

Alumno/a: José Manuel González Peña

Ajax en jQuery

1. Con respecto a las solicitudes HTTP GET y POST, indica cuál es destructiva y cuál no.

La solicitud GET se utiliza para operaciones no destructivas porque sólo recibe datos del servidor sin cambiar los datos.

La solicitud POST se utiliza para operaciones destructivas, es decir, cuando queremos modificar cualquier dato del servidor.

2. Indica los tipos de datos devueltos en las solicitudes AJAX.

Text, html, script, json, jsonp, xml

3. Indica por qué las respuestas AJAX sólo pueden manejarse mediante funciones callback.

Cuando realizamos una petición, ésta tiene que devolver un dato, es decir, al hacer el callback nos dará una respuesta correcta o un error al comunicándonos con el servidor.

4. El método ajaxSetup() establece los valores por defecto en las peticiones Ajax. Indica por qué su uso no está recomendado.

Porque el método \$.ajax() la mayoría de los parámetros son opcionales y con \$.ajaxSetup() hay parámetros que los tenemos que poner siempre.

5. Indica los 6 manejadores de eventos globales de AJAX.

- .ajaxComplete()
- .ajaxError()
- .ajaxSend()
- .ajaxStart()
- .ajaxStop()
- .ajaxSuccess()

6. Indica al objeto al que han de estar asociados los métodos anteriores.

\$(document)

7. Los métodos AJAX abreviados son 5. Indica cuáles son y utilidad de cada uno.

- jQuery.get(): Carga datos del servidor con una respuesta GET
- jQuerygetJSON(): Carga datos JSON del servidor con una respuesta HTTP
- jQuery.getScript(): Carga un fichero JS desde el servidor con una respuesta GET y lo ejecuta
- jQuery.post(): Carga datos del servidor con una respuesta POST
- .load(): Carga los datos del servidor y devuelve un elemento HTML

8. En jQuery, indica la diferencia entre .ajaxSucess() y .ajaxComplete()

.ajaxComplete() se realiza siempre que hay respuesta del servidor y .ajaxSuccess() cuando hay respuesta y es correcta (200).