**Module (HTML5)**

**• What are the new tags added in HTML5?**

**List of new semantic elements**

**The semantic elements added in HTML5 are:**

* **<article>**
* **<aside>**
* **<details>**
* **<figcaption>**
* **<figure>**
* **<footer>**
* **<header>**
* **<main>**
* **<mark>**
* **<nav>**
* **<section>**
* **<summary>**
* **<time>**

**Elements such as <header>, <nav>, <section>, <article>, <aside>, and <footer> act more or less like <div> elements. They group other elements together into page sections. However where a <div> tag could contain any type of information, it is easy to identify what sort of information would go in a semantic <header> region.**

• How to embed audio and video in a webpage?

**embed audio in HTML?**

To embed audio in HTML, we use the <audio> tag.

Before HTML5, audio cannot be added to web pages in the Internet Explorer era. To play audio, we used web plugins like Flash.

After the release of HTML5, it is possible.

<audio>

<source src="file\_name" type="audio\_file\_type">

</audio>

### embed video in HTML?

To embed video in HTML, we use the <video> tag.

It contains one or more video sources at a time using <source> tag.

**Syntax**

<video>

<source src="file\_name" type="video\_file\_type">

</video>

**• Semantic element in HTML5?**

A semantic element clearly describes its meaning to both the browser and the developer.

Examples of non-semantic elements: <div> and <span> - Tells nothing about its content.

Examples of semantic elements: <form>, <table>, and <article> - Clearly defines its content.

some semantic elements:

* <article>
* <aside>
* <details>
* <figure>
* <footer>
* <header>
* <main>
* <mark>
* <nav>
* <section>

**• Canvas and SVG tags??**

The <canvas> and <svg> tags are both commonly used in web development for creating graphics and visualizations, but they have different purposes and functionality

**Canvas**:

The **<canvas>** element is an HTML element that provides a drawable region defined in a resolution-dependent bitmap canvas.

It's primarily used for dynamic rendering of graphics using JavaScript.

With the canvas API, you can draw shapes, lines, text, images, and apply transformations to create complex visualizations.

The canvas is essentially a blank slate where you can paint whatever you want programmatically.

This makes it suitable for interactive games, data visualizations, and other complex graphics.

Html code:

<canvas id="myCanvas" width="500" height="500"></canvas>

**SVG**:

The **<svg>** element is an XML-based vector graphic format for two-dimensional graphics with support for interactivity and animation.

SVG stands for Scalable Vector Graphics. Unlike the canvas, SVG is resolution-independent, meaning it scales well and retains quality regardless of the display resolution.

SVG is often used for static graphics, such as icons, logos, illustrations, and charts, and it can be created and edited with vector graphic editing software like Adobe Illustrator or Inkscape.

html code:

<svg width="500" height="500"> <rect x="50" y="50" width="100" height="100" fill="red"/> </svg>