

# DESARROLLO DE SOFTWARE FRONTEND I

## ESPINOSA MARTINEZ CARLOS LEONEL

Aarón Hernández García

### 5.1 Ejercicio API Productos

## Realizar un análisis del proyecto adjunto que consume la api de productos por medio de fetch de Javascript.

### Objetivo de la Web (fakestoreapi):

Aprender a consumir y manipular datos de una API pública utilizando fetch en JavaScript, manejando respuestas y errores, y mostrando información en una aplicación web interactiva.

### Código HTML:

```
<> index.html X JS script.js # styles.css
<> index.html > html > body > table#tabla-productos > tbody
1
2 <!DOCTYPE html>
3 <html lang="es">
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Tabla de Productos</title>
8   <link rel="stylesheet" href="styles.css">
9 </head>
10 <body>
11
12 <h2>Tabla de Productos</h2>
13 <button id="btn-actualizar">Actualizar</button>
14 <table id="tabla-productos">
15   <thead>
16     <tr>
17       <th>ID</th>
18       <th>Nombre</th>
19       <th>Precio</th>
20       <th>Categoria</th>
21       <th>Imagen</th>
22     </tr>
23   </thead>
24   <tbody>
25   </tbody>
26 </table>
27 <script src="script.js"></script>
28 </body>
29 </html>
30
```

## Código CSS:

```
index.html  JS script.js  # styles.css X

# styles.css > ...
1  *{
2    font-family: Helvetica, sans-serif;
3  }
4
5  table {
6    border-collapse: collapse;
7    width: 100%;
8  }
9  th, td {
10   border: 1px solid #d00505;
11   text-align: left;
12   padding: 8px;
13 }
14 th {
15   background-color: #d28d8d;
16 }
17
```

## Código JS:

```
index.html  JS script.js X  # styles.css

JS script.js > document.addEventListener('DOMContentLoaded') callback > actualizarTabla > then() callback
1  document.addEventListener('DOMContentLoaded', function () {
2    const apiUrl = 'https://fakestoreapi.com/products/';
3    const btnActualizar = document.getElementById('btn-actualizar');
4    const tablaProductos = document.getElementById('tabla-productos');
5    function limpiarTabla() {
6      console.log('Entra');
7      tablaProductos.querySelector('tbody').innerHTML = '';
8    }
9    function actualizarTabla() {
10     // Uso de los métodos de las instancias
11     fetch(apiUrl)
12     .then(response => {
13       if (!response.ok) {
14         throw new Error('Ocurrió un error al recuperar los datos.');
```

```
// Limpiar la tabla antes de agregar nuevos datos
tablaProductos.querySelector('tbody').innerHTML = '';
// Llenar la tabla con los datos obtenidos
data.forEach(producto => {
  const fila = document.createElement('tr');
  fila.innerHTML = `
    <td>${producto.id}</td>
    <td>${producto.title}</td>
    <td>${producto.price}</td>
    <td>${producto.category}</td>
    <td></td>
    <!-- Agrega más celdas si la API devuelve más datos -->
  `;
  tablaProductos.querySelector('tbody').appendChild(fila);
});
.catch(error => {
  console.error('Hubo un problema con la solicitud fetch:', error);
});
}
// Agregar un evento de clic al botón que llame a la función 'ejecutarScript'
btnActualizar.addEventListener('click', limpiarTabla());
actualizarTabla();
});
```

Pagina Web (FakeStoreAPI)

Tabla de Productos

<div>Actualizar</div>				
ID	Nombre	Precio	Categoria	Imagen
1	Fjallraven - Foldsack No. 1 Backpack, Fits 15 Laptops	109.95	men's clothing	
2	Mens Casual Premium Slim Fit T-Shirts	22.3	men's clothing	
3	Mens Cotton Jacket	55.99	men's clothing	
4	Mens Casual Slim Fit	15.99	men's clothing	
5	John Hardy Women's Legends Naga Gold & Silver Dragon Station Chain Bracelet	695	jewelery	

**Experiencia:**

Trabajar en conjunto con el desarrollo web, consumiendo una API real, utilizando el Modelo de Fetch de JavaScript es una experiencia que me ha enseñado mucho, he visto y comprendido cosas que antes no respecto a las APIs

Como enseñanza nos deja la manera práctica de utilizar Fetch, el método GET para poder consumir un recurso e implementarlo en nuestra Página Web.