**MILSETONE 5**

- Selectors and Combinators

- Pseudo Classes and Elements

- !important

- CSS Functions

- Comments

1. **Selectors and Combinators:**

* Simple selectors
* Combinator selectors
* Pseudo – class selectors
* Pseudo – element selectors
* Attribute selectors
  + Simple selector – select elements based on name, id, class
    - <p> hello </p>

p{ color: red;} 🡪 p is a element selector

* + - <p id=”para1”> hello </p>

#para1{ color: red;} 🡪 para1 is id selector

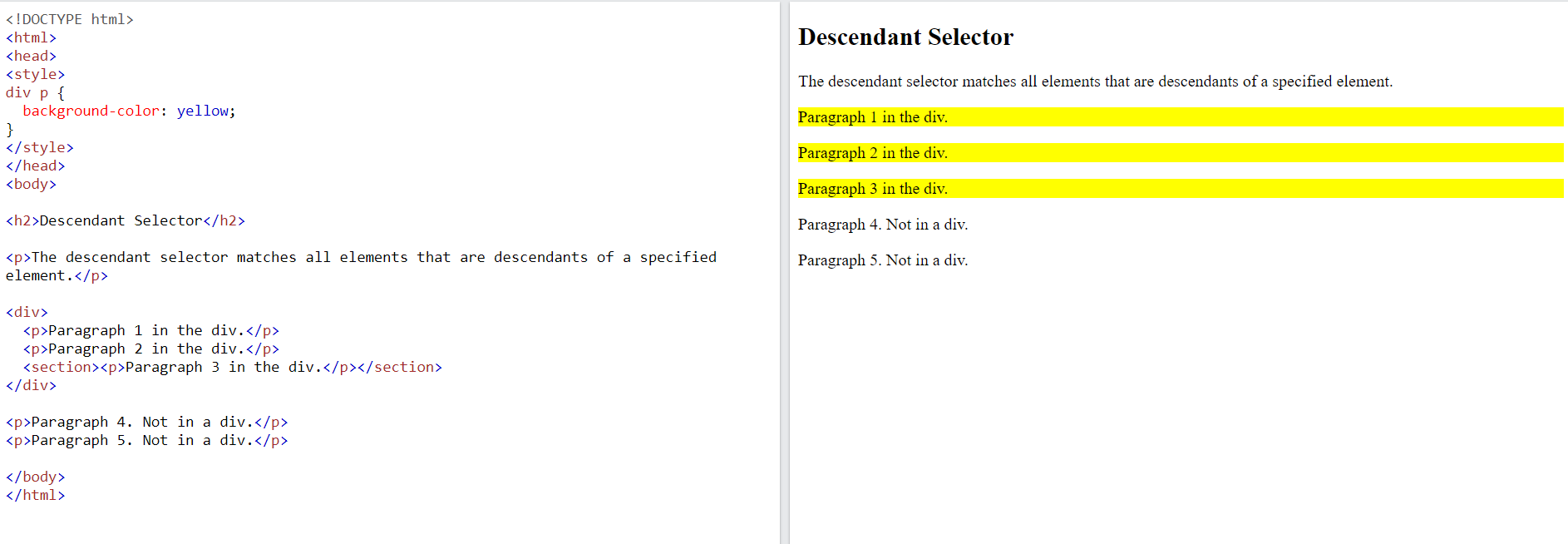
* + - <p class=”para1”> hello </p>

.para1{ color: red;} 🡪 para1 is class selector

* + - selector – selects all HTML elements on page

\*{ color: red;}

* + - Grouping selector – h1, p, h2{ color: red;}
* Combinator selector
  + descendant selector (space)
  + child selector (>)
  + adjacent sibling selector (+)
  + general sibling selector (~)
    - Descendant selector – matches all elements that are descendants of a specified element



As p1, p2, p3 is only under div. they have been changed to background-color yellow

* Child selector (>)

div > p { background-color: yellow; }

<div>

<p>Paragraph 1 in the div.</p>

<p>Paragraph 2 in the div.</p>

<section>

<!-- not Child but Descendant -->

<p>Paragraph 3 in the div (inside a section element).</p>

</section>

<p>Paragraph 4 in the div.</p>

</div>

* General sibling selector (~)- it selects next siblings of element

div ~ p {

background-color: yellow;

} <div>

<p>Paragraph 2.</p>

</div>

<p>Paragraph 3.</p>

<p>Some code.</p>

<p>Paragraph 4.</p>

* Adjacent sibling selector (+) – selects an element that is immediately after another specific element

div+p{ background-color: yellow};

<div>

<p>para 1 </p>

</div>

<p>para2</p>

1. **Pseudo classes**

* Pseudo-classes are used to define a special state of an element

***SYNTAX:*** *selector: pseudo-class {property: value;}*

**EX**: For anchor tag pseudo classes

a:link🡪a:visited🡪a:hover🡪a:active. Because active has least priority

* :first-child – matches an element that is first child of any element

p:first-child{color: blue};

<div>

<p>hello</p>

</div>

1. **Pseudo elements**

* Pseudo elements are used to style a specific part of an element.
* Syntax : selector::pseudo-element {  
    property: value;  
  }
* ::after, ::before, ::marked, ::selection:: first-line::first-letter

1. **!important**

* The !important rule in CSS is used to add more importance to a property/value than normal.
* It will override all previous styling rule for that specific property on that element.

1. **CSS Functions:**
2. " (" attr(href) ")" – adds URL which is taken from the HTML element.
3. conic-gradient()
4. counter()
5. linear-gradient()
6. max()
7. min()
8. rgba()

Pseudo classes –

* defined with single colon (:)
* Used to style target state like hover, link, visited, first-child etc.,
  + :active – when clicking whether the element is active/ not. Used in a, button
  + :fullscreen – matches an element that is in fullscreen mode requestFullScreen() API is used in JS.
  + :any-link – combination of :link(unvisited) & :visited
  + :autofill – if input is filled automatically by PC/ lap. Then its border/ smth can be changed
  + :checked – matches when an element such as checkboxes and radio button are toggled on
  + :modal – used to open and style dialog box created with <dialog> tag 🡪 uses JS API showmodal()
  + Input tags have
    - :enabled
    - :disabled
    - :read-only
    - :read-write
    - :placeholder-shown
    - :checked, :default 🡪 used to style/ highlight when a field is checked/ not.
    - **EX:** <input id="secret" name="secret" type="text" value="test" pattern="[a-z]+" /> **CSS –** input:invalid{ color:red;}
    - :in-range, :out-range 🡪 to check and validate the contents
    - :focus-within 🡪 to focus the complete element. No only input field. But also label by CSS label:focus-within{font-weight:bold};

Pseudo elements –

* defined with two colon (::)
* Used to target specific part of an element. Like ::first-line, ::first-letter
  + **EX:** ::marker { color: red;}

<ul>

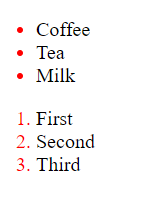
<li>Coffee</li>

<li>Tea</li>

<li>Milk</li>

</ul>

**OUTPUT:**



* **EX:** ::selection {

color: red;

background: yellow;

}

