# Module5 : HTML5

1. What are the new tags added in HTML5 ?

ANS:- HTML5 introduced several new semantically meaningful tags. These include <section>, <header>, <footer> <nav>, <mark>, <figure>, <aside> <figcaption>, <data>, <time>, <output>, <progress>, <meter> and <main>.

1. **<header>**: Used to define introductory content within a section or page, typically containing headings, logos, navigation links, etc.
2. **<footer>**: Represents a footer for a section or the whole document, typically containing copyright information, authorship, links to related documents, etc.
3. **<nav>**: Defines a section with navigation links, such as menus or tables of contents.
4. **<time>**: Represents a specific period in time or a date.
5. **<meter>**: Represents a scalar measurement within a known range, such as disk usage or relevance of a search result.
6. **<aside>**: Defines content aside from the main content, often presented as sidebars or call-out boxes.
7. **<progress>**: Represents the progress of a task, such as loading of an image or the progress of a quiz.

These tags help in making HTML documents more structured and semantically meaningful, aiding both search engines and accessibility technologies in understanding the content and its structure.

1. How to embed audio and video in a webpage ?

ANS:- To embed audio and video in a webpage using HTML5, you can use the <audio> and <video> elements respectively.

Example video:

<video controls width="500" height="300">

<source src="video\_file.mp4" type="video/mp4">

Your browser does not support the video tag.

</video>

* The <video> element is used to embed video content in a webpage.
* The controls attribute adds basic playback controls (play, pause, volume control, etc.) to the video player.
* The width and height attributes define the dimensions of the video player.
* The text "Your browser does not support the video tag." is displayed if the browser does not support the <video> element or any of the specified formats.

Example audio:

<audio controls>

<source src="audio\_file.mp3" type="audio/mpeg">

Your browser does not support the audio element.

</audio>

* The <audio> element is used to embed audio content in a webpage.
* The controls attribute adds basic playback controls (play, pause, volume control, etc.) to the audio player.
* The text "Your browser does not support the audio element." is displayed if the browser does not support the <audio> element or any of the specified formats.

1. Semantic element in HTML5 ?

ANS:- The semantic element clearly describes its meaning to both the browser and the developer.

There are two type of semantic element.

1.semantic elements

2.non-semantic elements

* **<header>**: Defines a header for a section or page, typically containing introductory content like headings, logos, navigation links, etc.
* **<nav>**Represents a section with navigation links, such as menus or tables of contents.
* **<main>**: Specifies the main content of the <body> of a document, useful for screen readers and other assistive technologies.
* **<section>**: Represents a thematic grouping of content within a document, typically with a heading.
* **<figcaption>**: Provides a caption or legend for a <figure> element.
* **<mark>**: Highlights text with a different background color or style, typically for indicating a portion of text to which attention is drawn.
* **<meter>**: Represents a scalar measurement within a known range, such as disk usage or relevance of a search result.

1. Canvas and SVG tags.

ANS:- The <canvas> and <svg> (Scalable Vector Graphics) tags are both used in HTML5 for creating graphics and visual content on web pages, but they have different approaches and use cases.

<canvas> Tag :-

**<canvas>**: Use when you need to draw complex graphics or animations that require frequent updates and interactions. It's more suitable for dynamic content where pixel-level control and performance are critical.

* **Purpose**: The <canvas> element is used to draw graphics, animations, and dynamic content on the fly using JavaScript.
* **Drawing Context**: It provides a drawing context (2D or 3D) which allows you to use JavaScript to draw shapes, paths, text, images, and more.
* **Examples**: Charts, animations, games, image manipulation tools.

Example of canvas tag :

<canvas id="myCanvas" width="400" height="200"></canvas>

<script>

var canvas = document.getElementById('myCanvas');

var ctx = canvas.getContext('2d');

ctx.fillStyle = 'green';

ctx.fillRect(10, 10, 100, 100);

</script>

<svg> Tag :-

**<svg>**: Use when you need scalable vector graphics that can be easily styled and manipulated using CSS or JavaScript. It's ideal for static or interactive graphical content that needs to maintain quality at various sizes.

* **Purpose**: The <svg> element is used to define vector-based graphics for the web.
* **DOM Elements**: SVG graphics are part of the HTML document's DOM and can be manipulated with CSS and JavaScript.
* **Interactivity**: Supports event handlers and can be animated using CSS animations or JavaScript.
* **Examples**: Icons, logos, scalable illustrations, data visualizations.

Example of svg tag :

<svg width="400" height="200">

<rect x="50" y="20" width="150" height="100" fill="blue" />

<circle cx="250" cy="70" r="50" fill="red" />

<text x="100" y="180" font-family="Arial" font-size="20" fill="black">SVG Example</text>

</svg>

Both <canvas> and <svg> have their strengths and are often used together in web applications where each is best suited to different parts of the interface or functionality.