

## PRÁCTICA 3.02 DOM II Hell on Earth

### 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 - El censor DOM**

Crear una función que recorra el **DOM** desde la etiqueta `<body>` del fichero `Ejercicio01.html` y si encuentra la palabra "sexo" elimine el texto y la sustituya por "**Contenido Bloqueado**" poniendo el texto en rojo, negrita y cursiva (utiliza clases).

**Ejercicio 2 - Primos DOM**

Escribe un programa mediante JavaScript que cree dinámicamente (al cargar la página) una tabla de 10 por 10 celdas. Cada celda de la tabla tendrá un número único que empezará en uno y se irá incrementando en uno.

Esta página, además, tendrá un botón encima de la tabla con el texto `Calcular números primos`, que, al pulsarlo, hará que todas las celdas de la tabla que tengan números primos se pongan con un fondo rojo y el texto en negrita (utiliza clases).

**Ejercicio 3 - Ocultar DOM**

Crea una página HTML que tenga cinco elementos `<p>`. Además, dota de algún estilo **CSS** para cada párrafo de forma que los párrafos sean distinguibles (altura, longitud, color y un borde).

En combinación con JavaScript realiza un programa que permita al hacer **clik** sobre los párrafos que estos desaparezcan (se oculten) y al hacer **mouse out** que cambie el color del fondo del párrafo por uno color aleatorio. Utiliza para ello una clase específica en el CSS.

Añade también un botón denominado `Reaparecer` que permitirá al pulsar que aparezcan todos los elementos desaparecidos en su aspecto original.

**NOTA:** utiliza las propiedades **CSS** `visibility:hidden/visible`; para desaparecer.

**Ejercicio 4 - Componente Carrusel DOM**

A partir de un *array* de imágenes como este:

```
var imagenes = ["img/feo.jpg", "img/fea.jpg", "img/feos.jpg", "img/feas.jpg"];
```

Escribir un programa que muestre la primera imagen en con una etiqueta `<img>` dentro de un `<div>`. Tras dos segundos de espera, se muestra la siguiente imagen. Una vez mostrada la última imagen el carrusel volverá a comenzar por la primera.

Añade alguna transición entre las imágenes utilizando las propiedades **CSS**. Por ejemplo desplazando las imágenes fuera de la página con **right** y **left** o que aparezcan y desaparezcan poco a poco con **opacity**.

Busca cuatro imágenes que estimes oportunas (aptas para menores) y colócalas en la carpeta `img` de tu ejercicio o, mejor todavía, puedes utilizar direcciones web para evitar almacenar esas imágenes en el disco.

Puedes hacer uso de la función `setInterval` que vimos en la unidad anterior.

**NOTA:** para realizar las transiciones se debe modificar el código **CSS** a través de **JavaScript**. No se puede utilizar características propias de **CSS** como **transition**.