

CSS 的工作原理

我们已经学习了 CSS 的基础知识、它的用途以及如何编写简单的样式表。在本课中，我们将了解浏览器如何使用 CSS 和 HTML 并将其转换为网页。

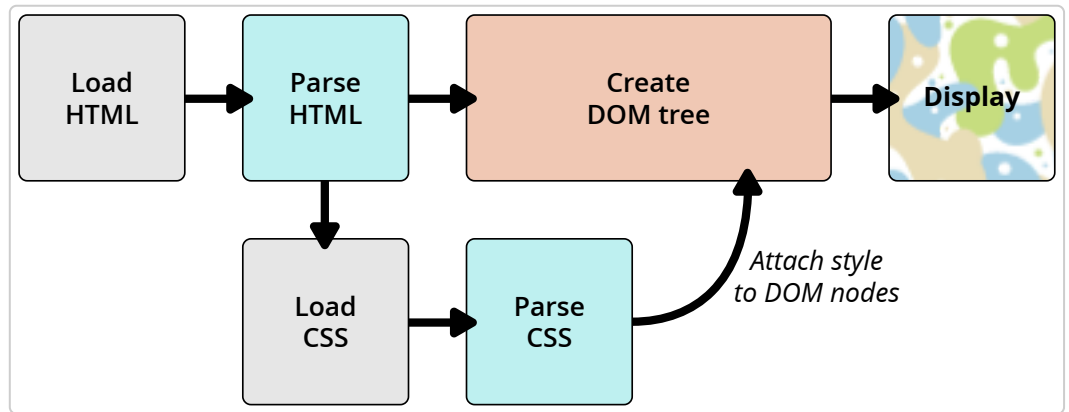
先决条件:	基本的计算机知识、 安装的基本软件 、 处理文件 的基本知识和 HTML 基础知识（学习 HTML 简介 。）
客观的:	了解浏览器如何解析 CSS 和 HTML 的基础知识，以及当浏览器遇到它不理解的 CSS 时会发生什么。

CSS实际上是如何工作的？

当浏览器显示文档时，它必须将文档的内容与其样式信息结合起来。它分多个阶段处理文档，我们在下面列出了这些阶段。请记住，这是浏览器加载网页时发生的情况的一个非常简化的版本，不同的浏览器将以不同的方式处理该过程。但这大致就是发生的事情。

1. 浏览器加载 HTML（例如从网络接收它）。
2. 它将[HTML](#)转换为[DOM](#)（文档对象模型）。DOM 表示计算机内存中的文档。下一节将更详细地解释 DOM。
3. 然后浏览器获取大部分由 HTML 文档链接到的资源，例如嵌入的图像、视频，甚至链接的 CSS！JavaScript 在这个过程中稍后处理，为了简单起见，我们不会在这里谈论它。
4. 浏览器解析获取的 CSS，并根据选择器类型将不同的规则分类到不同的“桶”中，例如元素、类、ID 等。基于它找到的选择器，它计算出哪些规则应该应用于 DOM 中的哪些节点，并根据需要将样式附加到它们（这个中间步骤称为渲染树）。
5. 渲染树按照规则应用后应出现的结构进行布局。
6. 页面的视觉显示显示在屏幕上（这个阶段称为绘画）。

下图还提供了该过程的简单视图。



关于 DOM

DOM 具有树状结构。标记语言中的每个元素、属性和一段文本都成为树结构中的一个 [DOM 节点](#)。节点由它们与其他 DOM 节点的关系定义。有些元素是子节点的父节点，子节点有兄弟节点。

了解 DOM 有助于您设计、调试和维护 CSS，因为 DOM 是您的 CSS 和文档内容相遇的地方。当您开始使用浏览器 DevTools 时，您将在选择项目时浏览 DOM，以查看适用的规则。

真实的 DOM 表示

与其进行冗长乏味的解释，不如让我们看一个示例，了解如何将真实的 HTML 片段转换为 DOM。

采用以下 HTML 代码：

```
<p>
  Let's use:
  <span>Cascading</span>
  <span>Style</span>
  <span>Sheets</span>
</p>
```

在 DOM 中，对应于我们 `<p>` 元素的节点是父节点。它的子节点是一个文本节点和对应于我们的元素的三个节点 ``。这些 SPAN 节点也是父节点，文本节点是它们的子节点：

```
P
├ "Let's use:"
├ SPAN
│   └ "Cascading"
├ SPAN
│   └ "Style"
└ SPAN
    └ "Sheets"
```

这就是浏览器如何解释前面的 HTML 片段——它呈现上面的 DOM 树，然后像这样在浏览器中输出它：

Let's use: Cascading Style Sheets

将 CSS 应用于 DOM

假设我们在文档中添加了一些 CSS 来设置样式。同样，HTML 如下所示：

```
<p>
  Let's use:
  <span>Cascading</span>
  <span>Style</span>
  <span>Sheets</span>
</p>
```

假设我们对其应用以下 CSS：

```
span {
  border: 1px solid black;
  background-color: lime;
}
```

浏览器解析 HTML 并从中创建 DOM。接下来，它解析 CSS。由于 CSS 中唯一可用的规则有一个 span 选择器，因此浏览器可以非常快速地对 CSS 进行排序！它将该规则应用于三个中的每一个 ，然后将最终的视觉表示绘制到屏幕上。

更新后的输出如下：

Let's use: `Cascading` `Style` `Sheets`

在我们下一个模块[的调试 CSS](#)文章中，我们将使用浏览器 DevTools 调试 CSS 问题，并将了解有关浏览器如何解释 CSS 的更多信息。

如果浏览器遇到它不理解的 CSS 会怎样？

[《什么是CSS》一文中的“浏览器支持信息”部分](#)提到，浏览器不一定同时实现新的CSS特性。此外，许多人没有使用最新版本的浏览器。鉴于 CSS 一直在发展，因此领先于浏览器可以识别的内容，您可能想知道如果浏览器遇到它无法识别的 CSS 选择器或声明会发生什么。

答案是它什么都不做，只是继续处理下一个 CSS！

如果浏览器正在解析您的规则，并遇到它不理解的属性或值，它会忽略它并继续下一个声明。如果您犯了错误并拼错了属性或值，或者属性或值太新而浏览器尚不支持，它将执行此操作。

同样，如果浏览器遇到一个它不理解的选择器，它会忽略整个规则并转到下一个。

在下面的示例中，我对颜色使用了英式英语拼写，这使得该属性无效，因为它无法被识别。所以我的段落没有被涂成蓝色。然而，所有其他 CSS 都已应用；仅忽略无效行。

```
<p>I want this text to be large, bold and blue.</p>
```

```
p {  
  font-weight: bold;  
  colour: blue; /* incorrect spelling of the color property */
```

```
font-size: 200%;  
}
```

I want this text to be
large, bold and blue.



使用 Kore.ai 无代码
对话式 AI 平台免费构
建智能虚拟助手。

[Mozilla 广告](#)

不想看广告？

这种行为非常有用。这意味着您可以使用新的 CSS 作为增强功能，知道如
果不理解也不会发生错误——浏览器要么获得新功能，要么不获得。这启用了
基本的回退样式。

当您想要使用一个非常新且并非在所有地方都受支持的值时，此方法特别有
效。例如，某些较旧的浏览器不支持 `calc()` 作为值。我可能会以像素为单
位给出框的总宽度，然后继续给出 `calc()` 值为 `100% - 50px`。旧浏览器



中。

```
.box {  
  width: 500px;  
  width: calc(100% - 50px);  
}
```

在后面的课程中，我们将研究更多支持各种浏览器的方法。

概括

您几乎完成了这个模块——我们只有一件事要做。在为[传记页面设计样式评
估](#)中，您将使用新知识重新设计示例样式，并在此过程中测试一些 CSS。

此页面最后修改于 2023 年 2 月 23 日由[MDN 贡献者](#)提供。

