

Códigos manual jQuery. Parte I, II y III

Entrega de los códigos del [manual jQuery](#). Parte 1, 2 y 3.

En un documento pdf responde a las siguientes preguntas:

1. En una línea, define qué es jQuery.

Es un framework Javascript, que sirve como base para la programación avanzada de aplicaciones, que aporta una serie de funciones o códigos para realizar tareas habituales.

2. Identifica la última versión jQuery

La última versión es la 2.2.0

3. Indica las diferencias entre la versión DEVELOPMENT y PRODUCTION

La versión de “development” tiene el código fuente bien formateado y estructurado por si se desea aprender su funcionamiento. Y la versión “production” está minificada para ocupar los menos bytes posibles y por consiguiente sobrecargar menos el navegador a la hora de visualizar la página.

4. Indica la línea donde introduces todo el código de la librería jQuery.

Dentro de la etiqueta “head” del “html”.

5. Indica qué es el jQuery CDN.

CDN es el acrónimo de *Content Delivery Network*, algo así como Red de Entrega de Contenido, es decir un servicio que, en el caso de jQuery, nos permite incluir las librerías de código del framework desde los servidores de un tercero.

6. Indica brevemente y con tus palabras las ventajas e inconvenientes del CDN de jQuery.

Ventajas:

- Mayor velocidad de carga, ya que jQuery está replicado en servidores de grandes empresas que probablemente envíen el script más rápido que cualquier servidor compartido por ejemplo.
- Cacheado probable: Si la persona que visita tu página ya ha visitado otra antes con el CDN de jQuery tendrá en la caché el script y no tenga que cargarlo.

Inconvenientes:

- Hay que tener conexión a internet para conectarse al CDN.
- No hay control total sobre el script. Esto significa que no lo puedes modificar.

7. Indica la línea donde introduces las últimas versiones de al menos dos jQuery CDN.

Dentro de la etiqueta “head” del HTML

```
<!-- Grab Google CDN's jQuery; fall back to local if necessary -->
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
<script src="//ajax.googleapis.com/ajax/libs/jquery/2.2.0/jquery.min.js"></script>
</head>
<body>
</body>
</html>
```

8. Indica cómo jQuery ejecuta un código cuando el árbol DOM está totalmente cargado. Indica el equivalente en JavaScript.

`$(document).ready()`

9. Función \$ o función jQuery. Indica brevemente los argumentos que puedes enviarle. Añádele a la explicación un breve código de ejemplo (distinto al del manual)

10. Indica cómo puedes reemplazar el clásico `$(document).ready(){...}` con jQuery.

`$(function())`

11. En una línea, explica qué hace el método `each()` de jQuery.

Un método para realizar acciones con todos los elementos que concuerdan con una selección realizada con la función jQuery -también llamada función `$()`.

12. Indica el argumento que ha de enviársele al método `each()`.

Recibe una función que es la que se ejecuta para cada elemento.

13. Englobado en el contexto del `each`:

1. Explica la utilidad de la palabra reservada `this`.

Con la variable “this” se accede a ese elemento del DOM.

2. Indica cómo se utiliza el índice de la iteración.

Recorre todos los elementos con el selector indicado.

3. Explica la utilidad de `return false`.

Si devuelve false se detiene por completo la iteración.

4. Indica la diferencia entre `return true` y no ponerlo. Explícalo mediante un trozo de código.

Si se le pone “return true” pasa directamente a la siguiente iteración del bucle. Por el contrario si no se pone nada sigue comprobando las demás líneas de código hasta que llegue a otra nueva iteración.

14. Indica las diferencias y semejanzas entre el método size() y la propiedad length. Indica las ventajas e inconvenientes de utilizar uno u otra.

Size devuelve el número de elementos seleccionados mientras que length aparte de devolverlo lo almacena por lo que es más rápido y más aconsejable.

15. Indica qué hace el método data().

Sirve tanto para guardar un dato en un elemento como para consultarlo

16. Tipos de datos admitidos por data()

Soporta cualquier tipo de variable, numérica, un array o incluso un objeto Javascript o jQuery

17. Indica qué significa que data() almacena valores por referencia.

Quiere decir que no se harían copias independientes del objeto a guardar, sino que permanecería tal cual y lo que se asignaría como dato es una referencia a ese único objeto.

18. Cuántos objetos se crean si data() opera sobre un conjunto de elementos.

Se guarda un dato por cada elemento del objeto jQuery seleccionado: En caso que en el objeto jQuery sobre el que estemos almacenando cosas con data() haya referencias a varios elementos de la página, el dato se almacena en todos los elementos.

19. Indica qué hace el método removeData(). ¿Está sobrescrito?

Este método sirve para eliminar un dato de un elemento y su funcionamiento es tan simple como enviar por parámetro el dato que se quiere eliminar del elemento.

20. Identifica con tu propio código (en una línea a ser posible) los distintos tipos de selectores. Indica cómo se recogen en una variable.

```
$(document).ready(function() {  
    $("#boton").click(function(evento) {  
        var selectorEtiqueta = $("h2");  
        var selectorId = $("#id").attr("value");  
        var selectorClase = $(".clase").attr("value");  
        var selectorVariasClases = $(".clase1.clase2").attr("value");  
        var todo = $("*");  
    });  
});
```