Taller – Consumo de API's

Objetivo general

El objetivo de este taller es integrar conocimientos de HTML, CSS y JavaScript para diseñar una página web responsiva que consuma datos desde una API externa y los muestre de forma clara y organizada en la interfaz.

Enunciado del reto

Diseñar y desarrollar una página web estática (HTML + CSS + JS) que consuma datos de la API pública: https://jsonplaceholder.typicode.com/users y muestre la información en pantalla con un diseño agradable y adaptado a dispositivos móviles.

Requisitos mínimos

- Estructura semántica en el HTML (uso de <header>, <main>, <section>, <footer>).
- Diseño responsive con CSS (adaptación a escritorio, tablet y móvil).
- Uso de fetch() o async/await en JavaScript para consumir la API.
- Presentación clara de los datos (ejemplo: nombre, correo y ciudad de cada usuario).
- Separación de responsabilidades: HTML (estructura), CSS (estilos), JS (lógica).

Puntos a desarrollar

Punto 1 | Estructura base HTML

Crea un archivo index.html con la estructura semántica mínima: encabezado, sección de contenido principal y pie de página.

Punto 2 | Estilos responsivos con CSS

Crea un archivo styles.css con estilos que permitan que la página se vea bien tanto en escritorio como en dispositivos móviles.

- Uso de flexbox o grid.
- Tipografías y colores legibles.

Punto 3 | Consumo de API

Crea un archivo app.js que:

- Use fetch() para obtener los datos de la API.
- Procese y muestre la información en tarjetas dentro del contenido principal.

Punto 4 | Interfaz dinámica

- Cada tarjeta debe mostrar al menos: nombre, email y ciudad del usuario.
- Las tarjetas deben generarse dinámicamente con JavaScript.

Punto 5 | Entregable final

Estructura recomendada del proyecto:

```
users-api/
├── index.html
├── styles.css
└── app.js
```

El diseño de las tarjetas queda sujeto a la autonomía creativa de cada estudiante.

Preguntas de reflexión

- ¿Qué desafíos surgen al trabajar con datos externos provenientes de una API?
- ¿Cómo manejarías los errores si la API no responde o envía datos en un formato inesperado?
- ¿Qué ventajas y limitaciones encuentras en separar la lógica (JS), la estructura (HTML) y los estilos (CSS)?