

Ejercicios fetch

Ejercicio 1: Obtener Información de Usuarios con fetch

Enunciado:

Desarrolla una función que utilice fetch para obtener información de los usuarios desde <https://jsonplaceholder.typicode.com/users>. Luego, muestra en la consola el nombre y correo de cada usuario.

Requisitos:

1. Muestra el nombre y correo de cada usuario en la consola.
2. Si la solicitud falla, muestra un mensaje de error indicando que no se pudo obtener la información.

Ejercicio 2: Publicar Datos en una API con fetch

Enunciado:

Crea un formulario simple en HTML que permita al usuario ingresar un título y un cuerpo de una publicación. Al enviar el formulario, utiliza fetch para hacer una petición POST a <https://jsonplaceholder.typicode.com/posts> con los datos ingresados.

Requisitos:

1. Utiliza el método POST y envía los datos en formato JSON.
2. Muestra en la consola la respuesta del servidor.
3. Si hay un error, muestra un mensaje adecuado en la consola.

Ejercicio 3: Filtrar Resultados de una API con fetch

Enunciado:

Realiza una petición fetch a <https://jsonplaceholder.typicode.com/todos> para obtener una lista de tareas. Luego, filtra y muestra solo aquellas tareas que estén marcadas como completadas.

Requisitos:

1. Muestra las tareas completadas en la consola.
2. Si la solicitud falla, muestra un mensaje de error adecuado.

Ejercicio 4: Manejo de Múltiples Consultas con Promise.all y fetch

Enunciado:

Realiza tres solicitudes fetch a las siguientes URLs:

1. <https://jsonplaceholder.typicode.com/posts>
2. <https://jsonplaceholder.typicode.com/users>
3. <https://jsonplaceholder.typicode.com/comments>

Usa Promise.all para esperar que todas las solicitudes se completen y muestra el número total de publicaciones, usuarios y comentarios en la consola.

Requisitos:

1. Muestra el número total de publicaciones, usuarios y comentarios.
2. Si alguna de las solicitudes falla, captura el error y muestra un mensaje adecuado.

Ejercicio 5: Simulación de Carga de Datos con Indicador de Progreso

Enunciado:

Crea una aplicación que realice una solicitud fetch a <https://jsonplaceholder.typicode.com/photos> para obtener una lista de fotos. Mientras se realiza la solicitud, muestra un mensaje en la consola que diga "Cargando...". Una vez que la solicitud se complete, muestra los títulos de las primeras 10 fotos.

Requisitos:

1. Muestra "Cargando..." antes de que la solicitud se complete.
2. Muestra los títulos de las primeras 10 fotos en la consola.
3. Si hay un error, muestra un mensaje adecuado en la consola.

Ejercicio 6: Actualizar Información con PUT usando fetch

Enunciado:

Utiliza fetch para actualizar la información de un usuario en <https://jsonplaceholder.typicode.com/users/1>. Cambia el nombre del usuario a "Nombre Actualizado" y su correo a "actualizado@correo.com".

Requisitos:

1. Utiliza el método PUT para actualizar los datos.
2. Muestra en la consola la respuesta del servidor.
3. Si la solicitud falla, muestra un mensaje de error adecuado.

Ejercicio 7: Eliminar Datos con DELETE usando fetch

Enunciado:

Realiza una solicitud fetch para eliminar una publicación con ID 1 en <https://jsonplaceholder.typicode.com/posts/1>.

Requisitos:

1. Utiliza el método DELETE para eliminar la publicación.
2. Muestra un mensaje en la consola confirmando que la publicación ha sido eliminada.
3. Si la solicitud falla, muestra un mensaje adecuado en la consola.

Ejercicio 8: Manejo de Errores HTTP con fetch

Enunciado:

Realiza una solicitud fetch a una URL inexistente (<https://jsonplaceholder.typicode.com/invalid-url>). Captura el error y muestra un mensaje personalizado que indique que la página no fue encontrada.

Requisitos:

1. Detecta el error HTTP y muestra un mensaje de "Página no encontrada".
2. Asegúrate de que la aplicación no se detenga al ocurrir el error.

Ejercicio 9: Uso de Parámetros en la URL con fetch

Enunciado:

Realiza una solicitud fetch para obtener los comentarios de la publicación con ID 1 usando <https://jsonplaceholder.typicode.com/comments?postId=1>. Muestra en la consola el nombre del autor y el contenido de cada comentario.

Requisitos:

1. Muestra los comentarios correspondientes en la consola.
2. Si la solicitud falla, muestra un mensaje adecuado en la consola.

Ejercicio 10: Renderizado Dinámico de Datos con fetch

Enunciado:

Crea una aplicación que realice una solicitud fetch a <https://jsonplaceholder.typicode.com/albums> para obtener una lista de álbumes. Luego, muestra los títulos de los primeros 10 álbumes en una lista HTML ().

Requisitos:

1. Inserta los títulos en el DOM dentro de una lista desordenada ().
2. Si hay un error en la solicitud, muestra un mensaje de error en la página.