Web avanzada – taller 2 – corte 1 – código 989389

Por: Moises Zabaleta Cruz

Responderé cada pregunta detalladamente:

1. La principal diferencia entre JS Vanilla y Node JS:

   - JS Vanilla (JavaScript puro) se ejecuta en el navegador del cliente (frontend) y se utiliza para manipular el DOM y crear interactividad en el lado del cliente.

   - Node JS es un entorno de ejecución que permite ejecutar JavaScript en el servidor (backend), permitiendo acceso al sistema de archivos, manejo de solicitudes HTTP, conexiones a bases de datos, etc.

2. Servicio Web, API y REST:

   - Un servicio web es una tecnología que permite la comunicación e intercambio de datos entre aplicaciones a través de protocolos web estándar.
   - Una API (Application Programming Interface) es un conjunto de reglas y protocolos que permite la comunicación entre diferentes componentes de software.
   - REST (Representational State Transfer) es un estilo arquitectónico para diseñar servicios web que utiliza HTTP como protocolo de comunicación y define un conjunto de principios y restricciones para crear servicios web escalables.

3. Los dos principales formatos de datos que retorna un servicio Web son:

   - JSON (JavaScript Object Notation)
   - XML (Extensible Markup Language)

4. Ventajas de aplicaciones basadas en servicios web:

   - Desacoplamiento entre frontend y backend
   - Escalabilidad mejorada
   - Reutilización de servicios
   - Independencia de tecnología (el frontend puede estar en cualquier lenguaje/framework)
   - Mejor mantenibilidad
   - Facilidad para implementar arquitecturas distribuidas
   - Posibilidad de crear múltiples clientes (web, móvil, desktop) consumiendo los mismos servicios

5. Archivo JSON: Un archivo JSON es un formato almacenamiento de texto ligero para el intercambio de datos organizados. Ejemplo:

```json
{
  "nombre": "Juan Pérez",
  "edad": 30,
  "activo": true,
  "hobbies": ["lectura", "deportes", "música"],
  "direccion": {
    "calle": "Av. Principal",
    "numero": 123,
    "ciudad": "Madrid"
  }
}
```

6. Diferencias entre paquete, librería y módulo:

   - Módulo: Unidad de código reutilizable que encapsula funcionalidad relacionada
   - Librería: Colección de módulos relacionados que proporcionan funcionalidades específicas
   - Paquete: Conjunto de archivos y recursos que incluyen librerías y módulos, junto con metadatos y dependencias

7. AJAX (Asynchronous JavaScript And XML): AJAX permite realizar solicitudes asíncronas al servidor sin recargar la página. Ejemplo de implementación usando Fetch:

```javascript
fetch('https://api.ejemplo.com/datos', {
  method: 'GET',
  headers: {
    'Content-Type': 'application/json'
  }
})
.then(response => response.json())
.then(data => {
  console.log(data);
})
.catch(error => console.error('Error:', error));
```

8. Estrategias para pintar datos de un servicio REST en HTML:

   - ❖ Manipulación del DOM: Crear elementos HTML dinámicamente usando JavaScript
   - ❖ Template Strings: Crear plantillas HTML con interpolación de variables
   - ❖ innerHTML: Insertar HTML directamente en un elemento

Ejemplo:

```
fetch('https://api.ejemplo.com/usuarios')
  .then(response => response.json())
  .then(usuarios => {
    const contenedor = document.getElementById('usuarios');
    const html = usuarios.map(usuario => `
      <div class="usuario">
        <h2>${usuario.nombre}</h2>
        <p>${usuario.email}</p>
      </div>
    `).join(' ');
    contenedor.innerHTML = html;
  });
```