**Difference between HTTP and HTTPS?**

HTTP used for transmits data in plain text, That is where HTTPS comes in.

HTTP by using SSL for security protocol, encryption, data confidentiality, integrity, and authentication.

HTTPS is securing sensitive information, boosting user trust, and with data protection.

**What are semantic HTML elements?**

Semantic HTML elements clearly describe their meaning to both the browser and the developer.

Eg.

* <header>: Represents introductory content.
* <article>: Represents a self-contained piece of content.
* <nav>: Defines navigation links.
* <footer>: Represents the footer of a document or section.

**What is the box model in CSS?**

* Content: The actual content of the box (text, images).
* Padding: Space between the content and the border, inside the box.
* Border: The outer line around the padding (if any).
* Margin: Space outside the border, separating the element from others.

**difference between == and === in JavaScript.**

* == is the equality operator that checks for value equality, performing type coercion if necessary.
* === is the strict equality operator that checks for both value and type equality without coercion. It's generally recommended to use === to avoid unexpected results.

**What is AJAX? How does it work?**

AJAX (Asynchronous JavaScript and XML) is a technique for creating dynamic web applications. This is typically using the **XMLHttpRequest** object or the fetch API to communicate with a server and update parts of a web page.

**How do you optimize a website for performance?**

* Minimize HTTP requests: Combine files (CSS, JS) to reduce the number of requests.
* Optimize images: Use appropriate formats and compression.
* Use caching: Leverage browser caching and CDNs (Content Delivery Networks).
* Minify resources: Remove unnecessary characters from code (HTML, CSS, JS).
* Load scripts asynchronously: Use the **async** or **defer** attributes in <script> tags.

**What is the difference between inline, block, and inline-block**

* **Inline elements** (e.g., <span>, <a>) do not start on a new line and only take up as much width as necessary. **Margins and paddings will not affect other elements.**
* **Block elements** (e.g., <div>, <p>) start on a new line and take up the full width available. **They can have margins and paddings that affect surrounding elements.**
* **Inline-block elements** (e.g., set with display: inline-block) are like inline elements but can have width and height. **They allow elements to sit next to each other while still respecting margins and paddings.**

**What are the advantages of using a CSS framework?**

* **Consistency**: They provide a standard design language across your project.
* **Efficiency**: Pre-defined styles and components can speed up development.
* **Cross-browser compatibility**: Frameworks are usually tested on multiple browsers.
* **Responsive design**: Most frameworks include responsive design features out of the box.

**What is a RESTful API?**

A RESTful API (Representational State Transfer) is an architectural style for designing networked applications. It uses standard HTTP methods (GET, POST, PUT, DELETE). RESTful APIs typically communicate using JSON or XML.

**How can you improve the SEO of a website?**

* Use semantic HTML and proper heading structures.
* Optimize page load speed and mobile responsiveness.
* Include relevant keywords in titles, meta descriptions, and content.
* Use alt attributes for images and descriptive anchor text for links.
* Create a sitemap and use robots.txt to guide search engines.

**What is the purpose of the alt attribute in images?**

The alt attribute provides alternative text for an image. It improves accessibility for screen readers and helps search engines understand the content of the image. If an image fails to load, the alt text will be displayed instead.

**What is the significance of the viewport meta tag?**

The viewport meta tag controls the layout on mobile browsers. It allows developers to specify how their content should be scaled and displayed on different devices. A common usage is <meta name="viewport" content="width=device-width, initial-scale=1.0">, which makes the page responsive.

**What is the role of the z-index property in CSS?**

The z-index property controls the stacking order of overlapping elements. It only works on positioned elements (those with position set to relative, absolute, or fixed). Higher z-index values appear in front of lower ones.