Full Stack Development-I

Experiment -3

All Units CO’s

CO	Course Outcome	BTL
CO1	Illustrate HTML elements and their attributes for designing static web pages.	K3
CO2	Apply appropriate CSS styles to HTML elements.	K3
CO3	Demonstrate JavaScript Pre-defined and User-Defined Objects.	K3
CO4	Develop dynamic web pages and validate forms using JavaScript.	K3

Reference Books

Reference Books:

1. Programming the World Wide Web, 7th Edition, Robert W Sebesta, Pearson, 2013.
2. Web Programming with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, 2019 (Chapters 1-11).
3. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, A Press, O’Reilly.

Online Learning Resources:

1. https://www.w3schools.com/html
2. https://www.w3schools.com/css
3. https://www.w3schools.com/js/
4. https://www.w3schools.com/nodejs

Week 3

3. HTML 5 and Cascading Style Sheets, Types of CSS

a) Develop a HTML program, that makes use of <article>, <aside>,<figure>,<figcaption>,<footer>,<header>, <main>, <nav>, <section>, <div>, <span> tags.

b) Develop a HTML program, to embed audio and video into HTML web page.

c) Develop a program to apply different types (or levels of styles or style specification formats)- inline, internal, external styles to HTML elements. (identify selector, property and value).

Cascading Style Sheets(CSS)

- CSS stands for **Cascading Style Sheets**
- Defines how our page looks
- Ways to apply Styles to our webpage

The CSS can be declared in 3 ways:

- – Inline Style
- – Internal Style
- – External Style

1. Inline Style Sheet

- Applies style at tag level.
- <style> attribute is used for inline style.

Syntax: **style=“property1:value1; property2:value2”**

Example:

```
<p style="color: red; margin-left: 20px;background:pink">
                This is a paragraph
</p>
```

2. Internal Style Sheet

- Applies style to entire web page.
- Define internal styles in the head section with the <style> tag.

Syntax:

```
<style>
      selector{
            property1:value1;
            property2:value2;
            }
      </style>
```

- The selector is normally the HTML tag/id/class… you wish to define.
- The property is the attribute you wish to change, and each property can take a value.
- The property and value are separated by a colon, and surrounded by curly braces.

2. Internal Style Sheet

Example:

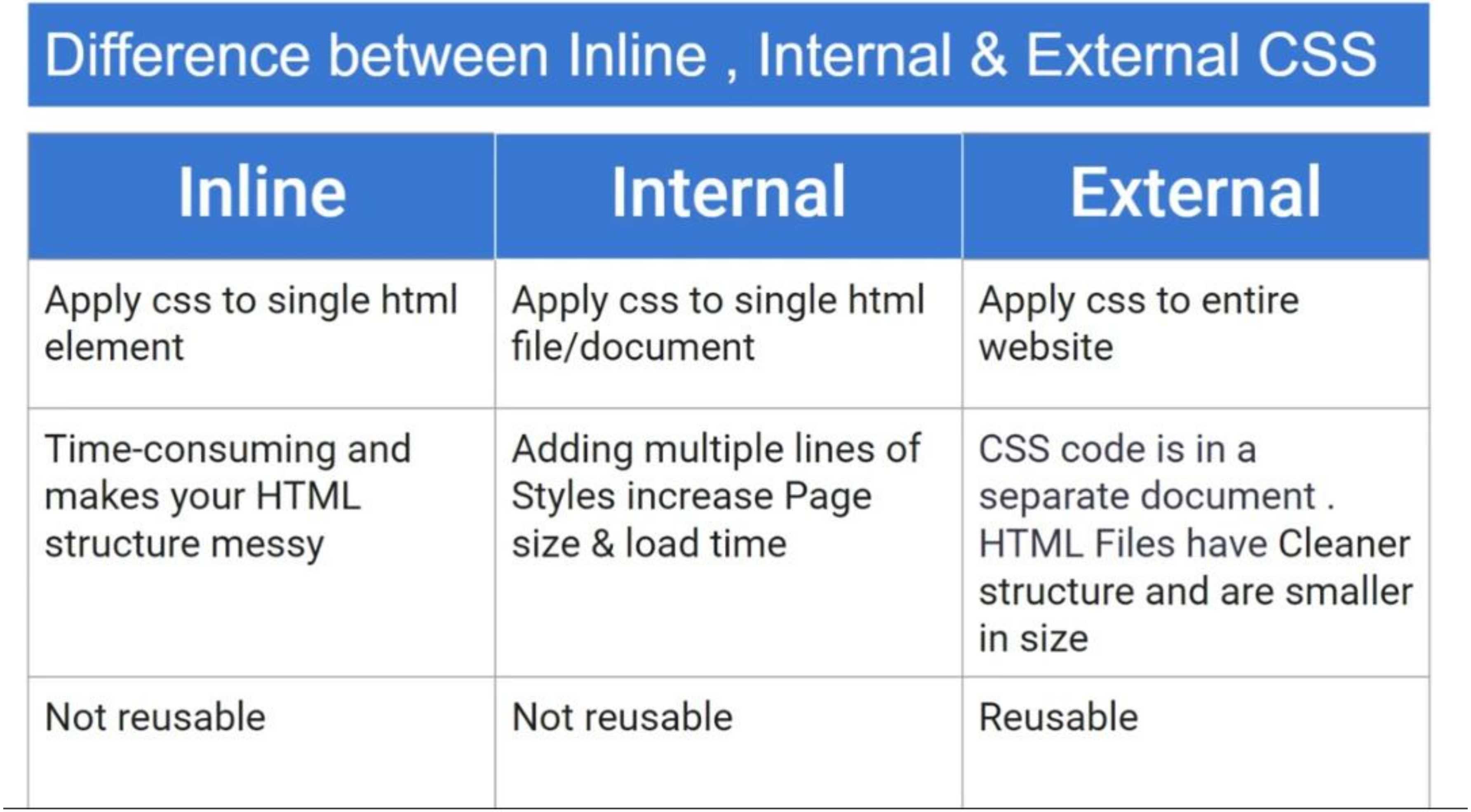
```
<style>
        p {
                color:white;
                background:green;
                }
   h1{
        color:green;
        }

</style>
```

3. External Style Sheet

- Applies style to multiple web pages.
- Styles should be in a separate file with .css extension.
- Apply (link css file to html) styles in a css file to any web page using <link> tag.

Syntax to link:

```
<head>
        <link rel="stylesheet" href=“path of css file“ type="text/css” >
        </link>
</head>
```



Difference between Inline , Internal & External CSS

Inline	Internal	External
Apply css to single html element	Apply css to single html file/document	Apply css to entire website
Time-consuming and makes your HTML structure messy	Adding multiple lines of Styles increase Page size & load time	CSS code is in a separate document . HTML Files have Cleaner structure and are smaller in size
Not reusable	Not reusable	Reusable

HTML Semantic & Non Semantic Elements

- Elements with meaning.
- Help to structure code & makes more readable.
- Easy to maintain the code.
- A semantic element clearly describes its meaning to both the browser and the developer.

Semantic Vs. Non Semantic Elements

Semantic	**Non Semantic**
Elements with meaning	Elements doesn’t have any meaning
Specific attribute for their structure	Can be used with different attributes
Ex: <img>, <table>, and <article> - Clearly defines its content.	**Ex:**<div> and <span> - Tells nothing about its content.

HTML <div> Non Semantic Element

- The <div> tag defines a division or a section in an HTML document.
- It is a **block-level element**(content will be displayed in a new line).
- The <div> tag is easily styled by using the class or id attribute.

- **Syntax:**

  <div> ……..</div>

HTML <span> Non Semantic Element

- The <span> element which is used to color a part of a text.
- It is used for in-line styles.

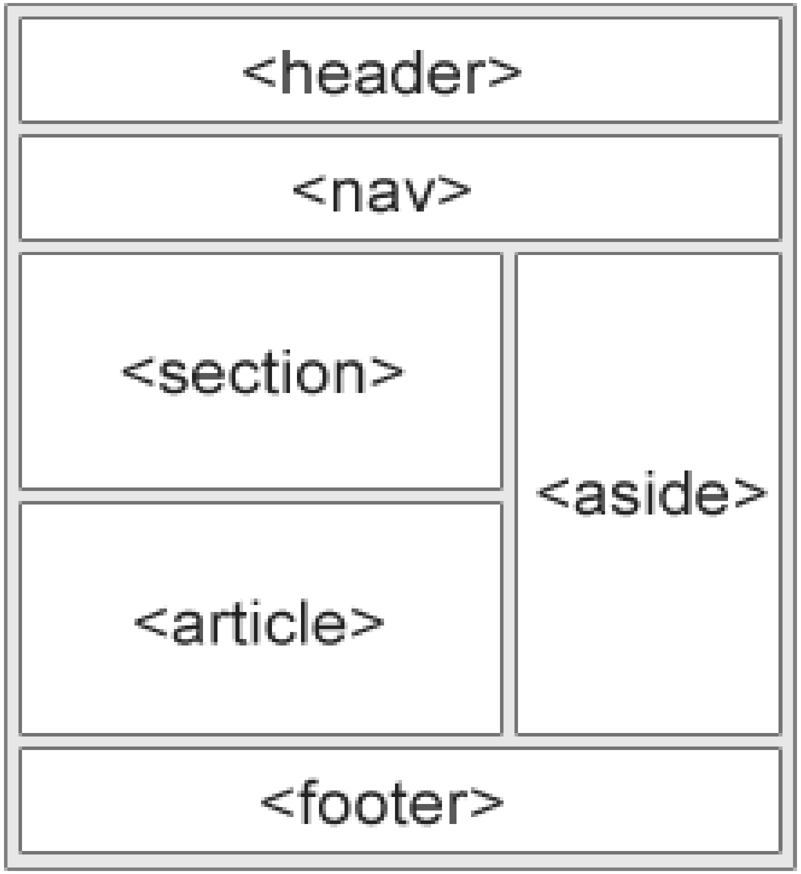
- **Syntax:**

```
<p>This is <span style="color:blue">paragraph</span> </p>
```

Semantic Elements in HTML

In HTML there are several semantic elements that can be used to define different parts of a web page:

- <section>
- <article>
- <header>
- <footer>
- <aside>
- <figure>
- <figcaption>
- <main>
- <nav>



HTML <section> Element

- The <section> element defines a section in a document. Such as Chapters, headers, footers or any other sections of the document.

- It is a paired tag .

- No Element specific attributes

Syntax:

<section>…….. </section>

HTML <article> Element

- It is used to represent an article.More specifically the content within the <article> tag is independent from the other content on the site.
- The <article> element can be used in forum post, a magazine or a newspaper article, a blog entry.
- It is a paired tag .
- No Element specific attributes

Syntax:

<article>…….. </article>

HTML < header> Element

- This element is used to display headers for subheadings, version information, navigation controls represents a container for introductory content or a set of navigational links.
- It is a paired tag .
- No Element specific attributes

Syntax:

<header>…….. </header>

Note: You can have several <header> elements in one HTML document. However, **<header> cannot be placed within a <footer>, <address> or another <header> element**.

HTML <footer> Element

- The <footer> element defining the footer of an HTML document or section.
- A <footer> element typically contains authorship information, copyright information, contact information, links to terms of use, privacy policy, etc.
- You can have several <footer> elements in one document.
- It is a paired tag .
- No Element specific attributes

Syntax:

<footer>…….. </footer>

HTML <aside> Element

- The <aside> element defines some content aside from the content it is placed in (like a sidebar).
- The <aside> content should be indirectly related to the surrounding content.

- Syntax:

    <aside>……. </aside>

HTML <nav> Element

- The <nav> element defines a set of navigation links.
- Write all anchor tags inside <nav> tag .

Example:

```
<nav>
    <a href="/html/">HTML</a>
    <a href="/css/">CSS</a> |
    <a href="/js/">JavaScript</a>
    <a href="/jquery/">jQuery</a>
  </nav>
```

HTML <figure> and <figcaption> Elements

- The <figure> tag specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.
- The <figcaption> tag defines a caption for a <figure> element. The <figcaption> element can be placed as the first or as the last child of a <figure> element.
- The <img> element defines the actual image/illustration.

Example:

```
<figure>
    <img src=“one.jpg" >
    <figcaption>Fig1. - Trulli, Puglia, Italy.</figcaption>
  </figure>
```

HTML <main> Element

- The <main> tag specifies the main content of a document.
- The content inside the <main> element should be unique to the document. It should not contain any content that is repeated across documents such as sidebars, navigation links, copyright information, site logos, and search forms.

Note: There must not be more than one <main> element in a document. The <main> element must NOT be a descendant of an <article>, <aside>, <footer>, <header>, or <nav> element.

HTML <audio> Element

- The HTML <audio> element is used to play an audio file on a web page.
- The controls attribute adds audio controls, like play, pause, and volume.
- The <source> element allows you to specify alternative audio files which the browser may choose from. The browser will use the first recognized format.

Syntax:

```
<audio>
        <source  src=“path”>
</audio>
                (or)

<audio src=“path”> </audio>
```

HTML <audio> Element

Attributes:

- controls
- loop (to play continuously)
- muted
- autoplay
- preload

Note: Add muted after autoplay to let your audio file start playing automatically (but muted)

HTML <video> Element

- The HTML <video> element is used to show a video on a web page.
- The controls attribute adds video controls, like play, pause, and volume.
- The <source> element allows you to specify alternative video files which the browser may choose from. The browser will use the first recognized format.

Syntax:

```
<video>
          <source  src=“path”>
</video>
          (or)

<video src=“path”> </video>
```

HTML <video> Element

Attributes:

- controls
- loop (to play continuously)
- muted
- Autoplay
- width
- height

- To start a video automatically, use the autoplay attribute:
- Add muted after autoplay to let your video start playing automatically (but muted):

THANK YOU