



APRENDERAPROGRAMAR.COM

ARCHIVOS DE EXTENSIÓN  
JS. INCLUIR JAVASCRIPT  
EN WEBS  
REFERENCIANDO UN  
ARCHIVO EXTERNO CON  
SRC (CU01110E)

Sección: Cursos

Categoría: Tutorial básico del programador web: JavaScript desde cero

Fecha revisión: 2029

**Resumen:** Entrega nº10 del Tutorial básico “JavaScript desde cero”.

Autor: César Krall

## JAVASCRIPT EN ARCHIVOS JS

Hemos visto que podemos incluir JavaScript “embebido” dentro de una línea de HTML o dentro de la cabecera del documento HTML. Pero esto será poco práctico cuando tengamos muchas páginas web que necesiten usar las mismas funciones JavaScript. ¿Vamos a estar repitiendo el código en cada página?



## JAVASCRIPT EXTERNO EN ARCHIVOS JS

De la misma forma que el código CSS puede estar contenido en archivos externos con extensión css, el código JavaScript puede independizarse del documento HTML introduciéndolo en un archivo de texto con extensión .js y que se invocará desde el código HTML de la forma que mostramos a continuación:

```
<head>
...
...
<script type="text/javascript" src="rutaDelArchivo1.js"></script>
<script type="text/javascript" src="rutaDelArchivo2.js"></script>
<script type="text/javascript" src="rutaDelArchivo3.js"></script>
...
...
</head>
```

Podemos invocar un solo archivo, o dos, tres, cuatro... tantos como resulten necesarios.

En el atributo src tenemos que especificar la ruta del archivo referenciado (si no indicamos nada, se sobreentiende que está en el mismo directorio que el documento HTML). Supongamos que un archivo externo con código JavaScript se llamara functions.js y estuviera dentro de una carpeta denominada jsf. En ese caso escribiríamos: src="/jsf/functions.js"

En general tener el código JavaScript en archivos independientes será lo más práctico desde el punto de vista del mantenimiento de un sitio web. Tendremos las funciones agrupadas y ordenadas, pudiendo ser invocadas desde cualquier parte de nuestra página web. Esto nos ahorrará tiempo de mantenimiento: si tenemos 200 documentos html, no tendremos que hacer 200 modificaciones sino únicamente modificar el archivo o los archivos con las funciones JavaScript que usamos en todo nuestro sitio web.

Veamos un ejemplo. Escribe este código y guárdalo en un archivo de extensión html:

```
<html>
<head>
<title>Portal web - aprenderaprogramar.com</title> <meta charset="utf-8">
<script type="text/javascript" src="functions.js"></script>
</head>
<body onload="alert('Completada la carga de la página')">
<div>
<p>Aquí un párrafo de texto situado antes de la imagen, dentro de un div contenedor</p>

<p onclick="mostrarMensaje2()">Aquí otro párrafo de texto. JavaScript es un lenguaje utilizado para dotar
de efectos dinámicos a las páginas web.
</p>
</div>
</body>
</html>
```

Crea un archivo de nombre `functions.js` (créalo con un editor de textos como el bloc de notas, Notepad++ o cualquier otro) y escribe el siguiente código:

```
function mostrarMensaje1(){
alert('Bienvenido al curso JavaScript de aprenderaprogramar.com');
}

function mostrarMensaje2(){
alert('Ha hecho click sobre el párrafo inferior');
}
```

El resultado esperado al cargar la página html será el siguiente:

1) Nada más cargarse la página aparecerá una ventana informativa con el texto “Completada la carga de la página”. Esto se debe a que hemos incluido JavaScript en línea asociado a la etiqueta `body`. El evento `onload` se produce cuando el elemento `body` se ha cargado completamente en el navegador. En ese momento se dispara el código JavaScript asociado.

2) Al hacer click sobre la imagen se muestra un mensaje (el mensaje 1), debido a que hemos indicado que cuando se produzca el evento `onclick` sobre la imagen, debe ejecutarse la función `mostrarMensaje1()`. Esta función no se encuentra dentro del documento html, pero el navegador ha incorporado el código JavaScript externo porque así se lo hemos indicado al indicar `src = "functions.php"`. Por tanto el código externo funciona como si fuera código incluido en el propio documento html.

3) Al hacer click sobre el párrafo inferior se mostrará otro mensaje (el mensaje 2), debido a que hemos indicado que cuando se produzca el evento `onclick` sobre el segundo párrafo debe ejecutarse la función `mostrarMensaje2()`.

Nos podemos hacer algunas preguntas como:

### **¿Por qué tenemos una función definida en línea y otras definidas en archivos externos?**

En general será más interesante tener todo el código organizado en un archivo externo en lugar de tenerlo “desperdigado” en scripts dentro de html, código en línea, etc. No obstante, acostúmbrate a que por múltiples motivos te puedes encontrar con páginas web donde la organización del código no sea buena. El navegador aceptará el código que se encuentre, tanto en línea, como interno, como externo. Como programadores deberemos intentar mantener el código tan ordenado como sea posible. Para ello es útil usar archivos externos y agrupar las funciones de un mismo tipo dentro de carpetas y archivos con nombres descriptivos. Por ejemplo si tenemos un reloj javascript, todas las funciones asociadas al reloj pueden ir dentro de un archivo de nombre reloj.js. Si tenemos una calculadora con diferentes funciones, podemos crear una carpeta calculadora y dentro de ella poner los archivos calculosAritmeticos.js, calculosFinancieros.js y calculosCientificos.js por ejemplo.

### **¿No hay que incluir las etiquetas `<script> ... </script>` en el archivo js?**

No, de hecho si lo hacemos el código JavaScript es probable que no funcione al no ser capaz el intérprete del navegador de entender su significado en ese contexto. En los archivos js se incluye código JavaScript únicamente. La etiqueta `<script> ... </script>` es código html que se utiliza para delimitar el código JavaScript dentro de un documento html. Esas etiquetas no forman parte del lenguaje JavaScript, por tanto no tiene sentido incluirlas en un documento JavaScript. Si el navegador encuentra cosas extrañas en un archivo js posiblemente ignore todo su contenido (con lo cual no ejecutará ninguna función y no veremos resultado alguno).

### **¿Puedo ver el código JavaScript si visualizo el código fuente de la página web?**

Depende. El código JavaScript en línea o interno forma parte del documento HTML, por tanto si visualizas el código fuente de la página sí podrás verlo. En cambio el código en archivos externos no podrás verlo directamente dentro del código HTML de la web, aunque sí podrás ver el archivo al que se referencia. Podrás acceder al código JavaScript escribiendo la ruta correspondiente. Por ejemplo `file:///C:/EjemplosCursoJavaScript/functions.js` podría ser una ruta en local (en nuestro ordenador) o `http://aprenderaprogramar.com/media/system/js/modal.js` podría ser una ruta en una página web online. De este modo podemos visualizar el código JavaScript asociado a una página web.

### **¿Entonces me pueden copiar otras personas mi código JavaScript (y yo copiarlo de otras webs)?**

Sí. Ten en cuenta que JavaScript se ejecuta del lado del cliente, es decir, en el ordenador del usuario. Lo mismo que el código HTML y el código CSS.

### **¿Puede robarse información del usuario o introducir virus usando JavaScript?**

En general no, ya que los navegadores y sistemas operativos incluyen medidas de seguridad que impiden que a través de código JavaScript se pueda acceder al ordenador del usuario. No obstante,

siempre hay “mentes perversas” que buscan fallos de seguridad en los sistemas e intentan hacer cosas de este tipo. Es muy difícil, pero a veces lo consiguen (y seguidamente son perseguidos por la policía).

### ¿Deshabilitar JavaScript puede ser una buena medida de seguridad?

Podría serlo, pero no tiene demasiado sentido. Es como no salir de casa para evitar que te roben. Puedes hacerlo, pero te perderías todo lo que hay fuera de tu casa y nadie quiere vivir sin disfrutar de lo que nos puede ofrecer la vida.

### ¿Es posible ejecutar código JavaScript sin estar asociado a una respuesta a un evento?

Sí, escribe este código en un archivo html y visualiza el resultado en tu navegador.

```
<html>
<head>
<title>Portal web - aprenderaprogramar.com</title> <meta charset="utf-8">
</head>
<body>
<div>
<p>Hola</p>
<script type="text/javascript">
alert('Estamos cargando los contenidos de la web...');
</script>
<p>Aquí estamos</p>
</div>
</body>
</html>
```

El resultado esperado es que se cargue en pantalla el texto “Hola”. A continuación se ejecuta el código JavaScript que al no estar incluido dentro de una función se ejecuta directamente. El resultado es que se detiene la carga de la página web y aparece una ventana con el mensaje “Estamos cargando los contenidos de la web...”. Una vez pulsamos aceptar, se terminará la carga de la página y aparecerá el texto “Aquí estamos”. En este caso hemos hecho cosas extrañas (¡sólo para ver un ejemplo de cosas que te puedes encontrar!): hemos incluido un script fuera de las etiquetas <head> ... </head> y hemos dado lugar a la ejecución de un código JavaScript secuencialmente con el progreso de la carga del html, en lugar de dar lugar a su ejecución en respuesta a un evento (que sería lo más normal).

**Próxima entrega:** CU01111E

**Acceso al curso completo** en [aprenderaprogramar.com](http://aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente:  
[http://aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=78&Itemid=206](http://aprenderaprogramar.com/index.php?option=com_content&view=category&id=78&Itemid=206)