

Professional Skills
Agile Fundamentals
Jira
Git
Databases Introduction
Java Beginner
Maven
Testing (Foundation)
Java Intermediate
HTML
CSS
<div><div></div> Introduction to CSS3</div>
<div><div></div> Fonts</div>
<div><div></div> Transform</div>
<div><div></div> Selectors</div>
<div><div></div> Selectors - Pseudo-Class / Pseudo-Elements</div>
<div><div></div> Spatial Attributes</div>
<div><div></div> Alignment</div>
<div><div></div> Overflow</div>
<div><div></div> Display and Positioning</div>
<div><div></div> Background and Borders</div>
<div><div></div> Bootstrap Introduction</div>
<div><div></div> Bootstrap Nav</div>
<div><div></div> Bootstrap Collapse</div>
<div><div></div> Bootstrap Grid</div>
<div><div></div> Bootstrap Forms & Inputs</div>
<div><div></div> Bootstrap Modal</div>
<div><div></div> Bootstrap Cards</div>
Javascript
Spring Boot
Selenium
Sonarqube
Advanced Testing (Theory)

Selectors - Pseudo-Class / Pseudo-Elements

Contents

- [Overview](#)
 - [Pseudo classes](#)
 - [Choosing elements by their attribute](#)
 - [Pseudo elements](#)
 - [::first-line pseudo-element](#)
 - [::first-letter pseudo-element](#)
 - [Multiple Pseudo-elements](#)
 - [::before pseudo-element](#)
 - [::after pseudo-element](#)
 - [::selection pseudo-element](#)
 - [Documentation](#)
- [Tutorial](#)
- [Exercises](#)

Overview

In this module, we will be exploring Pseudo-classes and Pseudo-Elements.

Pseudo classes

A pseudo-class is used to define a special state of an element.

It can be used to:

- Style an element when a user mouses over it
- Style visited and unvisited links differently
- Style an element when it gets focus

```
a {color: black; }
a:link {color: blue; }
a:visited {color: red; }
a:hover {color: green; }
a:active {color: orange; }
```

***Note:** `a:hover` MUST come after `a:link` and `a:visited` in the CSS definition in order to be effective!
a:active MUST come after a:hover in the CSS definition in order to be effective! Pseudo-class names are not case-sensitive.

Choosing elements by their attribute

- = operator finds attributes whose value exactly matches

```
a[href="http://www.qa.com"] { color: blue; }
```

- ^= operator finds attributes starting with a value

```
a[href^="http:"] { color: blue; }
```

- \$= operator finds any element attributes ending with a value

```
[src$=".png"] { color: red; }
```

- *= operator finds attributes containing the value

Cucumber
MongoDB
Express
NodeJS
React
Express-Testing
Networking
Security
Cloud Fundamentals
AWS Foundations
AWS Intermediate
Linux
DevOps
Jenkins Introduction
Jenkins Pipeline
Markdown
IDE Cheatsheet

```
[id*="stuff"] { color: red; }
```

Pseudo elements

A CSS pseudo-element is used to style specific parts of an element.
Such as:

- Style the first letter, or line of an element
- Insert content before, or after the content of an element.

Syntax is as follows:

```
selector::pseudo-element{  
    property:value;  
}
```

::first-line pseudo-element

The **::first-line** pseudo-element is used to add a special style to the first line of a text.

The following example formats the first line of the text in all **<p>** elements:

```
p::first-line{  
    color: red;  
    font-variant: small-caps;  
}
```

*Note: The **::first-line** pseudo-element can only be applied to block-level elements.*

Notice the double colon notation - **::first-line** versus **:first-line**

The double colon replaced the single-colon notation for pseudo-elements in CSS3. This was an attempt from W3C to distinguish between pseudo-classes and pseudo-elements.

The single-colon syntax was used for both pseudo-classes and pseudo-elements in CSS2 and CSS1.

For backward compatibility, the single-colon syntax is acceptable for CSS2 and CSS1 pseudo-elements.

::first-letter pseudo-element

The **::first-letter** pseudo-element is used to add a special style to the first letter of a text.

The following example formats the first letter of the text in all **<p>** elements:

```
p::first-letter{  
    color: #00ff00;  
    font-size: xx-large;  
}
```

Multiple Pseudo-elements

Several pseudo-elements can be combined. In the following example, the first letter of a paragraph will be red, in an xx-large font size. The rest of the first line will be green, and in small-caps. The rest of the paragraph will be the default font size and colour:

```
p::first-letter {
  color: #ff0000;
  font-size: xx-large;
}

p::first-line {
  color: #84b400;
  font-variant: small-caps;
}
```

::before pseudo-element

The **::before** pseudo-element can be used to insert some content before the content of an element.

The following example inserts an image before the content of each **<h1>** element:

```
h1::before{
  content:
url(https://media.tenor.com/images/7db260732bbe2a6024d78ac75a3cb18d/tenor.gif);
}
```

::after pseudo-element

The **::after** pseudo-element can be used to insert some content after the content of an element.

The following example inserts an image after the content of each **<h1>** element:

```
h1::after{
  content:
url(https://media.tenor.com/images/3b388fe03da271d2674faf85eb7c3fcd/tenor.gif);
}
```

::selection pseudo-element

The **::selection** pseudo-element matches the portion of an element that is selected by a user.

The following example makes the selected text red on a yellow background:

```
::selection {
  color: red;
  background: yellow;
}
```

Documentation

For more information on pseudo-classes please refer [to this document](#).

For more information on pseudo-elements please refer [to this document](#).

Tutorial

1. Create the following HTML document

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Selectors - Tutorial</title>
</head>
<body>

<p><b><a href="#" target="_blank">This is a link</a></b></p>

<p><b>Note:</b> a:hover MUST come after a:link and a:visited in the
CSS definition in order to be effective.</p>
<p class="start">To stop the degradation of the planet's natural
environment
  and to build a future in which humans live in harmony with nature, by;
  conserving the world's biological diversity, ensuring that the use of
  renewable natural resources is sustainable, and promoting the
reduction
  of pollution and wasteful consumption.</p>
<p class="end"></p>

</body>
</html>

```

2. Set the colour of a link to red

```

<style>
  a:link{
    color: #ff0000;
  }
</style>

```

3. Set the colour of a visited link to purple

```

<style>
  ...
  a:visited{
    color: #4800ff;
  }
</style>

```

4. Set the colour of the link to grey on hover.

```

<style>
  ...
  a:hover{
    color: #808080;
  }
</style>

```

5. Set the colour of the link to green when active.

```

<style>
  ...
  a:active{
    color: #4cff00;
  }
</style>

```

6. Modify the `<p>` with the class `start` to insert the content `Read this --` before the start of the paragraph

```
<style>
...
p.start::before{
    content: "Read this -- ";
}
</style>
```

7. Modify the `<p>` with the class `end` to append the content `More to come --` at the end of the paragraph

```
<style>
...
p.end::after{
    content: "More to come -- ";
}
</style>
```

8. Change the last two steps to choose the `p` element by their `class` attribute using the `=` operator

```
<style>
p[class="start"]::before{
    content: "Read this -- ";
}
p[class="end"]::after{
    content: "More to come --";
}
</style>
```

The final code should look like this:

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Selectors - Tutorial</title>
  <style>
    a:link{
      color: #ff0000;
    }
    a:visited{
      color:#4800ff;
    }
    a:hover{
      color:#808080;
    }
    a:active{
      color:#4cff00;
    }
    /* p.start::before{
      content: "Read this -- ";
    }
    p.end::after{
      content: "More to come -- ";
    } */
    p[class="start"]::before{
      content: "Read this -- ";
    }
    p[class="end"]::after{
      content: "More to come -- ";
    }
  </style>
</head>
<body>

  <p><b><a href="#" target="_blank">This is a link</a></b></p>

  <p><b>Note:</b> a:hover MUST come after a:link and a:visited in the
    CSS definition in order to be effective.</p>
  <p class="start">To stop the degradation of the planet's natural environment
    and to build a future in which humans live in harmony with nature,
by;
    conserving the world's biological diversity, ensuring that the use
of
    renewable natural resources is sustainable, and promoting the
reduction
    of pollution and wasteful consumption.</p>
  <p class="end"></p>

</body>
</html>

```

Exercises

1. Set text colour to blue, for the first line of the `<p>` element
2. Set text colour to red, and text size to "xx-large" for the first letter of the `<p>` element
3. Insert something before, and after `<p>` elements, using the `::before` and `::after` pseudo-elements
4. Change the background colour, when a user hovers over the last `<p>` element with the class "highlight".
5. Set the background color of `<input>` elements that are in focus (clicked or active), to "lightblue".

```
<!DOCTYPE html>
<html>
<head>
  <style>

  </style>
</head>
<body>

  <h1>This is a Heading</h1>
  <p>In my younger and more vulnerable years my father gave me some
advice that I've been turning over in my mind ever since. </p>
  <p>'Whenever you feel like criticizing anyone,' he told me, 'just
remember that all the people in this world haven't had the advantages that
you've had.'</p>
  <p class="highlight">This is another paragraph.</p>
  <form>
    <label>First name:</label>
    <input type="text" name="fname"><br/>
    <label>Last name:</label>
    <input type="text" name="lname"><br/>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

► Solutions