

Ejercicio 01 (html)

Utilizando el fichero *exercise-01.html* utiliza jquery para que al cargarse la página todos los textos cambien a inglés.

Utiliza estos ejemplos como referencia:

- <https://codepen.io/agomcue486/pen/BvgWwP>
- <https://codepen.io/agomcue486/pen/pqXPKR>
- <https://codepen.io/agomcue486/pen/dwBWrr>

Ejercicio 02 (alert)

Utilizando el código que tenemos en *exercise-02.html* completa el código para que al hacer clic en el botón aparezca una alerta diciendo “Has pulsado en el botón”.

Utiliza como referencia estos ejemplos:

- <https://codepen.io/agomcue486/pen/omPvdv>

Ejercicio 03 (+)

Utilizando el código que tenemos en *exercise-03.html* completa el código para que al hacer clic en el botón aparezca una alerta que contenga un mensaje como el siguiente:

El alto del elemento h1 es ____px y el ancho es ____px.

Utiliza como referencia estos ejemplos:

- <https://codepen.io/agomcue486/pen/JxajEG>

Ejemplo 04

Utilizando el código que tenemos en *exercise-04.html* crea un fichero *exercise-04.js* que contenga el código jquery necesario para que:

- dentro del elemento `` se coloque el tamaño del elemento h1.
- dentro del elemento `` se coloque el tamaño del elemento section (clase div.section)

Ejemplo 05 (this)

Utiliza el código que tenemos en *exercise-05.html*. Crea un código jquery que haga que al hacer clic en cualquier elemento de la rejilla, éste cambie su color de fondo, permaneciendo este color hasta que se recargue la página.

Es decir, partimos de:

Y después de unos clics tenemos:

Toma como referencia estos ejemplos:

- <https://codepen.io/agomcuez486/pen/dwBWrr>
- https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_click
- https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_dblick

Ejercicio 06 (toggle)

Utiliza el fichero *exercise-06.html*. Crea el código necesario para que los artículos se muestren y desaparezcan haciendo varios clics sobre el botón que los acompaña.

Para hacer este ejercicio necesitas de la función “siblings()”. Busca en la ayuda de jquery información de esta función y utilízala.

Ejemplo 07 (hide y show)

Utiliza el fichero *exercise-07.html*. Crea el código jquery necesario para que cuando se haga clic en el div que es un artículo (clase `.article`), éste se oculte. Cuando se haga clic en el botón que está justo a la izquierda, el artículo, se tiene que volver a abrir.

Para hacer este ejercicio necesitas de la función “siblings()”. Busca en la ayuda de jquery información de esta función y utilízala.

Ejemplo 08

Modifica el ejercicio 06 para que haya una duración “slow” en el efecto de cuando el elemento *article* se muestra y se oculta.

Ejemplo 09

Modifica el ejercicio 07 para que haya una duración de 5 segundos en el efecto de cuando el elemento *article* se muestra y se oculta.

Ejemplo 10 (fadeToggle)

Utiliza el código del ejercicio 06 y aplica el método *fadeToggle* para que se use este en lugar de *toggle*.

Ejemplo 11 (fadeIn, fadeOut)

Utiliza el código del ejercicio 07 y aplica en lugar de los métodos de mostrar y ocultar, los métodos de *fadeIn* y *fadeOut*, en lugar de *hide* and *show*.

Ejemplo 12 (slideToggle)

Coloca un botón en el código del ejercicio 06 de forma que todos los artículos se “cierren” cuando se haga clic en él. Del mismo modo, al volver a hacer clic en ese botón, todos los artículos tienen que volver a mostrarse.

Ejemplo 13 (slideDown y slideUp)

Coloca un par de botones en el código del ejercicio 07 de forma que todos los artículos se “cierren” cuando se haga clic en uno y que se “abran” cuando se haga clic en el otro.

Ejemplo 14 (animate)

Partiendo del código que tenemos en *example-14.html*, tienes que hacer que cuando el ratón entre en el panel lateral, éste amplíe su anchura en 200 píxeles. Cuando el ratón salga del panel lateral, tiene que perder esos 200 píxeles en su anchura para retornar a su tamaño original. La duración de esas animaciones tiene que durar al menos un segundo.

Ejemplo 15 (animate)

Al igual que el ejemplo anterior, el 14, tenemos que hacer que el panel lateral se amplíe y vuelva a su estado original. En este caso, tenemos el código que proporcionamos en el fichero *example-15.html*.

En el resultado final, únicamente se tiene que mostrar un único botón al mismo tiempo, es decir, mientras el panel esté cerrado, se tiene que mostrar el botón de abrir o ampliar.

Del mismo modo, cuando el panel esté abierto, sólo tiene que aparecer el botón para cerrar o reducir el tamaño del panel lateral.

Ejemplo 16 (animate)

Partiendo del código que tenemos en *example-16.html*, vamos a hacer que los diferentes botones que aparecen en la página controlen la posición del bloque rojo.

Cuando se haga clic en el botón “Arriba”, el elemento rojo del centro tiene que cambiar su posición y que suba.

Cuando se haga clic en el botón “Izquierda”, el elemento rojo del centro tiene que cambiar su posición y que se acerque al borde izquierdo de la pantalla.

Con los otros dos botones, tienes que hacer lo que queda para que el elemento se pueda mover hacia abajo y a la derecha.

Ejemplo 17 (animate/stop)

Partiendo del código del fichero *exercise-17.html*.

Tienes que añadir el código necesario para que al pulsar sobre el botón “To 100%”, el ancho del elemento `#percentage` tome el 100% del ancho en 5 segundos. Haz también, que las propiedades *border-top-right-radius* y *border-bottom-right-radius* se pongan con valor *15px*.

Tienes que añadir el código necesario para que al pulsar sobre el botón “To 1%”, el ancho del elemento `#percentage` tome el 1% del ancho en 5 segundos. En este caso, que las propiedades del radio del borde derecho se pongan a *0px*.

Tienes que añadir el código necesario para que al pulsar sobre el botón “Stop” se pare la animación.

Ejemplo 18 (console, callbacks)

Utilizando el código del ejemplo 17, añade los callbacks necesarios para que cuando termine cualquiera de las tres funciones jquery que hemos puesto, se escriba un mensaje en log (sigue leyendo).

El log en jquery y en javascript es un método de un objeto llamado “console” el cual siempre está disponible. Es decir, en cualquier parte de tu código jquery puedes escribir lo siguiente:

```
console.log(“Mensaje de log”);
```

Ejemplo 19 (chaining)

Partiendo del código del fichero *exercise-19.html* haz que cuando se clique en el botón “Play” sucedan los siguientes efectos, uno detrás de otro, sobre el elemento `#percentage`:

1. Que el ancho sea el 100% en 1 segundo.
2. Que la altura cambie aumente en 200 píxeles en 1 segundo.
3. Que se le aplique la animación “slideUp” y que dure 2 segundos.
4. Que se le aplique la animación “slideDown” y que dure 2 segundos.
5. Que se le aplique la animación “fadeOut” y que dure 2 segundos.
6. Que se le aplique la animación “fadeIn” y que dure 2 segundos.
7. Que el ancho cambie al 1% en 1 segundo.
8. Que la altura cambie a 50px en 1 segundo.

Que no se te olvide incluir la función

```
$(document).ready(function () {  
    });
```

Ejemplo 20 (text, html y val)

Partiendo del código del fichero *exercise-20.html* tienes que hacer que pulsando en los botones de la parte inferior se abra una alerta con el contenido que indica cada uno.

El botón “Obtiene el valor del párrafo `#name`” tiene que mostrar una alerta con el contenido de ese párrafo.

El botón “Obtiene el código HTML de `#info`” tiene que mostrar una alerta con el código HTML de ese identificador.

El botón “Obtiene el valor del formulario del email” tiene que mostrar una alerta con el contenido que está dentro del formulario, en la parte del email.

Ejemplo 21 (text, html y val)

Partiendo del código del fichero *exercise-21.html* tienes que hacer que los tres botones de abajo hagan sus funciones.

El botón “Traducir a inglés” implica que al pulsarlo, el contenido de los párrafos de cada elemento del formulario tiene que traducirse a inglés.

El botón “Cambiar formato” tiene que hacer que los títulos de los párrafos se pongan todos en negrita.

El botón “Valores por defecto” tiene que hacer que se cambie el valor de cada uno de los elementos de formulario a un valor concreto. Nota, para especificar un valor de fecha válido, tienes que hacerlo en el siguiente formato: YYYY-MM-DD, por ejemplo: 2018-12-01, para el 1 de diciembre de 2018.

Ejercicio 22 (attr)

Partiendo del código del fichero *exercise-22.html* tienes que hacer que los tres botones de abajo hagan sus funciones.

Cada botón indica el id de un elemento que debería existir en el código HTML del ejercicio anterior. Cada botón, además, indica un atributo que es el que hay que consultar. Fíjate que el objetivo de este ejercicio es que utilices la función “attr” de jquery, es decir, que consultes el valor de ciertos atributos del código HTML que se propone.

Cada botón tiene un texto autoexplicativo de la función que tiene que realizar.

Ejercicio 23 (append, remove)

Partiendo del código del fichero *exercise-23.html*, tiene que suceder que cuando se haga clic en el botón de “Añadir fila al final”, se inserte una fila nueva, con el mismo contenido que la fila original (todas las celdas tienen una X).

Al pulsar sobre el botón “Borrar fila del final”, se tiene que eliminar la última fila de la tabla. Para ello, tendrás que aplicar el método “remove” a la última ocurrencia de la etiqueta “<tr>” del objeto tabla “<table>”. Busca un selector que te permita obtener la última ocurrencia.

Ejercicio 24 (append)

Partiendo del código del fichero *exercise-24.html*, haz que al hacer clic Añadir elementos a los ya existentes con el métodos append.

Ejercicio 25 (toggleClass)

Partiendo del código del fichero *exercise-25.html* añade el código necesario para que al hacer clic sobre un botón, el elemento padre del botón quede dentro de un borde rojo de 2 píxeles de ancho.

De la misma forma que hicimos para el ejercicio donde buscamos información para el método *siblings*, busca en este caso, información sobre el método *parent*. De esta forma, no aplicamos *toggleClass* sobre el botón donde hemos hecho clic, sino sobre el elemento padre de este.

Ejercicio 26 (eventos) - Ejercicio opcional.

Crea una tabla como la de la imagen siguiente. Cada cuadro corresponde a un evento. Procura que se ejecute una función de jquery asociada a cada evento.

Por ejemplo, la primera casilla es “clic”. Por lo tanto, abajo, aparecerá un mensaje siempre que se haga clic en esa casilla.

La segunda casilla es “dblclick”. Por lo tanto, abajo, debe salir un mensaje siempre que se haga doble clic en esa casilla.

La tercera casilla es “mouseenter”. Por lo tanto, abajo, debe aparecer un mensaje siempre que el ratón entre en esa casilla.

Lo que estamos llamando “abajo” es un div con fondo y borde azules. Utilizando la función “prepend” podemos registrar cualquier evento y que este aparezca en la primera posición del div. No hace falta que incluyas la fecha y la hora, pues eso implica ciertos conocimientos de javascript que no son importantes por el momento.

Busca información de aquellos eventos como “submit”, “change”, “focus” y en la casilla que les toque, introduce un pequeño formulario para que cuando se cumpla la condición necesaria para que se lance el evento, muestren un mensaje en el div azul del final de la página.

Nota. Esta tabla también la puedes encontrar en la última diapositiva de la presentación de jquery de clase.

Nota. Podéis utilizar el fichero que se llama “simple-events.html”. Buscad ahí cómo poner la fecha para el resto de elementos, pues el de click y el del doble click os los doy hechos.

click	keypress	submit <input type="button" value="Click to submit"/>	load
dblclick	keydown	change <input type="text" value="Change me"/>	resize
mouseenter	keyup <input type="text" value="Press any key on the input below"/>	focus <input type="text" value="Bring the focus to me"/>	scroll
mouseleave	<input type="text" value="press any key here"/>	blur <input type="text" value="Take the focus out"/>	unload

19:02:50 - Mouse leave!
 19:02:41 - Mouse entered!
 19:02:41 - Mouse leave!

Nota. El método “load” de jquery es diferente del evento “load” de javascript.

La diferencia puede ser un poco complicada de entender, pero en cualquier caso, buscad la información de “load” de javascript si queréis implementar el evento. Si no, no importa, podéis no hacer ese evento.

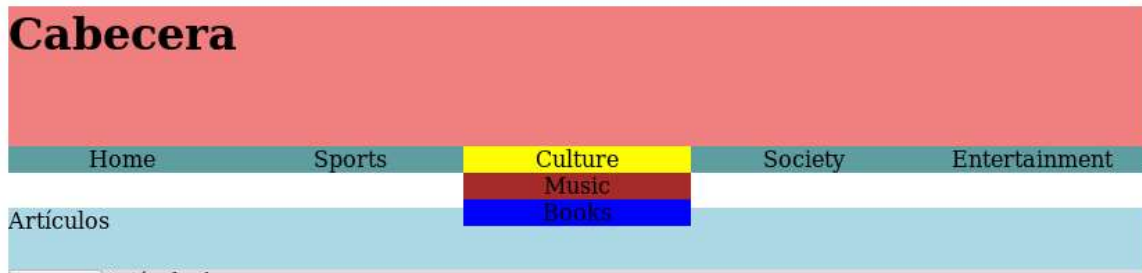
Ejemplo menús (children)

~~Vamos a crear, utilizando animate, unos menús contextuales.~~

~~Utilizamos el código de *example-menu.html*. Definimos un menú con las líneas que están debajo de las secciones *Sport*, *Culture*, *Society* y *Entertainment*.~~

Cabecera				
Home	Sports	Culture	Society	Entertainment
Artículos	Soccer	Music	Divinities	Movies
	Basket	Books	Famous people	Concerts
Open	Artículo 1			
-	Artículo 2			

Se trata de conseguir que, cuando el ratón pase por encima de uno de esos menús, se abra sólo la parte de debajo:



Fíjate en cómo está construida la página. En los "div" que utilizan la propiedad "position: relative" y "position: absolute".

Utilizando los métodos *show* y *hide* puedes conseguirlo. Además, busca en la API de jquery información del método "children". Con este método podrás hacer selecciones en los hijos de algún elemento que ya hayas seleccionado. Por ejemplo:

```
$("#mydiv").children("p").css("color","blue")
```

Con ese código, pondríamos el color azul a aquellos elementos "p" que cuelgan del elemento con identificador "#mydiv".