¿Qué es HTML5 y algunas de sus características principales?

HTML5 es la última versión del lenguaje de marcado HTML, que estructura el contenido web. Es como el esqueleto de una página web, pero en lugar de huesos, está compuesto de etiquetas que definen diferentes tipos de contenido y estructuras. Las características destacadas de HTML5 incluyen:

1. Nuevas Etiquetas Semánticas: Introduce etiquetas como <header>, <footer>, <article>, y <section>, que ayudan a estructurar mejor el contenido y mejorar la accesibilidad y el SEO.

2. Soporte para Multimedia: Ofrece elementos nativos como <video> y <audio> para manejar multimedia sin necesidad de plugins externos.

3. Canvas y SVG: Permite el dibujo dinámico y la creación de gráficos mediante el <canvas> y SVG (Scalable Vector Graphics), ideal para gráficos interactivos y visualizaciones.

4. Almacenamiento Local: Incluye API como Local Storage y Session Storage para guardar datos en el navegador, lo que facilita el desarrollo de aplicaciones web más avanzadas.

5. Compatibilidad con Aplicaciones Offline: La API de Web Storage y la API de Cache permiten que las aplicaciones web funcionen sin conexión a internet.

Explica la diferencia entre CSS Grid y Flexbox y cuándo usarías uno sobre el otro.

CSS Grid: Es un sistema de diseño bidimensional que permite crear complejas estructuras de cuadrículas en una página. CSS Grid es ideal para diseñar layouts más complejos y estructurados, como las páginas de un periódico o un dashboard, donde necesitas alinear elementos en ambas direcciones (horizontal y vertical).

Flexbox: Es un sistema de diseño unidimensional que se centra en la alineación de elementos en una sola dirección, ya sea en una fila o en una columna. Es para diseños más simples y lineales, como un menú de navegación o una lista de ítems donde los elementos deben distribuirse uniformemente o alinear de manera flexible en una sola línea.

¿Cuándo usar uno sobre el otro?

- Usa CSS Grid cuando necesites un layout más estructurado y con mayor control sobre ambas dimensiones (filas y columnas).

Usa Flexbox para diseños más simples y lineales, especialmente cuando solo necesitas ajustar los elementos en una sola dirección.

¿Qué es el responsive design y por qué es importante?

El diseño responsive (o diseño adaptable) es un enfoque de diseño web que asegura que una página web se vea y funcione bien en una variedad de dispositivos y tamaños de pantalla. En vez de crear versiones separadas de un sitio para diferentes dispositivos, el diseño responde automáticamente al tamaño de la pantalla y otras características del dispositivo.

Importancia:

1. Accesibilidad Universal: Permite que los usuarios accedan al contenido de tu sitio web en cualquier dispositivo, desde un teléfono móvil hasta un monitor de escritorio, mejorando la experiencia del usuario.

3. Costos Eficientes: Diseñar un solo sitio web que se adapte a todos los dispositivos es más económico que desarrollar y mantener versiones separadas para cada tipo de dispositivo.

4. Mejor Experiencia de Usuario: Garantiza una experiencia coherente y atractiva sin importar cómo accedan los usuarios a tu sitio, lo que puede aumentar la retención y la satisfacción del usuario.

¿Cuál es la diferencia entre JavaScript y TypeScript?

JavaScript es un lenguaje de programación que se ejecuta en el navegador y es esencial para el desarrollo web moderno. Permite la creación de interactividad y dinámica en las páginas web.

TypeScript es un super de JavaScript que añade tipado estático opcional al lenguaje. Básicamente, TypeScript es JavaScript con tipos. Se compila a JavaScript y se usa para mejorar el desarrollo de aplicaciones grandes y complejas al permitirte definir tipos y estructuras más claras.

Diferencias clave:

Tipado Estático vs Dinámico: TypeScript ofrece tipado estático, lo que significa que puedes definir los tipos de variables, parámetros y objetos. Esto ayuda a detectar errores en tiempo de compilación y mejora la autocompletación en editores de código. JavaScript es dinámico y no tiene este tipo de chequeo en tiempo de compilación.

Desarrollo de Grandes Aplicaciones: TypeScript es muy útil en proyectos grandes y colaborativos donde el mantenimiento y la escalabilidad son cruciales, gracias a su capacidad de definir interfaces y tipos que facilitan la comprensión y gestión del código.

Compatibilidad: Todo código TypeScript es JavaScript válido, pero para ejecutarlo en un navegador o entorno JavaScript, debe ser compilado a JavaScript, mientras que JavaScript se ejecuta directamente en el navegador.

¿Cuándo usar TypeScript?

- Si estás trabajando en un proyecto grande, TypeScript puede proporcionar ventajas significativas en términos de mantenimiento y prevención de errores.

- Para proyectos más pequeños o cuando se busca rapidez en el desarrollo sin necesidad de una compleja infraestructura de tipos, JavaScript puede ser suficiente.