



APRENDERAPROGRAMAR.COM

DISABLED JAVASCRIPT EN
INPUT TEXT, SUBMIT,
BUTTON, SELECT, LINK,
STYLE... IMPEDIR ENVÍO
DUPLICADO FORM.
DESACTIVAR CSS
(CU01183E)

Sección: Cursos

Categoría: Tutorial básico del programador web: JavaScript desde cero

Fecha revisión: 2029

Resumen: Entrega nº83 del Tutorial básico "JavaScript desde cero".

Autor: César Krall

DISABLED JAVASCRIPT

Diferentes objetos de tipo HTML Element tienen disponible la propiedad disabled, que permite que estén inactivos o deshabilitados de modo que el usuario no podrá pulsar sobre ellos o elegirlos. Esto puede tener distintas utilidades: impedir el envío duplicado de un formulario por error, desactivar temporalmente algunas opciones, etc.



DESACTIVAR EL BOTÓN DE ENVÍO DE UN FORMULARIO. IMPEDIR ENVÍO DUPLICADO

Cuando un usuario pulsa el botón enviar de un formulario deberíamos esperar que no ejecute ninguna acción adicional hasta que el formulario sea enviado. No obstante, podría ocurrir que por error haga dos veces click, o que la conexión con el servidor se demore y tenga la impresión de que el formulario no se está enviando y vuelva a hacer click sobre el botón de envío. Estas situaciones pueden dar lugar a el envío duplicado de un formulario, situación no deseable en cuanto que duplica procesos, consume recursos y puede dar lugar a problemas como errores en procesos de compra o contabilización de solicitudes, etc.

Para evitar un posible click en la fracción de tiempo que transcurre entre el click en el botón de envío y la carga de la página de destino podemos usar la propiedad disabled (deshabilitado) aplicable a los objetos HTML Element de tipo input y a objetos HTML de tipo button entre otros.

La sintaxis a utilizar es la siguiente:

```
nombreDelObjetoADeshabilitar.disabled = true;
```

Lo normal será deshabilitar el botón de envío en cuanto detectemos el evento click sobre el botón de envío ó submit sobre un formulario. Escribe este código y comprueba cómo se desactivan los botones cuando se pulsa el botón enviar (en este caso lo detectamos usando el evento submit):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head><title>Ejemplo aprenderaprogramar.com</title><meta charset="utf-8">
<style type="text/css"> body {margin-left:30px; font-family: sans-serif;}
.estiloForm {background-color: #f3f3f3; border: solid 2px black; margin-left:20px; width: 330px; padding:10px; }
.estiloForm label {display: block; width: 120px; float: left; text-align:right; margin-bottom: 35px; padding-right: 20px;}
br {clear: left;} input[type="submit"], input[type="reset"] {margin:25px 5px 10px 5px;}
</style>

<script type="text/javascript">
window.onload = function () {
document.forms['formularioContacto'].addEventListener('submit', avisarUsuario);
}
</script>
```

```
function avisarUsuario(evObject) {
  evObject.preventDefault();
  var botones = document.querySelectorAll('.botonFormulario');
  for (var i=0; i<botones.length; i++) {botones[i].disabled = true; }
  var nuevoNodo = document.createElement('h2');
  nuevoNodo.innerHTML = '<h2 style="color:orange;">Enviando... espere por favor</h2>';
  document.body.appendChild(nuevoNodo);
  var retrasar = setTimeout(procesaDentroDe2Segundos, 1000);
}

function procesaDentroDe2Segundos() {document.forms['formularioContacto'].submit();}
</script>
</head>
<body>
<h2>Cursos aprenderaprogramar.com</h2>
<h3>Ejemplos JavaScript</h3>
  <div class="estiloForm">
    <form name = "formularioContacto" method="get" action="http://aprenderaprogramar.com">
      <label>Nombre</label><input id="nombre" name="nombre" type="text"/><br/>
      <label>Apellidos</label><input id="apellidos" name="apellidos" type="text"/><br/>
      <label>Dirección</label><input id="direccion1" name="direccion1" type="text"/>
      <input class="botonFormulario" type="submit" value="Enviar"/>
      <input class="botonFormulario" type="reset" value="Cancelar"/>
    </form>
  </div>
</body></html>
```

En el código anterior hemos activado un retardo utilizando `setTimeout` simplemente para poder comprobar cómo se desactivan los botones hasta que el formulario es enviado (es decir, el `setTimeout` no sirve realmente para nada, lo hemos incluido para poder ver el efecto de desactivar botones).

DESACTIVAR INPUTS, SELECTS O CHECKBOX

En ocasiones nos puede interesar desactivar elementos de un formulario hasta que el usuario no elija alguna opción previa requerida.

Podemos desactivar inputs de tipo text, inputs de tipo checkbox, o elementos de tipo select. Para ello debemos aplicar la sintaxis ya conocida:

```
nombreDelObjetoADeshabilitar.disabled = true;
```

Escribe el siguiente código donde vemos un ejemplo de cómo mantener desactivados ciertos elementos hasta que el usuario no escoja una opción que queremos sea escogida antes que las que mantenemos desactivadas. Comprueba los resultados.

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html><head><title>Ejemplo aprenderaprogramar.com</title><meta charset="utf-8">
<style type="text/css"> body, p {margin-left:30px; font-family: sans-serif;}
.estiloForm {background-color: #f3f3f3; border: solid 2px black; margin-left:20px; width: 330px; padding:10px; }
.estiloForm label {display: block; width: 120px; float: left; text-align:right; margin-bottom: 35px; padding-right: 20px;}
br {clear: left;} input[type="submit"], input[type="reset"] {margin:25px 5px 10px 5px;}
</style>
<script type="text/javascript">
window.onload = function () {
// Variable global por estar declarada sin hacer uso de var
elementosEnForm = document.forms['formularioContacto'].elements;
for (var i=0; i< elementosEnForm.length; i++) {
    if (elementosEnForm[i].type != 'radio') {elementosEnForm[i].disabled = true;}
    else {elementosEnForm[i].addEventListener('click', activarElementos);}
}
}

function activarElementos() { for (var i=0; i<elementosEnForm.length; i++) {elementosEnForm[i].disabled=false;}}
</script>
</head>
<body><h2>Cursos aprenderaprogramar.com</h2><h3>Ejemplos JavaScript</h3>
    <div class="estiloForm">
        <form name = "formularioContacto" method="get" action="http://aprenderaprogramar.com">
        <p>Debe elegir tratamiento para activar el formulario</p>
        <label>Tratamiento</label>
        <input type="radio" name="tratamiento" id="tratarSr" value="Sr." />Sr.
        <input type="radio" name="tratamiento" id="tratarSra" value="Sra." />Sra.<br/>
        <label>Nombre</label><input id="nombre" name="nombre" type="text"/><br/>
        <label>Apellidos</label><input id="apellidos" name="apellidos" type="text"/><br/>
        <label>Dirección</label><input id="direccion1" name="direccion1" type="text"/><br/>
        <label>Ciudad</label><select id="ciudad" name="ciudad">
        <option value="">Elija opción</option><option value="Mexico">México D.F. (MX)</option>
        <option value="Madrid">Madrid (ES)</option><option value="Santiago">Santiago (CL)</option>
        </select><br/>
        <label>Preferencias</label><input name="Libros" type="checkbox" />Libros
        <input name="Películas" type="checkbox" />Películas
        <input class="botonFormulario" type="submit" value="Enviar"/>
        <input class="botonFormulario" type="reset" value="Cancelar"/>
    </form>
    </div>
</body></html>

```

El resultado esperado es que el usuario no pueda escribir en los input de tipo text, ni seleccionar opciones del select o de tipo checkbox, ni pulsar botón de envío o reset. ¿Por qué? Porque inicialmente establecemos la propiedad disabled de todos los elementos del formulario (excepto los input de tipo radio) a true. Esta situación no cambiará hasta que el usuario elija una opción de las presentadas como input de tipo radio. Cuando detectamos el evento click sobre un objeto de tipo radio (el usuario ha elegido una de las dos opciones que le pedimos que elija) hacemos que la propiedad disabled de todos los elementos del formulario quede establecida en true, con lo cual ya pueden introducirse textos, elegir opciones o pulsar los botones de envío o cancelación.

DESACTIVAR ELEMENTOS OPTION DE UN SELECT

En ocasiones nos puede interesar desactivar no un select de forma completa sino determinados elementos del select. Por ejemplo, supongamos que se trata de una aplicación web para un hotel donde existen tres habitaciones suite: "Mediterráneo", "Atlántico" y "Pacífico". Supongamos que la habitación Atlántico está en obras y queremos mantener la opción visible pero deshabilitada, de forma que el usuario no pueda elegir esa opción.

Escribe este código y comprueba cómo la opción "Atlántico" no está disponible para ser seleccionada.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html><head><title>Ejemplo aprenderaprogramar.com</title><meta charset="utf-8">
<style type="text/css"> body, p {margin-left:30px; font-family: sans-serif;}
.estiloForm {background-color: #f3f3f3; border: solid 2px black; margin-left:20px; width: 330px; padding:10px; }
.estiloForm label {display: block; width: 120px; float: left; text-align:right; margin-bottom: 35px; padding-right: 20px;}
br {clear: left;} input[type="submit"], input[type="reset"] {margin:25px 5px 10px 5px;}
</style>
<script type="text/javascript">
window.onload = function () {
var habitacionesSelect = document.forms['formularioContacto'].elements['habitacion'];
// habitacionesSelect.options[2].disabled=true; ¿Qué ocurre si cambia el índice y no es 2?
for (var i=0; i<habitacionesSelect.length; i++) {
    if (habitacionesSelect[i].value =='atlantico') {habitacionesSelect[i].disabled=true;}
}
}
</script>
</head>
<body><h2>Cursos aprenderaprogramar.com</h2><h3>Ejemplos JavaScript</h3>
    <div class="estiloForm">
        <form name = "formularioContacto" method="get" action="http://aprenderaprogramar.com">
            <p>La suite Atlántico está en obras</p>
            <label>Habitación suite:</label><select id="habitacion" name="habitacion">
            <option value="">Elija opción</option>
            <option value="mediterraneo">Mediterráneo</option>
            <option value="atlantico">Atlántico</option>
            <option value="pacifico">Pacífico</option>
            </select><br/>
            <input class="botonFormulario" type="submit" value="Enviar"/>
            <input class="botonFormulario" type="reset" value="Cancelar"/>
        </form>
    </div>
</body></html>
```

Fíjate que podemos desactivar la opción directamente usando la sintaxis `habitacionesSelect.options[2].disabled=true`; pero que sin embargo lo hemos hecho con un bucle `for`. Usar un bucle `for` tiene un coste (recorrer los elementos) pero nos proporciona más seguridad al basar la selección en el `value` 'atlantico' en lugar de en un índice numérico que podría cambiar. No podemos decir que una opción sea buena y otra mala: simplemente son alternativas que tenemos para llegar a un mismo objetivo y como programadores deberemos elegir la que consideremos más adecuada.

DESACTIVAR OTROS ELEMENTOS COMO BUTTON ó LINK

Hay más elementos a los que se le puede aplicar la propiedad disabled, por ejemplo los elementos button. La sintaxis a utilizar es la misma que hemos visto anteriormente.

También podemos desactivar elementos link como los que se emplean para referenciar un archivo CSS externo o elementos style como los que se emplean para incluir CSS interno. Usando esta posibilidad podemos activar o desactivar CSS de una página web de forma bastante sencilla. Escribe este código y comprueba cómo se activa o desactiva CSS pulsando en el botón:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html><head><title>Ejemplo aprenderaprogramar.com</title><meta charset="utf-8">
<style type="text/css"> .miEstilo{background-color: yellow; font-size:3em;} input{margin:10px;}
</style>
<script type="text/javascript">
function cambiarCSS() {
var nodoStyle = document.getElementsByTagName('style');
nodoStyle[0].disabled = !nodoStyle[0].disabled;
}
</script>
</head>
<body><h2>Cursos aprenderaprogramar.com</h2><h3>Ejemplos JavaScript</h3>
<input type="button" value="Activar/Desactivar Css" onclick="cambiarCSS()"/><br>
<div class="miEstilo">Esto es un Ejemplo</div>
</body></html>
```

El resultado esperado es que el texto "Esto es un ejemplo" se muestre con fondo amarillo y texto grande al cargar la página. Cuando pulsemos en el botón "Activar/Desactivar CSS" el texto dejará de tener el fondo amarillo y se mostrará con tamaño de letra normal al quedar el nodo style con la propiedad disabled establecida a true. Si se vuelve a pulsar el botón, la propiedad disabled toma el valor opuesto al que tuviera anteriormente. Esto lo conseguimos con la sintaxis de la línea `nodoStyle[0].disabled = !nodoStyle[0].disabled;`

RESUMEN

La propiedad disabled de los objetos HTML Element del DOM nos permite deshabilitar elementos de formularios pero también dejar inhabilitados otros nodos del DOM, con lo que podemos conseguir efectos interesantes.

Hay ciertos tipos de nodos, por ejemplo un elemento div, que no admiten la propiedad disabled.

EJERCICIO

Crear el código HTML y JavaScript que cumpla con las siguientes funciones:

a) Deberá presentarse un formulario con 2 elementos select. El primer elemento select permitirá elegir país y podrá elegirse entre México, España, Perú y Colombia.

- b) El segundo elemento select permitirá elegir ciudad y podrá elegirse entre México D.F., Guadalajara, Madrid, Barcelona, Lima, Trujillo, Bogotá y Cali.
- c) Si el usuario selecciona ciudad sin haber elegido país, deberá mostrarse como opción elegida de país el país correspondiente a la ciudad. Por ejemplo, si el usuario selecciona Bogotá deberá aparecer como país seleccionado Colombia.
- d) Si el usuario selecciona un país en primer lugar, deberán deshabilitarse todas las opciones que no correspondan a ciudades de ese país. Por ejemplo si el usuario elige España, sólo podrá elegir entre Madrid y Barcelona, debiendo estar México D.F., Guadalajara, Lima, Trujillo, Bogotá y Cali deshabilitadas. Si el usuario cambia el país, deberán cambiar las ciudades cuya elección es posible.
- e) Ampliación opcional para el ejercicio: introduce medidas de seguridad adicionales para que no puedan existir incoherencias entre el país seleccionado y la ciudad. Por ejemplo, imagina que el usuario elige en primer lugar como ciudad México D.F. y luego elige como país Perú. Introduce medidas de seguridad que impidan que esto suceda.

Para comprobar si tus respuestas y código son correctos puedes consultar en los foros aprenderaprogramar.com.

Próxima entrega: CU01184E

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