htmledition/images/icon/appmsg/qqmusic/icon_qqmusic_source393e3a.png

一个帮你提升 CSS 技巧的收藏集

**专业技巧**

1. 使用CSS复位
2. 继承 box-sizing
3. 使用 :not() 选择器来决定表单是否显示边框
4. 为 body 元素添加行高
5. 垂直居中任何元素
6. 逗号分隔的列表
7. 使用负的 nth-child 来选择元素
8. 使用 SVG 图标
9. 使用 “形似猫头鹰” 的选择器
10. 使用 max-height 来建立纯 CSS 的滑块
11. 创造格子等宽的表格
12. 利用 Flexbox 去除多余的外边距
13. 利用属性选择器来选择空链接
14. 给 “默认” 链接定义样式
15. 一致的垂直节奏
16. 内在比例盒
17. 为破碎图象定义样式
18. 用 rem 来调整全局大小；用 em 来调整局部大小
19. 隐藏没有静音、自动播放的影片
20. 使用选择器 :root 来控制字体弹性
21. 为更好的移动体验，为表单元素设置字体大小

**1. 使用CSS复位**

CSS复位可以在不同的浏览器上保持一致的样式风格。您可以使用CSS reset 库Normalize等，也可以使用一个更简化的复位方法：

1. \* {
2. box-sizing: border-box;
3. margin: 0;
4. padding: 0;
5. }

现在元素的 margin 和 padding 已为0， box-sizing可以管理您的CSS盒模型布局。

注意：如果你遵循接下来继承 box-sizing讲解的这个技巧, 你不需要在以上代码中添加 box-sizing 属性。

**2. 继承 box-sizing**

从 html 元素继承 box-sizing ：

1. html {
2. box-sizing: border-box;
3. }
4. \*, \*::before, \*::after {
5. box-sizing: inherit;
6. }

如此在插件或其它组件里改变 box-sizing 变得简单。

**3. 使用**:not()**选择器来决定表单是否显示边框**

先为元素添加边框

1. /\* 添加边框 \*/
2. .nav li {
3. border-right: 1px solid #666;
4. }

为最后一个元素去除边框

1. /\* 去掉边框 \*/
2. .nav li:last-child {
3. border-right: none;
4. }

不过不要这么做，使用 :not() 伪类来达到同样的效果：

1. .nav li:not(:last-child) {
2. border-right: 1px solid #666;
3. }

当然，你也可以使用 .nav li+li 或者 .nav li:first-child~li ，但是 :not() 更加清晰，具有可读性。

**4. 为**body**元素添加行高**

不必为每一个 <p>， <h\*> 元素逐一添加 line-height，直接添加到 body 元素：

1. body {
2. line-height: 1.5;
3. }

文本元素可以很容易地继承 body 的样式。

**5. 垂直居中任何元素**

不！这绝不是黑魔法，真的可以垂直居中任何元素：

1. html, body {
2. height: 100%;
3. margin: 0;
4. }
5. body {
6. -webkit-align-items: center;
7. -ms-flex-align: center;
8. align-items: center;
9. display: -webkit-flex;
10. display: flex;
11. }

这还不够？垂直方向，水平方向？任何元素，任何时间，任何地点？CSS-Tricks 有篇好文 讲到了各种居中的技巧。

**注意：** IE11 对 flexbox 的支持有点 bug。

**6. 逗号分隔列表**

使列表的每项都由逗号分隔：

1. ul > li:not(:last-child)::after {
2. content: ",";
3. }

因最后一项不加逗号，可以使用 :not() 伪类。

**注意：**这一技巧对于无障碍，特别是屏幕阅读器而言并不理想。而且复制粘贴并不会带走CSS生成的内容,需要注意。

**7. 使用负的**nth-child**来选择元素**

使用负的 nth-child 可以选择 1 至 n 个元素。

1. li {
2. display: none;
3. }
4. /\* 选择第 1 至第 3 个元素并显示出来 \*/
5. li:nth-child(-n+3) {
6. display: block;
7. }

或许你已经掌握了如何使用 :not()这个技巧，试下这个：

1. /\* 选择除前3个之外的所有项目，并显示它们 \*/
2. li:not(:nth-child(-n+3)) {
3. display: none;
4. }

如此简单！

**8. 使用 SVG 图标**

没有理由不使用 SVG 图标：

1. .logo {
2. background: url("logo.svg");
3. }

SVG 在所有分辨率下都可以良好缩放，并且支持所有 IE9 以后的浏览器，丢掉你的 .png, .jpg, 或 .gif-jif-whatev 文件吧。

**注意：** 针对仅有图标的按钮，如果 SVG 没有加载成功的话，以下样式对无障碍有所帮助：

1. .no-svg .icon-only::after {
2. content: attr(aria-label);
3. }

**9. 使用 “形似猫头鹰” 的选择器**

这个名字可能比较陌生，不过通用选择器 ( \*) 和 相邻兄弟选择器 ( +) 一起使用，效果非凡：

1. \* + \* {
2. margin-top: 1.5em;
3. }

在此示例中，文档流中的所有的相邻兄弟元素将都将设置 margin-top:1.5em 的样式。

更多 “形似猫头鹰” 的选择器，可参考 A List Apart 上面 Heydon Pickering 的文章

**10. 使用**max-height**来建立纯 CSS 的滑块**

max-height 与 overflow hidden 一起来建立纯 CSS 的滑块：

1. .slider {
2. max-height: 200px;
3. overflow-y: hidden;
4. width: 300px;
5. }
6. .slider:hover {
7. max-height: 600px;
8. overflow-y: scroll;
9. }

鼠标移入滑块元素时增大它的 max-height 值，便可以显示溢出部分。

**11. 创造格子等宽的表格**

table-layout:fixed 可以让每个格子保持等宽：

1. .calendar {
2. table-layout: fixed;
3. }

无痛的 table 布局。

**12. 利用 Flexbox 去除多余的外边距**

与其使用 nth-， first-， 和 last-child 去除列之间多余的间隙，不如使用 flexbox 的 space-between 属性：

1. .list {
2. display: flex;
3. justify-content: space-between;
4. }
5. .list .person {
6. flex-basis: 23%;
7. }

列之间的间隙总是均匀相等。

**13. 利用属性选择器来选择空链接**

当 <a> 元素没有文本内容，但有 href 属性的时候，显示它的 href 属性：

1. a[href^="http"]:empty::before {
2. content: attr(href);
3. }

相当简便。

**14. 给 “默认” 链接定义样式**

给 “默认” 链接定义样式：

1. a[href]:not([class]) {
2. color: #008000;
3. text-decoration: underline;
4. }

通过 CMS 系统插入的链接，通常没有 class 属性，以上样式可以甄别它们，而且不会影响其它样式。

**15. 一致垂直节奏**

通用选择器 ( \*) 跟元素一起使用，可以保持一致的垂直节奏：

1. .intro > \* {
2. margin-bottom: 1.25rem;
3. }

一致的垂直节奏可以提供视觉美感，增强内容的可读性。

**16. 固定比例盒子**

要创建具有固定比例的一个盒子，所有你需要做的就是给 div 设置一个 padding：

1. .container {
2. height: 0;
3. padding-bottom: 20%;
4. position: relative;
5. }
6. .container div {
7. border: 2px dashed #ddd;
8. height: 100%;
9. left: 0;
10. position: absolute;
11. top: 0;
12. width: 100%;
13. }

使用20％的padding-bottom使得框等于其宽度的20％的高度。与视口宽度无关，子元素的div将保持其宽高比（100％/ 20％= 5:1）。

**17. 为破碎图象定义样式**

只要一点CSS就可以美化破碎的图象：

1. img {
2. display: block;
3. font-family: Helvetica, Arial, sans-serif;
4. font-weight: 300;
5. height: auto;
6. line-height: 2;
7. position: relative;
8. text-align: center;
9. width: 100%;
10. }

以添加伪元素的法则来显示用户信息和URL的引用：

1. img::before {
2. content: "We're sorry, the image below is broken :(";
3. display: block;
4. margin-bottom: 10px;
5. }
6. img::after {
7. content: "(url: " attr(src) ")";
8. display: block;
9. font-size: 12px;
10. }

了解更多关于这类样式的技巧 Ire Aderinokun的 原帖.

**18. 用**rem**来调整全局大小；用**em**来调整局部大小**

在根元素设置基本字体大小后 ( html{font-size:100%;}), 使用 em 设置文本元素的字体大小:

1. h2 {
2. font-size: 2em;
3. }
4. p {
5. font-size: 1em;
6. }

然后设置模块的字体大小为 rem:

1. article {
2. font-size: 1.25rem;
3. }
4. aside .module {
5. font-size: .9rem;
6. }

现在，每个模块变得独立，更容易、灵活的样式便于维护。

**19. 隐藏没有静音、自动播放的影片**

这是一个自定义用户样式表的不错的技巧。避免在加载页面时自动播放。如果没有静音，则不显示视频：

1. video[autoplay]:not([muted]) {
2. display: none;
3. }

再次，我们利用了 :not() 的优点。

**20. 使用选择器**:root**来控制字体弹性**

在响应式布局中，字体大小应需要根据不同的视口进行调整。你可以计算字体大小根据视口高度的字体大小和宽度，这时需要用到 :root:

1. :root {
2. font-size: calc(1vw + 1vh + .5vmin);
3. }

现在，您可以使用 root em

1. body {
2. font: 1rem/1.6 sans-serif;
3. }

**21. 为更好的移动体验，为表单元素设置字体大小**

当触发 <select>的下拉列表时，为了避免表单元素在移动浏览器（IOS Safari 等等）上的缩放，加上 font-size：

1. input[type="text"],
2. input[type="number"],
3. select,
4. textarea {
5. font-size: 16px;
6. }

**## 支持情况**

这些技巧适用于最新版的 Chrome, Firefox, Safari, Opera, Edge, 以及 IE11。