

# Árbol DOM

## Atributos de HTML

- 1.- Crea un enlace que vaya a google. Crea dos botones junto al enlace:
  - Al pulsar el primero, muestra la URL a dónde va el enlace y su texto.
  - Al pulsar el segundo, cambia la URL a otra diferente y su texto por “he cambiado”
  
- 2.- Crea una tabla en de 4 filas y 4 columnas. Usando atributos HTML, coloca un borde, unas medidas de 300x300px y una alineación al centro. A continuación crea tres botones:
  - El primero muestra los valores de anchura y altura de la tabla, pide unos valores nuevos para cada propiedad y los aplica a la tabla. ¿se aplican correctamente los valores? ¿Falla alguno?
  - El segundo muestra el valor actual del borde de la tabla y solicita un valor nuevo para aplicarlo. ¿Funciona correctamente el nuevo valor?
  - El tercer botón va cambiar la alineación de la tabla cada vez que se pulse. Es decir, si la tabla está alineada al centro, al pulsar el botón, la tabla quedará alineada a la derecha. Si se pulsa el botón otra vez, se alineará a la izquierda y si vuelvo a pulsarlo, la tabla se pondrá en el centro.
  
- 3.- Crea un DIV con el número 0 en su interior. Bajo el DIV, coloca dos botones: uno para sumar y otro para restar.
  - Si se pulsa el botón para sumar, se incrementa en uno el número que hay en el DIV en ese momento.
  - Si se pulsa el botón restar, se decrementa en uno el número que hay en el DIV en ese momento.
  
- 4.- Crea una galería de fotos (al menos 5 fotos) que posea un botón para avanzar y otro para retroceder.
  - El botón para avanzar hará que se muestre la siguiente foto cada vez que se pulse. Si se llega al final de la lista de imágenes, el botón deja de funcionar.
  - El botón para retroceder hará que se muestre la foto anterior cada vez que se pulse. Si se llega al principio de la lista de imágenes, el botón deja de funcionar.

b) Mejorar el ejercicio haciendo que el botón correspondiente se deshabilite cuando se alcanza el final/principio de la lista de fotos.
  
- 5.- Crea un párrafo en HTML y ponle la clase “uno”. Usando CSS, coloca un fondo, uno borde, una anchura y un color/tam de letra para esa clase. Crea la clase “dos” similar a la clase “uno” pero con valores diferentes.
  - a) Cuando se haga click sobre el párrafo, cambia de la clase “uno” a la clase “dos”.
  - b) Cada vez que haga click en el párrafo, cambia de clase alternativamente.

6.- Crea cinco botones y coloca como `value` un color diferente a cada uno. Cada vez que se pulse uno de los botones, el fondo de la página debe cambiar al color indicado por ese botón.

7.- Crea una tabla con 4 filas y 4 celdas por fila. Usando atributos de HTML, coloca un borde y un tamaño de 400x400px y deja el fondo en blanco.

- a) Crea el código Javascript necesario para que cada vez que se pulse sobre una celda, esa celda se vuelva de color negro. *Nota: hay que usar funciones anónimas y `this`.*
- b) Mejora el ejercicio anterior para que cada vez que se pulse sobre una celda, esta cambie a negro o a blanco alternativamente según corresponda.
- c) Super mejora: define una lista de 4 colores y ahora cada vez que se pulsa sobre una celda, esta va cambiando de color siguiendo la lista de colores definida (si llega al final de la lista, vuelve al principio).

8.- Implementa el efecto ROLLOVER sobre una imagen: al pasar el ratón por encima, cambia la imagen. Al sacar el cursor fuera, la imagen vuelve a su estado original.

## Propiedades CSS

9- Repite el ejercicio 5 de esta relación pero esta vez no uses clases (toca directamente todas las propiedades CSS que necesites).

10.- Crea tres DIV de tamaño 100x100px y con un color de fondo diferente cada uno. Al pasar por encima de alguno de esos DIVs, el color de fondo de la página debe cambiar al color que tenga el DIV sobre el que está el cursor del ratón.

b) Añade el código necesario para que el color de la página vuelva a su color original si el cursor del ratón no está encima de ninguno de los DIVs del ejercicio.

11.- Crea tres DIV de tamaño 300x100px con un color de fondo distinto cada uno y con un texto en su interior que indique “Soy el DIV X” (siendo X un número del uno al tres).

- a) Oculta el segundo DIV usando la propiedad `visibility` de CSS. Consigue que ese DIV “vuelva a aparecer” sí y solo sí, el cursor del ratón se coloca en el primer DIV.
- b) Oculta el segundo DIV usando la propiedad `display` de CSS. Consigue que ese DIV “vuelva a aparecer” sí y solo sí, el cursor del ratón se coloca en el primer DIV.

- 12.- Coloca una imagen dentro de un DIV con position: absolute. Crea dos botones:
- Al pulsar uno de ellos la imagen debe moverse 5px hacia la derecha de la pantalla.
  - Al pulsar el otro, la imagen debe moverse 5px hacia la izquierda de la pantalla.

*Nota: Recuerda la posición (0,0) de la página web al usar top/left/right/bottom*

b) Mejora el ejercicio haciendo que la imagen no pueda salirse de los límites de la pantalla. *Nota: realiza el cálculo a ojo, ya aprenderemos a obtener las medidas de la página correctamente.*

c) Realizar el mismo ejercicio pero esta vez la imagen se moverá hacia la derecha mientras el ratón esté encima de ella. Coloca un botón para resetear los valores de posición de la imagen.

13.- Crea el efecto SPOILER que se puede ver en foros y otras páginas.

Se oculta el contenido de un elemento (pej, un párrafo) y sólo se muestra el mensaje “Mostrar”. Al hacer click sobre dicho mensaje, se muestra el contenido del elemento y el mensaje cambia a “Ocultar”. Si se pulsa sobre ese mensaje, el elemento vuelve a su estado original (contenido oculto y mensaje “Mostrar”).

14.- Crear un menú vertical con efecto acordeón usando los siguientes apartados:

- Inicio
  - Portada.
  - Acerca de mi.
  - Mis fotos.
- Proyectos
  - Web Apps.
  - Mobile Apps.
  - Juegos.
- Portfolio
  - Páginas webs.
  - Software.
- Contacto

A tener en cuenta:

- El menú debe tener una maquetación en CSS.
- Los subapartados permanecen ocultos hasta que se pulsa sobre el apartado correspondiente.
- Al abrir un bloque de subapartados, se cierran los demás.

15.- Crear el mismo menú que el ejercicio anterior pero en horizontal. Ahora, al pasar el ratón por encima de cada apartado, aparece el bloque de subapartados correspondiente.

A tener en cuenta:

- El menú debe tener una maquetación en CSS.
- El bloque de subapartados quedará por encima del resto de elementos que tenga debajo (pej, sections, articles...), no desplazará al resto de elementos.
- Un bloque de subapartados desaparecerá cuando se pase por encima de otro apartado.

16.- Usando las funciones `setInterval` y `clearInterval` de Javascript. Coloca una etiqueta `meter`, una etiqueta `progress` y un botón al lado de cada una. Al pulsar el botón se debe simular la “carga” de la barra. Muestra un mensaje cuando se alcance el máximo.



b) Mejora el ejercicio haciendo que cuando pulsemos uno de los botones, este quede bloqueado hasta que la barra del elemento `progress/meter` se complete del todo.

#### 17.- Carrera de DIVS

Coloca 4 divs de 100x100px y color de fondo distinto en la parte izquierda de la pantalla (uno debajo de otro). Crea un botón que se llame CORRER y al pulsarlo, los divs se moverán hacia la derecha de la pantalla de manera automática. Cada DIV debe moverse a una velocidad aleatoria que cambiará en cada carrera. El primer DIV que llegue a la derecha de la pantalla, gana. *Nota: realiza el cálculo a ojo, ya aprenderemos a obtener las medidas de la página correctamente.*

Coloca otro botón para resetear todo y volver a comenzar la carrera.

#### 18.- Modifica el ejercicio 16 de esta relación de la siguiente forma:

- Usa sólo el elemento `progress` y el botón que se encarga de realizar el funcionamiento explicado en el ejercicio 16. Cambia el nombre del botón por “Generar moneda”
- Crea usando HTML un div en la parte superior izquierda llamado MONEDAS, cuyo contenido inicial será el número 0.
- Cada vez que se pulse sobre el botón ‘Generar moneda’, se bloqueará dicho botón y se cargará la barra de `progress` como se vio en el ejercicio 17.
- Al terminar la carga, se sumará una moneda al contenido del div MONEDAS y se desbloqueará el botón ‘Generar Moneda’.