

MÓDULO

Diseño de Interfaces Web

UNIDAD 2-2

Elementos Semánticos HTML5 y propiedades CSS3

1. Introducción

En el capítulo anterior vimos un repaso a HTML5 y CSS3 haciendo hincapié en aquellas cosas que no han cambiado con respecto a las versiones anteriores de estas tecnologías.

En este segundo capítulo, comenzaremos a estudiar las nuevas características de ambas.

En lo que respecta a CSS, veremos los nuevos selectores CSS3, pseudo-elementos, pseudo-clases, etc. También veremos algunas nuevas características que nos facilitarán la labor a la hora de crear diseños más complejos, como es el caso de los bordes redondeados, sombras en los textos y en las cajas, transparencias, imágenes de borde, etc.

2. HTML5

HTML5 incorpora nuevos elementos que ayudan a identificar cada sección del documento y organizar su cuerpo. Estas secciones son diferenciadas y la estructura principal ya no depende más de los elementos `<div>`

Normalmente, una página o aplicación web está dividida entre varias áreas visuales para mejorar la experiencia del usuario y facilitar la interactividad. Las palabras claves que representan cada nuevo elemento de HTML5 están íntimamente relacionadas con estas áreas.

Cuando ponemos un elemento `<h1>` estamos indicando que contiene un título de nivel 1, cuando introducimos un elemento `<p>` estamos indicando que contiene un párrafo de texto.

2. HTML5

El atributo id es un identificador único, pero, en lugar de una cadena aleatoria, el identificador suele ser una palabra significativa, puede tener valor semántico. Por el contrario, el atributo class no es único, distintos elementos pueden tener el mismo valor en este atributo, además, un elemento puede tener aplicadas múltiples clases. A continuación, veremos algunos ejemplos:

Etiqueta	Significado
<code><p></code>	Un párrafo
<code><p id="autor"></code>	Un párrafo que representa un autor particular
<code><p class="biografia"></code>	Un párrafo que representa una biografía
<code><p class="autor biografia"></code>	Un párrafo que representa la biografía de un autor

2. HTML5

Ya vimos en el capítulo anterior que la mayoría de los sitios web tienen un diseño similar:

1. Cabecera
2. Barra de Navegación
3. Sección de Información Principal
4. Barra Lateral
5. El pie o la barra Institucional.



2. HTML5

HTML5 considera esta estructura básica y provee nuevos elementos para diferenciar y declarar cada una de sus partes. A partir de ahora, podemos decir al navegador para qué es cada sección:

El elemento header

Un elemento `<header>` puede ser utilizado para incluir contenido introductorio o ayudas a la navegación que sean específicas de una sola sección de la página, que se aplican a la totalidad de la página, o ambas cosas.



Normalmente, irá en la parte superior de una página o sección.

2. HTML5

El elemento section

“El elemento section representa una sección genérica de un documento o aplicación. Una sección, en este contexto, es una agrupación temática de los contenidos, por lo general, con un título”.

Una sección no debe ser utilizada como un contenedor genérico que existe sólo con fines de estilo o scripting, en esos casos utilizaremos el elemento `<div>` que es para lo que se creó.

Algunos usos aceptables para elementos `<section>` son:

- Secciones individuales de una interfaz con pestañas.
- Segmentos de una página "Acerca de", por ejemplo, la página "Acerca de" de una empresa podría incluir secciones sobre la historia de la

2. HTML5

empresa, su misión y su equipo.

- Diferentes partes de los “términos de servicio” de la página.
- Distintas secciones de un sitio de noticias en línea, por ejemplo, los artículos se podrían agrupar en secciones que cubren los deportes, los asuntos mundiales, y las noticias económicas.

`<section>` es genérico, por lo que, si un elemento semántico más específico es apropiado (como `<article>`, `<aside>`, o `<nav>`), lo utilizaremos en su lugar.

`<section>` tiene significado semántico. Si no podemos describir el contenido que estamos tratando de poner en una sección utilizando sólo unas pocas palabras, lo más probable es que necesitemos un contenedor semánticamente neutral (`<div>`) en su lugar.

Es correcto anidar secciones dentro de otras existentes.

2. HTML5

El elemento article

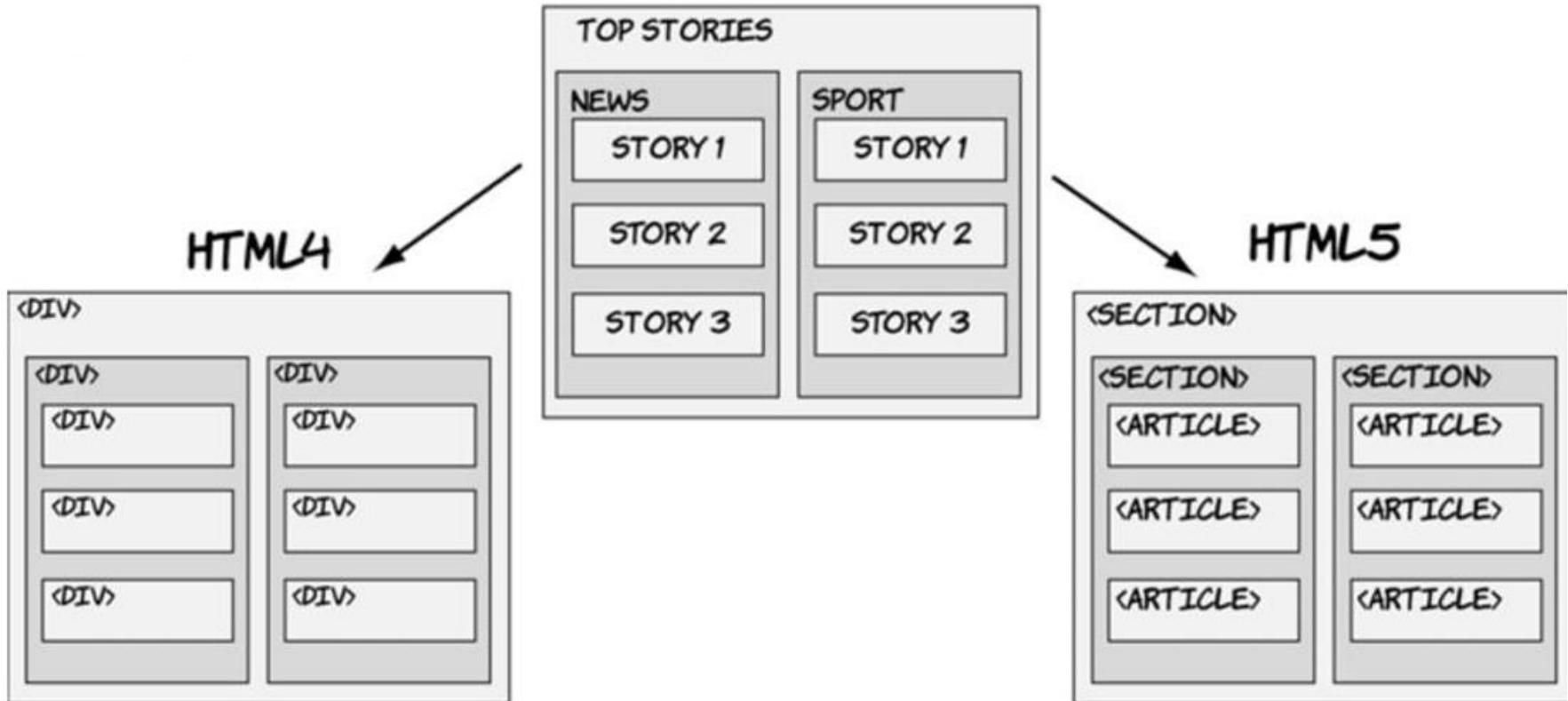
El elemento `<article>` es similar al elemento `<section>`, pero hay algunas diferencias.

Mientras que `<section>` puede contener cualquier contenido que se puede agrupar temáticamente, `<article>` debe ser una sola pieza de contenido. Algunas sugerencias de uso de `<article>`:

- Mensajes en el foro.
- Artículos de revistas o periódicos.
- Las entradas de un blog.
- Los comentarios enviados por los usuarios, etc

Los elementos `<article>` se pueden anidar. También podemos anidar una sección dentro de un artículo, y viceversa.

2. HTML5



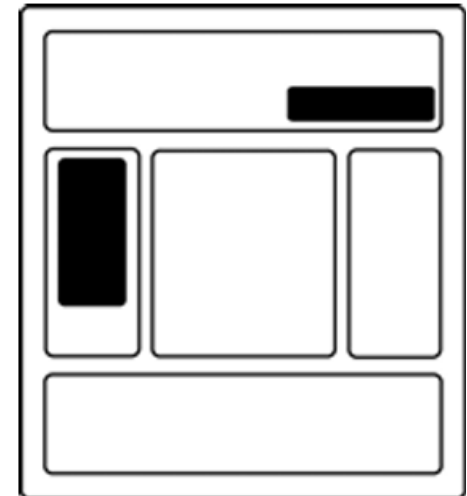
2. HTML5

El elemento nav

Este elemento aparecerá en, prácticamente, todos los proyectos. `<nav>` representa un grupo de vínculos de navegación. Lo más habitual será que contenga una lista desordenada de enlaces, aunque hay otras opciones.

Será adecuado un elemento `<nav>` en cualquiera de los siguientes casos:

- Si tenemos una barra de navegación principal del sitio web.
- Si tenemos un conjunto secundario de enlaces que apuntan a diferentes partes de la página actual (mediante anclajes).
- Para un formulario de búsqueda que constituye el principal medio de navegación de un sitio (como es el caso de Google).



2. HTML5

No será adecuado `<nav>`, en los enlaces del pie de página.

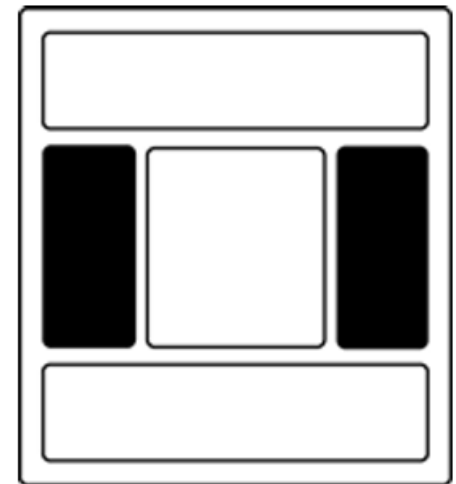
El elemento aside

Representa una parte de la página que está "tangencialmente relacionado con el contenido que se encuentra alrededor, y que podría considerarse separado de ese contenido".

El elemento `<aside>` nunca debe ser usado para envolver las secciones de la página que son parte del contenido principal.

Usos posibles para un `<aside>` serían: una barra lateral, una lista secundaria de enlaces, o un espacio para la publicidad.

El elemento `<aside>` no se define por su posición en la página.

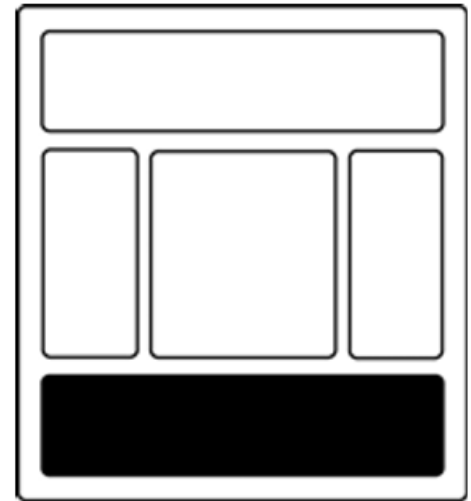


2. HTML5

El elemento footer

“Representa un pie de página de la sección de contenido que es su ancestro más cercano”.

A menudo, un pie de página contendrá información sobre el copyright, las listas de enlaces, información del autor, y contenido similar que, normalmente, consideramos como el final de un bloque de contenido. Sin embargo, al igual que `<aside>` y `<header>`, un elemento `<footer>` no se define en términos de su posición en la página, por lo que no tiene por qué aparecer al final de una sección, o en la parte inferior de una página. Lo más probable es que sí, pero esto no es necesario.



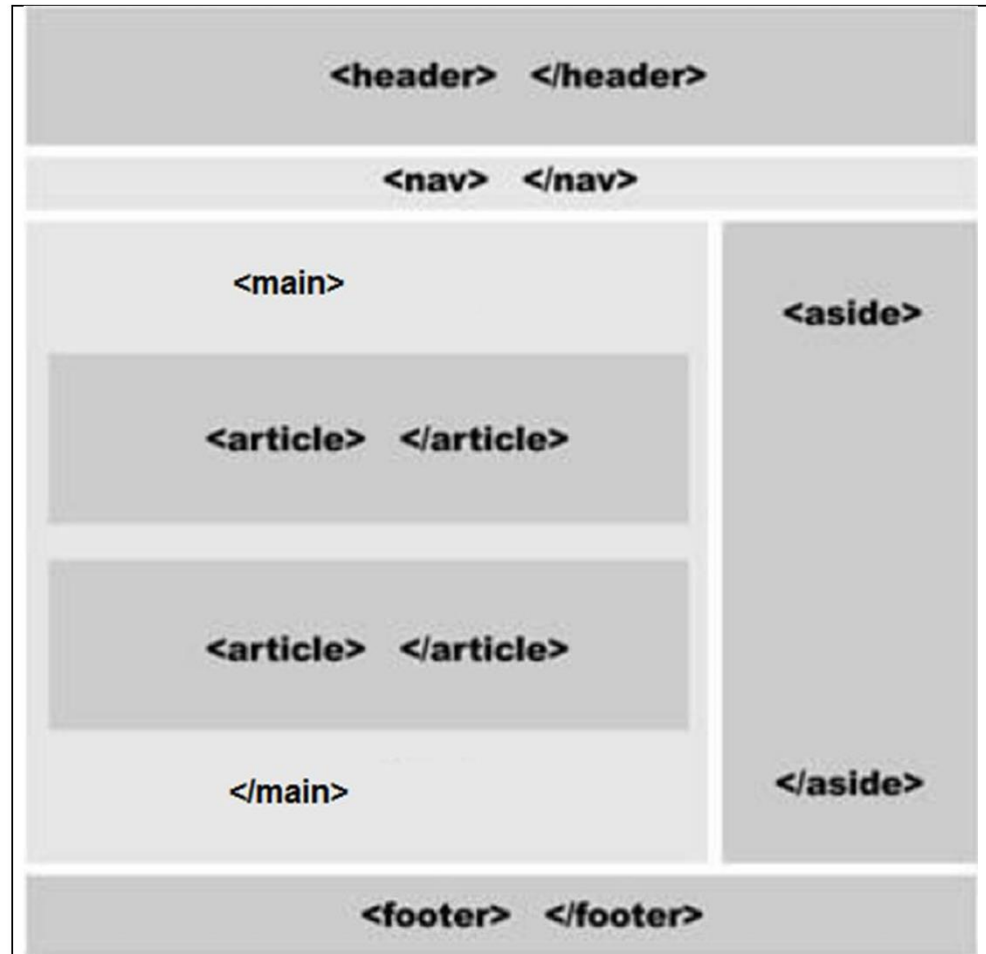
2. HTML5

El elemento main

El elemento `<main>` especifica el contenido principal del documento. Las características principales de este elemento son:

- Representa el contenido principal del cuerpo (`<body>`) de una web o aplicación.
- Incluye el contenido que es único en la página, y excluye contenido que se repite en todas las páginas de la web (menú, pie de página, barra lateral, etc.).
- No debe incluirse más de un elemento `<main>` por página.
- No debe incluirse el elemento `<main>` dentro de elementos como `<article>`, `<aside>`, `<footer>`, `<header>` o `<nav>`.

3. Organizando el diseño de la página con HTML5



4. Otros elementos semánticos

Los elementos figure y figcaption

El elemento `<figure>` se utilizará para anotar ilustraciones, diagramas, fotos, listados de código, etc, a los cuales se refiera el contenido principal del documento.

`<figcaption>` es, simplemente, una manera de marcar un título.

```
<figure>
  <figcaption>Screen Reader Support for WAI-ARIA</figcaption>
  
</figure>
```

El elemento mark

Indica una parte del documento que ha sido remarcada por la importancia que tiene.

```
<h1>Sí, usted puede utilizar <mark> HTML5 </mark> Hoy!</h1>
```


4. Otros elementos semánticos

El elemento time

El elemento `<time>` ha sido diseñado específicamente para tratar con la lectura de las fechas y horas. Por ejemplo:

```
<p>La siguiente conferencia de HTML5 será el próximo 12 de  
Octubre.</p>
```

introduciendo el elemento `<time>`:

```
<p> La siguiente conferencia de HTML5 será el próximo  
  <time datetime="2014-10-12">12 de Octubre</time>.  
</p>
```

5. Elementos con la semántica redefinida

El elemento small

En HTML5, el nuevo propósito del elemento `<small>` es presentar impresiones legales, descargos, etc... (letra pequeña de contratos)

```
<small>Derechos Reservados &copy; 2024 Javier Tárrega</small>
```

El elemento cite

Ahora las etiquetas `<cite>` encierran el título de un trabajo, como un libro, una película, una canción, etc...

```
<span>Me encanta la película <cite>Gladiator</cite></span>
```

El elemento address

Para situaciones en las que debemos presentar información de contacto

```
<address>Escrito por: <a href="http://www.mario.com">Mario</a></address>
```

6. Categorías de elementos HTML5

HTML5 clasifica los elementos y los divide en las siguientes categorías:

- Metadatos: no están presentes en la página, pero afectan a la presentación o incluyen otra información (<title>, <link>, <meta> y <style>).
- Elementos que afectan al flujo de contenidos de la página: casi todos los elementos que se utilizan en el cuerpo (<body>), por ejemplo <header>, <footer>, e incluso <p>.
- Elementos de seccionamiento. Secciones.
- Encabezados (<h1>, <h2>, etc.)
- Contenido de frase (, , <cite> y similares)
- Contenido incrustado (, <object>, <embed>, <video>, <canvas> y otros).
- Contenido interactivo, elementos de formulario, así como enlaces y otros elementos que son interactivos cuando tienen ciertos atributos.

7. CSS3

Nuevos selectores CSS3

- Selector general de elementos hermanos. `h1 ~ h2 { ... }`

Selecciona a todos los h2 hermanos de h1

- Nuevos selectores de atributo

```
p[name^="mi"] {font-size: 20px; } /*todo elemento <p> que  
contenga un atributo name con un valor que comience en "mi"
```

```
p[name$="mi"] { font-size: 20px;} /*todo elemento <p> que  
contenga un atributo name con un valor que acabe en "mi"
```

```
p[name*="mi"] { font-size: 20px;} /*todo elemento <p> que  
contenga un atributo name con un valor que incluya en "mi"
```

- Pseudo-elementos. Los pseudo-elementos de CSS 2.1 cambian su sintaxis y ahora se utilizan `::` en vez de `:` delante del nombre de cada pseudo-elemento (`::first-line`, `::first-letter`, `::before`...)

7. CSS3

- Nuevo pseudo-elemento. `::selection` que selecciona el texto que ha seleccionado un usuario con su ratón o teclado.

```
::selection {color:red;background:yellow;}
```

- Nuevas pseudo-clases. se puede utilizar `even` y `odd` como índice para seleccionar los hijos pares o impares

Psudo-clase	descripción
<code>:nth-child(n)</code>	Selecciona el enésimo hijo de su padre
<code>:nth-last-child(n)</code>	Selecciona el enésimo hijo de su padre contando desde el último
<code>:nth-of-type(n)</code>	Selecciona el enésimo hermano de su tipo
<code>:nth-last-of-type(n)</code>	Selecciona el enésimo hermano de su tipo comenzando desde el último
<code>:first-child</code>	Selecciona el primer hijo de su padre
<code>:last-child</code>	Selecciona el último hijo de su padre
<code>:first-of-type</code>	Selecciona el primer hermano de su tipo
<code>:last-of-type</code>	Selecciona el último hermano de su tipo
<code>:only-child</code>	Selecciona los que sean hijos únicos
<code>:only-of-type</code>	Selecciona los que sean los únicos hermanos de su tipo
<code>:empty</code>	Selecciona los elementos que no tienen hijos. Si un elemento contiene sólo texto no se considera vacío.

8. Nuevas propiedades CSS3

Bordes con esquinas redondeadas (border-radius).

Si todas las esquinas tienen la misma curvatura podemos utilizar un solo valor:

```
border-radius: 20px;
```

Curso HTML5 y CSS3

Como ocurre con las propiedades margin y padding, podemos también declarar un valor diferente para cada esquina. Los valores se aplicarán en el orden: esquina superior izquierda, esquina superior derecha, esquina inferior derecha y esquina inferior izquierda:

```
border-radius: 20px 10px 30px 50px;
```

Curso HTML5 y CSS3

8. Nuevas propiedades CSS3

`border-radius` también puede trabajar sólo con dos valores: el primer valor será asignado a la primera y tercera esquina (superior izquierda, inferior derecha), y el segundo valor a la segunda y cuarta esquina (superior derecha, inferior izquierda):

```
border-radius: 20px 50px;
```

Curso HTML5 y CSS3

También podemos dar forma a las esquinas declarando un segundo grupo de valores separados por una barra. Los valores a la izquierda de la barra representarán el radio horizontal, mientras que los valores a la derecha representan el radio vertical:

```
border-radius: 50px / 20px;
```

<http://www.cssmatic.com/border-radius>

Curso HTML5 y CSS3

8. Nuevas propiedades CSS3

Sombras en las cajas (box-shadow).

La propiedad box-shadow necesita al menos tres valores: Los dos primeros valores, expresados en píxeles, establecen el desplazamiento de la sombra y el tercero indicará el color de la misma. Este desplazamiento puede ser positivo o negativo:

```
box-shadow: 5px 5px rgb(150, 150, 150);
```

Curso HTML5 y CSS3

Otro valor que se puede agregar a la propiedad es la distancia de difuminación. Este será un valor expresado en píxeles que indicará como se va a difuminar la sombra:

```
box-shadow: 5px 5px 10px rgb(150, 150, 150);
```

Curso HTML5 y CSS3

8. Nuevas propiedades CSS3

Si añadimos un cuarto valor en píxeles antes del color, veremos cómo aumenta el tamaño de la sombra en todas las direcciones:

```
box-shadow: 5px 5px 10px 10px rgb(150, 150, 150);
```

Curso HTML5 y CSS3

El último valor posible para box-shadow no es un número, sino una palabra clave: inset. Esta palabra clave la indicaremos después del color, y convierte la sombra externa en una sombra interna, lo cual provee un efecto de profundidad al elemento afectado:

```
box-shadow: 5px 5px 10px rgb(150, 150, 150) inset;
```

Curso HTML5 y CSS3

<http://www.cssmatic.com/box-shadow>

8. Nuevas propiedades CSS3

Sombras en los textos (text-shadow).

Los valores para text-shadow son similares a los usados para box-shadow. Podemos declarar la distancia horizontal y vertical de la sombra con respecto al objeto, el radio de difuminación y el color de la sombra:

```
text-shadow: 5px 5px 5px rgb(150, 150, 150);
```



Curso HTML5 y CSS3

Para probar de una forma sencilla cómo quedaría esta propiedad puedes utilizar la herramienta disponible en la url:

<http://css3gen.com/text-shadow/>

8. Nuevas propiedades CSS3

Fondos con gradiente.

Los gradientes son configurados como fondos, por lo que podemos usar las propiedades background o background-image para declararlos. Tenemos dos tipos de gradientes: gradiente lineal y gradiente radial:

Gradiente lineal. Los indicamos con la función linear-gradient, que recibe los siguientes parámetros:

- Punto de comienzo: donde comenzará el gradiente. Puede ser especificado en píxeles, porcentaje o usando las palabras clave top, bottom, left y right. También puede ser reemplazado por un ángulo para declarar una dirección específica del gradiente.
- Color inicial: indica con que color comenzará el gradiente.
- Color final: indica con que color terminará el gradiente

8. Nuevas propiedades CSS3

```
background: linear-gradient(to bottom, #ffffff, #006699);
```



Curso HTML5 y CSS3

```
background: linear-gradient(30deg, #ffffff, #006699);
```



Curso HTML5 y CSS3

También podemos declarar los puntos de terminación para cada color de la siguiente forma:

```
background: linear-gradient(30deg, #ffffff 50%, #006699 90%);
```



Curso HTML5 y CSS3

8. Nuevas propiedades CSS3

Gradiente radial. Debemos usar la función `radial-gradient()` y un nuevo atributo para la forma.

```
background: radial-gradient(circle, #ffffff 0%, #006699 200%);
```

Curso HTML5 y CSS3

Existen dos posibles valores para la forma: `circle` y `ellipse` (el valor por defecto será `ellipse`). Además, como puedes observar en el ejemplo, para cada color se debe indicar la posición donde las transiciones comienzan (en el ejemplo, el gradiente comenzaría en la posición 0% y terminaría en la posición 200%).

Para practicar con gradientes puedes utilizar la herramienta:

<http://css3gen.com/gradient-generator/>

8. Nuevas propiedades CSS3

Outline.

La propiedad outline es una vieja propiedad CSS que ha sido expandida en CSS3 para incluir un valor de desplazamiento.

Esta propiedad se usaba para crear un segundo borde, y ahora ese borde puede ser mostrado alejado del borde real del elemento.

Por ejemplo, podemos añadir un segundo borde de 2 píxeles y un desplazamiento de 15 píxeles a la caja de nuestro título.

```
outline: 2px dashed #000099;  
outline-offset: 15px;
```



8. Nuevas propiedades CSS3

Border-image.

Ofrece la alternativa de utilizar imágenes propias en los bordes.

Imagen de borde: border-image-source.

Establece la imagen a utilizar como imagen de borde.

```
border-image-source: url("fo_bola_100.png");
```



Curso HTML5 y CSS3



```
border: 27px;  
border-style: solid;  
border-image-source:  
    url("fo_pastilla_60.png");  
border-image-slice: 27;
```

Curso HTML5 y CSS3

8. Nuevas propiedades CSS3

```
border-width: 50px 80px 20px 80px;  
border-style: solid;  
border-image-source:  
    url("fo_ventana.png");  
border-image-slice: 50 80 20 80;
```

```
border: 27px;  
border-style: solid;  
border-image-source:  
    url("fo_pastilla_60.png");  
border-image-slice: 27 fill;
```

```
border: 40px;  
border-style: solid;  
border-image-source:  
    url("fo_hormigas_210.png");  
border-image-slice: 70;  
border-image-repeat: stretch;
```



9. Recordatorio de posicionamientos

Posiciones predeterminadas

Por defecto son elementos en bloque que tienden de forma natural a ocupar todo el ancho del contenedor en el que están se puede asignar ancho y alto, pero no se pueden situar uno junto a otro. Para centrarlos podemos ajustar los márgenes en css.

Posiciones predeterminadas con float

Le asigno float left en css y sale de su ubicación natural subiendo el bloque de abajo arriba al "dejarle sitio" aunque normalmente puede quedar tapado por el flotante y a veces no se ve. Puedo usar los márgenes y desplazarlo horizontalmente en pantalla. Para "romper" la flotación clear (puede ser left, right o both)

9. Recordatorio de posicionamientos

Posiciones predeterminadas con display (inline, inline-block, block, none)

Dejamos de considerar las etiquetas como bloques y al ser inline pasan a ocupar el ancho de su contenido. Con inline-block podemos asignar un ancho.

Tiene un pequeño inconveniente de márgenes que no se quitan fácilmente. (<https://escss.blogspot.com/2012/03/display-inline-block-y-sus-empeno-en.html>).

Posiciones relativas

Puedo usar las propiedades top, left, right y bottom por lo que puedo ubicar las etiquetas "dónde quiero" pero siempre con respecto a la ubicación natural que le correspondía.

9. Recordatorio de posicionamientos

Posiciones absolutas

Para posicionarlos cambiamos la posición del contenedor padre a relativa y las etiquetas hijas en absolutas y ya tengo disponible top left right y bottom en cada una de ellas.