

开始使用 CSS

在本文中，我们将获取一个简单的 HTML 文档并对其应用 CSS，并在此过程中学习有关该语言的一些实用知识。

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:	To understand the basics of linking a CSS document to an HTML file, and be able to do simple text formatting with CSS.

Starting with some HTML

Our starting point is an HTML document. You can copy the code from below if you want to work on your own computer. Save the code below as `index.html` in a folder on your machine.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <title>Getting started with CSS</title>
  </head>

  <body>
    <h1>I am a level one heading</h1>

    <p>
      This is a paragraph of text. In the text is a
      <span>span element</span> and also a
      <a href="https://example.com">link</a>.
    </p>

    <p>
```

```
    This is the second paragraph. It contains an
    <em>emphasized</em> element.
  </p>

  <ul>
    <li>Item <span>one</span></li>
    <li>Item two</li>
    <li>Item <em>three</em></li>
  </ul>
</body>
</html>
```

Note: If you are reading this on a device or an environment where you can't easily create files, then don't worry — live code editors are provided below to allow you to write example code right here in the page.

Adding CSS to our document

The very first thing we need to do is to tell the HTML document that we have some CSS rules we want it to use. There are three different ways to apply CSS to an HTML document that you'll commonly come across, however, for now, we will look at the most usual and useful way of doing so — linking CSS from the head of your document.

Create a file in the same folder as your HTML document and save it as `styles.css`. The `.css` extension shows that this is a CSS file.

To link `styles.css` to `index.html`, add the following line somewhere inside the `<head>` of the HTML document:

```
<link rel="stylesheet" href="styles.css" />
```

This `<link>` element tells the browser that we have a stylesheet, using the `rel` attribute, and the location of that stylesheet as the value of the `href` attribute. You can test that the CSS works by

adding a rule to `styles.css` . Using your code editor, add the following to your CSS file:

```
h1 {  
  color: red;  
}
```

Save your HTML and CSS files and reload the page in a web browser. The level one heading at the top of the document should now be red. If that happens, congratulations — you have successfully applied some CSS to an HTML document. If that doesn't happen, carefully check that you've typed everything correctly.

You can continue to work in `styles.css` locally, or you can use our interactive editor below to continue with this tutorial. The interactive editor acts as if the CSS in the first panel is linked to the HTML document, just as we have with our document above.

Styling HTML elements

By making our heading red, we have already demonstrated that we can target and style an HTML element. We do this by targeting an *element selector* — this is a selector that directly matches an HTML element name. To target all paragraphs in the document, you would use the selector `p` . To turn all paragraphs green, you would use:

```
p {  
  color: green;  
}
```

You can target multiple selectors at the same time by separating the selectors with a comma. If you want all paragraphs and all list items to be green, your rule would look like this:

```
p,  
li {
```

```
color: green;  
}
```

Try this out in the interactive editor below (edit the code boxes) or in your local CSS document.

I am a level one heading

This is a paragraph of text. In the text is a span element and also a [link](#).

This is the second paragraph. It contains an *emphasized* element.

- Item one
- Item two
- Item *three*

Interactive editor

```
h1 {  
  
}  
  
p {  
  
}
```

```
<h1>I am a level one heading</h1>  
  
<p>This is a paragraph of text. In  
the text is a <span>span  
element</span>  
and also a <a  
href="http://example.com">link</a>.  
</p>  
  
<p>This is the second paragraph. It  
contains an <em>emphasized</em>  
element.</p>  
  
<ul>  
  <li>Item one</li>  
  <li>Item two</li>
```

Changing the default behavior of elements

When we look at a well-marked up HTML document, even something as simple as our example, we can see how the browser is making the HTML readable by adding some default styling. Headings are large and bold and our list has bullets. This happens because browsers have internal stylesheets containing default styles, which they apply to all pages by default; without them all of the text would run together in a clump and we would have to style everything from scratch. All modern browsers display HTML content by default in pretty much the same way.

但是，您通常会想要浏览器做出的选择以外的东西。这可以通过选择要更改的 HTML 元素并使用 CSS 规则更改其外观来完成。一个很好的例子是 ``，一个无序列表。它有列表项目符号。如果你不想要这些项目符号，你可以像这样删除它们：

```
li {  
  list-style-type: none;  
}
```

现在尝试将其添加到您的 CSS 中。

该 `list-style-type` 属性是一个很好的属性，可以在 MDN 上查看以查看支持哪些值。查看页面 [list-style-type](#)，您会在页面顶部找到一个交互式示例以尝试一些不同的值，然后在页面下方详细说明所有允许的值。

查看该页面，您会发现除了删除列表项目符号之外，您还可以更改它们 — 尝试通过使用值将它们更改为方形项目符号 `square`。

添加一个类

到目前为止，我们已经根据元素的 HTML 元素名称设置了样式。只要您希望文档中该类型的所有元素看起来都相同，此方法就有效。要选择元素的子集而不更改其他元素，您可以将一个类添加到 HTML 元素并在 CSS 中定位该类。

1. 在您的 HTML 文档中，将[类属性](#)添加到第二个列表项。您的列表现在将如下所示：

```
<ul>
  <li>Item one</li>
  <li class="special">Item two</li>
  <li>Item <em>three</em></li>
</ul>
```

2. `special` 在您的 CSS 中，您可以通过创建以句号开头的选择器来定位类。将以下内容添加到您的 CSS 文件中：

```
.special {
  color: orange;
  font-weight: bold;
}
```

3. 保存并刷新以查看结果。

您可以将 `special` 类应用于 `special` 页面上您希望与此列表项具有相同外观的任何元素。例如，您可能希望 `` 段落中的也为橙色和粗体。尝试向其中添加一个 `class of`，然后重新加载您的页面，看看会发生什么。 `special`

有时您会看到带有选择器的规则，其中列出了 HTML 元素选择器和类：

```
li.special {
  color: orange;
  font-weight: bold;
}
```

此语法的意思是“定位任何 `li` 具有特殊类的元素”。如果您要这样做，那么您将无法再 `` 通过将类添加到元素来将类应用于一个或另一个元素；您必须将该元素添加到选择器列表中：

```
li.special,
span.special {
  color: orange;
  font-weight: bold;
}
```

正如您可以想象的那样，某些类可能会应用于许多元素，并且您不希望每次需要采用该样式的新内容时都必须继续编辑您的 CSS。因此，有时最好绕过元素并引用类，除非您知道要单独为一个元素创建一些特殊规则，并且可能要确保它们不应用于其他事物。

根据它们在文档中的位置设置样式

有时您会希望某些内容根据其在文档中的位置看起来有所不同。这里有许多选择器可以为您提供帮助，但现在我们只看几个。在我们的文档中，有两个 `` 元素——一个在段落中，另一个在列表项中。`` 要仅选择嵌套在元素内的 `an`，您可以使用称为**后代组合器**的 `` 选择器，它采用两个其他选择器之间的空格形式。

将以下规则添加到您的样式表中：

```
li em {  
  color: rebeccapurple;  
}
```

此选择器将选择 ``（的后代）内部的任何元素 ``。因此，在您的示例文档中，您应该发现 `` 第三个列表项中的现在是紫色的，但段落中的那个没有变化。

您可能想尝试的其他方法是在 HTML 中同一层次结构级别的标题后直接出现的段落样式化。为此，请在选择器之间放置一个 `+`（**相邻的兄弟组合器**）。

也尝试将此规则添加到您的样式表中：

```
h1 + p {  
  font-size: 200%;  
}
```

下面的实例包括上面的两个规则。尝试添加一条规则，使段落内的跨度变为红色。你会知道你是否做对了，因为第一段中的跨度是红色的，但第一个列表项中的跨度不会改变颜色。

I am a level one heading

This is a paragraph of text. In the text is a span element and also a [link](#).

This is the second paragraph. It contains an *emphasized* element.

- Item one
- Item two
- Item *three*

Interactive editor

```
li em {  
  color: rebeccapurple;  
}  
  
h1 + p {  
  font-size: 200%;  
  color: red  
}
```

```
<h1>I am a level one heading</h1>  
  
<p>This is a paragraph of text. In  
the text is a <span>span  
element</span>  
and also a <a  
href="http://example.com">link</a>.  
</p>  
  
<p>This is the second paragraph. It  
contains an <em>emphasized</em>  
element</p>
```

注意：如您所见，CSS 为我们提供了多种定位元素的方法，到目前为止我们只触及了皮毛！我们将在课程后面的[选择器](#)文章中仔细研究所有这些选择器以及更多选择器。

根据状态设计事物的样式

我们将在本教程中看到的最后一种样式是根据事物的状态来设计样式的能力。一个简单的例子就是样式化链接。当我们为链接设置样式时，我们需要以 [<a>](#) (anchor) 元素为目标。它具有不同的状态，具体取决于它是未访问、已访问、悬停、通过键盘聚焦还是正在被单击（激活）的过程中。您可以使用 CSS 来定位这些不同的状态——下面的 CSS 将未访问的链接设置为粉红色，将已访问的链接设置为绿色。

```
a:link {  
  color: pink;  
}  
  
a:visited {  
  color: green;  
}
```

您可以更改当用户将鼠标悬停在链接上时链接的外观，例如通过删除下划线，这是通过下一条规则实现的：

```
a:hover {  
  text-decoration: none;  
}
```

在下面的实例中，您可以为链接的各种状态使用不同的值。我已将上面的规则添加到其中，现在意识到粉红色非常浅且难以阅读——为什么不将其更改为更好的颜色呢？你能把链接加粗吗？

I am a level one heading

This is a paragraph of text. In the text is a span element and also a [link](#).

This is the second paragraph. It contains an *emphasized* element.

- Item one
- Item two
- Item *three*

Interactive editor

```
a:link {
  color: pink;
}

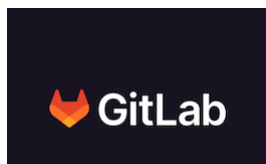
a:visited {
  color: green;
}

a:hover {
  text-decoration: none;
}
```

```
<h1>I am a level one heading</h1>

<p>This is a paragraph of text. In
the text is a <span>span
element</span>
and also a <a
href="http://example.com">link</a>.
</p>

<p>This is the second paragraph. It
contains an <em>emphasized</em>
```



```
<li>Item two</li>
```

Reset

用于软件创新的 One DevOps 平台。消除点解决方案蔓延。30 天免费试用。

Mozilla 广告

不想看广告？

我们删除了悬停时链接上的下划线。您可以从链接的所有状态中删除下划线。然而，值得记住的是，在真实站点中，您要确保访问者知道链接就是链接。保留下划线可能是一个重要线索，让人们意识到可以点击段落中的某些文本——这是他们习惯的行为。与 CSS 中的所有内容一样，您的更改可能会降低文档的可访问性——我们的目标是在适当的地方突出显示潜在的陷阱。

注意：您经常会在这些课程和整个 MDN 中看到提及可访问性。当我们谈论可访问性时，我们指的是让每个人都能理解和使用我们的网页的要求。

您的访客很可能正在使用带有鼠标或触控板的计算机，或带有触摸屏的手机。或者他们可能正在使用屏幕阅读器来读出文档的内容，或者他们可能需要使用更大的文本，或者仅使用键盘来浏览网站。

普通的 HTML 文档通常可供所有人访问——当您开始设计该文档的样式时，重要的是不要使其难以访问。

组合选择器和组合器

值得注意的是，您可以将多个选择器和组合器组合在一起。例如：

```
/* selects any <span> that is inside a <p>, which is inside an
<article> */
article p span {
}

/* selects any <p> that comes directly after a <ul>, which comes
directly after an <h1> */
h1 + ul + p {
}
```

您也可以将多种类型组合在一起。尝试将以下内容添加到您的代码中：

```
body h1 + p .special {
  color: yellow;
  background-color: black;
```

```
padding: 5px;  
}
```

这将为具有类的任何元素设置样式 `special`，该类位于 `a` 内，`<p>` 紧接在 `an` 之后 `<h1>`，位于 `a` 内 `<body>`。呸！

在我们提供的原始 HTML 中，唯一样式化的元素是 ``。

如果此刻这看起来很复杂，请不要担心——随着您编写更多 CSS，您很快就会开始掌握它。

概括

在本文中，我们了解了使用 CSS 设置文档样式的多种方法。我们将在学习其余课程时发展这些知识。但是，您现在已经足够了解如何设置文本样式、根据文档中定位元素的不同方式应用 CSS，以及在 MDN 文档中查找属性和值。

在下一课中，我们将了解[CSS 的结构](#)。

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