

PRÁCTICA 3.01 DOM

Normas de entrega

- En cuanto al **código**:
 - en la **presentación interna**, importan los **comentarios**, la claridad del código, la significación de los nombres elegidos; todo esto debe permitir considerar al programa como **autodocumentado**. No será necesario explicar que es un **if** un **for** pero sí su funcionalidad. Hay que comentar las cosas destacadas y, sobre todo, las **funciones** y **clases** empleadas. La ausencia de comentarios será penalizada,
 - en la **presentación externa**, importan las leyendas aclaratorias, información en pantalla y avisos de ejecución que favorezcan el uso de la aplicación,
 - si no se especifica lo contrario, la información resultante del programa debe aparecer en la consola del navegador **console.log(información)**,
 - los ejercicios deben realizarse usando **JavaScript ES6** y usar el modo estricto (**use strict**) No se podrá utilizar *jQuery* ni cualquier otra biblioteca (si no se especifica lo contrario en el enunciado),
 - para el nombre de **variables**, **constantes** y **funciones** se utilizará *lowerCamelCase*,
- En cuanto a la **entrega** de los archivos que componen los ejercicios:
 - todos los ejercicios en **una carpeta** (creando las **subcarpetas** necesarias para documentación anexa como imágenes o estilos) cuyo nombre queda a discreción del discente,
 - el nombre de los ficheros necesarios para resolver el ejercicio será el número de ejercicio que contenga,
 - el código contendrá ejemplos de ejecución, si procede, y
 - la carpeta será comprimida en formato **ZIP** y será subida a **Aules** de forma puntual.

Ejercicio 1 - Contando elementos del DOM

Dispones de una página web en **Ejercicio01.html**. Añade el código necesario en un fichero **js** aparte para mostrar la siguiente información:

- el número de párrafos de la página,
- el texto del segundo párrafo,
- el número de enlaces de la página,
- la dirección del primer enlace y
- la dirección del penúltimo enlace.

El texto del resumen será añadido al **<div>** con **id=info** y debe estar debidamente formateado.

Ejercicio 2 - Números DOM

Crea una página web que tenga un listado de tipo **** con un **** de muestra y el texto "Soy una muestra fea" (o algo similar). Mediante **JS**, añade encima del componente anterior un botón **Nuevo número** que cuando se pulse añada un nuevo elemento **** con un número aleatorio entre 1 y 1 000.

Ejercicio 3 - Tabla dinámica DOM

A partir del fichero **Ejercicio03.html** genera el código necesario para que al pulsar sobre los botones realice las siguientes acciones:

- **Añadir fila**, añade el contenido del input con id texto de formulario como última fila de la tabla,
- **Caniar**, transforma el texto de la celda a su izquierda mediante la función **toCani**. Debe mostrarse en el mismo lugar que la frase original.

NOTA: para seleccionar el contenido de un **input** se hace utilizando su atributo value del siguiente modo:
`document.getElementById("texto").value;`

Ejemplo resultado:

```
<td id="fila1">Ejemplo de contenido</td>
<td><button onclick="toCani()">Caniar</button></td>
```

Ejercicio 4 - Creando funciones DOM

Aunque parezca mentira, no existe ningún método para añadir un elemento después de otro como sí lo hace **insertBefore**. Crea una función que reciba como parámetros dos elementos (el primero debe ser nuevo en el **DOM**) y coloque el primero después del segundo. La llamada se realizará de este modo:

```
function insertAfter(nuevoElemento, existenteElemento);
```