***Web Designing Assignment***

***Module (HTML5) – 3***

1.What are the new tags added in HTML5?

* HTML5 introduced several new tags to enhance web development. Some of the key new tags include:

1. <article>
2. <section>
3. <nav>
4. <header>
5. <footer>
6. <aside>
7. <figure>
8. <figcaption>
9. <main>
10. <mark>
11. <time>
12. <progress>
13. <meter>

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

* To embed audio and video in a webpage, you can use HTML5’s <audio> and <video> elements.
* Key Attributes:

1. Controls: Adds play, pause, and volume controls.
2. Width: Sets the width of the video player.
3. Src: Specifies the path to the audio or video file.
4. Type: Indicates the format of the media file.

* Tips:

1. Formats: Use multiple formats for better browser compatibility.
2. Responsive Design: Use CSS to make your audio and video players responsive if needed.
3. Accessibility: Provide alternative text for better accessibility.

3.Semantic element in HTML5?

* Semantic elements in HTML5 are designed to clearly describe their meaning in a human and a machine-readable way. Some common Semantic elements:

1. <header>
2. <nav>
3. <main>
4. <article>
5. <section>
6. <aside>
7. <footer>
8. <figure>
9. <figcaption>
10. <time>

4.Canvas and SVG tags?

* Canvas and SVG are both used for rendering graphics in web applications, but they have different approaches and use cases.
* Canvas:

1. Element <canvas>
2. Rendering: Immediate mode, graphics are drawn pixel by pixel.
3. Use Canvas: Ideal for animations, games and applications requiring dynamic and real-time rendering.
4. API: Uses a JavaScript API for drawing shapes, images, and text.
5. Performance: Generally faster for rendering many objects due to less overhead, but can be complex for interactive graphics.
6. Accessibility: Not inherently accessible, requires additional work for screen readers.

* SVG:

1. Elements: <svg>
2. Rendering: Retained mode, graphics are defined in XML and can be manipulated via the DOM.
3. Use Cases: Best for static graphics, icons, and graphics that need to be scalable without loss of quality.
4. API: Can be styled with CSS and manipulated with JavaScript.
5. Performance: May be slower with a large number of elements due to DOM overhead.
6. Accessibility: More accessible by default, elements can be made interactive and are part of the document structure.

* Use Canvas for high-performance graphics and complex animations.
* Use SVG for scalable graphics, simpler shapes, and scenarios where you need to manipulate graphics with CSS and the DOM.