

Ejercicios Unidades 6 y 7 (1) – Modelo de Objetos del Documento (DOM). Introducción a los eventos

EJERCICIO: u7e01_analisisPagina

Completa la página web **analisisPagina.html**, para mostrar en el `div` con `id="info"`, la siguiente información de la web:

- El número de párrafos de la página.
- El texto del segundo párrafo.
- El número de enlaces de la página.
- La dirección del primer enlace.
- La dirección del penúltimo enlace.
- El número de enlaces que apuntan a `http://prueba`.
- El número de enlaces del primer párrafo.

EJERCICIO: u7e02_modTexto

Completa el documento **modTexto.html**, para que, al hacer clic sobre el botón **Cambiar**, se modifique el tamaño de la fuente y el color del texto de los párrafos del documento. Los valores permitidos de encuentran definidos en los desplegables.

EJERCICIO: u7e03_intercambioInformacion

Completa la página web **intercambioInformacion.html**, para que, al hacer clic sobre el botón **Añadir**, se añada la opción seleccionada en el `select` al `textarea`. Has de tener en cuenta que, cuando se almacene un elemento nuevo en el `textarea`, los que hemos seleccionado anteriormente, deben permanecer hasta que se recargue la página.

EJERCICIO: u7e04_selectAsociados

Completa la página web **selectAsociados.html**, para que, al seleccionar una Comunidad Autónoma en el primer `select`, se carguen las provincias que correspondan en el segundo `select`. Para resolverlo, deberás utilizar el evento `change`, que se lanza cada vez que cambiamos la opción seleccionada en un `select`.

EJERCICIO: u7e05_parrafo

Crea una página web que tenga un `textarea` y un botón **Crear párrafo**. Cuando el usuario haga clic sobre el botón se creará un nuevo párrafo que contendrá el texto del `textarea`. En caso de que el `textarea` esté vacío, mostrará un mensaje de error y no creará ningún párrafo.

EJERCICIO: u7e06_floristeria

Queremos crear una página en la que generar filas de una tabla con información de plantas de una floristería, haciendo uso del DOM del documento.

Para ello, deberás crear un archivo HTML y un script JavaScript que tenga tres botones con las siguientes funcionalidades:

- **Generar planta:** solicitará al usuario la información de la planta (nombre, ubicación, ejemplares y flor) con 4 `prompt` y una vez introducidos los datos se creará una nueva fila en la tabla.
- **Borrar fila:** borrará la última fila introducida.
- **Mostrar número de filas:** al hacer clic en este botón, se mostrará el número de filas de la tabla.

EJERCICIO: u7e07_animales

Queremos crear una página en la que colocar imágenes de animales haciendo uso del DOM del documento.

Para ello, deberás crear un archivo HTML y un script JavaScript que tenga los siguientes botones con las siguientes funcionalidades:

- **Generar personaje:**
 - Cuando hacemos clic en el botón deberá aparecer un prompt con el siguiente texto: "Elige animal: 1. Perro, 2. Gato, 3. Hurón."
 - Cuando el usuario elija un número, mediante un switch, mostrará la imagen del animal elegido (dispones de las 3 imágenes en la carpeta **imgAnimales**).
- **Borrar personaje:** este botón borrará la última imagen introducida.
- Botones **Nº perros**, **Nº gatos**, **Nº hurones**: al hacer clic en cada uno de ellos, mostrará el número de imágenes que hay de cada animal. No utilices un contador al generar las imágenes, cuenta el número de nodos de cada imagen que haya.

EJERCICIO: u7e08_berlin

Partiendo de la página web **berlin.html** y de las imágenes almacenadas en la carpeta **berlin**, crea un script JavaScript, para conseguir la siguiente funcionalidad en la página:

- Cuando el **ratón pase por encima** de cualquiera de las imágenes de las paradas del metro de Berlín, debes mostrar en el párrafo correspondiente, información de esa parada, por ejemplo: *Esta es la parada de Westhafen*.
- Cuando se haga clic en los botones **Línea roja**, **Línea verde**, **Línea azul**; las paradas por las que pasa cualquiera de esas líneas, se tienen que resaltar con el color correspondiente (fíjate en las clases CSS que ya tienes creadas y asociadas).
- Cuando se haga clic en el botón **Borrar selección**, las paradas se deben dejar de resaltar en los colores de cualquiera de las líneas de metro.

EJERCICIO: u7e09_colores

Simula, mediante JavaScript, el comportamiento de una paleta de colores. Para ello, definirás una tabla de forma dinámica con 20 colores. Utiliza un elemento `div` para que al hacer clic sobre un color de la tabla, el color de fondo de ese elemento `div`, se corresponde con el color elegido. Además, mostrarás el valor del color seleccionado en otro elemento HTML de tu elección.

EJERCICIO: u7e10_listaTareas

Crea una pequeña aplicación web para crear y mantener una lista de las tareas que tengamos pendientes.

La página constará de un formulario con una caja de texto para indicar el nombre de la tarea, un botón **Añadir** para añadir la tarea y otro **Eliminar todas** para eliminar todas las tareas existentes.

Por otro lado, las tareas se mostrarán dentro de una lista `` e irán junto a un **icono** que, al hacer clic sobre él, nos permitirá eliminar solamente la tarea correspondiente.

Una vez que compruebes que todo funciona bien, modifica el programa para que no se pierdan las tareas cada vez que cerramos el navegador. Para ello, podemos utilizar `localStorage`.

*** Para almacenar las tareas, se puede crear un array de strings. Y se puede utilizar `JSON.stringify(miArray)` que devuelve una cadena que se puede almacenar con `localStorage`. Para obtener el array de nuevo desde la cadena almacenada en `localStorage`, puedes utilizar `JSON.parse(varLocalStorage)`.