**MODULE ( HTML 5) – 3**

* What are the new tags added in HTML 5?
* HTML5 introduced several new tags (or elements) that enhance the structure, functionality, and semantics of web pages. These tags help developers create cleaner, more accessible, and maintainable code while improving the user experience. Here's a list of the new HTML5 tags and their purposes:

**1.**<article>

* **Purpose**: Represents a self-contained piece of content that could be distributed or reused independently (e.g., blog posts, news articles).
* **Use Case**: Wrap content like articles, forum posts, or other stand-alone sections that could make sense outside of the page's context.

**2.**<aside>

* **Purpose**: Represents content that is tangentially related to the main content, such as sidebars, pull quotes, or additional information.
* **Use Case**: Used for content that is not part of the main narrative but adds value or context (e.g., related links, advertisements).

**3.**<audio>

* **Purpose**: Embeds audio content (e.g., music, sound effects) in the document. It allows users to control playback with built-in controls.
* **Use Case**: Used to embed audio files without relying on external plugins.

**4.**<canvas>

* **Purpose**: Provides a drawable region that can be used for rendering graphics (such as 2D graphics, charts, games, and animations) via JavaScript.
* **Use Case**: Used for drawing graphics dynamically in a web page.

**5.**<details>

* **Purpose**: Represents a disclosure widget that users can click to reveal or hide additional information.
* **Use Case**: Used for collapsible content like FAQs or hidden sections.

**6.**<figcaption>

* **Purpose**: Represents a caption for a <figure> element (e.g., an image, video, or diagram).
* **Use Case**: Provides a description or caption for the content inside a <figure>.

**7.**<footer>

* **Purpose**: Represents the footer section of a document or section. It typically contains metadata, copyright information, or links.
* **Use Case**: Used at the bottom of pages or sections.

**8.**<header>

* **Purpose**: Represents the introductory content or navigation for a document or section.
* **Use Case**: Used to group introductory elements like logos, navigation menus, or page titles.

**9.**<main>

* **Purpose**: Represents the main content of the document, excluding headers, footers, sidebars, and navigation. There should be only one <main> element per document.
* **Use Case**: Used to wrap the primary content that is unique to the page.

**10.**<nav>

* **Purpose**: Represents a section of navigation links. It groups links to other pages or sections within the site.
* **how to embed audio and video in a webpage?**

* To embed audio and video content in a webpage, HTML5 provides the <audio> and <video> tags, respectively. These elements allow you to directly embed media files into your webpage without relying on plugins like Flash or external players. You can also provide controls for users to play, pause, and adjust the volume of the media**.**

**1. Embedding Audio with the**<audio>**Tag**

* The <audio> tag is used to embed sound files, such as music or speech, in a webpage. It can contain multiple <source> elements to offer different formats of the audio file, and it provides built-in controls for playback.

**2. Embedding Video with the**<video>**Tag**

* The <video> tag allows you to embed video files. Similar to the <audio> tag, it can contain multiple <source> elements to provide different video formats, ensuring compatibility across browsers.
* **semantic element in html?**
* Semantic elements in HTML refer to elements that have a clear meaning or significance about the content they enclose. These elements provide context to the browser, search engines, and developers about the type of content they represent. By using semantic elements, you create a more structured, readable, and accessible web page, which is essential for both users and search engine optimization (SEO).
* Before HTML5, many elements like <div> and <span> were widely used for layout and styling purposes, but they didn’t convey any specific meaning about the content they held. With HTML5, a number of semantic elements were introduced to give web pages a more meaningful structure.

**Benefits of Using Semantic Elements:**

* **Improved Accessibility**: Semantic tags provide more meaningful structure for assistive technologies, like screen readers, which helps people with disabilities better navigate and understand web content.
* **Better SEO**: Search engines can more easily understand the structure and meaning of a page. Proper use of semantic elements improves the discoverability and ranking of your page.
* **Easier to Maintain**: A well-structured page with semantic elements is easier for developers to maintain, as it’s more intuitive and organized.
* **Cross-Browser Consistency**: By using standard, well-supported HTML5 elements, your page will be more consistent across different browsers and devices.
* **Readability and Understanding**: Semantic elements give developers and others reading the code a better understanding of the structure and purpose of the content, making the code more readable and maintainable.
* **Canvas and SVG tags?**
* Both the <canvas> and <svg> elements in HTML are used for drawing graphics on a web page, but they are different in terms of how they work, their use cases, and their advantages. Here’s a detailed comparison and explanation of each:

**1. The**<canvas>**Element**

* The <canvas> element in HTML5 is used to draw graphics via JavaScript. It is a bitmap-based element, meaning you can use it to render pixel-based graphics (e.g., images, animations, charts, games, etc.).

**How It Works:**

* The <canvas> element itself is just a container with a specified width and height.
* You draw on the canvas using JavaScript and the Canvas API, which provides a set of methods for drawing shapes, images, and text.
* Once something is drawn on the canvas, it becomes part of the bitmap and cannot be edited as individual elements (unless you clear or redraw the canvas).

**2. The**<svg>**Element**

* The <svg> (Scalable Vector Graphics) element is used to draw graphics using XML-based markup. Unlike <canvas>, which is pixel-based, <svg> is vector-based, meaning it represents graphics in terms of paths, shapes, and coordinates. This allows for scalability without loss of quality, making SVG perfect for creating crisp graphics at any size.

**How It Works:**

* SVG uses XML markup to describe the shapes and paths in a graphic.
* Each shape or graphic in SVG is a part of the DOM, which means you can manipulate them using CSS or JavaScrip .

.

* You can apply styles (like color, borders) and animations directly within the SVG code.