CSS Pseudo-classes & Pseudo-elements

Dr D V Naga Raju

Introduction

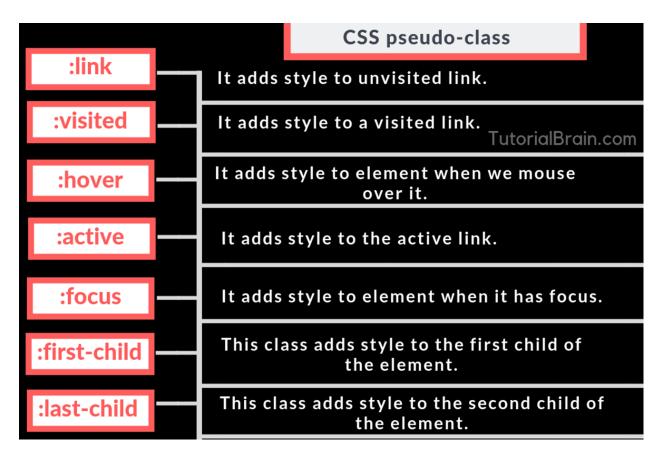
A Pseudo class in CSS is used to define the special state of an element.

A state pseudo-class usually comes into play when the user performs an action.

It can be combined with a CSS selector to add an effect to existing elements based on their states.

For Example, changing the style of an element

- -when the user hovers over it, or
- when a link is visited.



```
<!DOCTYPE html>
<html>
<head>
<title>Pseudo-class Selector Practice</title>
 <style>
 /* Add your CSS rules here */
 </style>
</head>
<body>
 <header>
  <h1>Welcome to the Pseudo-class Selector Practice</h1>
 </header>
  <nav>
  <a href="#">Home</a>
   <a href="#">About</a>
   <a href="#">Services</a>
  <a href="#">Contact</a>
  </nav>
  <main>
  <section>
   <h2>About Us</h2>
   We are a web development company...
```

```
</section>
 <section>
 <h2>Our Services</h2>
 Web Design
  Front-end Development
  Back-end Development
 </section>
</main>
<div id="additional-info">
<div class="paragraphs">
 Paragraph 1
 Paragraph 2
 Paragraph 3
 </div>
 <div class="links">
 <a href="#">Link 1</a>
 <a href="#">Link 2</a>
</div>
</div>
<footer>
© 2023 Pseudo-class Selector Practice
</footer></body></html>
```

Exercises:

- 1. Change the font color of the `hover` state of the `<a>` elements inside the `<nav>`.
- 2. Apply a different background color to the `active` state of the `<a>` elements inside the `<nav>`.
- 3. Change the font color of the `link` state of the `<a>` elements inside the `<nav>`.
- 4. Apply italic font style to the `visited` state of the `<a>` elements within the `<div>` with the class "links".
- 5. Apply a 'hover' effect to the 'h2' elements, changing their background color.
- 6. Add an underline to the 'active' state of the 'h2' elements.
- 7. Change the font color of the `last-child` of the `<a>` elements inside the `<div>` with the class "links".
- 8. Apply a background color to the `nth-child(odd)` of the `` elements within the `` inside the `<nav>`.
- 9. Change the font color of the `nth-child(3)` of the `` elements inside the `<div>` with the class "paragraphs".
- 10. Apply a 'focus' effect to the 'input' elements, adding a border when they are selected.

Pseudo-elements

A CSS pseudo-element is used to style specified parts of an element.

For example, it can be used to:

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

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

::first-line - add a special style to the first line of a text.

::first-letter - add a special style to the first letter of a text.

::before - to insert some content before the content of an element.

::after - to insert some content after the content of an element.

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

::marker - selector selects the marker of a list item
```

- 1. Apply a different font size and color to the `first letter` of the `` elements inside the `<section>`.
- 2. Change the font style of the `first line` of the text within the `<h1>` element in the `<header>`.
- 3. Add a decorative element before the content of the `<h2>` elements in the `<main>` using the `before` pseudo-element.
- 4. Add a content marker (e.g., a star or bullet) after the content of the `` elements inside the `` in the `<nav>` using the `after` pseudo-element.
- 5. Change the background color of the `selection` (text highlight) within the `` elements in the `<div>` with the class "paragraphs".
- 6. Apply a different font color to the text in the `first-line` of the `` elements within the `<div>` with the class "paragraphs".

- 7. Insert a content element before the `<a>` elements inside the `<nav>` using the `before` pseudo-element.
- 8. Add a content marker after the content of the `last-child` of the `` elements inside the `` in the `<nav>` using the `after` pseudo-element.
- 9. Change the font style of the content of the `marker` pseudo-element in the `` within the `<nav>`.
- 10. Apply a content element after the content of the `<footer>` using the `after` pseudo-element.