

Ejercicios Unidad 7 – Tecnologías actuales: jQuery.

EJERCICIO: u11e01_selectores

Modifica la página que hay en la carpeta **ej1**, para probar los selectores de jQuery. No se modificará el HTML de la página, ya que se trata de **utilizar diferentes selectores**.

EJERCICIO: u11e02_eventos

Modifica la página que está en la carpeta **ej2**.

En la página hay tres zonas: izquierda centro y derecha. En las dos primeras hay una imagen y un párrafo. En la de la derecha, un input y un párrafo.

Cuando se produzca un evento sobre las imágenes o sobre el input, en el párrafo se mostrará un mensaje que indique qué evento se ha producido sobre la imagen/input.

Los eventos que se probarán en cada parte de la página son:

- En la parte izquierda: `click()`, `dblclick()`, `mouseenter()`, `mouseleave()`.
- En la parte centro: `mousedown()`, `mouseup()` y `hover()`.
- En la parte derecha: `focus()` y `blur()`.

EJERCICIO: u11e03_efectos

En la página de la carpeta **ej3**, crea el código necesario para que, al hacer clic sobre los elementos del menú de la izquierda, se apliquen los efectos que se piden sobre todas las imágenes.

Los ejercicios **u11e04 - u11e11**, se realizan sobre el HTML de la carpeta
/ejercicios4a11-16-17/index.html.

EJERCICIO: u11e04_selecciones

1. Seleccionar todos los elementos `div` que poseen la clase `module`.
2. Seleccionar el tercer ítem de la lista desordenada `#myList`, de tres modos distintos.
3. Seleccionar el elemento `label` del elemento `input` utilizando un selector de atributo.
4. Averiguar cuantos elementos en la página están ocultos.
5. Averiguar cuantas imágenes en la página poseen el atributo `alt`.

EJERCICIO: u11e05_recorrerDOM

1. Seleccionar todas las imágenes en la página y mostrar por consola el atributo `alt` de cada imagen.
2. Seleccionar el elemento `input`, después moverse hacia el formulario y añadirle una clase al mismo.
3. Seleccionar el ítem que posee la clase `current` dentro de la lista `#myList` y eliminar dicha clase en el elemento, después añadir la clase `current` al siguiente ítem de la lista.
4. Seleccionar el elemento `select` dentro de `#specials`, después moverse hacia el botón `submit`.
5. Seleccionar el primer ítem de la lista en el elemento `#slideshow`, añadirle la clase `current` al mismo y luego añadir la clase `disabled` a los elementos hermanos.

EJERCICIO: u11e06_manipularDOM

1. Añadir 5 nuevos ítems al final de la lista desordenada `#myList`.

2. Eliminar los ítems impares de la lista.
3. Añadir otro elemento **h2** y otro párrafo al último **div.module**.
4. Añadir otra opción al elemento **select**; darle a la opción añadida el valor **Wednesday**.
5. Añadir un nuevo **div.module** a la página después del último; luego añadir una copia de una de las imágenes existentes dentro del nuevo **div**.

EJERCICIO: u11e07_ayudaInput

Este ejercicio lo realizarás en el archivo **/ejercicios4a11-16-17/js/inputHint.js**. Lo que se pide es utilizar el texto del **label** y para mostrar una ayuda en la caja de texto. Los pasos para realizar el ejercicio son los siguientes:

1. Establecer el valor del elemento input igual al valor del elemento **label**.
2. Añadir la clase **hint** al elemento **input**.
3. Eliminar el elemento **label**.
4. Vincular un evento **focus** en el input para eliminar el texto de sugerencia y la clase **hint**.
5. Vincular un evento **blur** en el input para restaurar el texto de sugerencia y la clase **hint** en caso de que no se haya introducido texto.

EJERCICIO: u11e08_mostrarOcultarDiv

Este ejercicio lo realizarás en el archivo **/ejercicios4a11-16-17/js/tabs.js**. Lo que se pide es crear un pequeño menú para mostrar u ocultar cada **div.module**. Los pasos para realizar el ejercicio son los siguientes:

1. Ocultar todos los elementos **div.module**.
2. Crear una lista desordenada antes del primer **div.module** para utilizar como pestañas.
3. Acceder a cada **div** utilizando **\$fn.each** para utilizar el texto del elemento **h2** como el texto para el ítem de la lista desordenada.
4. Vincular un evento **click** a cada ítem de la lista de forma que:
 - Muestre el **div** correspondiente y oculte el otro
 - Añada la clase **current** al ítem seleccionado
 - Elimine la clase **current** del otro ítem de la lista
5. Finalmente, mostrar la primera pestaña.

EJERCICIO: u11e09_mostrarOcultarTexto

Este ejercicio lo realizarás en el archivo **/ejercicios4a11-16-17/js/blog.js**. Lo que se pide es añadir algún efecto para mostrar u ocultar párrafos. Los pasos para realizar el ejercicio son los siguientes:

1. Al hacer clic en alguno de los titulares del **div #blog**, se debe mostrar el párrafo correspondiente con un efecto de deslizamiento.
2. Al hacer clic en otro titular, se debe ocultar el párrafo mostrado con un efecto de deslizamiento y mostrar nuevamente el párrafo correspondiente también con un efecto de deslizamiento. Ayuda: utilizar el selector **:visible**.

EJERCICIO: u11e10_menúDesplegable

Este ejercicio lo realizarás en el archivo **/ejercicios4a11-16-17/js/navigation.js**. Lo que se pide es poder desplegar los ítems del menú superior de la página. Los pasos para realizar el ejercicio son los siguientes:

1. Al pasar el puntero del ratón por encima de un ítem del menú, se debe mostrar su submenú en caso de que exista.
2. Al abandonar ítem, el submenú se debe ocultar.
3. Para poder realizarlo, utiliza el método `$.fn.hover` para añadir o eliminar una clase en el submenú para poder controlar si debe estar oculto o visible (el archivo `/ejercicios4a14/css/styles.css` incluye una clase `hover` para este propósito).

EJERCICIO: u11e11_slideShow

Este ejercicio lo realizarás en el archivo `/ejercicios4a11-16-17/js/slideshow.js`. Lo que se pide es añadir un *slideshow* a la página con JavaScript. Los pasos para realizar el ejercicio son los siguientes:

1. Mostrar los ítems del `ul#slideshow` de forma cíclica, mostrando un ítem por unos segundos, luego ocultándolo con un efecto `fadeOut` y mostrando el siguiente con un efecto `fadeIn`.
2. Una vez llegado al último ítem de la lista, comenzar de nuevo con el primero.
3. Añadir unos botones de navegación por debajo del *slideshow* que muestre cuantas imágenes existen y en cual se encuentra en cada momento (`$.fn.prevAll` puede resultar útil).

EJERCICIO: u11e12_arrays

Crea una página con 4 párrafos, cada uno con un texto diferente. En la página, habrá un script en jQuery que implementará las siguientes operaciones:

1. Crea un array con los números del 0 al 6. Muéstralo por consola.
2. Crea un array con los números del array anterior, multiplicados por 3. Para hacerlo, utiliza el método `$.each`. Muéstralo por consola.
3. Crea otro array, con los números del primer array, multiplicados por 4. Para hacerlo, utiliza el método `$.map`. Muéstralo por consola.
4. Crea un array formado por los elementos tres arrays anteriores, pero sin modificar el contenido de esos arrays. Muéstralo por consola.
5. Muestra por consola el número de párrafos que hay en el documento.
6. Extrae el tercer párrafo, utilizando el método `get()`, y muestra su texto en consola.
7. Crea un array cuyos elementos sean el texto de los párrafos (utiliza el método `$.makeArray()`). Muéstralo por consola.
8. Crea un objeto jQuery con los párrafos cuyo texto tiene una longitud mayor de 10 caracteres. Muestra por consola el texto de los párrafos seleccionados. Utiliza el método `$.grep()`.

EJERCICIO: u11e13_tabla

Crea un archivo HTML que tenga una tabla con 10 filas y 2 columnas. La primera y la última fila serán filas con celdas de tipo encabezado (`<th>`).

En la primera columna de la tabla, estará el nombre de un color CSS (ej. yellow, red, ...). En la segunda columna, el texto que quieras.

Crea un CSS con las siguientes reglas:

- `.rojo`: permite modificar el fondo del elemento a color `#F5A9BC`.
- `.azul`: permite modificar el fondo del elemento a color `#A9BCF5`.
- `.negrita`: permite modificar el tipo de letra a negrita (`bold`).
- asignar el estilo `border-collapse: collapse` a la tabla

Crea unos botones en la parte inferior del documento HTML que permitan realizar las siguientes operaciones con jQuery:

1. Asignar a todas las filas de la tabla un borde sencillo.
2. Asignar a las filas de encabezado de la parte superior un borde doble.
3. Asignar a las filas de encabezado de la parte inferior solo borde superior. El resto de los bordes no se verán.
4. Asignar a las celdas pares la clase `.rojo`.
5. Asignar a las celdas impares la clase `.azul`.
6. Asignar a las celdas de encabezado la clase `.negrita`.
7. Aumentar el alto de las celdas de encabezado para que sea diferente del predeterminado.
8. Ocultar la fila 5.
9. Asignar a cada fila el color indicado en el texto de la primera columna de dicha fila.

EJERCICIO: u11e14_imágenes

Crea un archivo HTML que tenga tres imágenes. Añade cuatro botones: **Restaurar**, **Rotar**, **Ocultar** y **Mostrar**.

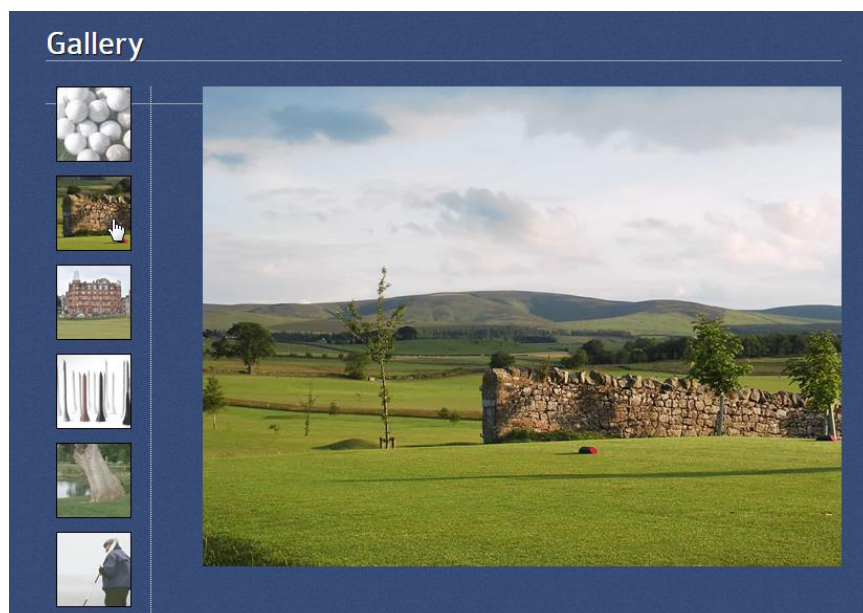
Necesitaremos además otras tres imágenes para sustituir a las originales en algunos de los efectos.

Realiza con jQuery las siguientes operaciones:

- Al colocarnos con el ratón sobre cualquiera de las imágenes, se mostrará otra imagen en su lugar. También mostrará un mensaje con la ruta `href` correspondiente a la imagen que se está mostrando.
- Al hacer clic sobre cualquiera de las imágenes, su tamaño se aumentará y mostrará un mensaje con el tamaño actual. Si volvemos a hacer clic sobre ella, volverá al tamaño original.
- Si hacemos clic sobre el botón **Restaurar**, las imágenes volverán al tamaño original.
- Si hacemos clic sobre el botón **Rotar**, las imágenes cambiarán de la primera a la segunda, de la segunda a la tercera, y de la tercera a la primera.
- Si hacemos clic sobre el botón **Ocultar**, las imágenes desaparecerán.

EJERCICIO: u11e15_galeríalmágenesEfectos

Crear una galería de imágenes utilizando los archivos de la carpeta **ej15-galeria**. Su funcionamiento será el siguiente: al hacer clic sobre una imagen de la izquierda, el usuario verá la imagen en grande en la parte derecha de la ventana:



EJERCICIO: u11e16_ajax

Este ejercicio lo realizarás en el archivo `/ejercicios4a11-16-17/js/load.js`. Trabajaremos en el `div#blog`, que contiene una lista.

En cada elemento de la lista hay un `<h3>`, con un `<a>`. Y después un `<p>` que está oculto.

El objetivo es cargar el contenido de un artículo de blog cuando el usuario haga clic en el título del ítem (en el `<h3>`). Los pasos para resolver el ejercicio son los siguientes:

1. Crear un elemento `div` después de cada título `h3`, y guardar una referencia hacia el `div` en el elemento del título utilizando `$.fn.data`.
2. Vincular un evento `click` a cada título, el cual utilizará el método `$.fn.load` para cargar en cada `div` creado el contenido apropiado desde el archivo `/data/blog.html`. No hay que olvidar deshabilitar el comportamiento predeterminado del evento `click`.

Cada título de artículo de blog en `index.html` incluye un enlace hacia el artículo. Necesitaremos aprovechar el atributo `href` de cada enlace para obtener el contenido propio de `blog.html`. Una vez obtenido el valor del atributo, se puede utilizar la siguiente forma para procesar la información y convertirla en un selector para utilizar en conjunto con `$.fn.load`:

```
var href = 'blog.html#post1';
var tempArray = href.split('#');
var id = '#' + tempArray[1];
```

EJERCICIO: u11e17_ajaxJson

Este ejercicio lo realizarás en el archivo `/ejercicios4a11-16-17/js/specials.js`. Trabajaremos en el `div#specials`, que contiene una lista.

El objetivo es mostrar los detalles del usuario para un día determinado cuando se selecciona desde la lista desplegable. Los pasos para resolver el ejercicio son los siguientes:

1. Añadir un elemento `div` después del formulario que se encuentra dentro del elemento `#specials`, para colocar la información a obtener.
2. Vincular el evento `change` en el elemento `select`; cuando se realiza un cambio en la selección, enviar una petición AJAX a `/data/specials.json`.
3. Cuando la petición devuelve una respuesta, utilizar el valor seleccionado en el `select` (ayuda: `$.fn.val`) para buscar la información correspondiente en la respuesta JSON.
4. Añadir algún HTML con la información obtenida en el `div` creado anteriormente.
5. Finalmente eliminar el botón `submit` del formulario.