

3.3.9. Pseudo-clases

Se utilizan para seleccionar elementos en función, no del código como hasta ahora, sino del comportamiento posterior del usuario (eventos), por ejemplo, en función de los enlaces que visita, el movimiento del ratón, etc.

Algunas pseudo-clases son:

- :link (se aplica a los enlaces nunca visitados)
- :visited (se aplica a los enlaces visitados al menos una vez)
- :hover (se aplica a un elemento seleccionado por el usuario pero sin activarlo, como por ejemplo al situar el ratón encima de un objeto sin hacer clic)
- :active (se aplica a los elementos que están siendo activados, por ejemplo, mientras el usuario presiona el botón izquierdo del ratón)
- :focus (se aplica a un elemento que tiene el foco, por ejemplo, un cuadro de entrada en un formulario)

Algunos ejemplos de pseudo-clases serían los siguientes:

<code>a:link {color: black;}</code>	Enlaces sin visitar -> negro
<code>a:visited {color: blue;}</code>	Enlaces visitados -> azul
<code>a:hover {color: green;}</code>	Con el ratón encima -> verde
<code>a:active {color: red;}</code>	Presionando el botón -> rojo
<code>a:focus {background-color: yellow;}</code>	Con el foco-> fondo amarillo

Actividad 3.2:

Crea una tabla con 4 enlaces y utiliza las tres primeras pseudo-clases, de modo que los enlaces aparezcan en negro al principio, en azul al ser visitados y en verde con el ratón encima.

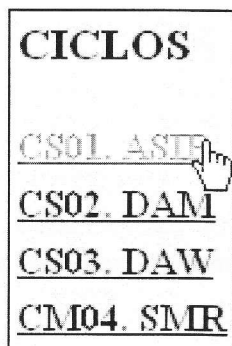


Figura 3.6: Selector pseudo-clase

3.3.10. Pseudo-elementos

Algunos pseudo-elementos son:

- **:first-line** (se aplica a la primera línea de un texto)
- **:first-letter** (se aplica a la primera letra de un texto)
- **:before** (sirve para generar texto antes del contenido de un elemento)
- **:after** (sirve para generar texto después del contenido de un elemento)

Ejemplo:

```
<html>
  <head>
    <style>
      h1:before {content: "Encabezado --- "; }
      h1:after {content: " --- Fin de encabezado"; }
      p:first-line {font-size: 20px; }
      p:first-letter {font-size: 3em; }
    </style>
  </head>
  <body>
    <h1>Texto1</h1>
    <p>
      Esta es la primera línea <br>
      Esta es la segunda línea <br>
      Esta es la tercera línea <br>
      Esta es la cuarta línea <br>
    </p>
  </body>
</html>
```



Figura 3.7: Selector pseudo-elemento

En CSS 3 se incorporan los siguientes selectores:

Selector general de hermanos, selectores de atributos (^=, \$=, *=), :first-of-type, :last-of-type, only-of-type, only-child, :nth-child (n), :nth-last-child (n), :nth-of-type (n), :nth-last-of-type (n), :first-child, :last-child, :root, :empty, :target, :enabled, :disabled, :checked, :not y ::selection.

A continuación, veremos el significado de los más relevantes.

- Selector general de hermanos (elemento1 ~ elemento2)

Selecciona los elemento2 precedidos por elemento1 y con el mismo padre (hermanos).

Ejemplo:

p ~ ul → Selecciona los elementos precedidos por <p>.

- Selectores de atributos:

- elemento[atributo^="valor"] selecciona los elementos cuyo "atributo" empieza por "valor".
- elemento[atributo\$="valor"] selecciona los elementos cuyo "atributo" termina por "valor".
- elemento[atributo*="valor"] selecciona los elementos cuyo "atributo" contiene "valor".

Ejemplo:

img[src^="android"] selecciona todas las imágenes cuyo atributo src empieza por "android".

- Pseudoclases:

- elemento:nth-child (n) selecciona los elementos que sea el hijo n-ésimo.
- elemento:nth-last-child (n) igual que el anterior pero se empieza a contar desde el último hijo.
- elemento:empty selecciona los elementos que no tenga ningún hijo ni contenido.
- elemento:first-child y elemento:last-child seleccionan los elementos que sean el primer hijo o el último hijo de su padre.
- elemento:nth-of-type (n) selecciona los elementos que sean el enésimo elemento de ese tipo.
- elemento:nth-last-of-type (n) idéntico al anterior pero el número indicado se empieza a contar desde el último hijo.
- :enabled, :disabled, :checked seleccionan campos de formulario según su estado.
- elemento:not (selector) selecciona elementos que no sean como "selector".
- ::selection selecciona el contenido marcado por el usuario.

Estos selectores son muy específicos por lo que se suelen utilizar con menos frecuencia que los anteriores de CSS 2.

3.4. Modelo de cajas

3.4.1. Fundamentos

El modelo de cajas es el elemento fundamental de CSS porque determina la composición de la página web. Todos los elementos que se insertan en el documento HTML se representan automáticamente mediante cajas rectangulares (excepto head).

En principio son cajas invisibles salvo que les asignemos bordes o color de fondo, para comprobar su apariencia podemos aplicarle bordes, como en la figura siguiente.

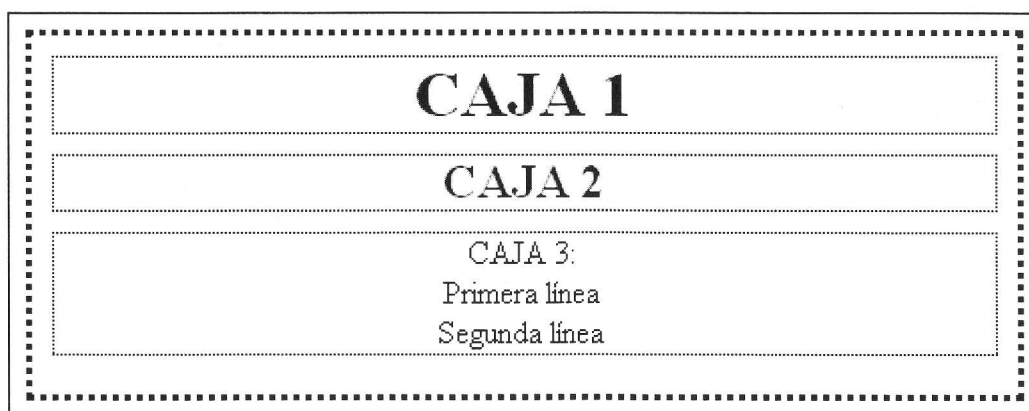


Figura 3.8: Modelo de cajas

La caja de puntos gruesos corresponde al elemento body, los puntos finos corresponden a los elementos h1, h2 y p. La caja exterior de borde sólido corresponde al elemento html.

```
<html>
<head>
  <style>
    html {border: 1px solid;}
    body {border: 3px dotted; text-align: center;}
    p,h1,h2 {border: 1px dotted; text-align: center; margin: 10px;}
  </style>
</head>
<body>
  <h1>CAJA 1</h1>
  <h2>CAJA 2</h2>
  <p>
    CAJA 3: <br>
    Primera línea <br> Segunda línea <br> Tercera línea <br>
    Cuarta línea <br> Quinta línea
  </p>
</body>
</html>
```

Las partes que componen cada caja y su orden de visualización son las siguientes:

1. **Contenido** (content): texto, imágenes, videos, listas, etc. del elemento.
2. **Relleno** (padding): espacio entre el contenido y el borde.
3. **Borde** (border): línea que encierra completamente el contenido y su relleno.
4. **Imagen de fondo** (background-image): imagen que se muestra por detrás del contenido y el espacio de relleno.
5. **Color de fondo** (background-color): color que se muestra por detrás del contenido y el espacio de relleno.
6. **Margen** (margin): espacio libre entre la caja y las cajas adyacentes.

Padding y margin son transparentes.

En el espacio ocupado por padding se muestra el color o imagen de fondo.

En el espacio ocupado por margin se muestra el color o imagen de fondo de su elemento padre.

Si ningún elemento padre tiene definido un color o imagen de fondo, se muestra el color o imagen de fondo de la página.

Si una caja define tanto un color como una imagen de fondo, la imagen tiene más prioridad (4 antes que 5) y se visualiza antes que el color.

Si la imagen de fondo no cubre totalmente la caja del elemento o si la imagen tiene zonas transparentes, entonces se visualiza el color de fondo en dichas zonas.

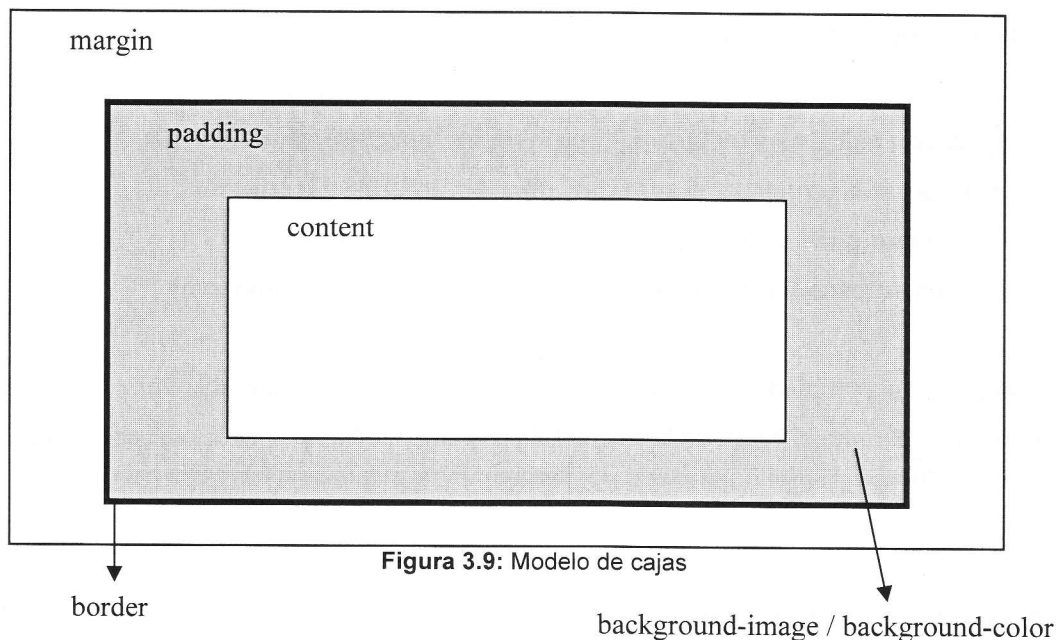


Figura 3.9: Modelo de cajas

3.4.2. Ancho, alto

Las propiedades que controlan el ancho y alto de una caja son `width` y `height`.

Propiedades: **`width/height`**

Significado: **ancho/alto de la caja**

Valores: **`<número px>` | `<número %>` | `inherit` | `auto`**

Aplicación: **elementos de bloque, imágenes y columnas de tabla**

`<número px>`: establece un ancho/alto fijo de la caja en píxeles.

`<número %>`: establece un ancho/alto fijo de la caja según un porcentaje del ancho/alto del elemento padre.

`<inherit>`: establece un ancho/alto fijo heredado del elemento padre.

`<auto>`: es el valor por defecto y determina que el navegador debe calcular el ancho/alto teniendo en cuenta el espacio disponible en la página.

Ejemplo:

```
#menu {width: 180px; height: 600px;}
```

```
<div id="menu">
```

```
...
```

```
</div>
```

3.4.3. Margin, Padding

Propiedades: **`margin`, `margin-top`, `margin-right`, `margin-bottom`, `margin-left`**

Significado: **margen, margen superior, derecho, inferior e izquierdo**

Valores: **`<número px, em>` | `<número %>` | `inherit` | `auto`**

Aplicación: **a todos los elementos** (excepto `margin-top` y `margin-bottom`)

Margin es una propiedad ‘shorthand’ que equivale a las otras cuatro a la vez.

El margen permite el centrado horizontal de una caja mediante → **`margin: 0 auto;`** que significa margen superior e inferior 0 y margen derecho e izquierdo automático.

Ejemplo:

```
#body {margin-top: 2em;}
```

```
#caja1 {margin: 10px;}
```

```
#centrado {margin: 0 auto;}
```

Propiedades: **padding, padding-top, padding-right, padding-bottom, padding-left**

Significado: **relleno, relleno superior, derecho, inferior e izquierdo**

Valores: **<numérico px, em> | <numérico %> | inherit | auto**

Aplicación: **a todos los elementos excepto cabeceras y pies de tablas**

Ejemplo:

```
#caja1 {padding: 5 10 5 10px;}
```

3.4.4. Bordes

Se pueden establecer tres tipos de propiedades a los bordes: el ancho, el color y el estilo.

· Ancho.

Propiedades: **border-width, border-top-width, border-right-width, border-bottom-width, border-left-width**

Significado: **ancho de borde, ancho de borde superior, ancho de borde derecho, ancho de borde inferior y ancho de borde izquierdo**

Valores: **<numérico px,em> | thin | medium | thick | inherit**

Aplicación: **a todos los elementos**

· Color.

Propiedades: **border-color, border-top-color, border-right-color, border-bottom-color, border-left-color**

Significado: **color de borde, color de borde superior, color de borde derecho, color de borde inferior y color de borde izquierdo**

Valores: **<color> | transparent | inherit**

Aplicación: **a todos los elementos**

· Estilo.

Propiedades: **border-style, border-top-style, border-right-style, border-bottom-style, border-left-style**

Significado: **estilo de borde, estilo de borde superior, estilo de borde derecho, estilo de borde inferior y estilo de borde izquierdo**

Valores: **none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset | inherit**

Aplicación: **a todos los elementos**

Si el estilo es none o hidden no aparece borde

3.4.5. Colores y fondos

Color

Básicamente hay dos formas de especificar los colores en CSS:

- Palabras reservadas y
- Números hexadecimales en el modelo RGB.

- Palabras reservadas:

Aliceblue, antiquewhite, aqua, aquamarine, azure, beige, bisque, black, blanchedalmond, blue, blueviolet, brown, , burlywood, cadetblue, chartreuse, chocolate, coral, cornflowerblue, cornsilk, crimson, cyan, darkblue, darkcyan, darkgoldenrod, darkgray, darkgreen, darkgrey, darkkhaki, darkmagenta, darkolivegreen, darkorange, darkorchid, ...

Por ejemplo:

```
p {color: yellow}
```

- Números en el modelo RGB:

Se trata de un modelo de colores aditivo, con el que se puede obtener cualquier color mediante la mezcla de los tres colores primarios (Red, Green, Blue) en la cantidad adecuada.

Existen varios formatos posibles para cada color: hexadecimal de 3 dígitos (rgb), hexadecimal de 6 dígitos (rrggbb), decimal con 3 valores, y porcentual con 3 valores.

Por ejemplo:

```
p {color: #ff0}
p {color: #ffff00}
p {color: rgb(255,255,0)}
p {color: rgb(100%,100%,0%)}
```

Los 4 formatos son equivalentes entre sí y equivalentes al color amarillo.

Para consultar todas las equivalencias:

<http://www.w3.org/TR/2010/PR-css3-color-20101028/#svg-color>

Fondo

El fondo solamente se visualiza en la zona ocupada por el contenido y su relleno (content + padding), ya que el borde tiene su propio color y los márgenes son transparentes.

Para establecer un color o imagen de fondo a la página entera, se debe aplicar fondo al elemento <body>. En este caso todos los elementos de la página se visualizan con el mismo fondo, a menos que algún elemento tenga el suyo propio.

CSS define cinco propiedades para establecer el fondo de cada elemento:

Propiedad: **background-color**

Significado: **color de fondo**

Valores: **<color> | transparent | inherit**

Aplicación: **a todos los elementos**

El valor inicial es transparent

Propiedad: **background-image**

Significado: **imagen de fondo**

Valores: **<url ("ruta a la imagen")> | none | inherit**

Aplicación: **a todos los elementos**

El valor inicial es none

Propiedad: **background-repeat**

Significado: **sirve para repetir la imagen de fondo hasta rellenar el elemento**

Valores: **repeat | repeat-x | repeat-y | no-repeat | inherit**

Aplicación: **a todos los elementos**

El valor inicial es repeat

Propiedad: **background-position**

Significado: **indica el desplazamiento de la imagen desde la esquina superior izquierda, mediante dos valores, el desplazamiento horizontal y el desplazamiento vertical**

Valores: **(<num%> | <medida> | left | center | right | inherit) y
(<num%> | <medida> | top | center | bottom | inherit)**

Aplicación: **a todos los elementos**

El valor inicial es 0% 0%. Si uno de los dos se omite, el valor por defecto es 50%.

Propiedad: **background-attachment**

Significado: **controla el comportamiento de la imagen de fondo respecto a la scroll-bar**

Valores: **fixed | scroll | inherit**

Aplicación: **a todos los elementos**

El valor inicial es scroll

Ejemplo de propiedades background:

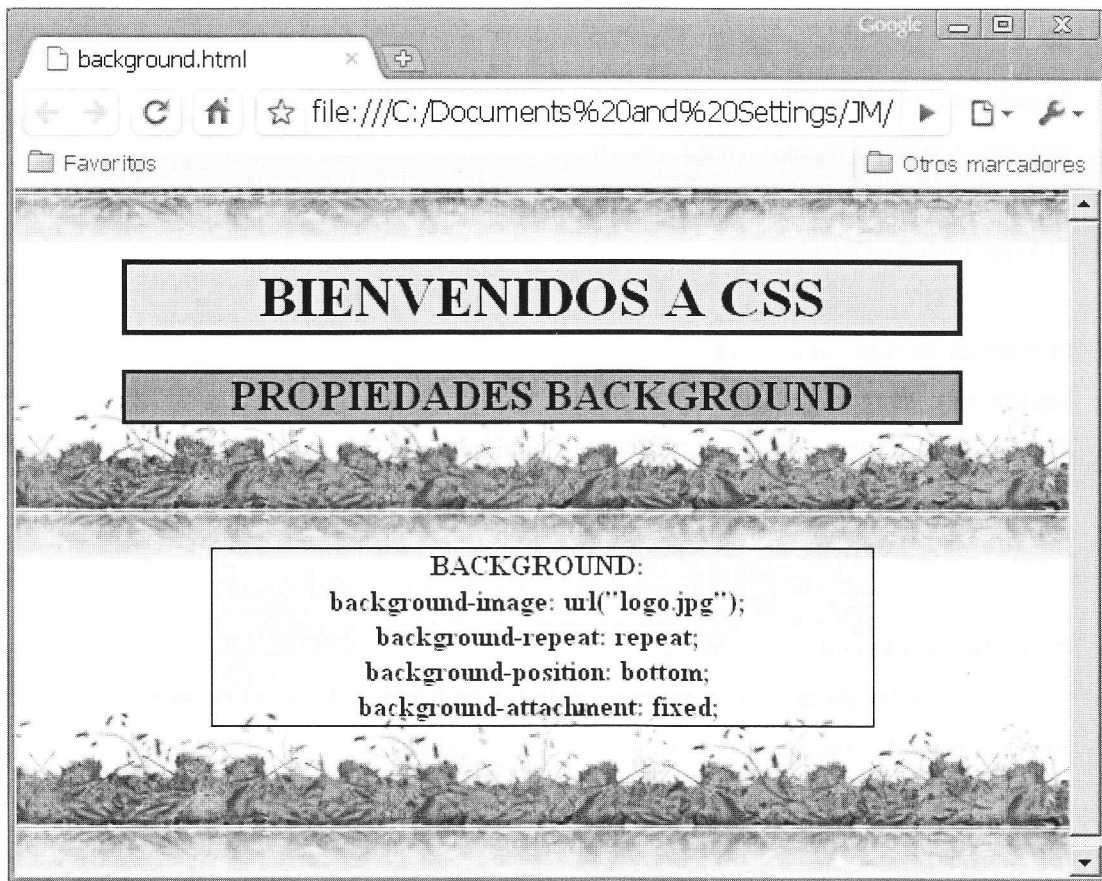


Figura 3.10: Propiedades background.

Actividad 3.3:

Crea una página similar a la Figura 3.5 con la imagen que prefieras, utilizando las propiedades que aparecen en la figura sobre el elemento <body>.

Comprueba como permanece fija la imagen de fondo aunque utilicemos la barra de desplazamiento.

(Reduce la altura de la ventana si fuera necesario para que aparezca la barra)

Las propiedades background también pueden utilizarse en formularios, incluso dentro de cada campo del formulario, como veremos más adelante.