
Práctica 3

CSS: Cascading Style Sheets y JavaScript

1. Menu

- Uno de los aspectos más usados en las web son los menús así como el desarrollo de distintos apartados en la página.
- El desarrollo de menús esta estrechamente relacionados con los elementos link y a (ancla).
 - ❖ Los elementos link describen relaciones a través de varios documentos.
 - ❖ Las anclas (también conocidas como elementos a) permiten enlazar con otro documento o con una sección concreta del documento actual.
- El desarrollo de menús también estan estrechamente relacionados con listas de viñetas, pues se suelen utilizar con esta estructuración.

1. Menu

- El uso de listas para crear menús es ideal por varias razones:
 - ❖ Todo el HTML se encuentra en un único elemento de lista (ul, por ejemplo), lo cual significa que podéis utilizar la cascada del CSS para aplicar un estilo diferente a cada uno de ellos y que es muy fácil llegar hasta los elementos en JavaScript bajando desde el nivel superior.
 - ❖ Las listas se pueden anidar, lo cual significa que podéis crear fácilmente una navegación con varios niveles de jerarquía.
 - ❖ Incluso si no se aplica ningún estilo a la lista, la representación del HTML por parte del navegador es lógica y para un visitante es muy fácil ver que estos enlaces van juntos y que forman una única unidad lógica.

1. Menu

- Hay varios tipos de menús cuando vais trabajando en diferentes sitios web.
 - ❖ Navegación por la página: por ejemplo, un contenido para una única página con enlaces que conducen a diferentes secciones de ésta.
 - ❖ Navegación por el sitio: una barra de navegación para todo el sitio web (o una subsección de éste) con enlaces que conducen a páginas diferentes del mismo sitio.
 - ❖ Navegación contextual según el contenido: una lista de enlaces que conducen a páginas estrechamente relacionadas con el tema de la página actual, tanto del mismo sitio como de otros.
 - ❖ Mapas del sitio: listas largas de enlaces que conducen a todas las páginas diferentes de un sitio web; los enlaces se agrupan en sublistas relacionadas para que sea más fácil entenderlo todo.
 - ❖ Paginación: enlaces que conducen a otras páginas que constituyen otras secciones o partes de todo un conjunto junto con la página actual, como por ejemplo parte 1, parte 2 y parte 3 de un artículo.

1. Menu

- Hay varios tipos de menús cuando vais trabajando en diferentes sitios web.
 - ❖ Navegación por la página: por ejemplo, un contenido para una única página con enlaces que conducen a diferentes secciones de ésta.
 - ❖ Navegación por el sitio: una barra de navegación para todo el sitio web (o una subsección de éste) con enlaces que conducen a páginas diferentes del mismo sitio.
 - ❖ Navegación contextual según el contenido: una lista de enlaces que conducen a páginas estrechamente relacionadas con el tema de la página actual, tanto del mismo sitio como de otros.
 - ❖ Mapas del sitio: listas largas de enlaces que conducen a todas las páginas diferentes de un sitio web; los enlaces se agrupan en sublistas relacionadas para que sea más fácil entenderlo todo.
 - ❖ Paginación: enlaces que conducen a otras páginas que constituyen otras secciones o partes de todo un conjunto junto con la página actual, como por ejemplo parte 1, parte 2 y parte 3 de un artículo.

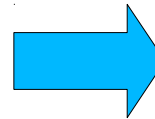
1. Menu

➤ Tipos de menú según el tipo de desarrollo

- ❖ Menú Vertical
- ❖ Menú Horizontal
- ❖ Menú con imagenes
- ❖ Menú dropdown

➤ Código inicial:

```
<ul class="navbar">
  <li><a href="indice.html">Página principal</a></li>
  <li><a href="meditaciones.html">Meditaciones</a></li>
  <li><a href="ciudad.html">Mi ciudad</a></li>
  <li><a href="enlaces.html">Enlaces</a></li>
</ul>
```



1. Menu

➤ Menú vertical : Los sitios web correctamente diseñados emplean las listas de elementos para crear todos sus menús de navegación.

1. Definir la anchura del menú:
2. Eliminar las viñetas automáticas y todos los márgenes y espaciados aplicados por defecto
3. Añadir un borde al menú de navegación y establecer el color de fondo y los bordes de cada elemento del menú
4. Aplicar estilos a los enlaces



```
ul.navbar {  
    border: 1px solid #7C7C7C;  
    border-bottom: none;  
    list-style: none;           //Paso 2  
    margin: 0;                 //Paso 2  
    padding: 0;                //Paso 2  
    width: 180px;              //Paso 1  
}  
ul.navbar li {                //Paso 3  
    background: #F4F4F4;  
    border-bottom: 1px solid #7C7C7C;  
    border-top: 1px solid #FFF;  
}  
ul.navbar li a {              //Paso 2  
    color: #333;  
    display: block;  
    padding: .2em 0 .2em .5em;  
    text-decoration: none;  
}
```

1. Menu

- Ejercicio 1: Usando el menú anterior y con el código XHTML proporcionado, aplicar las reglas CSS para obtener la siguiente página



- Añadir una barra de navegación
 - ❖ Mover la lista a la izquierda
 - ✓ `position: absolute`
 - ✓ `top: 2em;`
 - ✓ `left: 1em;`
 - ✓ `width: 9em`
 - ❖ Colocar el resto del texto hacia la derecha: Body
 - ✓ `padding-left: 11em`

6. Formato para Listas

➤ Menú horizontal :

1. Aplicar los estilos CSS básicos para establecer el estilo del menú
2. Establecer la anchura de los elementos del menú.
3. Establecer los bordes de los elementos que forman el menú:
4. Se elimina el borde derecho del último elemento de la lista

Elemento 1	Elemento 2	Elemento 3	Elemento 4	Elemento 5
------------	------------	------------	------------	------------

```
<ul>
  <li><a href="#">Elemento 1</a></li>
  <li><a href="#">Elemento 2</a></li>
  <li><a href="#">Elemento 3</a></li>
  <li><a href="#">Elemento 4</a></li>
  <li><a href="#" style="border-right: none">Elemento 5</a></li>
</ul>
```

Importante: Probar
a quitarlo ¿Qué sucede?

```
ul.navbar {          //Paso 1
  background: #F4F4F4;
  border: 1px solid #7C7C7C;
  list-style: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
}
ul.navbar li {       //Paso 2
  float: left;
  width: 20%;
}
ul.navbar li a {     //Paso 3
  border-left: 1px solid #FFF;
  border-right: 1px solid #7C7C7C;
  color: #333;
  display: block;
  padding: .3em;
  text-decoration: none;
}
```

1. Menu

- Ejercicio 3: Los menu previos se pueden mejorar añadiendo estilos más conseguidos.

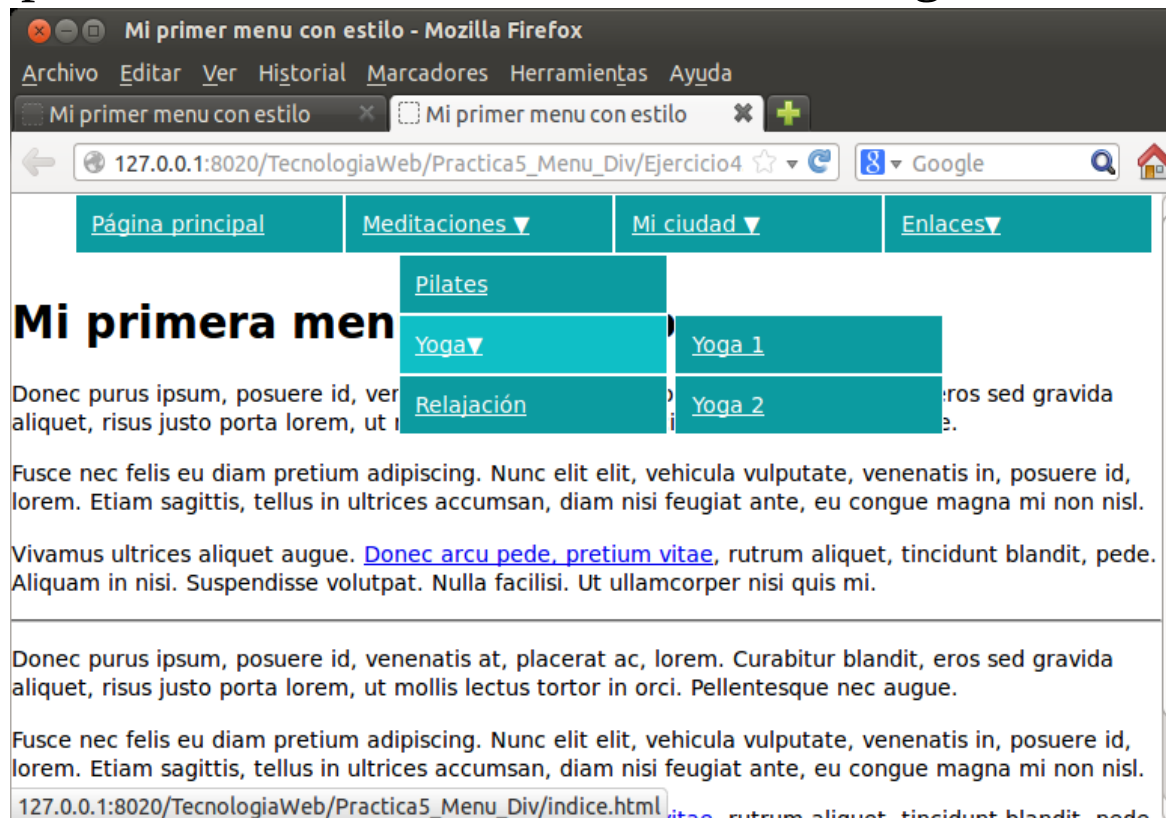


- Dar estilos a los enlaces

- ❖ `ul.navbar {`
 - ✓ `list-style-type: none;`
 - ✓ `padding: 0;`
 - ✓ `margin: 0;`
 - ✓ `position: absolute;`
 - ✓ `top: 2em;`
 - ✓ `left: 1em;`
 - ✓ `width: 9em }`
- ❖ `ul.navbar li {`
 - ✓ `background: white;`
 - ✓ `margin: 0.5em 0;`
 - ✓ `padding: 0.3em;`
 - ✓ `border-right: 1em solid black }`
- ❖ `ul.navbar a {`
 - ✓ `text-decoration: none }`
- ❖ `a:link {`
 - ✓ `color: blue }`
- ❖ `a:visited {`
 - ✓ `color: purple }`

1. Menu

- Menú dropdown o submenús
 - ❖ Se desplegaran cuando el usuario pase el cursor
 - ❖ Se estructuran con listas que contienen dentro listas (ul > li > ul).
 - ✓ Se puede realizar sobre texto o sobre imágenes



Obtenido y modificado de: <http://web.tursos.com/tutoriales/html-css/como-hacer-un-menu-desplegable-multinivel-usando-solo-css>

1. Menu

➤ Menu Dropdown

```
<!-- Menú de navegación del sitio -->
<ul class="navbar">
  <li><a href="indice.html">Página principal </a></li>
  <li>
    <a href="#">Meditaciones <span class="flecha">&#9660</span></a>
    <ul>
      <li><a href="indice.html" >Pilates</a></li>
      <li><a href="indice.html">Yoga<span class="flecha">&#9660</span></a>
        <ul>
          <li><a href="indice.html">Yoga 1</a></li>
          <li><a href="indice.html">Yoga 2</a></li>
        </ul>
      </li>
      <li><a href="indice.html">Relajación</a></li>
    </ul>
  </li>
  <li><a href="#">Mi ciudad <span class="flecha">&#9660</span></a>
    <ul>
      <li><a href="indice.html">Mérida</a></li>
      <li><a href="indice.html">Badajoz</a></li>
      <li><a href="indice.html">Cáceres</a></li>
    </ul>
  </li>
  <li><a href="">Enlaces<span class="flecha">&#9660</span></a>
    <ul>
      <li><a href="">Historia</a></li>
      <li><a href="">Mision</a></li>
    </ul>
  </li>
</ul>
```

1. Menu

➤ Menu Dropdown. Pasos:

- ❖ Establecer el formato para el body
 - ✓ Familia de fuente sans-serif;
 - ✓ Lista de estilos: ninguna
 - ✓ Decoración del texto: ninguno
 - ✓ Margen y relleno: 0
- ❖ ul.navbar
 - ✓ Lista de estilos ninguna
 - ✓ Margen:0
 - ✓ Alineación del texto: izquierda
 - ✓ Ancho: 800px
 - ✓ Relleno inferior: 50px
- ❖ ul.navbar li //Por cada viñeta
 - ✓ Estilo de lista lista: none;
 - ✓ Ancho: 200px;
- ❖ ul.navbar > li { // Al primer li
 - ✓ Flotante: izquierda;
- ❖

- ❖ ul.navbar li a { //Para los enlaces
 - ✓ Color de fondo: #0c9ba0;
 - ✓ Color: #FFF;
 - ✓ display: block; //tratamiento como bloque
 - ✓ Borde: 1px solid;
 - ✓ Relleno: 10px 12px;
- ❖ ul.navbar li a:hover { //pase el ratón
 - ✓ Color de fondo: #0fbfc6;
- ❖ ul.navbar li ul { //Que no se vean los submenu
 - ✓ display: none;
 - ✓ Posición: absoluta;
 - ✓ min-width: 140px;
- ❖ ul.navbar li:hover > ul { //Siguiendo descend.
 - ✓ display: block;
- ❖ ul.navbar li ul li { //Tratamiento para 3º submenu
 - ✓ Posición relativa
- ❖ ul.navbar li ul li ul { // Se situa
 - ✓ right: -205px; //Se mostraba 200px por eso 205
 - ✓ Top: 0;
- ❖ .nav li .flecha{
 - ✓ Tamaño del a fuente: 9px;
 - ✓ Relleno de izquierda: 6px;
 - ✓ display: none;
 - ✓ Alineación del texto:derecha;}

1. Menu

➤ Menu Dropdown. Pasos:

```
body {  
  font-family: sans-serif;  
  list-style: none;  
  text-decoration: none;  
  margin: 0;  
  padding: 0;}
```

```
ul.navbar {  
  list-style-type: none;  
  margin: 0 auto;  
  text-align: left;  
  width: 800px;  
  padding-bottom: 50px;}
```

```
ul.navbar li {  
  list-style-type: none;  
  width: 200px;}
```

```
ul.navbar > li {  
  float: left;}
```

```
ul.navbar li a {  
  background: #0c9ba0;  
  color: #FFF;  
  display: block;  
  border: 1px solid;  
  padding: 10px 12px;}
```

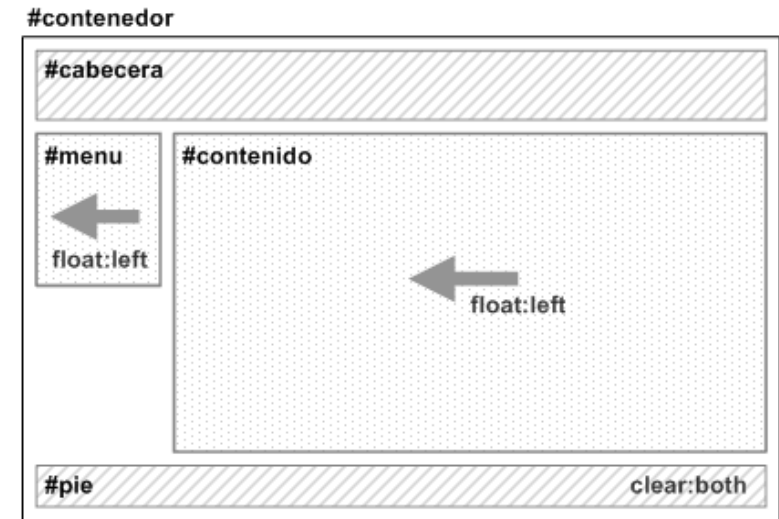
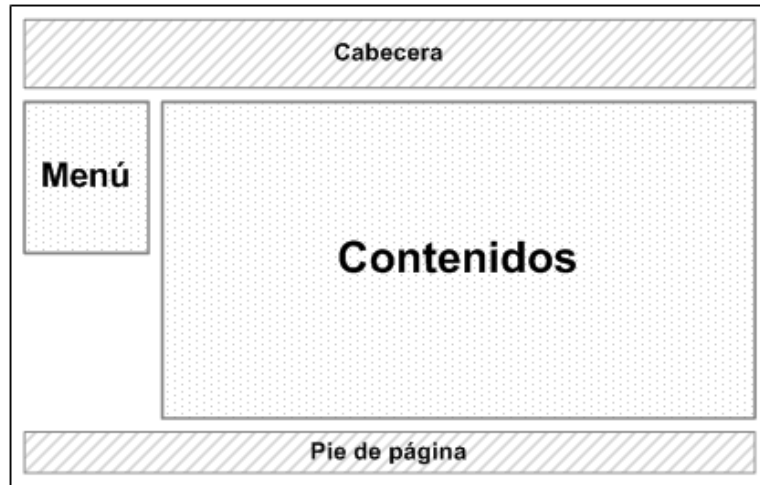
```
ul.navbar li a: hover {  
  background: #0fbfc6;  
}  
/*Paso 2. Quitar los submenus que no se vean */  
ul.navbar li ul {  
  display: none;  
  position: absolute;  
  min-width: 140px;  
}  
/*Paso 3 Cuando el cursor pase sobre cualquier li su  
descendiente ul se muestre(display: block). */  
ul.navbar li: hover > ul {  
  display: block;  
}  
/*3º SubMenu --> derecha de su respectivo ancestro  
li:*/  
ul.navbar li ul li {  
  position: relative;  
}  
ul.navbar li ul li ul {  
  right: -205px;  
  top: 0;  
}  
/*Paso 4: Mostrar la flecha*/  
.nav li .flecha {  
  font-size: 9px;  
  padding-left: 6px;  
  display: none;  
  text-align: right;  
}
```

2. Layouts

- El diseño de las páginas web habituales se divide en bloques: cabecera, menú, contenidos y pie de página donde se usa CSS para formatear cada apartado/página.
- La principal ventaja de usar CSS:
 - ❖ Mantenimiento: una página diseñada exclusivamente con CSS es mucho más fácil de mantener que una página diseñada con tablas.
 - ❖ Accesibilidad: las páginas creadas con CSS son más accesibles que las páginas diseñadas con tablas.
 - ❖ Velocidad de carga frente a otras soluciones

2. Layouts

➤ Diseño a 2 columnas con cabecera y pie de página



```
<body>
<div id="contenedor">
  <div id="cabecera">
  </div>

  <div id="menu">
  </div>

  <div id="contenido">
  </div>

  <div id="pie">
  </div>
</div>
</body>
```



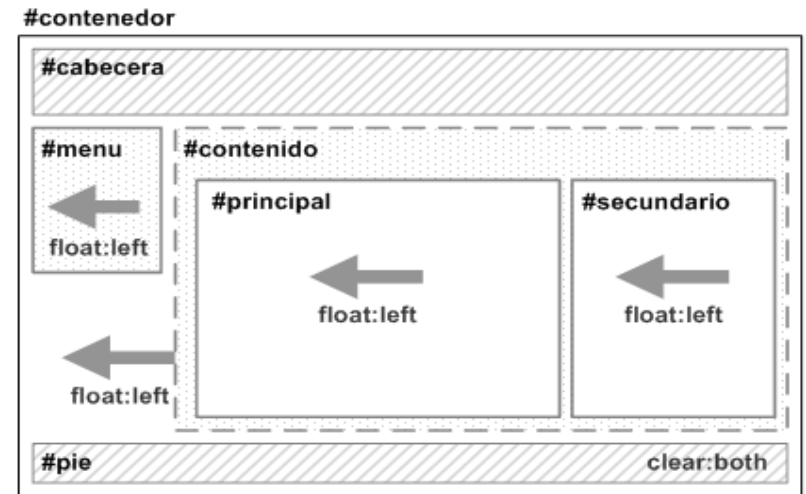
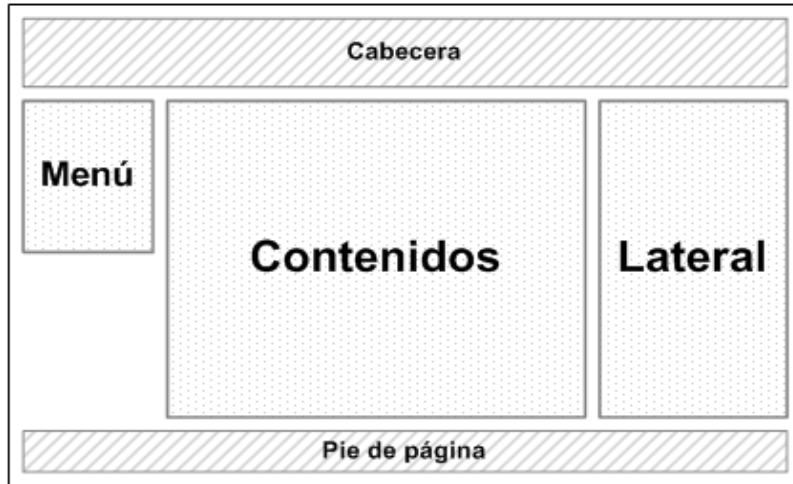
```
#contenedor {
  width: 700px;
}
#cabecera {
}
#menu {
  float: left;
  width: 150px;
}
#contenido {
  float: left;
  width: 550px;
}
#pie {
  clear: both;
}
```

Adaptable

```
#contenedor {
}
#cabecera {
}
#menu {
  float: left;
  width: 15%;
}
#contenido {
  float: left;
  width: 85%;
}
#pie {
  clear: both;
}
```


2. Layouts

➤ Diseño a 3 columnas con cabecera y pie de página



```
<div id="contenedor">
  <div id="cabecera">
  </div>

  <div id="menu">
  </div>

  <div id="contenido">
    <div id="principal">
    </div>

    <div id="secundario">
    </div>
  </div>

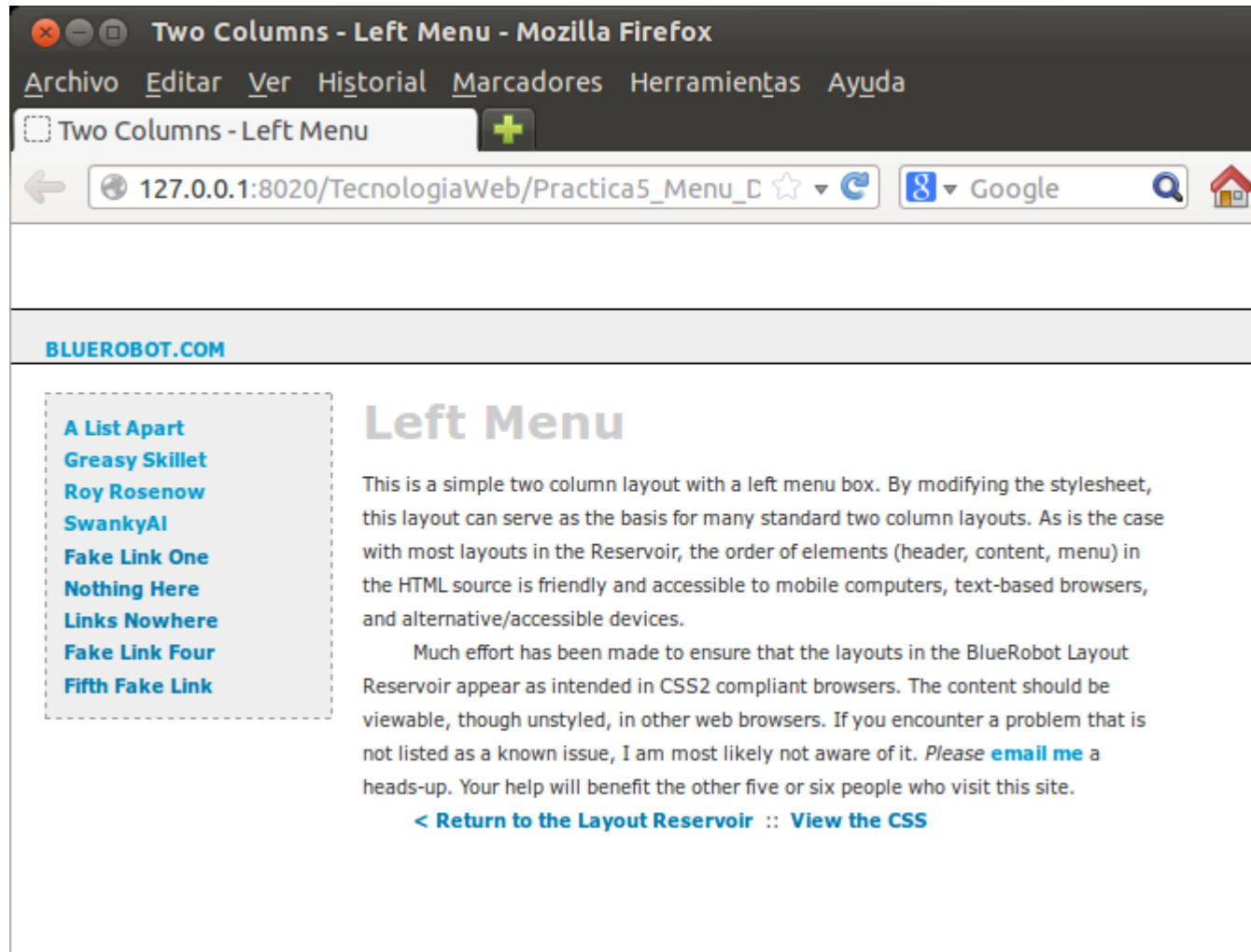
  <div id="pie">
  </div>
</div>
```

```
#contenedor {
}
#cabecera {
}
#menu {
  float: left;
  width: 15%;
}
#contenido {
  float: left;
  width: 85%;
}
#contenido #principal {
  float: left;
  width: 80%;
}
```

```
#contenido #secundario {
  float: left;
  width: 20%;
}
#pie {
  clear: both;
}
```

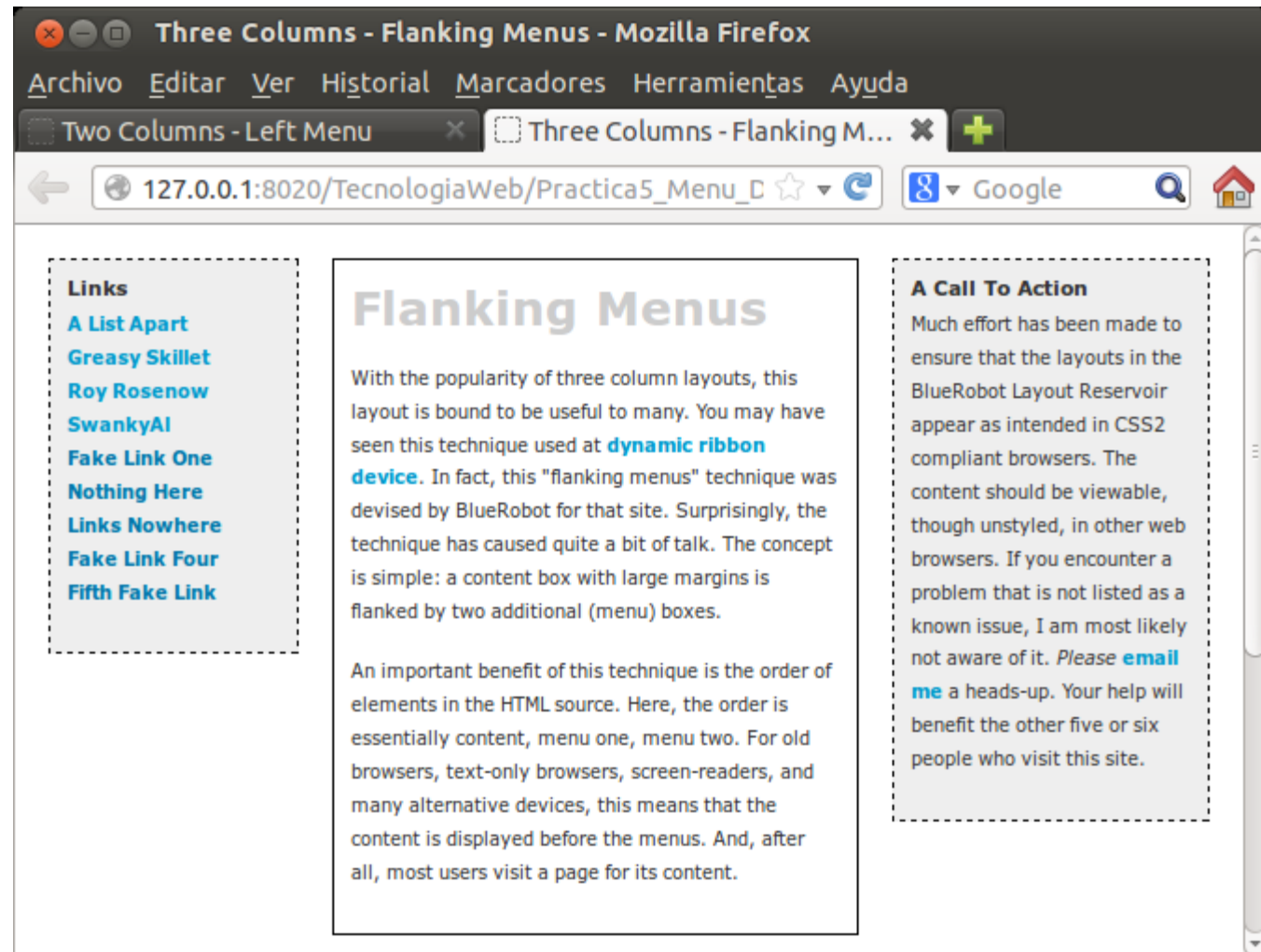
2. Layouts

➤ Ejercicio 7A. Analizar el código del siguiente layout



2. Layouts

➤ Ejercicio 7B. Analizar el código del siguiente layout



2. Layouts

- Ejercicio8. Dado el siguiente código XHTML + CSS , se pide modificar el css para conseguir el siguiente diseño



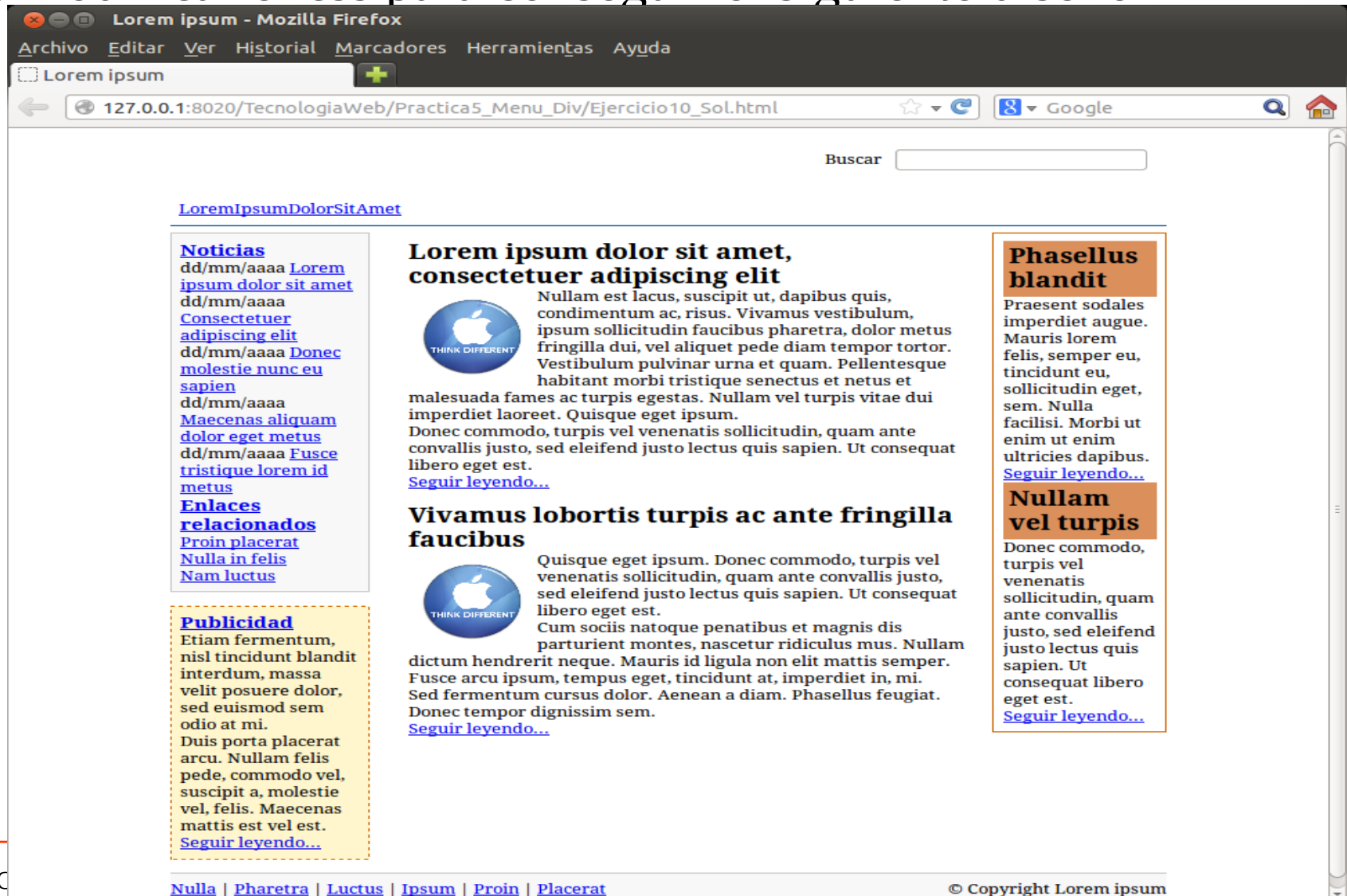
2. Layouts

- Ejercicio8. Dado el siguiente código XHTML + CSS , se pide modificar el css para conseguir el siguiente diseño



2. Layouts

- Ejercicio8B. Dado el siguiente código XHTML + CSS , se pide modificar el css para conseguir el siguiente diseño



2. Layouts

- Existen multitud de sitios con ejemplos de layout (en el buscador: layout examples). Algunas web recomendables:
 - ❖ <http://www.bluerobot.com/web/layouts/>
 - ❖ <http://www.benmeadowcroft.com/webdev/>
 - ❖ http://www.thenoodleincident.com/tutorials/box_lesson/boxes.html
 - ❖ <http://www.webproducer.at/flexible-layout>
 - ❖ etc
- Así como web que automáticamente generan layout
 - ❖ <http://www.wannabegirl.org/firdamatic/>
 - ❖ <http://csscreator.com/version2/pagelayout.php>
 - ❖ etc

Introducción JavaScript

- js_ejercicio1.html: Ejecuta el siguiente ejercicio

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>El primer script</title>
    <script type="text/javascript">
        alert("Hola Mundo!");
    </script>
</head>
<body>
<p>Esta página contiene el primer script</p>
</body>
</html>
```

- js_ejercicio1b: Haz lo mismo pero poniendo el script en un fichero externo (js/script1.js)

Introducción JavaScript

- js_ejercicio2.html y js/script2.js: Observa el código
- js_ejercicio2B.html y js/script2B.js: Observa el código
- js_ejercicio2C.html y y js/script2C.js: Observa el código
 - ❖ A continuación se debe pedir por teclado un número e ir indicando si es mayor o menor al número aleatorio hasta que se acierte con el número
- js_ejercicio2D.html y y js/script2D.js:
 - ❖ El mismo ejercicio que el anterior con un número máximo de intentos (5 por defecto)

Introducción JavaScript

- js_ejercicio3. Calculadora: Dado el código html, css y esqueleto de javascript de una calculadora, se pide realizar su implementación
 - ✓ hay 3 variables: dos números (num1 y num2) y una para realizar las operaciones (opera).
 - ✓ darNumero(numero): Concatena el número pulsado al cuadro de texto. Luego refresca.
 - ✓ darComa(): Añade una coma al texto del cuadro de texto
 - ✓ darC(): borrar todos los resultados, es decir, poner las variables a 0.
 - ✓ operar(valor): Recibe un número en base a la operación. El 1 indica que se trata de una suma, el 2 de una resta, el 3 de una multiplicación, el 4 de una división y el 5 de una potencia.
 - ✓ esIgual(): Realiza la operación y muestra el resultado.
 - ✓ refrescar(): Cambia el valor del número del cuadro de texto.