

# Código de clase de DOM comentario

## Código de HTML

Leer y modificar elementos desde el DOM hasta el HTML y crear Elementos

```
<!-- /<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="style.css">
  <title>Leer y Modificar</title>
</head>
<body>
  <h1>Hola al DOM</h1>
  <p id="parrafo"> Texto estatico</p>
  <h3>Titulo uno h3 sin clases ni ids</h3>
  <h3 class="h3-danger">Titulo dos h3 con clase danger</h3>
  <h3 id="h3-secondary">Titulo tres h3 con clase secondary</h3>

  <h4 class="h3-danger">Titulo dos h4 con clase danger</h4>
</body> -->

<h1>Hola al DOM</h1>
  <ul id="lista">
    <!-- No tenemos elementos acá desde HTML -->
  </ul>

  <!-- es recomendable ponerlo siempre abajo el template de
  <template id="template-li">
    <li class="list">
      <b>Nombre : </b> <span class="text-danger">${item}</span>
```

```
</li>
</template>
```

## Código de JS

Leer y modificar elementos desde el DOM hasta el HTML y crear Elementos

```
document.querySelector('h1').textContent = "Texto desde JS";
console.log(document.querySelector('h1'));

console.log(document.querySelectorAll('#parrafo'));
console.log(document.getElementById('h3-secondary'));
//Modificar elementos

const parrafo = document.querySelector('#parrafo');

console.log(parrafo);

parrafo.textContent = 'texto desde JavaScript we';
parrafo.innerHTML = '<strong>text with inner</strong>';

parrafo.classList.add('h3-danger');

//Manipulacion de DOM desde JavaScript
Creacion de elemento li en document lo asignamos como hijo
const lista = document.getElementById('lista');
console.log(lista);

const li = document.createElement('li');//ya creamos la variable
li.textContent = 'primer elemento';

lista.appendChild(li);

//Creamos el array

//no es recomendable hacerlo
const li = document.createElement('li'); //Creamos un li
li.textContent = item; //Su contenido será item, porque tiene
```

```

lista.appendChild(li)

//innerHTML es para poder modificar o leer los caracteres
<div id="mensaje">Hola, este es un mensaje.</div>

const mensaje = document.getElementById('mensaje');
mensaje.innerHTML = '<p>Mensaje actualizado con <stro

//cómo lo podemos crear?
arrayElement.forEach(item => {
    lista.innerHTML += `<li>${ item}</li>`
});

Insert Before
//codigo Eliminado

const childNode = fragment.firstChild;

fragment.insertBefore(li, childNode);

const lista = document.getElementById('lista');

const arrayElement = ['primer elemento', 'segundo', 'tercero',

const fragment = document.createDocumentFragment();

arrayElement.forEach((item) => {

    const li= document.createElement("li");
    li.textContent = item;

    fragment.appendChild(li);

    const childNode = fragment.firstChild;

    fragment.insertBefore(li, childNode);
});

```

```

//console.log(fragment);
lista.appendChild(fragment);

//segunda ves que se elimina parte delm cod
const fragment = document.createDocumentFragment();

arrayElement.forEach((item) =>{
    const li = document.createElement("li")
    li.classList.add("list");

    const b = document.createElement("b");
    b.textContent = "Nombre : ";

    li.appendChild(b);

    const span = document.createElement("span");
    span.classList.add("text-danger");
    span.textContent = item;
    li.appendChild(span);

    fragment.appendChild(li);

});
//Empujar el fragment
lista.appendChild(fragment);

//Las comillas inversas indican que se puede agregar template
fragment += `
    <li class = "list">
        <b> Nombre : </b> <span class ="text-danger">{item}</span>
    </li>
`;

lista.innerHTML= fragment
let fragment = "";

```

```

//tercera ves que elimino codigo

const lista = document.querySelector('#lista');

const arrayElement = ["primer elemento", "segundo", "tercer e

const template = document.querySelector(`#template-li`).conte
const fragment = document.createDocumentFragment();

arrayElement.forEach((item)=> {
    //Query seleccionador de etiquetas donde agregues es donde
    template.querySelector('.list span').textContent = item;

    //pasamos el parametro de "true"
    const clone = template.cloneNode(true);
    fragment.appendChild(clone);
});

lista.appendChild(fragment);

```