**Q 1. What are the new tags added in HTML5?**

**Ans :** **There is a new tags of HTML5**

**<article> --> Represents an independent piece of content of a document, such as a blog entry or newspaper article**

**<audio> --> Defines an audio file.**

**<canvas> --> This is used for rendering dynamic bitmap graphics on the fly, such as graphs or games.**

**<footer> --> Represents a footer for a section and can contain information about the author, copyright information, et cetera.**

**<mark> --> Represents a run of text in one document marked or highlighted for reference purposes, due to its relevance in another context.**

**<video> --> Defines a video file.**

**<progress> --> Represents a completion of a task, such as downloading or when performing a series of expensive operations.**

**Q 2. How to embed audio and video in a webpage?**

**Ans :** **Video:-**

**ANS:- <video controls width="500" height="300">**

**<source src="video-file.mp4" type="video/mp4">**

**<source src="./pexels-ivan-samkov-6955102-1920x1080-25fps.mp4" type="video/webm">**

**</video>**

**Audio:-**

**Ans:-**

**<audio controls>**

**<source src="audio-file.mp3" type="audio/mpeg">**

**<source src="./town-10169.mp3" type="audio/ogg">**

**</audio>**

**Q 3 : Semantic element in HTML5?**

**ANS:- In HTML5, semantic elements are specific HTML tags that carry meaning and describe the structure of the content they surround. These elements provide contextual information about the content, making it easier for search engines, screen readers, and other technologies to understand and interpret the web page.**

**<header>: Represents the introductory content or a container for the site's header, typically containing logos, navigation menus, or other site-specific information.**

**<nav>: Defines a section of navigation links, such as a menu or a list of links to other pages.**

**<main>: Represents the main content of a document. It should be unique to the document and not contain any content that is repeated across a set of documents such as site navigation, headers, or footers.**

**<article>: Represents a self-contained composition in a document, such as a blog post, a news article, or a forum post.**

**<section>: Defines a standalone section within a document. It can be used to group related content together.**

**<aside>: Represents content that is tangentially related to the main content, such as sidebars, pull quotes, or advertisements.**

**<footer>: Defines the footer of a document or a section. It typically contains information about the author, copyright information, links to related documents, or contact information.**

**<figure> and <figcaption>: <figure> is used to encapsulate media content, such as images or videos, along with an optional caption provided by the <figcaption> element.**

**These semantic elements help both developers and assistive technologies understand the structure and purpose of the content, making the web and accessible and improving search engine optimization (SEO).**

**Q 4 : Canvas and SVG tags**

**ANS:-** **<canvas>: The <canvas> element is part of the HTML5 specification and provides a drawing surface for rendering graphics dynamically with JavaScript. It allows you to draw and animate graphics, create interactive games, charts, and visualizations. The content inside a <canvas> element is rendered using JavaScript and the HTML5 Canvas API, which provides methods for drawing shapes, lines, text, images, and applying transformations. The canvas is a bitmap-based technology, meaning that once something is drawn, it is essentially a pixel representation and cannot be directly manipulated as individual elements. To create complex visuals, you need to write JavaScript code to update and redraw the canvas as needed.**

**Example:**

**CANVASE:**-

**HTML**

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

**JAVASCRIPT**

**const canvas = document.getElementById("myCanvas");**

**const ctx = canvas.getContext("2d");**

**<svg>: The <svg> (Scalable Vector Graphics) element is an XML-based markup language for describing two-dimensional vector graphics. It allows you to create and manipulate graphical objects using XML tags and attributes. SVG is resolution-independent, meaning that graphics scale smoothly to any size without losing quality. With SVG, you can draw shapes, lines, curves, text, and apply transformations. You can also add interactivity and animation using JavaScript or CSS.**

**Example:**

**SVG:-**

**HTML:-**

**<svg xmlns="http://www.w3.org/2000/svg" width="500" height="500"></svg>**