

Chapter 21: Pseudo-Elements

pseudo-element	Description
<code>::after</code>	Insert content after the content of an element
<code>::before</code>	Insert content before the content of an element
<code>::first-letter</code>	Selects the first letter of each element
<code>::first-line</code>	Selects the first line of each element
<code>::selection</code>	Matches the portion of an element that is selected by a user
<code>::backdrop</code>	Used to create a backdrop that hides the underlying document for an element in the top layer's stack
<code>::placeholder</code>	Allows you to style the placeholder text of a form element (Experimental)
<code>::marker</code>	For applying list-style attributes on a given element (Experimental)
<code>::spelling-error</code>	Represents a text segment which the browser has flagged as incorrectly spelled (Experimental)
<code>::grammar-error</code>	Represents a text segment which the browser has flagged as grammatically incorrect (Experimental)

Pseudo-elements, just like pseudo-classes, are added to a CSS selectors but instead of describing a special state, they allow you to scope and style certain parts of an html element.

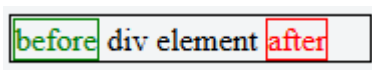
For example, the `::first-letter` pseudo-element targets only the first letter of a block element specified by the selector.

Section 21.1: Pseudo-Elements

Pseudo-elements are added to selectors but instead of describing a special state, they allow you to style certain parts of a document.

The `content` attribute is required for pseudo-elements to render; however, the attribute can have an empty value (e.g. `content: ""`).

```
div::after {  
  content: 'after';  
  color: red;  
  border: 1px solid red;  
}  
  
div {  
  color: black;  
  border: 1px solid black;  
  padding: 1px;  
}  
  
div::before {  
  content: 'before';  
  color: green;  
  border: 1px solid green;  
}
```



Section 21.2: Pseudo-Elements in Lists

Pseudo-elements are often used to change the look of lists (mostly for unordered lists, `ul`).

The first step is to remove the default list bullets:

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

Then you add the custom styling. In this example, we will create gradient boxes for bullets.

```
li:before {  
  content: "";  
  display: inline-block;  
  margin-right: 10px;  
  height: 10px;  
  width: 10px;  
  background: linear-gradient(red, blue);  
}
```

HTML

```
<ul>  
  <li>Test I</li>  
  <li>Test II</li>  
</ul>
```

Result

