

软件工具：第 2 部分

(COMS10012 / COMSM0085)

上周 (第 6 周) : HTTP 和 HTML

本周 (第 7 周) : CSS

下周 (第 8 周) : JS

CSS的

级联样式表（我们将讨论“级联”）。

定义 HTML 元素显示方式的一组规则。

在大多数情况下，规则是指 HTML 的视觉属性的设置 document 元素（更技术地说：从 HTML 派生的 DOM 对象）。

CSS组合

CSS 条目遵循相当简单的语法：

```
selector {  
    property : value;  
}
```

- `property` 是对此元素类型有意义的任何显示属性。
- `value` 允许的值将取决于属性。
- `selector` 定义可以应用此规则的元素。

CSS 实现

你的CSS规则应该放在一个样式表文档中，例如，`.mystyle.css`

然后，可以通过引用通知浏览器使用特定的样式表 在 HTML 文档中。

```
<html>
  <head>
    <link rel="stylesheet" href="mystyle.css"/>
  </head>
  <body>
    ...
```

您也可以指示浏览器将自定义样式表应用于 HTML 默认情况下为文档（“用户”或“自定义”样式）。但作为网页设计师，你 假设您的网站访问者希望按照您预期的方式查看网页 被查看。

CSS简单示例

```
p {
  color : red;
}
```

对于所有元素，将（font color）设置为。 p color red

选择

广泛的选择能力。

最简单的选择器：单个标签的名称（例如，，，）。适用于所有元素 那种。 p a div

下一个最简单的选择器是标签列表：

```
p, div, main {
  color : red;
}
```

选择器：类

用于仅选择具有特定属性的元素的语法： class

```
p.important {
  color: red;
}
```

```
<p class='important'>This is red</p>
<p>This is not</p>
```

选择器：类

也可以适用于任何元素类型: class

```
.important {
  color: red;
}
```

```
<p class='important'>This is red</p>
<p>This is not</p>
<span class='important'>But this is</span>
```

选择器：ID

如果要将样式应用于特定文档元素：

```
p#uniquebox {
  color: red;
}
```

```
<p id='uniquebox'>This is red</p>
<p class='uniquebox'>This is not</p>
<p>This certainly is not</p>
```

与类相同 - 本身将适用于? #uniquebox

选择器：属性

可以概括为按任何属性选择元素。

```
p[name=tim] {
  color:red;
}
div[border=none] {
```

```
color: blue;
}
```

`p[class='important']` 将与 `.p.important`

也可以做一些花哨的部分匹配，例如，选择 属性包含字符串“flower”的所有图像。

```
img[title~='flower'] title
```

选择器：位置

```
<div class="container">
  <p>direct child</p>
  <div>
    <p>descendant</p>
  </div>
</div>
<p>para one</p>
<p>para two</p>
```

- 后代是在任何级别上“内部”另一个元素的元素。
- 子元素是直接包含在父元素中的元素。
- 如果一个元素在文档的同一级别上的任何时间点出现，则该元素在另一个元素之前。
- 如果一个元素是文档该级别的下一个元素，则该元素紧随另一个元素之后。

选择器：位置

- `this that`（空格）：选择所有作为 `that` 的后代的元素。
- `this > that`：选择所有元素，这些元素是父项的直接子元素。
- `this ~ that`：选择前面有一个元素的所有元素。
- `this + that`：选择直接跟在元素后面的所有元素。

只是为了增加复杂性：所有规则都可以组合在一起。

```
div.important > p, h1#main, [title=nowred] ~ span {
  color: red;
}
```

值得一看的[参考指南](#)

级联?

哪条规则适用?

```
<p class='important'>If you want to pass this unit then...</p>
```

```
p {  
    font-size: 12pt;  
}
```

```
p.important {  
    color: red;  
}
```

值

可以设置许多不同的属性，这些属性需要不同的值 - 您需要浏览 [MDN 文档](#)以掌握所有选项。

但是，一些常见的元素与颜色和元素布局有关。

颜色值

以及，您可以设置 和 元素，例如。 color background-color border-color

- 已经看到， 和 .常见基本颜色的其他一些关键字。 red blue
- 还有十六进制格式，它接受 R、G 和 B 中每个从 00 到 FF 的值。 #rrggbb
- 也可以调用函数，RGB 的值为 1-255，A 的值为 0-1。 rgba(r,g,b,a)

red 并且是相同的。但还是会看起来“红色”。 #FF0000 #FF0001 #FF1111

布局

在页面上布局元素时，一个常见问题与处理空间有关 “围绕”元素（或元素之间）。

每个页面元素都可以被认为是一个包含多个层的“框”：

- **内容**是元素本身的原材料（例如，文本的空间，或图像的空间）。 <p>
- **填充**是内容和边框之间的空间。
- **边框**是“围绕”元素的一条（有时是不可见的）线，标记其边界。它可以有厚度。
- **边距**是边界外需要保持畅通的空间——其他元素不得侵入此空间。

在填充和边距之间混淆是很常见的（使边框可见会有所帮助）。

开发人员工具很好地直观地演示了这些值（请参阅本周的视频）。

布局值

两者兼而有之，可以针对单个侧面指定，也可以按顺时针顺序一起指定。 `margin padding`

```
margin-top: 10px;
margin-right: 20px;
margin-bottom: 10px;
margin-left: 5px;

margin: 10px 20px 10px 5px;
```

计量单位

在 CSS 中，有许多不同的方法可以指定度量。

“绝对”单位尝试生成特定的实际大小：

- 1 px - 1 个“像素”（但解释为：1/96 英寸）。
- 1 pt - 1 个“点”（1/72 英寸）
- 1 cm - 1 厘米（也 10 mm）
- 1 in - 1 寸

“相对”单位生成相对于视口或某些视口的尺寸 页面的 reference 元素。

- 1 vh - 视口高度的 1%（也用于宽度） 1 vw
- 1 em - 1 x 无论字体的大小（“m”的宽度）是多少。
- 1 ex - 1 x 无论 x 的高度是多少。
- 1 rem - 1 x 无论文档根元素的字体大小如何。
- 1 % - 父元素相应尺寸的 1%。

很容易对单位感到困惑。

设计很难

本单元试图教你对CSS的一些基本理解。

CSS可能很难调试和理解 - 技术问题。

但是，在非技术意义上，为真实网站成功设计样式也可能很困难。有一些关键原则（与人体工程学基础、观众期望等的联系），但从根本上说，很多事情都归结为品味、风格、时尚的问题——网页设计是一门艺术。

您可能会发现一些方便的概念：

- 基于网格的页面布局（本周实验的重点）
- 让设计人员创建可以应用的框架（也可在实验室中应用）
- 偷窃

来自其他网站的想法

本周练习

1. 阅读 MDN 文档。
2. 将基本 CSS 应用于 HTML 文档。
3. 对粉红色的线条感到非常沮丧。
4. 使用现有的 CSS 框架。
5. **阅读更多**MDN文档。
6. 使用网格布局。
7. 创建响应式布局。