MODULE – 3 : Mern Stack – CSS and CSS 3

CSS Selectors & Styling:

Theory Assignment:

1) What is a CSS selector? Provide examples of element, class, and ID selectors.

→
A CSS selector is a pattern used to select specific elements in an HTML document and apply styles to them. Selectors allow developers to target HTML elements based on attributes such as element type, class, or ID.

1. Element Selector :
   → Target all element of a specific type.
   ```
   P {
           Color:blue;
     }
   ```
   This will target all the paragraph tag.
2. Class selector :
   → Target elements with class attribute.
   ```
   .name {
           Color:blue;
           }
   ```
   This will target tag with "name" class.
3. ID selector :
   → Target element with ID attribute.
   ```
   #name{
           Color:blue; }
   ```
   This will target tag with "name" ID.

2) Explain the concept of CSS specificity. How do conflicts between multiple styles get resolved?

→

CSS specificity determines which styles are applied when multiple rules target the same element. It is calculated based on the types of selectors used in a rule.

Specificity Hierarchy:

1. Inline styles (e.g., style="color: red;") have the highest specificity.
2. ID selectors (#id) are highly specific.
3. Class selectors (.class), attributes (e.g., [type="text"]), and pseudo-classes (e.g., :hover) have medium specificity.
4. Element selectors (e.g., div, p) and pseudo-elements (e.g., ::before) have the lowest specificity.
5. Universal selector (*), inherited styles, and browser defaults have the least specificity.

Rules with higher specificity take precedence. If specificity is the same, the rule defined later in the stylesheet is applied.

3) What is the difference between internal, external, and inline CSS? Discuss the advantages and disadvantages of each approach.

→

1. Internal CSS:
   ➔ CSS rules are defined within a <style> tag inside the <head> section of an HTML document.

```
<style>
  h1 {
    color: blue;
  }
</style>
```

→ Advantages :

- Useful for styling a single document.

Disadvantages:

- Cannot be reused across multiple files.

2. External CSS:
   → Rules are written in a separate .css file and linked to the HTML document using a <link> tag.

   ```
   <link rel=”stylesheet” href=”styles.css”>
   ```

   In styles.css:

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

   → Advantages :
   • Allows reusability across multiple HTML files.
   Disadvantages:
   • Requires an additional HTTP request for the CSS file.

3. Inline CSS:
   ➔ CSS is applied directly within an HTML element using the style attribute.
   ➔

```
<h1 style="color: blue;">Hello</h1>
```

   ➔ Advantages :
   • Quick to apply for individual elements.
   Disadvantages:
   • Leads to cluttered HTML code.
      - Not reusable, reducing maintainability.

CSS Box Model:

Theory Assignment:-

1) Explain the CSS box model and its components (content, padding, border ,margin). How does each affect the size of an element?

→

The CSS box model describes how elements are structured and rendered on a web page. It defines the space that an element occupies, including its content, padding, border, and margin.

Components:

Content: This is the actual content inside the element, such as text or images. It is the starting point of the box model.

Padding: The space between the content and the border. It increases the size of the element and pushes the content inward.

Border: The line surrounding the padding. Its width and style can be customized (e.g., solid, dashed, etc.).

Margin: The space outside the border that separates the element from other elements on the page.

2) What is the difference between border-box and content-box box-sizing in CSS? Which is the default?

→

1. Content-box(Default) :

   → Only the **content** is considered in the specified width and height.
   →**Padding** and **border** are added outside the dimensions.

2. Border-box:
   → The content, padding, and border are included in the specified width and height.

CSS Flexbox

Theory Assignment:

1) What is CSS Flexbox, and how is it useful for layout design? Explain the terms flex-container and flex-item.

   →

   CSS Flexbox (Flexible Box Layout) is a layout model in CSS that allows developers to create flexible and responsive layouts by arranging and aligning elements efficiently, even when their size is unknown or dynamic.

2) Describe the properties justify-content, align-items, and flex-direction used in Flexbox.

   →

   1. Justify-Content:
      →Define how flex items are distributed horizontally.
         - ➔flex-start: Items align at the start of the main axis.
         - ➔flex-end: Items align at the end of the main axis.
         - ➔center: Items are centered along the main axis.
         - ➔space-between: Items are evenly distributed, with the first item at the start and the last item at the end.
         - ➔space-around: Items are evenly distributed, with equal space around them.
         - ➔space-evenly: Items are distributed with equal space between and around them.

2. Align-Items:
   →Determines how the direction of the main axis and the order of the items.
   - ➔stretch (default): Items stretch to fill the container.
   - ➔flex-start: Items align at the start of the cross-axis.
   - ➔flex-end: Items align at the end of the cross-axis.
   - ➔center: Items are centered along the cross-axis.
   - ➔baseline: Items align with the baseline of the text.

3. Flex-direction:
   →Determines the direction of the main axis and the order of items.
   - ➔row (default): Items are placed left-to-right in a row.
   - ➔row-reverse: Items are placed right-to-left in a row.
   - ➔column: Items are placed top-to-bottom in a column.
   - ➔column-reverse: Items are placed bottom-to-top in a column.

CSS Grid

<u>Theory Assignment:</u>

1) Explain CSS Grid and how it differs from Flexbox. When would you use Grid over Flexbox?

   →
   CSS Grid is a two-dimensional layout system in CSS that enables developers to design complex layouts with rows and columns. It is ideal for creating grid-based designs, where precise control over both axes (horizontal and vertical) is needed.

   Flexbox, on the other hand, is a one-dimensional layout system designed to layout items in a row (horizontal) or column (vertical).

2) Describe the grid-template-columns, grid-template-rows, and grid-gap properties. Provide examples of how to use them.?
→

1. grid-template-columns:
   →Defines the structure of columns in the grid, specifying their width.

   Example:

   ```
   grid-template-columns: 1fr 2fr 1fr;
   ```

2. grid-template-rows:
   → Defines the structure of rows in the grid, specifying their height.

   Example:

   ```
   grid-template-rows: 100px 200px auto;
   ```

3. grid-gap:
   →Specifies the spacing between grid items.

   Example:

   ```
   grid-gap: 20px;
   ```