HTML 5

Difference b/w HTML & HTML5?

1. **Doctype Declaration**:
   * **HTML**: The doctype declaration for HTML is more complex, e.g., **<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">**.
   * **HTML5**: The doctype is simplified to **<!DOCTYPE html>**.
2. **New Elements**:
   * **HTML**: HTML has a limited set of elements.

**HTML5**: Introduces new semantic elements such as **<header>**, **<footer>**, **<article>**, **<section>**, **<nav>**, and **<aside>**, which help in structuring the content more meaningfully.

1. **Multimedia Support**:
   * **HTML**: Multimedia content (like audio and video) required third-party plugins (e.g., Flash).
   * **HTML5**: Provides native support for audio and video with the **<audio>** and **<video>** elements, allowing for easier embedding and control.
2. **APIs and Features**:
   * **HTML**: Limited in terms of APIs.
   * **HTML5**: Introduces several APIs, such as the Canvas API for drawing graphics, the Geolocation API for location-based services, and the Web Storage API for storing data locally in the browser.
3. **Form Enhancements**:
   * **HTML**: Forms were limited in functionality.
   * **HTML5**: Introduces new input types (e.g., **email**, **date**, **range**, **color**) and attributes (e.g., **placeholder**, **required**, **autofocus**) that enhance form usability and validation.
4. **Error Handling**:
   * **HTML**: Error handling was more complex and varied by browser.
   * **HTML5**: Provides better error handling and is designed to be more forgiving of errors, allowing browsers to render content even if there are minor mistakes in the markup.
5. **Backward Compatibility**:
   * **HTML**: Older versions of HTML may not be fully compatible with modern browsers.
   * **HTML5**: Designed to be backward compatible with older HTML versions, ensuring that older content can still be rendered correctly.

**What are the additional tags used in HTML5?**

1. **Semantic Elements**:
   * **<header>**: Represents the introductory content or a set of navigational links.
   * **<footer>**: Represents the footer for its nearest sectioning content or sectioning root element.
   * **<article>**: Represents a self-contained composition in a document, such as a blog post or news article.
   * **<section>**: Represents a thematic grouping of content, typically with a heading.
   * **<nav>**: Represents a section of navigation links.
   * **<aside>**: Represents content that is tangentially related to the content around it, often used for sidebars.
   * **<main>**: Represents the main content of the document, excluding headers, footers, and sidebars.
2. **Multimedia Elements**:
   * **<audio>**: Used to embed sound content in documents.
   * **<video>**: Used to embed video content in documents.
   * **<track>**: Used to specify text tracks for **<video>** and **<audio>** elements, such as subtitles or captions.
3. **Graphics and Animation**:
   * **<canvas>**: Used to draw graphics on the fly via scripting (usually JavaScript).
   * **<svg>**: Used to define vector-based graphics directly in the document.
4. **Form Enhancements**:
   * New input types:
     + **<input type="email">**: For email addresses.
     + **<input type="date">**: For date selection.
     + **<input type="time">**: For time selection.
     + **<input type="range">**: For selecting a value from a range.
     + **<input type="color">**: For selecting a color.
   * **<datalist>**: Provides an autocomplete feature for **<input>** elements.
   * **<output>**: Represents the result of a calculation or user action.
5. **Other Elements**:
   * **<progress>**: Represents the progress of a task.
   * **<meter>**: Represents a scalar measurement within a known range, or a fractional value.
   * **<details>**: Represents a disclosure widget from which the user can obtain additional information or controls.
   * **<summary>**: Represents a summary or heading for a **<details>** element.