

Menus desplegables

Menú desplegable: concepto

Un menú desplegable es el que se muestra al pasar el ratón por encima de un elemento. Suelen usarse en sitios con muchas páginas distribuidas en secciones y subsecciones, ya que esta es una manera rápida de acceder desde una página a cualquier otra del sitio, sin que el menú ocupe mucho espacio.

Normalmente se ponen en un menú visible las secciones principales del sitio. Al pasar el ratón por encima de cada sección aparecen los submenús que contienen las subsecciones. Los submenús se ocultan al salir el ratón de encima de ellos.

En este ejercicio veremos cómo hacer una barra de navegación con menús desplegables que se muestran u ocultan al entrar o salir el ratón en ellos.

La página una vez acabada debe quedar como en el siguiente enlace:



Planteamiento

La forma más sencilla de hacer un menú desplegable es crear primero el menú en la página de forma visible, y luego mediante javascript utilizar la propiedad CSS *display* para ocultar o mostrar el menú.

La página HTML

Partimos de una página en HTML tal como se muestra a continuación:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
<title>desplegable1.html</title>
<style type="text/css">
* { padding: 0; margin: auto; text-align: center; }
#cabecera { background-color: #fff0f0; }
h1 { font: bold 1.5em arial; padding: 0.5em; }
#navegador { background-color: #663366; padding: 0.5em; }
#navegador li { margin: 0px 5px; padding: 0.1em 1em 0.5em 1em;
background-color: #9933cc; width: 12%;float: left;
list-style-type: none; }
#navegador li:hover { background-color: #990033; }
#navegador li a:link, #navegador li a:visited {
```

```

        color: #feffe4; text-decoration: none; }
#navegador li a:hover, #navegador li a:active {
        color:#ffd7a9 ; text-decoration: none; }
.borrar { clear: both; }
#principal h2 { font: bold 1.5em arial; padding: 0.5em }
#principal p { font: normal 0.9em arial; text-align: justify;
        text-indent: 3em; padding: 0.5em 2em; }
</style>
</head>
<body>
<div id="cabecera">
    <h1>Mi p&aacute;gina</h1>
</div>
<div id="navegador">
    <ul>
        <li id="seccion1"><a href="#">Secci&oacute;n Uno</a></li>
        <li id="seccion2"><a href="#">Secci&oacute;n Dos</a></li>
        <li id="seccion3"><a href="#">Secci&oacute;n Tres</a></li>
        <li id="seccion4"><a href="#">Secci&oacute;n Cuatro</a></li>
    </ul>
    <div class="borrar"></div>
</div>
<div id="principal">
    <h2>Cuerpo principal de la p&aacute;gina</h2>
    <p>Lorem ipsum ad his scripta blandit partiendo, eum fastidii accumsan
euripidis
    in, eum liber hendrerit an. Qui ut wisi vocibus suscipiantur, quo dicit
ridens
    inciderint id. Quo mundi lobortis reformidans eu, legimus senserit definiebas
an eos.
    Eu sit tincidunt incorrupte definitionem, vis mutat affert percipit cu,
eirmod
    consectetur signiferumque eu per. In usu latine equidem dolores. Quo no
falli
    viris intellegam, ut fugit veritus placerat per.</p>

    <p>Ius id vidit volumus mandamus, vide veritus democritum te nec, ei eos
debet
    libris consulatu. No mei ferri graeco dicunt, ad cum veri accommodare.
    Sed at malis omnesque delicata, usu et iusto zzril meliore. Dicunt maiorum
eloquentiam cum cu, sit summo dolor essent te. Ne quodsi nusquam legendos
has,
    ea dicit voluptua eloquentiam pro, ad sit quas qualisque. Eos vocibus
deserunt quaestio ei.</p>
</div>
</body>
</html>

```

Modificación del código HTML

Nos centramos en el código de la barra de navegación de la página de ejemplo dada, este es el siguiente:

```

<div id="navegador">
    <ul>
        <li id="seccion1"><a href="#">Sección Uno</a></li>
        <li id="seccion2"><a href="#">Sección Dos</a></li>
        <li id="seccion3"><a href="#">Sección Tres</a></li>
        <li id="seccion4"><a href="#">Sección Cuatro</a></li>
    </ul>
    <div class="borrar"></div>
</div>

```

Vamos a centrarnos en el primer elemento de la lista. Lo ampliamos, poniéndole un "div" con subsecciones. El código de la barra de navegación quedará de la siguiente manera: (en color azul destacamos el código nuevo)

```
<div id="navegador">
  <ul>
    <li id="seccion1"><a href="#">Sección Uno</a>
      <div id="subseccion1">
        <p><a href="#"> Subsección uno uno</a></p>
        <p><a href="#"> Subsección uno dos</a></p>
        <p><a href="#"> Subsección uno tres</a></p>
        <p><a href="#"> Subsección uno cuatro</a></p>
      </div>
    </li>
    <li id="seccion2"><a href="#">Sección Dos</a></li>
    <li id="seccion3"><a href="#">Sección Tres</a></li>
    <li id="seccion4"><a href="#">Sección Cuatro</a></li>
  </ul>
  <div class="borrar"></div>
</div>
```

No es precisamente como queremos que quede la página, sin embargo no hay nada que no se pueda arreglar mediante el código CSS. La importancia de este paso está en crear un elemento hijo en el que ponemos los enlaces de las subsecciones.

Cada elemento de la lista, al igual que los "divs" de las subsecciones llevan el correspondiente atributo *id* con el que podremos tanto manejar el código CSS como el Javascript.

Modificación del código CSS

Para poder ver el elemento de la subsección anterior de manera correcta, modificaremos el código CSS de la página. En primer lugar modificamos el código CSS de manera que los elementos de la lista estén en posición relativa.

Para ello buscamos la regla CSS cuyo selector es `#navegador li` y le añadimos la siguiente propiedad (en verde el código nuevo):

```
#navegador li { margin: 0px 5px; padding: 0.1em 1em 0.5em 1em;
  background-color: #9933cc; width: 12%;float: left;
  list-style-type: none; position: relative; }
```

Ahora el div "subseccion1" lo marcamos con posición absoluta. Esto hará que el punto 0,0 de coordenadas para este div tenga su referencia en el elemento padre, es decir el elemento de la lista "seccion1". añadimos por tanto la siguiente regla CSS:

```
#subseccion1 { position: absolute; }
```

Para posicionar el elemento debajo de su elemento padre, en la misma regla anterior añadiremos las siguientes propiedades:

```
top: 100%; left: 0px;
```

Con esto desde la parte superior (top) del elemento padre (seccion1), buscamos la parte inferior del mismo, (el 100%), y colocamos ahí el elemento con los submenús (subseccion1). La propiedad `left: 0px;` alinea por la derecha el submenú con su elemento padre.

De la misma manera si quisieramos colocarlo en la parte derecha escribiríamos la propiedad `left: 100%; top: 0px,` y la propiedad `right: 100%; top 0px;` nos colocaría el elemento en la parte izquierda.

Por último en la misma regla colocamos otra serie de propiedades para dar un estilo más personal al submenú, como pueden ser cambiar el color de fondo, el tipo de letra, etc.

La regla CSS anterior quedará así:

```
#subseccion1 { position: absolute; top: 100%; left: 0px;
    background-color: #41338b; font:normal 0.8em arial;
    padding: 0.2em 0.5em ; }
```

Una vez modificado el código CSS la página la veremos siempre el submenú tal como quedaría desplegado. El siguiente paso está en ocultarlo, y mostrarlo sólo cuando el ratón pase por encima. Para eso utilizaremos el código javascript.

Código Javascript

En primer lugar lo que debemos hacer es ocultar el submenú. para ello en la regla anterior añadiremos la propiedad `display: none;`

```
#subseccion1 { position: absolute; top: 100%; left: 0px;
    background-color: #41338b; font:normal 0.8em arial;
    padding: 0.2em 0.5em ; display: none; }
```

A continuación en el código HTML añadiremos al elemento de la lista (seccion1) los atributos de los eventos `onmouseover` y `onmouseout` que detectan cuando el ratón está o no encima del elemento. En ambos casos iremos a unas funciones javascript que hagan que el submenú sea o no visible.

el código HTML quedará así:

```
<div id="navegador">
<ul>
  <li id="seccion1" onmouseover="ver(1)" onmouseout="ocultar(1)">
    <a href="#">Sección Uno</a>
    <div id="subseccion1">
      <p><a href="#"> Subsección uno uno</a></p>
      <p><a href="#"> Subsección uno dos</a></p>
      <p><a href="#"> Subsección uno tres</a></p>
      <p><a href="#"> Subsección uno cuatro</a></p>
    </div></li>
  ....
</ul>
</div>
```

El parámetro que pasamos a la función indica el número de elemento de la lista, y es el mismo que lleva el "id" de subsección. De esta manera podremos luego añadir otras subsecciones a otros elementos de la lista utilizando las mismas funciones.

para que esto funcione debemos crear ahora el código javascript con las funciones que visualizan y ocultan el submenú. Este es el siguiente:

```
<script type="text/javascript">
function ver(n) {
    document.getElementById("subseccion"+n).style.display="block"
}
function ocultar(n) {
    document.getElementById("subseccion"+n).style.display="none"
}
</script>
```

El hecho de pasar el número de la subsección como parámetro nos permitirá poder añadir más subsecciones sin tener que crear nuevas funciones.

Añadir más subsecciones

Para añadir subsecciones al resto de elementos del menú desplegable seguiremos los mismos pasos que para el primer elemento, por lo tanto primero modificamos el código HTML, en el cual añadimos los menús y hacemos una llamada a las funciones que visualizan y ocultan los menús:

```
<div id="navegador">
<ul>
  <li id="seccion1" onmouseover="ver(1)" onmouseout="ocultar(1)">
    <a href="#">Sección Uno</a>
    <div id="subseccion1">
      <p><a href="#"> Subsección uno uno</a></p>
      <p><a href="#"> Subsección uno dos</a></p>
      <p><a href="#"> Subsección uno tres</a></p>
      <p><a href="#"> Subsección uno cuatro</a></p>
    </div>
  </li>
  <li id="seccion2" onmouseover="ver(2)" onmouseout="ocultar(2)">
    <a href="#">Sección Dos</a>
    <div id="subseccion2">
      <p><a href="#"> Subsección dos uno</a></p>
      <p><a href="#"> Subsección dos dos</a></p>
      <p><a href="#"> Subsección dos tres</a></p>
    </div>
  </li>
  <li id="seccion3" onmouseover="ver(3)" onmouseout="ocultar(3)">
    <a href="#">Sección Tres</a>
    <div id="subseccion3">
      <p><a href="#"> Subsección tres uno</a></p>
      <p><a href="#"> Subsección tres dos</a></p>
      <p><a href="#"> Subsección tres tres</a></p>
      <p><a href="#"> Subsección tres cuatro</a></p>
      <p><a href="#"> Subsección tres cinco</a></p>
    </div>
  </li>
  <li id="seccion4" onmouseover="ver(4)" onmouseout="ocultar(4)">
    <a href="#">Sección Cuatro</a>
    <div id="subseccion4">
      <p><a href="#"> Subsección cuatro uno</a></p>
      <p><a href="#"> Subsección cuatro dos</a></p>
      <p><a href="#"> Subsección cuatro tres</a></p>
    </div>
  </li>
</ul>
<div class="borrar"></div>
</div>
```

Como puedes ver el código es similar al primero para cada uno de los elementos de la lista del navegador.

Ahora, para que los submenús queden iguales, debemos aplicar al resto de subsecciones el mismo código CSS que hemos aplicado a la primera, para ello en la regla CSS con el selector `#subseccion1` le añadimos los selectores correspondientes al resto de subsecciones: la regla CSS quedará así:

```
#subseccion1, #subseccion2, #subseccion3, #subseccion4
{ position: absolute; top: 100%; left: 0px;
  background-color: #41338b; font:normal 0.8em arial;
  padding: 0.2em 0.5em ; display: none; }
```

Con esto ya lo tenemos todo para visualizar el menú, ya que en el script javascript que hemos escrito anteriormente, las funciones nos sirven para todos los elementos del menú.

Actividades:

1. Modificar el ejemplo anterior para que el menú desplegable se genere desde código javascript. En html solo tendremos el `<div id=navegador>`. Las secciones y subsecciones del menú se añadirán al DOM desde javascript a partir de un array bidimensional que contendrá los literales de cada menú y submenu.

Quienes Somos	Productos	Pedidos	Facturación	Contacto
	Tipo A	Nuevo	Nueva	
	Tipo B	Consulta	Consulta	
	Tipo C	Totales	Totales	