**jQuery**

jQuery es una de las librerías de JavaScript más utilizadas en el mundo.

En 2006, cuando se lanzó, los principales navegadores manejaban JavaScript de forma ligeramente diferente. jQuery simplificó el proceso de escribir JavaScript del lado del cliente, y también garantizó que tu código funcionara de la misma manera en todos los navegadores.

En este curso, aprenderás a usar jQuery para seleccionar, eliminar, copiar y modificar diferentes elementos de la página.

**- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -**

Paso 1 - Aprende cómo funciona la etiqueta script y la función "document ready"

Ahora estamos listos para aprender jQuery, la herramienta de JavaScript más popular de todos los tiempos.

Antes de empezar a usar jQuery, necesitamos agregar algunas cosas a nuestro HTML.

Primero, agrega un elemento script en la parte superior de tu página. Asegúrate de cerrarlo en la siguiente línea.

Tu navegador ejecutará cualquier JavaScript dentro de un elemento script, incluyendo jQuery.

Dentro de tu elemento script, agrega este código: $(document).ready(function() { a tu script. Luego ciérralo en la siguiente línea (todavía dentro de tu elemento script) con: });

Aprenderemos más sobre functions más tarde. Lo importante a saber es que el código que pongas dentro de esta function (función) se ejecutará tan pronto como tu navegador haya cargado tu página.

Esto es importante porque sin tu document ready function, tu código podría ejecutarse antes de que se procese tu HTML, lo que causaría errores.

Paso 2 - Apunta a elementos HTML con selectores usando jQuery

Ahora tenemos una función document ready.

Ahora vamos a escribir nuestra primera declaración de jQuery. Todas las funciones de jQuery comienzan con un $, usualmente conocido como un operador de signo de dólares.

jQuery a menudo selecciona un elemento HTML con un selector, y luego le hace algo a ese elemento.

Por ejemplo, hagamos que todos tus elementos button reboten. Simplemente agrega este código dentro de tu función document ready:

$("button").addClass("animated bounce");

Ten en cuenta que ya hemos incluido tanto la librería jQuery como la librería Animate.css en segundo plano para que puedas usarlos en el editor. Así que estás usando jQuery para aplicar la clase Animate.css bounce a tus elementos button.

Paso 3 - Apunta a elementos por clase usando jQuery

¿Ves cómo hemos hecho que todos tus elementos button reboten? Los seleccionamos con $("button"), luego agregamos algo de clases CSS a ellos con .addClass("animated bounce");.

Acabas de usar la función .addClass() de jQuery, que te permite agregar clases a los elementos.

Primero, apuntemos tus elementos div con la clase well usando el selector $(".well").

Ten en cuenta que, al igual que con las declaraciones CSS, escribes un . antes del nombre de la clase.

Luego usa la función .addClass() de jQuery para agregar las clases animated y shake.

Por ejemplo, puedes hacer que todos los elementos con la clase text-primary se agiten agregando lo siguiente a tu función document ready function:

$(".text-primary").addClass("animated shake");

Paso 4 - Apunta a elementos por id usando jQuery

También puedes apuntar a elementos por su atributo id.

Primero apunta a tu elemento button con el id target3 usando el selector $("#target3").

Ten en cuenta que, al igual que con las declaraciones CSS, escribes un # antes del nombre del id.

Luego usa la función .addClass() de jQuery para agregar las clases animated y fadeOut.

Así es como harás que el elemento button con el id target6 se desvanezca:

$("#target6").addClass("animated fadeOut");

Paso 5 - Elimina tus funciones jQuery

Estas animaciones fueron geniales al principio, pero ahora distraen un poco.

Elimina las tres funciones de jQuery que están dentro de tu document ready function, pero deja intacta tu función document ready function en sí.

Paso 6 - Apunta al mismo elemento con múltiples selectores jQuery

Ahora conoces tres maneras de seleccionar elementos: por tipo: $("button"), por clase: $(".btn"), y por id $("#target1").

Aunque es posible agregar múltiples clases en una sola llamada .addClass(), vamos a agregarlos al mismo elemento de tres maneras diferentes.

Usando .addClass(), agrega solo una clase a la vez al mismo elemento, de tres maneras diferentes:

Agrega la clase animated a todos los elementos con el tipo button.

Agrega la clase shake a todos los botones con la clase .btn.

Agrega la clase btn-primary al botón con id #target1.

Nota: Solo debes apuntar a un elemento y agregar solo una clase a la vez. En total, tus tres selectores individuales terminarán agregando las tres clases shake, animated, y btn-primary a #target1.

$('button').addClass('animated');

$('.btn').addClass('shake');

$('#target1').addClass('btn-primary');

Paso 7 - Elimina clases de un elemento con jQuery

De la misma manera en que puedes agregar clases a un elemento con la función addClass() de jQuery, puedes eliminarlos con la función removeClass() de jQuery.

Así es como lo harías para un botón específico:

$("#target2").removeClass("btn-default");

Eliminemos la clase btn-default de todos nuestros elementos button.

Paso 8 - Cambia el CSS de un elemento usando jQuery

También podemos cambiar el CSS de un elemento HTML utilizando directamente jQuery.

jQuery tiene una función llamada .css() que te permite cambiar el CSS de un elemento.

Así es como cambiaríamos su color a azul:

$("#target1").css("color", "blue");

Esto es ligeramente diferente a una declaración CSS normal, porque la propiedad CSS y su valor están entre comillas y separados con coma en lugar de con dos puntos.

Elimina tus selectores de jQuery, dejando document ready function vacío.

Selecciona target1 y cambia su color a rojo.

Paso 9 - Deshabilita un elemento usando jQuery

También puedes cambiar las propiedades que no son CSS de tus elementos HTML con jQuery. Por ejemplo, puedes desactivar botones.

Cuando deshabilitas un botón, se mostrará en gris y no se podrá hacer clic sobre este.

jQuery tiene una función llamada .prop() que te permite ajustar las propiedades de los elementos.

Así es como deshabilitarías todos los botones:

$("button").prop("disabled", true);

Deshabilita solo el botón target1.

Paso 10 - Cambia el texto dentro de un elemento usando jQuery

Con jQuery, puedes cambiar el texto entre las etiquetas de abertura y cierre de un elemento. Puedes incluso cambiar el código HTML.

jQuery tiene una función llamada .html() que te permite agregar etiquetas HTML y texto dentro de un elemento. Cualquier contenido anterior dentro del elemento será completamente reemplazado con el contenido que proporciones usando esta función.

Así es como reescribirías y enfatizarías el texto de nuestro título:

$("h3").html("<em>jQuery Playground</em>");

jQuery también tiene una función similar llamada .text() que solamente altera el texto sin agregar etiquetas. En otras palabras, esta función no evaluará ninguna etiqueta HTML que le pasemos, pero en cambio, la tratará como el texto por el que quieres reemplazar el contenido existente.

Cambia el botón con id target4 enfatizando su texto.

Vea nuestro artículo de noticias para <em> aprender la diferencia entre <i> y <em> y sus usos.

Ten en cuenta que mientras la etiqueta <i> tradicionalmente se usaba para enfatizar texto, fue adaptada para ser usada como etiqueta para iconos. La etiqueta <em> ahora es ampliamente aceptada como la etiqueta de énfasis. Cualquiera de las dos servirá para este desafío.

Paso 11 - Remueve un elemento usando jQuery

Ahora vamos a eliminar un elemento HTML de su página utilizando jQuery.

jQuery tiene una función llamada .remove() que quitará un elemento completamente.

Remueve el elemento #target4 de la página utilizando la función .remove().

Paso 12 - Usa appendTo para mover elementos con jQuery

Ahora intentemos mover elementos de un div a otro.

jQuery tiene una función llamada appendTo() que te permite seleccionar elementos HTML y agregarlos a otro elemento.

Por ejemplo, si quisiéramos mover target4 desde nuestro "right well" a nuestro "left well", usaríamos:

$("#target4").appendTo("#left-well");

Mueve tu elemento target2 desde tu left-well a tu right-well.

Paso 13 - Clona un elemento usando jQuery

Además de mover elementos, también puedes copiarlos de un lugar a otro.

jQuery tiene una función llamada clone() que hace una copia de un elemento.

Por ejemplo, si quisiéramos copiar target2 de nuestro left-well a nuestro right-well, usaríamos:

$("#target2").clone().appendTo("#right-well");

¿Notaste que esto implica pegar dos funciones jQuery juntas? Esto es llamado function chaining (encadenamiento de funciones) y es una forma conveniente de lograr cosas con jQuery.

Clona tu elemento target5 y agrégalo a tu left-well.

Paso 14 - Apunta al padre de un elemento usando jQuery

Cada elemento HTML tiene un elemento parent (padre) del cual inherits (hereda) propiedades.

Por ejemplo, el elemento h3 en tu jQuery Playground tiene el elemento padre de <div class="container-fluid">, que en sí mismo tiene el elemento padre de body.

jQuery tiene una función llamada parent() que te permite acceder al padre del elemento que hayas seleccionado.

Aquí hay un ejemplo de cómo usarías la función parent() si quieres dar al elemento padre del elemento left-well un color de fondo azul:

$("#left-well").parent().css("background-color", "blue")

Dale al padre del elemento #target1 un color de fondo rojo.

Paso 15 - Apunta al hijo de un elemento usando jQuery

Cuando los elementos HTML se colocan un nivel por debajo de otro, se denominan hijos de ese elemento. Por ejemplo, los elementos botón en este desafío con el texto #target1, #target2, y #target3 son todos hijos del elemento <div class="well" id="left-well">.

jQuery tiene una función llamada children() que te permite acceder a los hijos del elemento que hayas seleccionado.

Aquí hay un ejemplo de cómo usarías la función children() para dar a los hijos de tu elemento left-well el color blue:

$("#left-well").children().css("color", "blue")

Dale a todos los hijos de tu elemento right-well el color naranja.

Paso 16 - Apunta a un hijo específico de un elemento usando jQuery

Has visto por qué los atributos id son muy convenientes para apuntar con los selectores de jQuery. Pero no siempre tendrás tan buenos id's con los que trabajar.

Afortunadamente, jQuery tiene algunos otros trucos para apuntar a los elementos correctos.

jQuery usa selectores de CSS para apuntar elementos. El selector target:nth-child(n) de CSS, te permite seleccionar todos los enésimos elementos con el tipo de elemento o la clase apuntada.

Así es como le darías al tercer elemento en cada "well" la clase de rebote:

$(".target:nth-child(3)").addClass("animated bounce");

Haz que el segundo niño en cada uno de sus elementos de "well" rebote. Debes seleccionar los hijos de los elementos con la clase target.

Paso 17 - Apunta a elementos pares utilizando jQuery

También puedes apuntar a elementos basado en sus posiciones, usando los selectores :odd o :even.

Ten en cuenta que jQuery es indexado desde el cero, lo que significa que el primer elemento de una selección tiene la posición 0. Esto puede ser un poco confuso, ya que contra-intuitivamente, :odd selecciona el segundo elemento (posición 1), cuarto elemento (posición 3), y así sucesivamente.

Así es como apuntas a todos los elementos impares con la clase target y les das clases:

$(".target:odd").addClass("animated shake");

Intenta seleccionar todos los elementos pares target y darle las clases de animated y shake. Recuerda que even se refiere a la posición de los elementos con un sistema de indexado desde cero en mente.

Paso 18 - Usa jQuery para modificar la página completa

Hemos terminado de jugar con nuestra área de juego de jQuery. ¡Vamos a derribarlo!

jQuery también puede seleccionar al elemento body.

Así es como haríamos que el cuerpo entero se desvaneciera: $("body").addClass("animated fadeOut");

Pero hagamos algo más dramático. Agrega las clases animated y hinge a tu elemento body.