



En la carpeta compartida de ejemplos de esta unidad tenéis disponible los documentos HTML de todos los ejercicios del boletín.

1. Dado el formulario con los datos de tres actores seleccionables mediante un radiobutton, añadir un manejador de eventos al botón para que al pulsar sobre él muestre por consola los datos del actor seleccionado.
2. Dado el formulario con los datos de cuatro provincias gallegas seleccionables mediante un select, añadir un manejador de eventos al botón para que al pulsar sobre él muestre por consola los datos de la provincia seleccionada.
3. Igual que el ejercicio anterior pero esta vez con un select múltiple.
4. Dado el formulario que permite la entrada de una provincia con su código correspondiente, y dos listas múltiples que inicialmente estarán vacías. Una vez pulsado el botón “agregar provincia” deberá introducirse la provincia en la lista múltiple de la izquierda. Los botones con las flechas deberán pasar de izquierda a derecha o viceversa las provincias seleccionadas de una de las listas múltiple de origen a la de destino. Habrá que controlar que si la provincia ya existe en alguna de las dos listas múltiples, no se permitirá que se agregue de nuevo.
5. Dado el formulario con un checkbox que puede ser marcado o desmarcado pulsando el botón correspondiente. Inicialmente dicho botón no hace nada, hay que añadirle el manejador de eventos correspondiente usando uno de los dos botones que están debajo. El otro es para eliminar dicho manejador de eventos. Si el botón tiene el manejador de eventos permitirá marcar o desmarcar el checkbox, en caso contrario no hará nada.
6. Dado el documento HTML facilitado con este ejercicio, añadir los manejadores de eventos necesarios para:
 - El cuadrado se ponga de color amarillo cuando el cursor del ratón se coloque encima del mismo, para ello usa el atributo classList añadiendo la clase amarillo preparada de antemano. Además deberá informar en la capa de salida de texto del tipo de evento, el objeto en el que se produce, así como las coordenadas del cursor en el momento que se desencadena. Cuando el cursor salga del cuadrado deberá volver todo a la situación original, cuadrado blanco y salida de texto vacía.
 - Cuando el cursor se coloque en el input de texto, cada vez que se pulse una tecla se informará en la capa de salida de texto de la tecla pulsada.
7. Dado el documento HTML facilitado con este ejercicio, añadir un único manejador de eventos que gestione la pulsación de las teclas de los diferentes dígitos mostrándolos en el input de solo lectura.
8. Dado el documento HTML facilitado con este ejercicio, añadir un manejador de eventos que impida que se introduzcan dígitos en el input.
9. Dado el documento HTML facilitado con este ejercicio, añadir un manejador de eventos que impida copiar el texto que haya en el input.



10. Dado el documento HTML facilitado con este ejercicio, añadir los manejadores de eventos necesarios para mostrar un mensaje informativo en la capa de salida acerca de cuál de los dos botones del ratón se ha pulsado.
11. Dado el documento HTML facilitado con este ejercicio, añadir un manejador de eventos al formulario que muestre por consola el atributo tagName de las propiedades target y currentTarget del objeto evento, además incluye también la referencia al objeto this, para que quede claro a qué objeto se referencia al hacer click sobre el documento. Hacer pruebas haciendo click en el párrafo, la capa o el formulario.
12. Dado el documento HTML facilitado con este ejercicio, añadir un manejador de eventos que permita validar si el input del formulario está vacío. En ese caso no se procesa el formulario y se mostrará un mensaje de alerta informando de tal circunstancia.
13. Dado el documento HTML facilitado con este ejercicio, añadir los atributos y expresiones regulares necesarias para que se valide el formulario con las siguientes especificaciones:
 - Todos los inputs deben ser de tipo texto y ninguno de ellos podrá quedarse en blanco.
 - Tanto el nombre como los apellidos deben tener una o más palabras que comienzan por letra mayúscula y siguen con minúsculas. Habría que tener en cuenta la Ñ y los caracteres tildados.
 - La fecha de nacimiento tendrá el formato DD/MM/AAAA, por ejemplo: 17/01/2024.
 - El DNI tendrá 7 u 8 dígitos y una letra mayúscula.
 - El correo electrónico tendrá solo letras minúsculas, en la parte del usuario (previo a la arroba) podrá tener puntos, guiones y guiones bajos. Entre la arroba y el punto solo podrá tener letras, guiones y guiones bajos. Finalmente tras el punto podrá tener de dos a cuatro caracteres. Ejemplo: correo.prueba@dominio.info
 - El usuario IDEA se compone de la inicial del nombre, las tres primeras letras del primer apellido, las tres primeras del segundo y las tres últimas cifras del DNI. Por ello, deberá comenzar con siete caracteres en minúscula, evitando Ñ y tildados, y finalizar con tres dígitos.
 - El teléfono deberá tener 9 dígitos, pero solo serán válidos los números que comiencen por 6, 7, 8 o 9.
 - El usuario de Twitter empezará por una arroba y le seguirán de 4 a 15 caracteres.
14. Dado el documento HTML facilitado con este ejercicio, desarrolla el código necesario para validar el formulario siguiendo las especificaciones del ejercicio anterior. En la capa de salida se mostrará una lista con los campos vacíos y otra con los campos con errores. En el caso de que todos los campos sean correctos se procesará el formulario.



15. Dado el documento HTML facilitado con este ejercicio, añade manejadores de eventos a los botones para que tengan las siguientes funcionalidades:

- Guardar: leerá lo que haya en los campos del formulario y guardará una cookie con el par clave=valor. Tras esto se reseteará el formulario.
- Recuperar: leerá el nombre de la cookie introducido como clave en el formulario y levantará un alert con el valor guardado en dicha cookie. Tras esto se reseteará el formulario.
- Eliminar: leerá el nombre de la cookie introducido como clave en el formulario y la borrará. Tras esto se reseteará el formulario.