

Trucos para mejorar tu página Web. JavaScript.

1. INTRODUCCIÓN AL JAVASCRIPT

En este tema se estudiarán algunos de los efectos más usados comúnmente para el diseño de las páginas Web. Algunos de estos efectos no son posibles con HTML por lo que se ha tenido que recurrir al lenguaje de programación JavaScript que por otra parte es compatible con el HTML.

El lenguaje JavaScript surge como respuesta a la necesidad de aumentar el dinamismo de las páginas desarrolladas con HTML, permitiendo a los usuarios una mayor interactividad con las mismas.

Aunque los programas CGI posibilitan la realización de aplicaciones Web, el uso del lenguaje JavaScript descarga la realización de muchas de las funciones relacionadas con la gestión de las interfaces de usuario, ya que el código en JavaScript se ejecuta en el ordenador del que accede a la página y no en el servidor. Esto conlleva a que el servidor no se sobrecarge ejecutando infinidad de programas de los usuarios que accedan.

JavaScript no es un lenguaje de propósito general, sino que se incrusta dentro del documento HTML contenido entre las etiquetas `<SCRIPT>` y `</SCRIPT>` colocadas entre las etiquetas `</TITLE>` y `</HEAD>`. Existen en el mercado otros lenguajes que ofrecen las mismas posibilidades (como VBScript) pero JavaScript es en la actualidad el más extendido pasando a ser un estándar.

La sintaxis para incluir un programa en JavaScript en un documento HTML es la siguiente:

```
<SCRIPT LANGUAGE="JavaScript">
```

Instrucciones JavaScript </SCRIPT>

En este capítulo sólo estudiaremos el conjunto de instrucciones JavaScript necesario para llevar a cabo un determinado efecto.

Bien, una vez sabemos qué es JavaScript pasemos a estudiar los demás apartados que componen el tema.

2. CARGAR VARIOS FRAMES CON UN ÚNICO ENLACE

En el tema cuatro estudió que puede cargar mediante un enlace el contenido de una determinada página en uno de los marcos que componen la página. No obstante, lo que no estudió es cómo puede cargar varias páginas distintas en distintos frames con tan sólo un enlace.

Esto no era posible en el tema cuatro porque su realización es mediante JavaScript.

Para cargar varios frames mediante un enlace en primer lugar debe incluir en la cabecera del documento la declaración de la función que llevará a cabo esta acción.

```
<SCRIPT LANGUAGE="JavaScript">  
  
function fnUpdate(){  
    top.nombre del frame1.location="documento a cargar1.html";  
    top.nombre del frame2.location="documento a cargar2.html";  
    ...  
    top.nombre del frameN.location="documento a cargarN.html";  
}  
  
</SCRIPT>
```

Así en *nombre del frameN* deberá indicar el nombre del marco donde se va a cargar el documento *documento a cargarN*.

Una vez ha declarado la función que llevará a cabo la carga, en el enlace que llevará a cabo esta función debe indicar la llamada a esta función y el evento que producirá la ejecución.

```
<A HREF="documento a cargar 1.html" TARGET="nombre del frame1"  
onClick= "fnUpdate();">  
  
    Texto del enlace  
</A>
```

Aunque en el enlace se referencia un solo documento, al incluir el atributo *onClick="fnUpdate();"* se está indicando que al hacer clic en el enlace se ejecute la función *fnUpdate* que cargará todos los frames que se han especificado en la subrutina.

3. PERSONALIZAR LA BARRA DE ESTADO

Como ya estudió en el tema de enlaces, al posicionar el puntero del ratón sobre un enlace en la barra de estado se le mostrará la dirección a la que le lleva el enlace.

No obstante, mediante una rutina JavaScript puede personalizar el texto que muestre la barra de estado cuando posicione el cursor del ratón sobre el enlace.

Para ello, deberá incluir la siguiente línea de código dentro de la etiqueta del enlace.

onMouseOver="window.status='Escriba aquí el texto que quiere que aparezca en la barra de estado';return true;"

Así, por ejemplo, en la siguiente figura se muestra el texto *Enlace a la página principal de Antakira Grupo de Empresas* al posicionar el puntero del ratón sobre el enlace.

```
<A HREF="http://www.antakira.com" onMouseOver="window.status='Enlace a la página principal de Antakira Grupo de Empresas' ;return true;" >
  Antakira Grupo de Empresas
</A>
```

El resultado en el navegador es el siguiente:



El problema es que este texto se quedará fijo en la barra de estado y lo que interesa es que cuando el puntero del ratón deje de estar posicionado en el enlace cambie el texto, para ello puede utilizar la función *onMouseOut* de la misma forma, tal y como se muestra a continuación.

onMouseOut="window.status='Escriba aquí el texto que quiere que aparezca en la barra de estado';return true;"

4. LIMITAR EL ACCESO A UNA PÁGINA POR MEDIO DE UN PASSWORD

En algunas páginas Web puede que quiera limitar el acceso de los usuarios a una determinada página. Por ejemplo, esta situación puede darse en una universidad donde se informe de la calificación de los alumnos por internet y para acceder a una página el alumno tenga que introducir su DNI.

La forma más profesional de hacerlo es por medio de un CGI, es decir, un programa que se coloca en el servidor y que cada vez que intente acceder a la página se ejecutará actuando de filtro.

Pero los servidores se muestran reacios a que los usuarios instalen programas de este tipo en su ordenador, por lo que usaremos JavaScript.

4 CURSO DE HTML

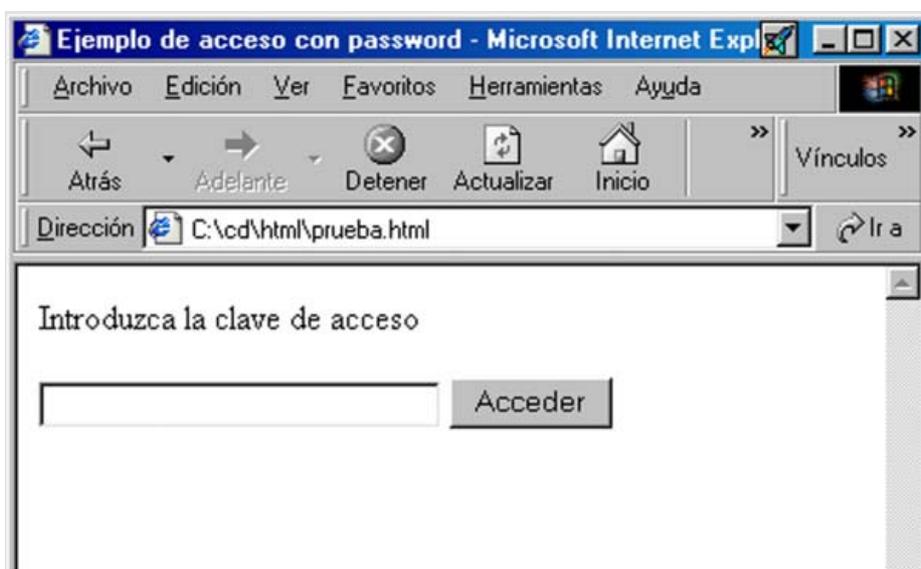
```
<HTML>
<HEAD>
  <TITLE>Ejemplo de acceso con password</TITLE>
  <SCRIPT LANGUAGE="JavaScript">
    function Saltar(pal) {
      window.location=pal+".html"
    }
  </SCRIPT>
</HEAD>
<BODY>
  <FORM>
  Introduzca la clave de acceso
  <P>
  <INPUT TYPE="password" NAME="claveacceso" SIZE=25 VALUE="">
  <INPUT TYPE = "button" VALUE = "Acceder" onClick = "Saltar
(this.form.claveacceso.value) ">
  </FORM>
</BODY>
</HTML>
```

Este código funciona de la siguiente forma: una vez escrita la palabra clave y hecho clic en el botón *Acceder* el programa llamará a una página con nombre *claveacceso.html* siendo la *clave de acceso* el texto a introducir en la caja de texto tipo *password*. Es decir se debe dar a la página el nombre que se establezca como clave de acceso. Por ejemplo, si la clave establecida es *prueba* la página debe denominarse *prueba.html*.

Tal y como está escrito el código, la extensión de la página debe ser *.html*, no obstante, si la extensión de las páginas es *.htm* deberá cambiar la extensión en la siguiente línea de código.

window.location=pal+".htm"

El resultado de las instrucciones HTML es el siguiente:



5. ABRIR UNA VENTANA DIMENSIONABLE

También es muy frecuente que al acceder a una página se abra una pequeña ventana de dimensiones predeterminadas y que ofrezca por ejemplo un resumen de lo que se va a encontrar en la página.

No hay que confundir que esta ventana sea tal elemento, ya que en realidad es una página Web de dimensiones fijadas de antemano y sin las barras de *scroll* ni de herramientas (aunque también pueden dejarse).

La creación de esta página Web, que por otra parte puede contener todo lo relativo a las páginas Web, es posible insertando la siguiente rutina en JavaScript.

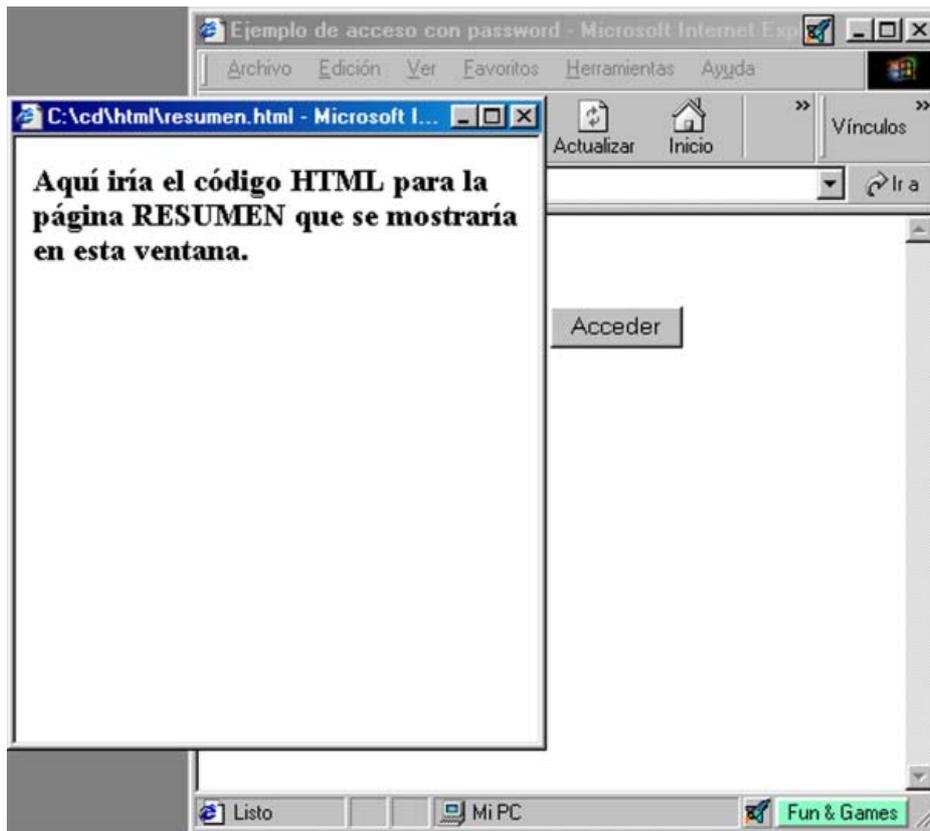
```
</TITLE>
<SCRIPT LANGUAGE="JavaScript">
    open('nombre de la ventana.html', 'Sizewindow', 'width=300, height=350,
scrollbars=no, toolbar=no')
</SCRIPT>
</HEAD>
```

En el parámetro *nombre de la ventana.html* deberá especificar la página Web que desea incluir en la ventana. Así mismo, compruebe cómo puede especificar la anchura y altura de la misma. En el caso de que la ventana no incluya barras de herramientas ni barras de desplazamiento asigne a los parámetros *scrollbars* y *toolbar* el valor *no*.

En la siguiente figura se muestra un ejemplo donde se ha incluido la ventana *Resumen* en la carga de la página principal de una página Web. Como puede comprobar, esta ventana no tiene barras de herramientas aunque sí barras de desplazamiento.

```
</TITLE>
<SCRIPT LANGUAGE="JavaScript">
    open('resumen.html', 'Sizewindow', 'width=300, height=350,
scrollbars=no, toolbar=si')
</SCRIPT>
</HEAD>
```

El resultado del código anterior es el siguiente:



6. REDIRECCIONAMIENTO AUTOMÁTICO DE LAS PÁGINAS

Como se ha mencionado ya en repetidas ocasiones Internet es algo dinámico, es decir algo que varía con el tiempo. Por ello, es frecuente cambiar las páginas de sitio, es decir de URL.

Pero si una página se mueve de sitio, de algún modo debe informar a los usuarios que no sepan el cambio. Esto es posible redireccionando la página antigua a la nueva dirección tras X segundos una vez el usuario haya entrado en ella.

Para ello deberá incluir la etiqueta META en el encabezado de la página y tras X segundos que deberá especificar, se redireccionará a la página que tenga en una de sus propiedades. La sintaxis de esta etiqueta es la siguiente:

```
<META HTTP-EQUIV = "Refresh" CONTENT = " x ; URL = URL_de_la_siguiete_página " >
```

En el parámetro x deberá especificar el número de segundos tras los cuales se cambiará la dirección a la nueva URL que especifique en el atributo URL.

Por ejemplo, en la siguiente figura se cambia de dirección a *antakira.com* tras 20 segundos que comienzan a contar al cargar la página.

```
<HEAD>
  <TITLE> Redireccionamiento de páginas </TITLE>
  <META HTTP-EQUIV = "Refresh" CONTENT = " 20 ;
  URL =http://www.antakira.com " >
</HEAD>
<BODY>
...

```

7. CAMBIAR LA IMAGEN DE UN ENLACE

En la mayor parte de las páginas actuales se suele incluir un efecto que al pasar el puntero del ratón sobre un determinado botón o una determinada opción ésta cambie de forma, color, etc.

Para llevar a cabo este cambio de aspecto es necesario primero tener dos archivos de imágenes, uno que contenga el aspecto que presentará normalmente y otro que contenga el aspecto que presentará cuando se pase el puntero del ratón por encima.

Este efecto también se lleva a cabo a través de JavaScript. El código que lo implementa se muestra a continuación:

```
<A href ="doc1.html" >
<img src = "boton1.gif" width = "75" height ="50"
onMouseOver = " src ='boton2.gif' " onMouseOut = " src ='boton1.gif' " >
</A>

```

Como puede comprobar, está formado por un enlace a través de una imagen, donde a la etiqueta de esta última se le han incluido las funciones *OnMouseOver* y *OnMouseOut*. La primera función cambia de la imagen *boton1.gif* a la imagen *boton2.gif* cuando se pasa el puntero del ratón por encima del enlace y la segunda cuando el puntero del ratón no se encuentre situado encima del enlace, vuelva a mostrar el contenido del archivo *boton1.gif*.

EJERCICIOS PROPUESTOS.

1. *Vuelva a abrir el archivo 'Formulario de alta.html' de su disco de trabajo.*

2. *Realice un enlace al final del formulario que le lleve a la página 'Marcos.html', con el texto 'Enlace a la página Marcos'. El enlace deberá mostrar en la barra de estado el texto 'Enlace a la página Marcos' cuando se posicione el cursor del ratón encima de él y el texto 'Formulario de alta' cuando el cursor del ratón se posicione en cualquier otro lugar del formulario.*

3. *Sustituya el enlace anterior por la imagen 'boton2-n.jpg' que se encuentra en el directorio 'D:\datos'. Así mismo, cuando el cursor del ratón pase por encima del botón éste debe cambiar de aspecto al indicado en el archivo 'boton2-a.jpg'. El botón debe tener un tamaño porcentual del 5% y no debe tener borde.*

4. *Abra una ventana dimensionable de 300 píxeles de anchura y 350 de altura sin barras de herramientas y sin barras de deslizamiento. Carge en la ventana dimensionable la página 'Marcos.html'*

5.- *Cierre el documento y guarde los cambios.*

