1.How are inline and block elements different from each other?

In simple, a <span> element is used as an inline element and a <div> element as a block level element.

An inline element does not cause a line break (start on a new line) and does not take up the full width of a page, only the space bounded by its opening and closing tag. It is usually used within other HTML elements.

Example:

<img> <a>

A block-level element always starts on a new line and takes up the full width of a page, from left to right. A block-level element can take up one line or multiple lines and has a line break before and after the element.

Block elements example.

* Heading tags <h1> to <h6>
* List (Ordered, Unordered, Description and List Item) tags <ol> , <ul> ,<dl> , <li>
* Pre-formatted text tag <pre>
* Blockquote tag <blockquote>

2.Explain the difference between visibility:hidden and display:none

display: none; is commonly used with JavaScript to hide and show elements without deleting and recreating them. Take a look at our last example on this page if you want to know how this can be achieved.

The <script> element uses display: none; as default.

Visibility: hidden

The element box is invisible (not drawn), but still affects layout as normal. Descendants of the element will be visible if they have visibility set to visible.

3. Explain the clear and float properties.

**Float** is a CSS positioning property. To understand its purpose and origin, we can look to print design. In a print layout, images may be set into the page such that text wraps around them as needed. This is commonly and appropriately called "text wrap".

Float's sister property is clear. An element that has the clear property set on it will not move up adjacent to the float like the float desires, but will move itself down past the float.

4. explain difference between absolute, relative,fixed and static.

## **position: static**

HTML elements are positioned static by default.

Static positioned elements are not affected by the top, bottom, left, and right properties.

An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page.

## **position: relative;**

An element with position: relative; is positioned relative to its normal position.

Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.

## **position: fixed;**

An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.

A fixed element does not leave a gap in the page where it would normally have been located.

## **position: absolute;**

An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.

6. Why do we use meta tags?

Metadata is data (information) about data.

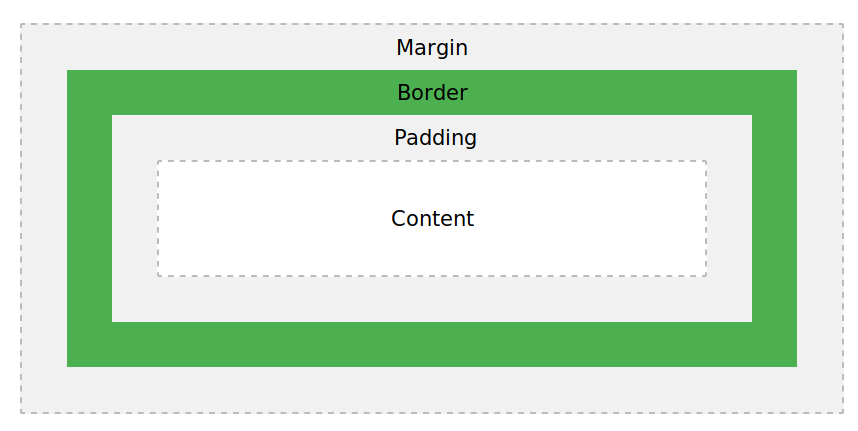
The <meta> tag provides metadata about the HTML document. Metadata will not be displayed on the page, but will be machine parsable.

Meta elements are typically used to specify page description, keywords, author of the document, last modified, and other metadata.

The metadata can be used by browsers (how to display content or reload page), search engines (keywords), or other web services.

HTML5 introduced a method to let web designers take control over the viewport (the user's visible area of a web page), through the <meta> tag (See "Setting The Viewport" example below).

7. Explain box model.



Explanation of the different parts:

* **Content** - The content of the box, where text and images appear
* **Padding** - Clears an area around the content. The padding is transparent
* **Border** - A border that goes around the padding and content
* **Margin** - Clears an area outside the border. The margin is transparent

The box model allows us to add a border around elements, and to define space between elements.

8. What are the different types of CSS Selectors?

CSS selectors are used to select the content you want to style. Selectors are the part of CSS rule set. CSS selectors select HTML elements according to its id, class, type, attribute etc.

There are several different types of selectors in CSS.

1. CSS Element Selector
2. CSS Id Selector
3. CSS Class Selector
4. CSS Universal Selector
5. CSS Group Selector

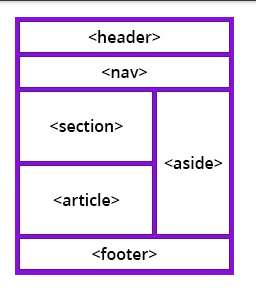
9. Define Doctype.

The <!DOCTYPE> declaration is not an HTML tag; it is an instruction to the web browser about what version of HTML the page is written in.

In HTML 4.01, the <!DOCTYPE> declaration refers to a DTD, because HTML 4.01 was based on SGML. The DTD specifies the rules for the markup language, so that the browsers render the content correctly.

HTML5 is not based on SGML, and therefore does not require a reference to a DTD.

10. Explain 5 HTML5 semantic tags.



In earlier versions of HTML, there were no globally accepted names for structural elements, and each developer used their own. That made it very hard for search engines to index web page content correctly.

When a browser communicates with the code, it looks for some specific information to help with the display. Hence, HTML5 introduced a consistent list of semantic elements to help search engines and developers.

HTML5 semantic tags define the purpose of the element. By using semantic markup, you help the browser understand the meaning of the content instead of just displaying it. By providing this extra level of clarity, HTML5 semantic elements also help search engines to read the page and find the required information faster.