

DWEC – Ejercicios propuestos UD5 y UD6

Contenido

| | |
|---------------------|---|
| Instrucciones | 1 |
| Ejercicios | 1 |
| Ejercicio 1 | 1 |
| Ejercicio 2 | 2 |
| Ejercicio 3 | 2 |
| Ejercicio 4 | 2 |

Instrucciones

Lee atentamente las siguientes indicaciones y pregunta cualquier duda que tengas.

Te **recomiendo encarecidamente** que leas el ejercicio completo antes de hacer los distintos apartados indicados, ya que te dará ideas de cómo plantear la solución.

Se te ha facilitado una estructura de carpetas con todos los documentos.

Los documentos **HTML** solo podrás modificarlos para para añadir las referencias a los archivos JS que corresponda, cualquier otra modificación que se pida será a través de JavaScript. **Todos los HTML han de tener su referencia al JS sino se considerará que el ejercicio no está hecho**

Ejercicios

Ejercicio 1

Se te facilita un documento HTML con el body completamente vacío en el que no podrás escribir nada, solo enlazar el archivo JS correspondiente.

En este ejercicio tendrás que crear una serie de elementos mediante DOM, no HTML, directamente en la carga de la página.

1. Crea un título h2 con el texto “Ejercicio 1” y otro h1 con tu nombre y apellidos en el interior.
2. Crea un input de tipo texto acompañado del texto “Nombre”
3. Crea un input de tipo number acompañado del texto “Cantidad”
4. Crea un select con las siguientes opciones: Patatas, Tomates, Manzanas
5. Crea un botón “Crear” que cuando se haga click en él cree una división nueva con fondo rojo y el texto “*valorNombre* come *valorCantidad* seleccion*Comida*”, siendo el *valorNombre* lo escrito en primer input, *valorCantidad* lo escrito en el segundo input y *seleccionComida* lo seleccionado en el select.

Cada vez que se pulse el botón “crear” se deberá eliminar cualquier división previa que existiese.

Ejercicio 2

Se te facilita un documento HTML con la estructura ya montada.

Lo primero modifica a través de JS la etiqueta h1 identificada con “cambiaNombre” para poner tu nombre y apellidos.

En esta estructura encontrarás una serie de divisiones de colores, todas perfectamente identificadas tal que su id es “div_” seguido de su color en inglés.

Además, encontrar un selector de colores perfectamente definido e identificado con “inColor”, un botón Resetear identificado con “btnResetear” y una división de salida “div_salida”.

Debes de implementar el código JS de tal manera que:

1. Al pasar por encima de una división de colores dentro del legend “Pulsadores” la división se salida tomará como color de fondo el mismo que el de la división sobre la que se esté. Ten cuidado, dado que la división “container” y la propia división de salida no han de tenerse en cuenta para esto. Crea un único manejador para este evento.
 2. Cuando se haga click en cualquiera de las divisiones de colores dentro del legend “Pulsadores”, ésta tomará de color de fondo el escogido en el desplegable de colores “inColor”. Al igual que antes, las divisiones container y de salida no se verán afectadas por el click. Crea un único manejador para este evento.
 3. Cuando se pulse el botón “btnResetear” todas las divisiones de colores volverán a tener su color original. Container y div_salida no se ven afectadas por esta acción. Aprovecha el id de las divisiones...
-

Ejercicio 3

Se te facilita un documento HTML con la estructura ya montada.

Lo primero modifica a través de JS la etiqueta h1 identificada con “cambiaNombre” para poner tu nombre y apellidos. Además, tienes asociado un estilo mediante estilos.css.

Desarrolla un manejador de tal forma que cada vez que se teclee algo con el teclado físico se cree un elemento de tipo div que contenga la letra o tecla pulsada y que tenga como clase “cuadrado-cuqui”, del css.

Ejercicio 4

Se te facilitan una serie de documentos HTML y JS completamente limpios.

Desde Javascript, crea un pequeño formulario con dos inputs de tipo texto, acompañados de las etiquetas *ID* y *Nombre*, un botón *Guardar*, otro botón *Cargar* y una división de salida donde mostrar resultados.

Añade a los inputs manejadores del evento *change*, de tal manera que cuando se modifique alguno de los esto se compruebe automáticamente, mediante las funciones que se indican en la tabla siguiente, si el dato introducido es correcto, y que de no ser así ponga el fondo del input en un tono rojo para que el usuario se dé cuenta.

Cuando se pulse el botón *Guardar* se comprobará de nuevo si los dos inputs tiene un valor correcto y se guardará en sessionStorage la pareja de valores ID y nombre como un objeto literal.

El botón Cargar leerá del sessionStorage todos los elementos guardados y los mostrará por la división de salida.

Las funciones verificadoras de los inputs serán las siguientes y ambas han de ser realizadas mediante expresiones regulares.

| Función | Recibe | Devuelve | Descripción |
|--------------------|----------------------|-------------|---|
| checkID | id:string | bool | <p>Mediante una expresión regular comprobará si el <i>id</i> recibido empieza por “BCH”, seguido de un '-' opcional y continuado de entre 2 y 6 dígitos.</p> <p>Ejemplos de id válidos:</p> <ul style="list-style-type: none"> - BCH-12 - BCH12 - BCH-123456 |
| checkNombre | nombre:string | bool | <p>Mediante una expresión regular comprobará si el <i>nombre</i> tiene un mínimo de 3 letras y un máximo de 30 y contiene al menos dos palabras. No son admitidos ni caracteres especiales, ni dígitos ni ningún otro carácter que no sea una letra.</p> |