

# CSS 选择器

在CSS中，选择器用于定位我们想要设置样式的网页上的HTML元素。有各种各样的 CSS 选择器可用，在选择要设置样式的元素时允许细粒度的精度。在本文及其子文章中，我们将详细介绍不同的类型，了解它们的工作原理。

|       |   |
|-------|---|
| 先决条件: | 基本的计算机知识、 <a href="#">安装的基本软件、使用文件</a> 的基本知识、HTML 基础知识（学习 <a href="#">HTML 简介</a> ）以及 CSS 的工作原理（学习 <a href="#">CSS 第一步</a> ）。 |
| 客观的:  | 详细了解 CSS 选择器的工作原理。  |

## 什么是选择器？

CSS 选择器是 CSS 规则的第一部分。它是一种元素和其他术语的模式，告诉浏览器应选择哪些 HTML 元素以将规则中的 CSS 属性值应用于它们。选择器选择的一个或多个元素称为*选择器的主题*。

```
h1 {
  color: blue;
  background-color: yellow;
}

p {
  color: red;
}
```

在其他文章中，您可能遇到过一些不同的选择器，并了解到有些选择器以不同的方式定位文档——例如，通过选择一个元素（例如）`h1` 或一个类（例如 `.special`）。

在 CSS 中，选择器在 CSS 选择器规范中定义；像 CSS 的任何其他部分一样，它们需要在浏览器中得到支持才能工作。您将遇到的大多数选择器都是在[Level 3 Selectors specification](#) 和[Level 4 Selectors specification](#) 中定义的，它们都是成熟的规范，因此您会发现浏览器对这些选择器的支持非常好。

## 选择器列表

如果您有多个使用相同 CSS 的事物，则可以将各个选择器组合到一个 *选择器列表* 中，以便将规则应用于所有单个选择器。例如，如果我对 an h1 和 class of 有相同的 CSS `.special`，我可以将其写为两个单独的规则。

```
h1 {  
  color: blue;  
}  
  
.special {  
  color: blue;  
}
```

我还可以通过在它们之间添加逗号将它们组合成一个选择器列表。

```
h1, .special {  
  color: blue;  
}
```

逗号前后的空格有效。如果每个选择器都在一个新行上，您可能还会发现选择器更具可读性。

```
h1,  
.special {  
  color: blue;  
}
```

在下面的实例中，尝试组合具有相同声明的两个选择器。合并后的视觉显示应该是一样的。

## Type selectors

Veggies es bonus vobis, proinde vos postulo essum magis kohlrabi welsh onion daikon amaranth tatsoi tomatillo melon azuki bean garlic.

Gumbo beet greens corn soko **endive** gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini.

Turnip greens yarrow ricebean rutabaga **endive cauliflower** sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach

### Interactive editor

```
span {  
  background-color: yellow;  
}  
  
strong {  
  color: rebeccapurple;  
}  
  
em {  
  color: rebeccapurple;  
}
```

```
<h1>Type selectors</h1>  
<p>Veggies es bonus vobis, proinde  
vos postulo essum magis  
<span>kohlrabi welsh onion</span>
```



当您以这种方式对选择器进行分组时，如果任何选择器在语法上无效，则整个规则将被忽略。

```
melon azuki bean garlic.</p>  
<p>Gumbo beet greens corn soko
```

在下面的示例中，无效的类选择器规则将被忽略，而 h1 仍将被设置样式。

```
h1 {  
  color: blue;  
}  
  
..special {  
  color: blue;  
}
```

但是，当组合时，无论是 the h1 还是 class 都不会被设置样式，因为整个规则被视为无效。

```
h1, ..special {  
  color: blue;  
}
```

## 选择器的类型

选择器有几种不同的分组，了解您可能需要哪种类型的选择器将有助于您找到适合工作的工具。在本文的子文章中，我们将更详细地了解不同的选择器组。

### 类型、类和 ID 选择器

该组包括以 HTML 元素为目标的选择器，例如 <h1>。

```
h1 {  
}
```

它还包括针对类的选择器：

```
.box {  
}
```

或者，一个 ID：

```
#unique {  
}
```

## 属性选择器

这组选择器为您提供了不同的方法来根据元素上某个属性的存在来选择元素：

```
a[title] {  
}
```

或者甚至根据具有特定值的属性的存在进行选择：

```
a[href="https://example.com"]  
{  
}
```

## 伪类和伪元素

这组选择器包括伪类，它们为元素的某些状态设置样式。例如，伪 `:hover` 类仅在鼠标指针悬停在元素上时才选择元素：

```
a:hover {  
}
```

它还包括伪元素，伪元素选择元素的某个部分而不是元素本身。例如，`::first-line` 始终选择元素内的第一行文本（`<p>` 在下面的例子中是 `a`），就好像 `a <span>` 环绕在第一行格式化行上然后被选中。

```
p::first-line {  
}
```

## 组合器

最后一组选择器结合其他选择器以定位我们文档中的元素。例如，以下内容 `<article>` 使用子组合符（`>`）选择作为元素的直接子元素的段落：



不想看广告？

```
article > p {  
}
```

## 概括

在本文中，我们介绍了 CSS 选择器，它使您能够定位特定的 HTML 元素。接下来，我们将仔细研究[类型、类和 ID 选择器](#)。



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