

# Chuleta para el examen

## “Peticiones AJAX a un servidor”

1. ¿Cómo podemos crear o generar un objeto “AJAX”?  
Con `new XMLHttpRequest()`;
2. ¿Qué métodos hemos visto hasta ahora para enviar una petición a un servidor?  
POST - GET
3. ¿Con qué métodos de los anteriores se puede enviar información a un servidor?  
POST - GET
4. Si enviamos información, al hacer una petición con POST, ¿dónde debemos poner la información que vamos a enviar?  
En el cuerpo de la solicitud
5. Si enviamos información, al hacer una petición con GET, ¿dónde debemos poner la información que vamos a enviar?  
Ira en la url
6. Pon un ejemplo del formato que hay que seguir para enviar información a un servidor si hacemos la petición con GET.  
`https://direccion?valor1=value&valor2=value;`
7. Pon un ejemplo del formato que hay que seguir si quieres enviar, junto a una petición, dos parámetros llamados “nombre” y “edad”. Primero con GET y después con POST.  
GET: `https://direccion?nombre=value&nombre=value;`  
  
POST:  
Establecemos el tipo de contenido en `setRequestHeader()`, si es `application/json` o `form urlenconded`, y lo enviamos con `send()`, pasandole por parámetro dicho contenido;
8. ¿Qué método del objeto “AJAX” utilizamos para preparar una petición a un servidor?  
CON `XMLHttpRequest()`;
9. ¿Qué necesitamos para preparar una petición a un servidor?  
El objeto Ajax, Configurar con `open()` el tipo de solicitud y enviar ese objeto con `send()`
10. ¿Qué ocurre si en la petición a un servidor se solicita un archivo de tipo PHP? ¿Qué enviará el servidor como respuesta?  
Enviar a un archivo json

11. ¿Qué ocurre si en la petición a un servidor se solicita un archivo de tipo TXT? ¿Qué enviará el servidor como respuesta?  
Pues un archivo txt
12. ¿Qué método del objeto “AJAX” se utiliza para enviar la petición al servidor?  
Send()
13. ¿Qué hay que hacer antes de enviar una petición a servidor si tenemos que hacerla con POST y enviamos alguna información?  
Open()
14. ¿En qué se diferencia una petición a servidor, en la que enviamos un dato, si la hacemos con POST o si la hacemos con GET?  
Post: cuerpo de la solicitud - Get: a través de la URL
15. ¿En qué se diferencia una petición a servidor, en la que NO enviamos ningún dato, si la hacemos con POST o si la hacemos con GET?  
Pues en Get la solicitud no llevará datos en el cuerpo de dicha solicitud mientras que en Post es capaz de enviar información vacía.
16. ¿En qué propiedad del objeto “AJAX” se almacena la definición de la función que se ejecutará cuando se reciba la respuesta del servidor?  
Onload
17. ¿Qué evento se dispara cuando se recibe en el cliente la respuesta del servidor?  
Load o onload
18. ¿Qué valores se almacenan en la propiedad readyState del objeto “AJAX”? Indica su valor y su significado correspondiente.  
0: sin iniciación de la solicitud  
1: no se ha enviado la solicitud, pero si ha habido conexión  
2: se ha enviado la solicitud, pero no se ha completado  
3: en este caso la respuesta esta en proceso, es decir el servidor esta en funcionamiento enviado los datos  
4: respuesta lista y procesada
19. ¿En qué propiedad del objeto “AJAX” se almacenará la respuesta enviada desde el servidor?  
En responseText
20. ¿Cuándo estará lista la respuesta enviada desde el servidor en la propiedad del objeto “AJAX” indicada en la cuestión anterior?  
Cuando status sea igual a 200
21. ¿Qué evento se disparará en el objeto “AJAX” en caso de que no se reciba ninguna respuesta del servidor?  
Onerror
22. Si se dispara el evento load del objeto “AJAX” ¿podemos estar seguros de que la respuesta recibida del servidor es una respuesta “adecuada” para nuestra aplicación?  
No del todo, ya que hay que comprobar que status sea 200 por ejemplo

23. ¿Cómo podemos estar seguros de que el servidor nos ha devuelto una respuesta correcta (pista: no basta con el evento load)?  
Cuando status sea igual a 200
24. ¿Cuál es el valor del campo status del objeto “AJAX” que me asegura que la conexión con el servidor ha terminado correctamente?  
Status == 200, es decir 200
25. Si el servidor ha devuelto un valor 404 en el campo status ¿significa eso que la conexión con el servidor ha fallado?  
No, la conexión no ha fallado, más bien es un error de archivo no encontrado.  
Conexión se realiza a nivel de HTTP
26. Si el servidor ha devuelto un valor 404 en el campo status ¿significa eso que la petición al servidor ha fallado?  
No, no ha fallado, hay respuesta, pero igual no hemos recibido el archivo, es decir la petición se ha realizado
27. Si el servidor ha devuelto un valor 404 en el campo status ¿tendremos una respuesta válida en la propiedad response del objeto “AJAX”?  
No, ha fallado la respuesta
28. ¿Es necesario establecer una conexión (a nivel del protocolo http) con el servidor para poder hacer una petición al servidor (a nivel de AJAX)?  
Obviamente, es necesario una conexión a nivel HTTP
29. Explica cuál es la diferencia entre establecer una conexión con el servidor y hacer una petición al servidor.  
Una es conexión a nivel HTTP (comunicación inicial) mientras que la otra es una petición que la programamos, es decir definimos nosotros el tipo de solicitud (get, post)
30. Piensa en uno de los ejercicios que has realizado en el que se hace una petición a una API.  
¿Cómo puedes forzar un error 404?  
  
Cuando hacemos la petición y ponemos equívocamente el nombre del archivo o una url incorrecta
31. Piensa en uno de los ejercicios que has realizado en el que se hace una petición a una API.  
¿Cómo puedes forzar un error que dispare el evento onerror del objeto “AJAX”?  
Cuando no recibimos respuesta, por lo tanto, el error sería provocado cuando introducimos incorrectamente la dirección del servidor, que no haya internet o que no funcione el servidor.