

CLIENT SIDE WEB DEVELOPMENT TEMA7: AJAX RESUME

Index

- Uso AJAX
- Objeto XMLHttpRequest
- Petición Servidor – Asíncrona
- Ejemplo de uso:
- \$.ajax
 - `peticion.php`

1. Uso de Ajax- Petición Servidor - Asíncronica

3

- ❑ Mientras se lanza la petición al servidor, el resto de nuestro código va a seguir ejecutándose.
- ❑ Para hacer una petición al servidor, por ejemplo con el método get, se pasan dos parámetros:
 - ❑ **pagina.php** y una **función** (llamada **callback**) que son necesarias para la gestión de la respuesta recibida por el servidor.
- ❑ La petición al servidor, a la pagina.php, hasta que no haya terminado la petición de ejecutarse no se recibirá la respuesta.

1. Uso de Ajax- Petición Servidor - Asíncrona

4

- ❑ No tiene sentido poner un `console.log (response)` ya que nos dirá `Undefined`, porque su respuesta no ha sido todavía obtenida.
- ❑ Las funciones de callback siempre reciben como parámetro la respuesta del servidor.
- ❑ Hasta que no se ejecuta `pagina.php` , la función de callback no obtiene la respuesta.

```
XMLHttpRequest - AJAX
1 XMLHttpRequest - AJAX
2
3 Petición Servidor - Asíncrona
4
5 var respuesta;
6 $.get('pagina.php', function(resp) { respuesta = resp; });
7 console.log(respuesta); // Indefinido
```

1. Uso de Ajax- XmlHttpRequest

5

- ❑ Para manejar XmlHttpRequest:
 - ❑ Hay que tener en cuenta las diferencias de cada navegador para gestionar la petición.

2. Uso de Ajax

6

- ❑ No es necesario recargar la página, mientras se envía la petición.
- ❑ La respuesta se cargará a través de las funciones callbacks, cuando el servidor haya terminado de darnos la respuesta.
- ❑ Mientras tanto el resto de código seguirá ejecutándose.

1. AJAX NO AJAX

7

SIMPLY EXPLAINED



NO AJAX



AJAX

1. Ejemplos de AJAX

8

- 1.1 Peticiones con el método 'Ajax'
- 1.2 Peticiones con el método GET
- 1.3 Peticiones con el método POST
- 1.4 Peticiones con el método getScript
- 1.5 Peticiones con el método getJSON

1.1 Ejemplos método 'ajax'


9

Formulario de Contacto

Nombre...

Correo electrónico...

Mensaje...

Enviar 

1.1 Ejemplos método 'ajax'

10

```
index.html 100% 100%
12 $(document).on('ready',function(){
13
14
15
16 })
17
18
19
20
21 </script>
22
23 <style>=
99 </head>
100
101 <body>
102 <div id="main">
103 <h1>Formulario de Contacto</h1>
104 <form action="peticion.php" name="fo" method="POST">
105 <input type="text" name="nombre" placeholder="Nombre..." required/>
106 <input type="email" name="mail" placeholder="Correo electrónico..." required />
107 <textarea name="mensaje" placeholder="Mensaje..." required></textarea>
108 <div style="display:block;width:60%;margin:0 auto">
109 <input type="submit" name="send" value="Enviar"/>
110 <div id="status"><!----></div>
111 </div>
112 </form>
113 </div>
114 </body>
115 </html>
```

1.1 Ejemplos método 'ajax'

11

El método ajax, tiene un argumento que es un objeto json
Va a tener todas sus propiedades en forma de un objeto json
Tiene unas 15 propiedades este método ajax.

```
index.html  spin.min.js
2  <html>
3    <head>
4      <meta charset="utf-8"/>
5      <title>Método Animate</title>
6
7      <script src="../../js/jquery-1.7.2.min.js"></script>
8      <script src="../../js/spin.min.js"></script>
9      <script>
10
11
12        $(document).on('ready',function(){
13
14          $.ajax({
15            beforeSend: function (){
16              $('#status').spin({radius:3,width:2,height:2,length:4});
17              |
18            }
19          })
20
21        })
22
23
24
25
26      </script>
27
28      <style>=
104    </head>
```

1. Ejemplos de funcionamiento

12

```
<script src="../../js/jquery-1.7.2.min.js"></script>
<script src="../../js/spin.min.js"></script>
<script>

    $(document).on('ready',function(){

        var pet = $('#main form').attr('action');
        var met = $('#main form').attr('method');
        $.ajax({
            beforeSend: function (){
                $('#status').spin({radius:3,width:2,height:2,length:4});
            },
            url: pet,
            type: met,
        })
    })
})
```

- **beforeSend:** se ejecuta antes de hacer el envío de la petición. Utilizo método **spin** de jquery.
- **url:** dirección de la página a la que reenvío la petición.
- **type:** método que utilize para el envío.

1.1

13

```
index.html  spin.min.js
12      $(document).on('ready',function(){
13
14
15
16      })
17
18
19
20
21      </script>
22
23      <style>=
99  </head>
100
101  <body>
102  <div id="main">
103    <h1>Formulario de Contacto</h1>
104    <form action="peticion.php" name="fo" method="POST">
105      <input type="text" name="nombre" placeholder="Nombre..." required/>
106      <input type="email" name="mail" placeholder="Correo electrónico..." required />
107      <textarea name="mensaje" placeholder="Mensaje..." required/></textarea>
108      <div style="display:block;width:60%;margin:0 auto">
109        <input type="submit" name="send" value="Enviar"/>
110        <div id="status"><!----></div>
111      </div>
112    </form>
113  </div>
114  </body>
115</html>
```

```
<script src="../../js/jquery-1.7.2.min.js"></script>
<script src="../../js/spin.min.js"></script>
<script>

$(document).on('ready',function(){

    var pet = $('#main form').attr('action');
    var met = $('#main form').attr('method');
    $.ajax({
        beforeSend: function (){
            $('#status').spin({radius:3,width:2,height:2,length:4});
        },
        url: pet,
        type: met,
    })
})
})
```

1.1 Ejemplos método 'ajax'

14

- **beforeSend:** se ejecuta antes de hacer el envío de la petición. Utilizo método **spin** de jquery.
- **url:** dirección de la página a la que reenvío la petición.
- **type:** método que utilice para el envío.
- **data:** aquí vamos a construir los datos que vamos a enviar
 - Construir directamente el objeto json
 - o
 - Utilizar serialize, método de jquery que nos permite recoger los datos rellenos en el formulario y pasarlos al servidor en formato json

1.1 Ejemplos método 'ajax'

15

- **beforeSend**: se ejecuta antes de hacer el envío de la petición. Utilizo método **spin** de jquery.
- **url**: dirección de la página a la que reenvío la petición.
- **type**: método que utilice para el envío.
- **data**: aquí vamos a construir los datos que vamos a enviar
 - Construir directamente el objeto json
 - o
 - Utilizar **serialize**, método de jquery que nos permite recoger los datos rellenos en el formulario y pasarlos al servidor en formato json, recuperaría los datos nombre-valor

```
14
15 var pet = $('#main form').attr('action');
16 var met = $('#main form').attr('method');
17 $.ajax({
18     beforeSend: function (){
19         $('#status').spin({radius:3,width:2,height:2,length:4});
20     },
21     url: pet,
22     type: met,
23     data: $('#main form').serialize()
24 })
25
26 })
```

1.1 Ejemplos método 'ajax'

16

➤ Construir directamente el objeto json

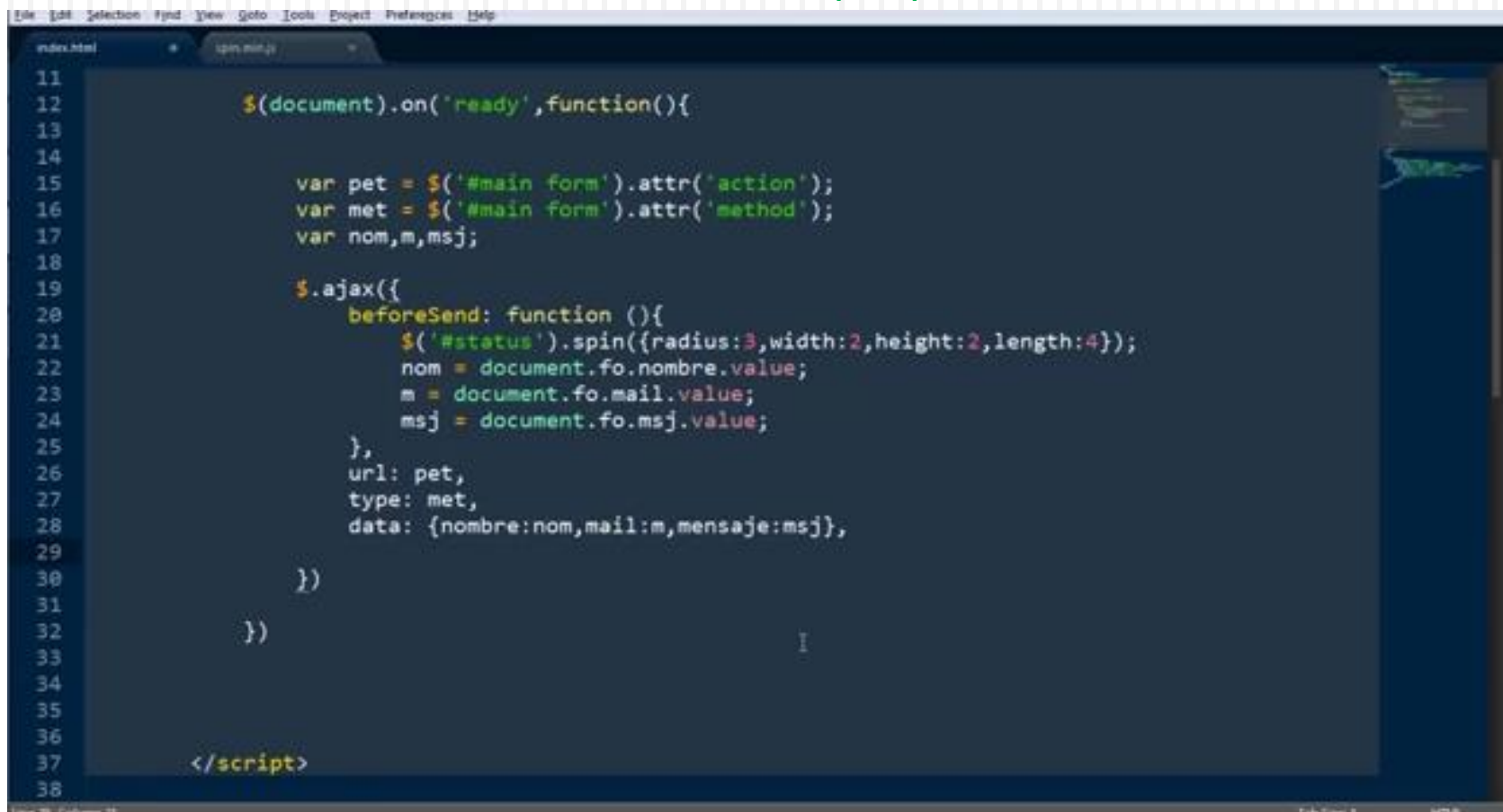
```
File Edit Selection Find View Goto Tools Project Preferences Help
index.html spin.min.js

11
12 $(document).on('ready',function(){
13
14
15     var pet = $('#main form').attr('action');
16     var met = $('#main form').attr('method');
17     var nom,m,msgj;
18
19     $.ajax({
20         beforeSend: function (){
21             $('#status').spin({radius:3,width:2,height:2,length:4});
22             nom = document.fo.nombre.value;
23             m = document.fo.mail.value;
24             msgj = document.fo.msj.value;
25         },
26         url: pet,
27         type: met,
28         data: {nombre:nom,mail:m,mensaje:msgj},
29     })
30 })
31
32
33
34
35
36
37 </script>
38
```


1.1 Ejemplos método 'ajax'

17

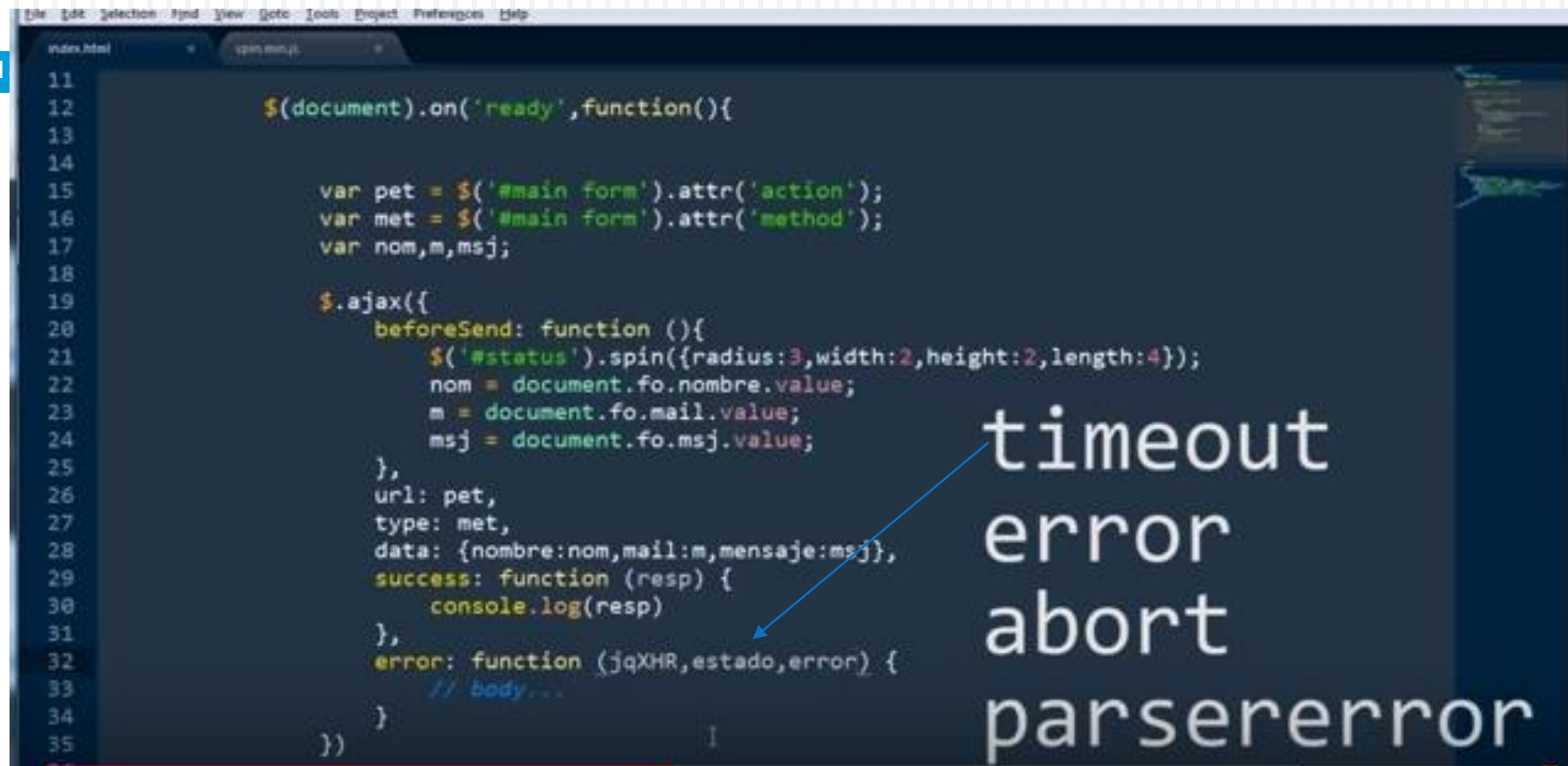
➤ Construir directamente el objeto json

A screenshot of a code editor window with a dark theme. The editor shows a JavaScript file named 'index.html' with jQuery code. The code is as follows:

```
11
12     $(document).on('ready',function(){
13
14
15         var pet = $('#main form').attr('action');
16         var met = $('#main form').attr('method');
17         var nom,m,msgj;
18
19         $.ajax({
20             beforeSend: function (){
21                 $('#status').spin({radius:3,width:2,height:2,length:4});
22                 nom = document.fo.nombre.value;
23                 m = document.fo.mail.value;
24                 msgj = document.fo.msj.value;
25             },
26             url: pet,
27             type: met,
28             data: {nombre:nom,mail:m,mensaje:msgj},
29
30         })
31     })
32
33
34
35
36
37 </script>
38
```

The code is enclosed in a script tag. The editor has a menu bar at the top with options like File, Edit, Selection, Find, View, Goto, Tools, Project, Preferences, and Help. The status bar at the bottom shows 'Line 38 / Column 31' and 'Tab Size: 2'.

1.1 Ejemplos método 'ajax'



The image shows a code editor window with a dark theme. The code is jQuery AJAX code. On the right side, there is a list of error types: 'timeout', 'error', 'abort', and 'parsererror'. A blue arrow points from the 'error' text to the 'error' function in the code.

```
11 $(document).on('ready',function(){
12
13
14
15     var pet = $('#main form').attr('action');
16     var met = $('#main form').attr('method');
17     var nom,m,msg;
18
19     $.ajax({
20         beforeSend: function (){
21             $('#status').spin({radius:3,width:2,height:2,length:4});
22             nom = document.fo.nombre.value;
23             m = document.fo.mail.value;
24             msg = document.fo.msj.value;
25         },
26         url: pet,
27         type: met,
28         data: {nombre:nom,mail:m,mensaje:msg},
29         success: function (resp) {
30             console.log(resp)
31         },
32         error: function (jqXHR,estado,error) {
33             // body...
34         }
35     })
36 }
```

timeout
error
abort
parsererror

success: se ejecuta en caso de que todo haya ido bien y nos devuelve la información del servidor en la vble **resp**.

error: se ejecuta en caso de que haya habido un fallo en el archivo de la petición o un fallo en el servidor, etc...la función tiene 3 parámetros: **jqXHR** (es el objeto XMLHttpRequest), **estado** (valores arriba indicados) y **error**.

1.1 Ejemplos método 'ajax'

19



complete: se ejecutará después de haberse ejecutado o bien success o bien error. Tiene dos parámetros: **jqXHR** y **estado**, éste último puede adquirir todos los valores arriba indicados.

1.1 Ejemplos método 'ajax'

20

```
index.html  spin.min.js
17      var nom,m,msj;
18
19      $.ajax({
20          beforeSend: function (){
21              $('#status').spin({radius:3,width:2,height:2,length:4});
22              nom = document.fo.nombre.value;
23              m = document.fo.mail.value;
24              msj = document.fo.msj.value;
25          },
26          url: pet,
27          type: met,
28          data: {nombre:nom,mail:m,mensaje:msj},
29          success: function (resp) {
30              console.log(resp)
31          },
32          error: function (jqXHR,estado,error) {
33              console.log(estado)
34              console.log(error)
35          },
36          complete: function (jqXHR,estado) {
37              console.log(estado)
38          },
39          timeout: 10000
40      })
41
```

timeout: es el tiempo que esperará la respuesta del servidor, antes de dar un error. En milisegundos.

1.1 Ejemplos método 'ajax'

21

```
index.html  x  spin.min.js  x
18
19      $('#main form').on('submit',function(e){
20
21
22          e.preventDefault();
23
24          $.ajax({
25              beforeSend: function (){
26                  $('#status').spin({radius:3,width:2,height:2,length:4});
27                  nom = document.fo.nombre.value;
28                  m = document.fo.mail.value;
29                  msj = document.fo.msj.value;
30              },
31              url: pet,
32              type: met,
33              data: {nombre:nom,mail:m,mensaje:msj},
34              success: function (resp) {
35                  console.log(resp)
36              },
37              error: function (jqXHR,estado,error) {
38                  console.log(estado)
39                  console.log(error)
40              },
41              complete: function (jqXHR,estado) {
```

En este caso lo que hay que hacer es meter `$.ajax` dentro de un evento de nuestro formulario. Por ejemplo: en el `on` de nuestro formu. Cuando se produzca `el submit`, se hará la `function` descrita como segundo argumento, a la cual se pasa `e` como evento. Con `prevent default`, se `evita` que se envíe el submit del botón de envío y se haga nuestra petición antes.

1.1 Ejemplos método 'ajax'

```
C:\wamp\www\cf\30\index.html - Sublime Text 2 (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

index.html x spin.min.js x
21
22     e.preventDefault();
23
24     $.ajax({
25         beforeSend: function (){
26             $('#status').spin({radius:3,width:2,height:2,length:4});
27             nom = document.fo.nombre.value;
28             m = document.fo.mail.value;
29             msj = document.fo.mensaje.value;
30         },
31         url: pet,
32         type: met,
33         data: {nombre:nom,mail:m,mensaje:msj},
34         success: function (resp) {
35             $("#status").html('');
36             console.log(resp)
37         },
38         error: function (jqXHR,estado,error) {
39             $("#status").html('');
40             console.log(estado)
41             console.log(error)
42         },
43         complete: function (jqXHR,estado) {
44             console.log(estado)
45         },
46         timeout: 10000
47     })
48 }
```

3 characters selected

Tab Size: 4

HTML

1.1 Ejemplos método 'ajax'

```
C:\wamp\www\cf\30\index.html - Sublime Text 2 (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

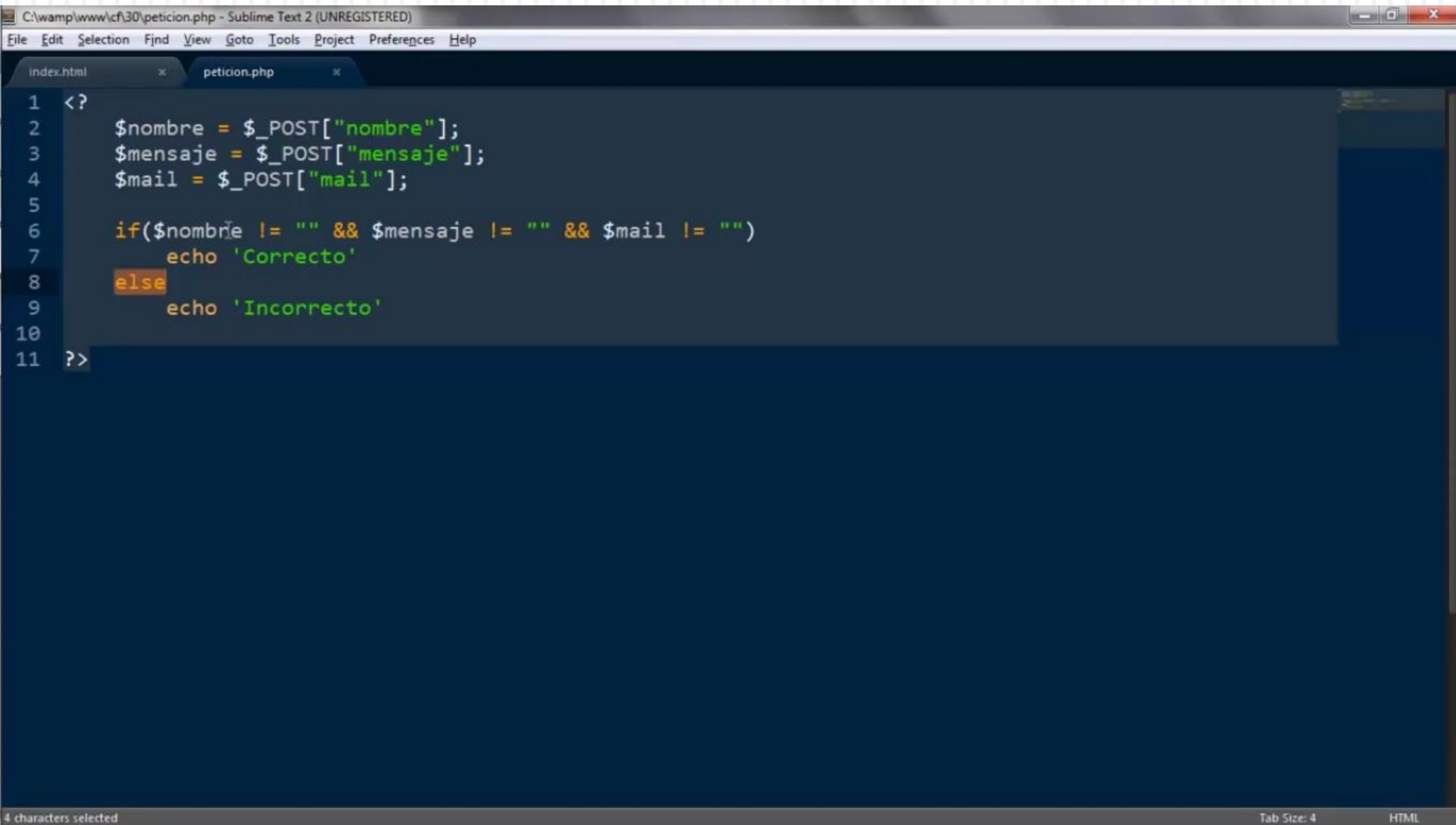
index.html x spin.min.js x
21
22     e.preventDefault();
23
24     $.ajax({
25         beforeSend: function (){
26             $('#status').spin({radius:3,width:2,height:2,length:4});
27             nom = document.fo.nombre.value;
28             m = document.fo.mail.value;
29             msj = document.fo.mensaje.value;
30         },
31         url: pet,
32         type: met,
33         data: {nombre:nom,mail:m,mensaje:msj},
34         success: function (resp) {
35             $("#status").html('');
36             console.log(resp)
37         },
38         error: function (jqXHR,estado,error) {
39             $("#status").html('');
40             console.log(estado)
41             console.log(error)
42         },
43         complete: function (jqXHR,estado) {
44             console.log(estado)
45         },
46         timeout: 10000
47     })
48 }
```

3 characters selected

Tab Size: 4

HTML

1.1 Ejemplos método 'ajax'



The image shows a screenshot of a Sublime Text 2 editor window. The title bar indicates the file path is 'C:\wamp\www\cf30\peticion.php - Sublime Text 2 (UNREGISTERED)'. The menu bar includes 'File', 'Edit', 'Selection', 'Find', 'View', 'Goto', 'Tools', 'Project', 'Preferences', and 'Help'. There are two tabs open: 'index.html' and 'peticion.php'. The 'peticion.php' tab is active, displaying the following PHP code:

```
1 <?
2 $nombre = $_POST["nombre"];
3 $mensaje = $_POST["mensaje"];
4 $mail = $_POST["mail"];
5
6 if($nombre != "" && $mensaje != "" && $mail != "")
7     echo 'Correcto'
8 else
9     echo 'Incorrecto'
10
11 ?>
```

At the bottom of the editor, a status bar shows '4 characters selected' on the left, 'Tab Size: 4' in the center, and 'HTML' on the right.

1.1 Ejemplos método 'ajax'

25

```
index.html
peticion.php

48     })
49
50     })
51
52
53
54
55     </script>
56
57     <style>=
133 </head>
134
135 <body>
136     <div id="main">
137         <h1>Formulario de Contacto</h1>
138         <form action="peticion.php" name="fo" method="POST">
139             <input type="text" name="nombre" placeholder="Nombre..." />
140             <input type="email" name="mail" placeholder="Correo electrónico..." />
141             <textarea name="mensaje" placeholder="Mensaje..." />
142             <div style="display:block;width:60%;margin:0 auto">
143                 <input type="submit" name="send" value="Enviar"/>
144                 <div id="status"></div>
145             </div>
146         </form>
147     </div>
```

Quitar el required

1.1 Ejemplos método 'ajax'

The screenshot shows a web browser window with the address bar at `localhost/cf/30/`. The page title is "Formulario de Contacto". It contains three input fields: "Nombre...", "Correo electrónico...", and "Mensaje...". Below these is a blue button labeled "Enviar" with a green checkmark icon.

The browser's developer console is open, showing a JavaScript error: "Notice: Undefined index: mail in C:\wamp\www\cf\30\peticion.php on line 5". The console also shows a "Success" message. Blue arrows point from the text "Incorrecto" and "Success" to the corresponding entries in the console.

Elements Resources Network Scripts Timeline Profiles Audits Console

```
<br />
<font size='1'><table class='xdebug-error xe-notice' dir='ltr' border='1' cellspacing='0' cellpadding='1'>
<tr><th align='left' bgcolor='#f57900' colspan='5'><span style='background-color: #cc0000; color: #fce94f; font-size: x-large;'>( ! )</span> Notice: Undefined index: mail in
C:\wamp\www\cf\30\peticion.php on line 5</i></th></tr>
<tr><th align='left' bgcolor='#e9b96e' colspan='5'>Call Stack</th></tr>
<tr><th align='center' bgcolor='#eeeeec'>#</th><th align='left' bgcolor='#eeeeec'>Time</th><th align='left' bgcolor='#eeeeec'>Memory</th><th align='left' bgcolor='#eeeeec'>Function</th><th
align='left' bgcolor='#eeeeec'>Location</th></tr>
<tr><td bgcolor='#eeeeec' align='center'>1</td><td bgcolor='#eeeeec' align='center'>0.0016</td><td bgcolor='#eeeeec' align='right'>251488</td><td bgcolor='#eeeeec'>{main}()</td><td
title='C:\wamp\www\cf\30\peticion.php' bgcolor='#eeeeec'>..\peticion.php<b>:</b><b>0</td></tr>
</table></font>
Incorrecto
Success
```

Incorrecto: nos devuelve el error, que nos esta dando el servidor php, se va por el error.
Success: nos lo da el complete

1.1 Ejemplos método 'ajax'

27

Actividad:

Crear el archive index.html

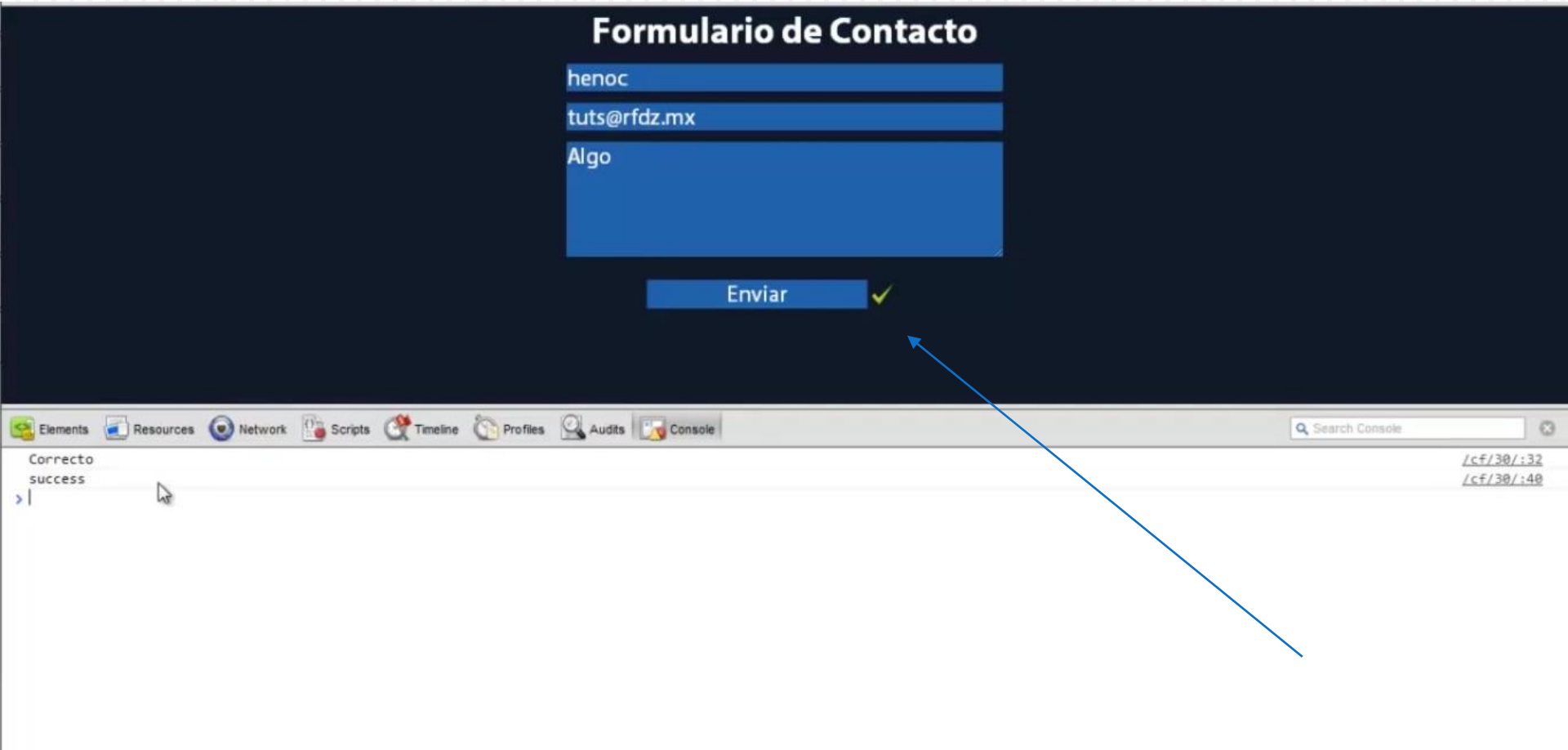
- Form
- Script con el objeto S.ajax

Crear el archive peticion.php

- Devuelva correcto si se rellena el contenido, e incorrecto al contrario.
- Hacer el ejemplo usando serialize
- Hacer que salga una imagen de ok en la div id=status, en caso de ir bien el envío.
- Hacer que salga la imagen de nook en la div id=status, en caso de ir mal el envío y muestre la respuesta por consola.

1.1 Ejemplos método 'ajax'

28



1.1 Ejemplos método 'ajax'

29

