

# **WEB DEVELOPMENT ASSIGNMENT-1**

**Q.1 <!DOCTYPE html> is it a tag of html? If not, what is it and why do we use it?**

<!DOCTYPE html> is not a tag in HTML. It is called a document type declaration or doctype declaration. It is used at the beginning of an HTML document to specify the version of HTML

The doctype declaration is not considered a tag because it doesn't have an opening or closing tag structure like other HTML elements.

**Q.2 Explain Semantic tags in html? And why do we need it?**

Semantic tags are introduced in HTML5. Semantic tags in HTML are a set of elements that provide meaning and structure to the content within a web page. These tags primarily define the appearance or formatting of the content, semantic tags convey the purpose or role of the content to both browsers and developers. the use of semantic tags promotes better **organization, clearer structure, and enhanced** accessibility in HTML documents.

Commonly used tags are:

**<header>**: Represents the introductory section or container at the top of a web page, typically containing branding, navigation, and introductory content.

**<nav>**: Represents a section of the web page that contains navigation links or menus.

**<section>**: Represents a standalone section of content within a web page, such as chapters, tabbed content, or groupings of related content.

**<article>**: Represents a self-contained composition or content that can be independently distributed or syndicated, such as blog posts, news articles, or forum posts.

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

**<footer>**: Represents the closing section or container at the bottom of a web page, typically containing copyright information, contact details, or related links.

**Q.3 Differentiate between HTML Tags and Elements?**

HTML tags are represented by keywords surrounded by angle brackets, such as <tagname>. They usually come in pairs, consisting of an opening tag and a closing tag, with content placed between them.

HTML Elements: HTML elements are made up of HTML tags, along with their content and any attributes. An HTML element consists of the opening tag, the content, and the closing tag. The content can include text, other nested elements, or a combination of both.

**<p>This is paragraph-1.</p>**

The <p> tags represent the opening and closing tags of the <p> element, and the content "This is a paragraph." is the text content within the <p> element. So, the entire <p> element, including the tags and content, forms the HTML element.

**Q.6 What are some of the advantages of HTML5 over its previous versions?**

HTML5 introduced several advancements and improvements over its previous versions. Some of the advantages of HTML5 include:

**1. Semantic markup and structure:** HTML5 introduced new semantic elements, such as ``<header>``, ``<nav>``, ``<article>``, ``<section>``, etc., which provide a clearer structure and meaning to the content. This improves accessibility, search engine optimization (SEO), and facilitates better organization and understanding of web page content.

**2. Offline web application support:** HTML5 introduced the Application Cache and Local Storage features, which allow web applications to store data locally on the user's device. This enables web applications to work offline and provides a better user experience by reducing the dependency on constant internet connectivity.

**3. Enhanced multimedia support:** HTML5 provides native support for audio and video elements, eliminating the need for third-party plugins like Flash. It allows seamless embedding of media content into web pages, making it easier to incorporate multimedia elements.

**4. Simpler and cleaner code:** HTML5 offers simplified and cleaner syntax compared to previous versions of HTML. It introduces new elements, attributes, and APIs that reduce the need for complex workarounds and hacks, leading to more maintainable and efficient code.

#### **Q.8 What is the difference between `<figure>` tag and `<img>` tag?**

The ``<figure>`` tag and ``<img>`` tag serve different purposes in HTML.

The ``<img>`` tag is used specifically for embedding images into an HTML document. It is a self-closing tag and requires the ``src`` attribute to specify the source URL of the image file. It represents a standalone image within the document.

Example:

```
html

```

On the other hand, the ``<figure>`` tag is used to group together and represent self-contained content, typically with a caption. It can contain various types of content, including images, illustrations, videos, code snippets, etc. The purpose of the ``<figure>`` element is to provide a semantic container for content that is referenced or explained in the document.

Example:

```
html
<figure>
  
  <figcaption>Caption for the image</figcaption>
</figure>
```

In the ``<figure>`` example, the ``<img>`` tag represents the image, and the ``<figcaption>`` tag is used to provide a caption for the image. The ``<figure>`` element

helps convey the relationship between the content and its caption, improving accessibility and understanding for both humans and machines.

In summary, while the `<img>` tag is specifically for embedding standalone images, the `<figure>` tag is used to group content with a caption, allowing for more meaningful and structured representation of content within an HTML document.

### **Q.9 What's the difference between html tag and attribute and give example of some global attributes?**

In HTML, a tag is used to define an element and provides the structure and meaning to the content within it. Tags are represented by opening and closing brackets, such as `<tag>` and `</tag>`. Examples of HTML tags include `<p>`, `<div>`, `<h1>`, and `<a>`.

On the other hand, attributes provide additional information or properties to the HTML elements. They are used within the opening tag of an element and are comprised of a name-value pair. Attributes modify the behavior or appearance of an element. Some attributes are specific to certain tags, while others can be used with multiple tags.

Examples of global attributes that can be used with various HTML tags include:

1. class: Specifies one or more class names for an element, allowing the element to be styled or targeted with CSS. Example: `<div class="container">`

2. id: Provides a unique identifier for an element, which can be used to target and manipulate the element with JavaScript or CSS.

Example: `<h1 id="title">`

3. style: Defines inline CSS styles for an element, allowing you to specify custom styling directly within the HTML.

Example: `<p style="color: blue;">`

4. data-\*: Allows you to store custom data attributes on an element for scripting or styling purposes.

Example: `<button data-action="submit">`

5. src: Specifies the source URL for embedded content, such as images or media files.

Example: ``