

Diseño, Evaluación y Desarrollo de Interfaces



CSS3

Luis Rodríguez Baena (luis.rodriguez@upsam.es)

Universidad Pontificia de Salamanca
Escuela Superior de Ingeniería y Arquitectura

Introducción

- A diferencia de versiones anteriores, la especificación CSS3 no está en un único documento.
 - Está dividida en módulos que cubren cada una de las partes de la especificación.
 - Algunos de estos módulos ya están en estado de recomendación (a febrero de 2014):
 - ✓ Atributos, las API de los selectores Level 1, Media Queries, selectores, color.
 - Otros todavía no son recomendaciones oficiales, por lo que su comportamiento y sintaxis no es definitiva.

Introducción (II)

- ❑ El problema actual de CSS3 es que ningún navegador cumple 100% ni las recomendaciones ni los borradores de los módulos.
 - No se puede confiar en dejar parte importante de la funcionalidad de la página a CSS3.
- ❑ Como con HTML5 hay que probar con distintos navegadores y versiones.
- ❑ Algunos módulos interesantes todavía no tienen soporte en navegadores modernos incluso con las extensiones propietarias. Por ejemplo:
 - Retículas (grid layout), algunas propiedades de alineación y decoración de texto, propiedades de la interfaz de usuario...
- ❑ Se puede ver el soporte HTML5 de los navegadores en:
 - Can I use (caniuse.com/)
 - En algunas secciones de CSS3 de W3Cschools (www.w3schools.com/css/css3_intro.asp) se puede ver qué navegadores soportan los atributos.

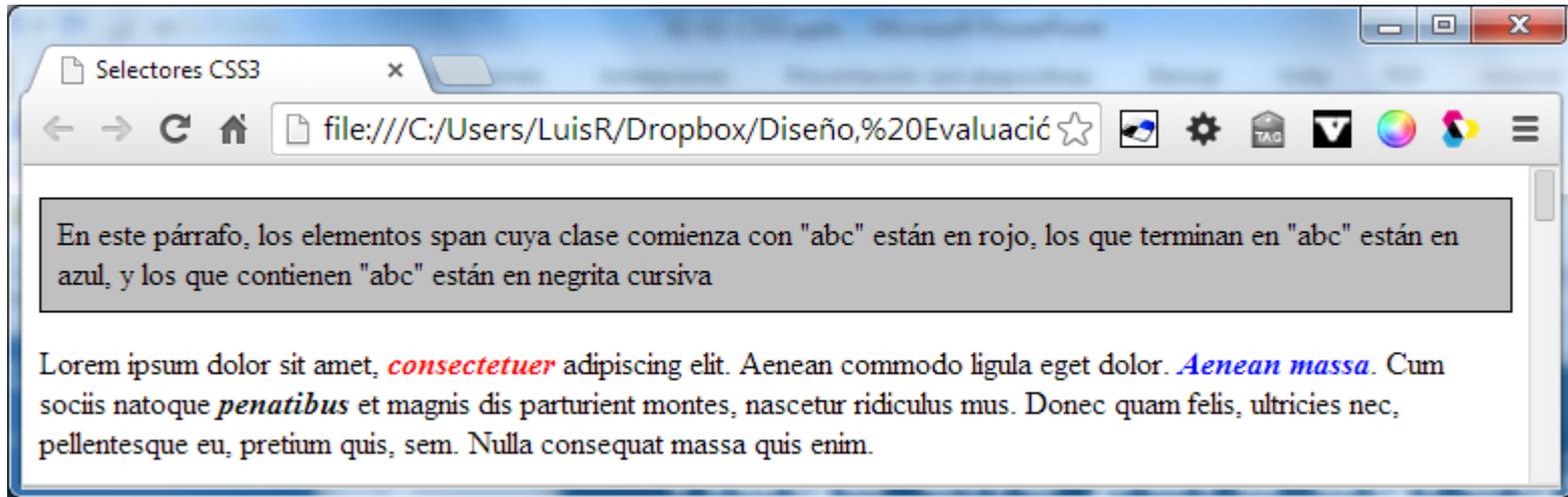
Nuevos selectores

Selectores de atributo

- ❑ El módulo correspondiente a los selectores es una recomendación oficial:
 - Selectors Level 3 (www.w3.org/TR/css3-selectors/).
- ❑ Permiten buscar a un elemento por el valor de un atributo:
 - `elemento[atributo^="valor"]`
 - ✓ Selecciona los elementos `elemento` cuyo valor del atributo `atributo` comience por "valor".
 - `elemento[atributo$="valor"]`
 - ✓ Selecciona los elementos `elemento` cuyo valor del atributo `atributo` termine por "valor".
 - `elemento[atributo*="valor"]`
 - ✓ Selecciona los elementos `elemento` cuyo valor del atributo `atributo` contenga la cadena por "valor".

Nuevos selectores

Selectores de atributo (II)



```
span[class^="abc"] {
  color: red;
}
span[class$="abc"] {
  color:blue;
}
span[class*="abc"] {
  font-style: italic;
  font-weight:bold;
}
```

```
<p>Lorem ipsum dolor sit amet, <span
class="abcxyz">consectetuer</span> adipiscing elit.
Aenean commodo ligula eget dolor. <span
class="xyzabc">Aenean massa</span>. Cum sociis natoque
<span class="xyzabcxyz">penatibus</span> et magnis
dis parturient montes, nascetur ridiculus mus. Donec
quam felis, ultricies nec, pellentesque eu, pretium
quis, sem. Nulla consequat massa quis enim. </p>
```

Nuevos selectores

Selectores de hermanos

□ CSS 2.1 incluía el selector de hermanos adyacentes:

- $E1+E2$

- ✓ Selecciona los elementos $E2$ que sean hermanos adyacentes de $E1$.

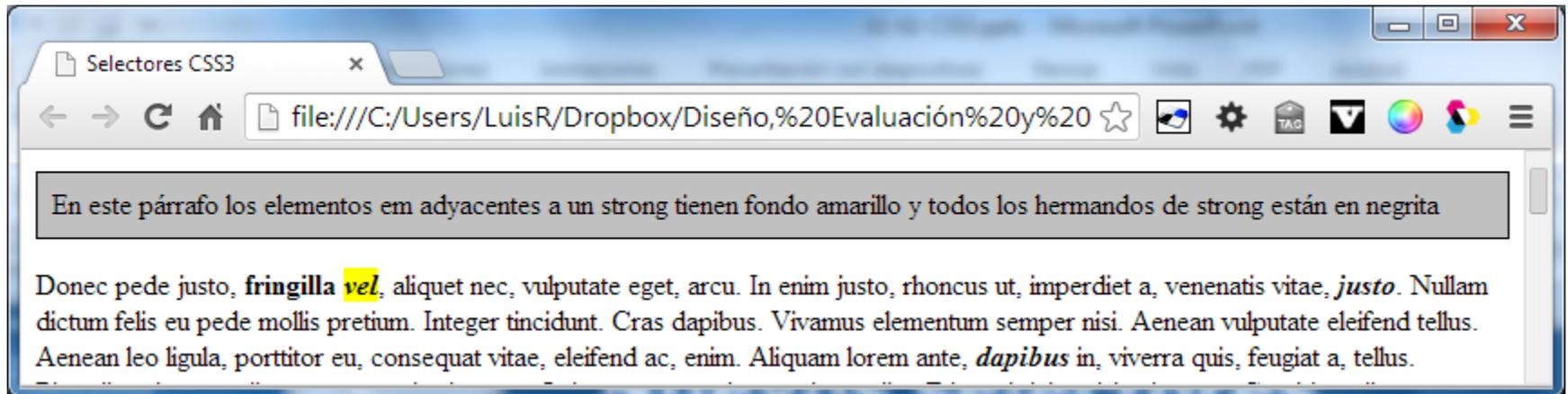
□ CSS3 tiene además el selector de hermanos.

- $E1 \sim E2$

- ✓ Selecciona todos los elementos $E2$ que sean hermanos (adyacentes o no) de $E1$.

Nuevos selectores

Selectores de hermanos (II)



```
#hermanosyadyacentes strong+em{
  background:yellow;
}

#hermanosyadyacentes strong~em{
  font-weight:bold;
}
```

```
<p id="hermanosyadyacentes">Donec pede justo,
<strong>fringilla</strong> <em>vel</em>, aliquet
nec, vulputate eget, arcu. In enim justo, rhoncus
ut, imperdiet a, venenatis vitae, <em>justo</em>.
Nullam dictum felis eu pede mollis pretium. Integer
tincidunt. Cras dapibus. Vivamus elementum semper
nisi. Aenean vulputate eleifend tellus. Aenean leo
ligula, porttitor eu,consequat vitae, eleifend ac,
enim. Aliquam lorem ante, <em>dapibus</em> in,
viverra quis, feugiat a, tellus. Phasellus viverra
nulla ut metus varius...ullamcorper ultricies nisi.
Nam eget dui.</p>
```

Nuevos selectores

Pseudo-clases dinámicas

- ❑ Una pseudo-clase es un elemento que no se puede seleccionar a partir del árbol DOM del documento o a partir de otros selectores.
 - Un enlace visitado, el primer o último hijo de un elemento, el enésimo hijo, etc.
- ❑ Algunas pseudo-clases se activan cuando se produce una acción del usuario o un cambio de estado en la interfaz:
 - Pseudo-clases dinámicas.
- ❑ La pseudo-clase `target` selecciona el destino cuando se activa un enlace con un URL con identificador de fragmento de un recurso.
 - Por ejemplo, en `http://www.servidor.com/index.html#contenido`, el identificador de fragmento sería `#contenido`.
 - `elemento:target`
 - ✓ Selecciona `elemento` cuando accedemos a él mediante un URL con un identificador de fragmento.
 - Se supone que `elemento` está identificado como destino de un URL.

Nuevos selectores

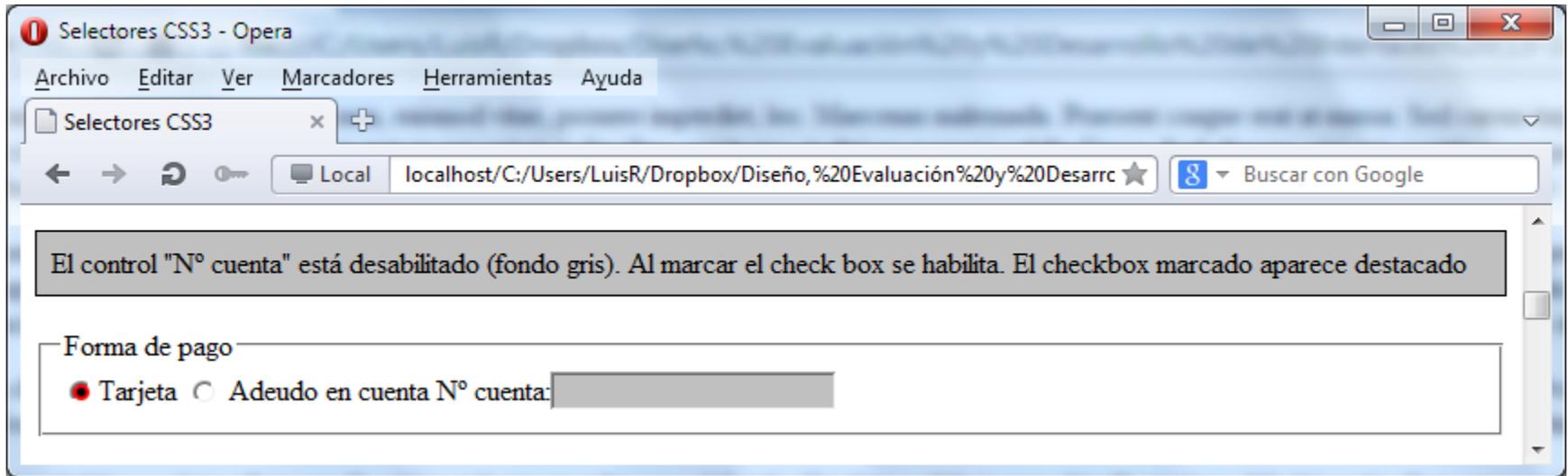
Pseudo-clases dinámicas (II)

❑ Para los controles de la interfaz de usuario:

- `elemento:disabled`
 - ✓ Selecciona el elemento de la interfaz si está desactivado.
- `elemento:enabled`
 - ✓ Selecciona el elemento de la interfaz si está activado.
- `elemento:checked`
 - ✓ Selecciona el botón de radio o el checkbox si está marcado.

Nuevos selectores

Pseudo-clases dinámicas (III)



```
input:disabled {
    background-color: silver;
}
input:enabled {
    background-color: white;
}
/* Sólo funciona en Opera */
input:checked{
    background:red;
}
```

```
<input type="radio" name="pago" id="tarjeta"
    checked="true"/><label for="tarjeta">Tarjeta</label>
<input type="radio" name="pago" id="cc"
    onclick="if(this.checked)
    {document.getElementById('nocc').disabled=false};"/>
<label for="cc">Adeudo en cuenta</label>
<label for="nocc">Nº cuenta:</label>
<input type="text" name="nocc" id="nocc"
    disabled="true" />
```

Nuevos selectores

Pseudo-clases estructurales

- ❑ Permiten la selección basándose en el árbol, pero que no se pueden representar por otros selectores.
 - `:root`
 - ✓ Representa el elemento raíz del documento.
 - ✓ En `html/xhtml` hace referencia al elemento `html`.
 - `elemento:nth-child(n)`
 - ✓ Selecciona `elemento` si es el `n`ésimo hijo de un elemento padre.
 - ✓ `n` puede ser un número entero.
 - El valor `n` indicaría todos los elementos.
 - El valor `2n` indicaría sólo los elementos pares.
 - El valor `2n+1` indicaría sólo los elementos impares.
 - ✓ Además puede tomar los valores `odd` para los hijos impares o `even` para los hijos pares.
 - `elemento:nth-of-type(n)`
 - ✓ Selecciona la `n`ésima aparición de `elemento` dentro de un elemento padre.

Nuevos selectores

Pseudo-clases estructurales (II)

```
<div>
  <p>Primer párrafo</p>
  <p>Segundo párrafo</p>
  <h2>Titulo 2</h2>
  <p>Tercer párrafo</p>
</div>
```

```
/*Selecciona el segundo hijo del
elemento padre si es un párrafo */
p:nth-child(2) {...}

/*Selecciona el segundo párrafo del
elemento padre */
p:nth-of-type(2) {...}

/* Seleccionan lo mismo */
```

```
<div>
  <h1>Titulo 1</h1>
  <p>Primer párrafo</p>
  <p>Segundo párrafo</p>
  <h2>Titulo 2</h2>
  <p>Tercer párrafo</p>
</div>
```

```
/*Selecciona el segundo hijo del
elemento padre si es un párrafo */
/*Selecciona "Primer párrafo" */
p:nth-child(2) {...}

/*Selecciona el segundo párrafo del
elemento padre */
p:nth-of-type(2) {...}
/* Seleccionan "Segundo párrafo"*/
```

Nuevos selectores

Pseudo-clases estructurales (III)

- `elemento:nth-last-child(n)`
 - Selecciona el elemento enésimo de la lista, tomando como referencia el último, siempre que sea de tipo `elemento`.
- `elemento:nth-last-of-type(n)`
 - Selecciona el elemento enésimo de tipo `elemento`, tomando como referencia el último.
- `elemento:last-child`
 - Selecciona el último elemento de tipo `elemento`.
 - ✓ Cómo en CSS 2.1 también se dispone del selector `:first-child`.

Nuevos selectores

Pseudo-clases estructurales (IV)

- ❑ `elemento:first-of-type` y `elemento:last-of-type`.
 - Seleccionan el primer y el último elemento de tipo `elemento`.
 - Harían lo mismo que `:nth-of-type(1)` y `:nth-last-of-type(1)`.
- ❑ `elemento:only-child`
 - Selecciona `elemento` si es hijo único.
- ❑ `elemento:only-of-type`
 - Selecciona `elemento` si es el único hijo de ese tipo.
- ❑ `elemento:empty`
 - Selecciona `elemento` si no tiene hijos.
- ❑ `elemento:not(selector)`
 - Selecciona los elementos de tipo `elemento` que no coincidan con el `selector`.

Nuevos selectores

Pseudo-clases estructurales (V)

El selector `li:nth-child(3)` selecciona el 3er. elemento de la lista (lo pone en negrita).

El selector `li:nth-last-child(2)` selecciona el penúltimo elemento de la lista, lo mismo que haría `li:nth-last-of-type(2)`, puesto que los dos últimos elementos son de tipo `li` (los dos van en rojo y subrayado).

El selector `li:only-child` selecciona "Elemento 2.1", ya que es el único elemento `li` que es hijo único (lo pone en azul).

El selector `dd:only-of-type` selecciona el elemento "Definición" porque es el único hijo de tipo `dd` (`dd:only-child` no seleccionaría nada puesto que el elemento `dd` no es hijo único) (lo enmarca).

El selector `ul:not(li)` selecciona a los descendientes de `ul` que no sean `li` (la lista de definición aparece en verde)

- Elemento 1
- Elemento 2
 - Elemento 2.1

- **Elemento 3**
- Elemento 4
- Elemento 5

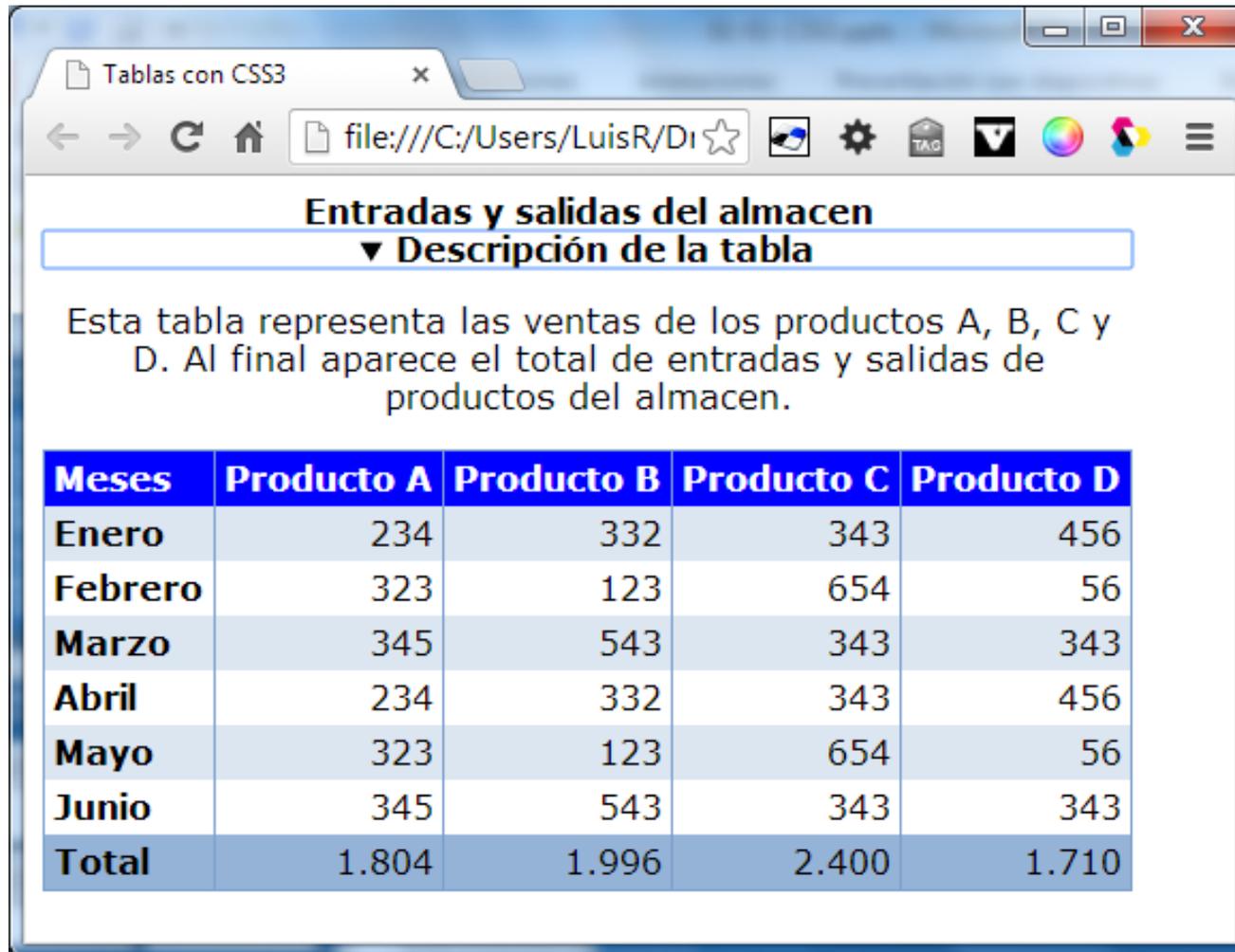
Termino de la definición

Definición

- Elemento 6
- Elemento 7
- Elemento 8

Nuevos selectores

Pseudo-clases estructurales: un ejemplo



Entradas y salidas del almacén

▼ Descripción de la tabla

Esta tabla representa las ventas de los productos A, B, C y D. Al final aparece el total de entradas y salidas de productos del almacén.

Meses	Producto A	Producto B	Producto C	Producto D
Enero	234	332	343	456
Febrero	323	123	654	56
Marzo	345	543	343	343
Abril	234	332	343	456
Mayo	323	123	654	56
Junio	345	543	343	343
Total	1.804	1.996	2.400	1.710

Nuevos selectores

Pseudo-clases estructurales: un ejemplo (II)

```
<table>
<caption>Entradas y salidas del almacen
  <details>
    <summary>Descripción de la tabla</summary>
    <p>Esta tabla representa las ventas de los productos A, B, C y D. Al final
      aparece el total de entradas y salidas de productos del almacen.</p>
  </details>
</caption>
  <tr>
    <th id="mes">Meses</th>
    <th id="prodA">Producto A</th>
    <th id="prodB">Producto B</th>
    <th id="prodC">Producto C</th>
    <th id="prodD">Producto D</th>
  </tr>
  ...
  <tr>
    <th id="tot">Total</th>
    <td headers="tot prodA">1.804</td>
    <td headers="tot prodB">1.996</td>
    <td headers="tot prodC">2.400</td>
    <td headers="tot prodD">1.710</td>
  </tr>
</table>
```

Nuevos selectores

Pseudo-clases estructurales: un ejemplo (III)

```
table {
  font-family: Verdana;
  border: #7BA0CD solid thin;
  border-collapse: collapse;
}
td, th {
  padding: 0.3em;
  border-right: #7BA0CD solid thin;
}
td {
  text-align:right;
}
th {
  text-align: left;
}
caption {
  font-weight:bold;
}
```

```
details p{
  font-weight: normal;
}
summary {
  display:block
}
tr:first-of-type{
  background: #0000ff;
  color:white;
}
tr:nth-of-type(even) {
  background-color:#DCE6F1;
}
tr:last-child{
  background-color:#95B3D7;
}
```

Nuevos selectores

Pseudo-elementos

- ❑ Crean abstracciones del árbol DOM que no están especificados por el lenguaje de marcas.
 - Permiten seleccionar a la primera línea o a la primera letra de un elemento.
 - Permiten añadir contenido no presente en el documento fuente.
- ❑ CSS 3 mantiene los mismos pseudo elementos:
`::first-letter`, `::first-line`, `::after`
y `::before`.
 - Pero en lugar de ir precedidos por el carácter dos puntos (:) van precedidos por **dos caracteres de dos puntos (::)**.

Color

□ El módulo de color es una recomendación oficial:

- CSS Color Module Level 3 (www.w3.org/TR/css3-color/).
- CSS 3 añade:
 - ✓ Una nueva especificación de colores: rgba.
 - A los colores tradicionales rgb añade el canal alpha para especificar la transparencia.
 - ✓ El modelo de color hsl (hue, saturation, lightness o tono saturación y brillo).
 - ✓ El modelo de color hsla.
 - Al igual que rgba, añade al modelo hsl el canal alpha para la transparencia.

Color

Colores rgba

- ❑ En CSS 2.1 se pueden especificar colores mediante:
 - `#rrggbb`, especifica el color indicando la cantidad de rojo, verde y azul mediante un número hexadecimal de 00 a FF.
 - `rgb(r, v, a)`, especifica el color indicando la cantidad de rojo, verde y azul mediante un número decimal de 0 a 255.
 - `rgb(%r, %v, %a)`, especifica el color mediante los porcentajes de rojo, verde y azul.
 - Estos dos últimos sistemas se pueden combinar:
 - ✓ `rgb(255, 100%, 0)` indicaría el color amarillo: 255 de rojo, 100% de verde y 0 de azul.

Color

Colores rgba (II)

□ En CSS 3 existe la especificación

`rgba(r, v, a, alpha)`.

- `r`, `v`, `a` se pueden especificar en números decimales (de 0 a 255 o porcentajes).
- `alpha` indica el nivel de transparencia con valores entre 0 y 1.
 - ✓ 0 para la máxima transparencia y 1 para la máxima opacidad.

Color

Colores rgba (III)

El texto del encabezado tiene un color rgba(255,255,255,1)

Lorem ipsum

El texto del encabezado tiene un color rgba(255,255,255,0.8)

Lorem ipsum

El texto del encabezado tiene un color rgba(255,255,255,0.5)

Lorem ipsum

Color

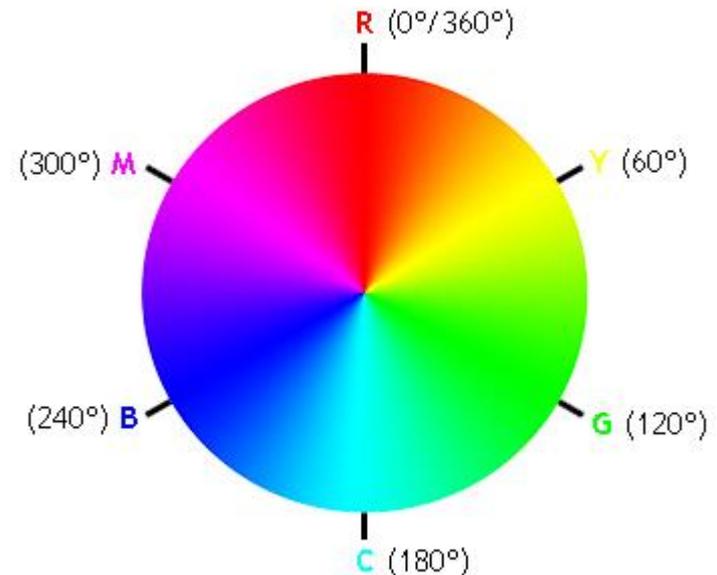
Colores hsl y hsla

- Hay que especificar en números decimales o porcentajes el tono (hue), la saturación (saturation), la luminosidad (lightness) y, opcionalmente la transparencia (canal alpha).

- `hsl(t, s, l)`
- `hsla(t, s, l, a)`

- El tono es lo que habitualmente llamamos color.

- Toma valores de 0 a 360:
 - ✓ 0 rojo.
 - ✓ 60 amarillo.
 - ✓ 120 verde.
 - ✓ 180 cian.
 - ✓ 240 azul.
 - ✓ 300 magenta.
 - ✓ 360 rojo.



Color

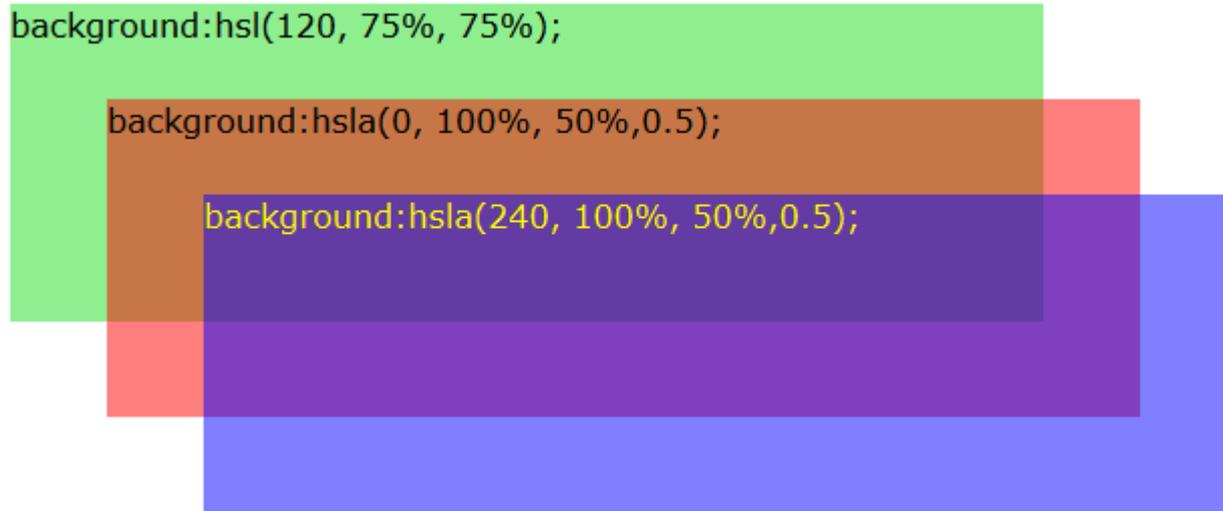
Colores hsl y hsla (II)

- ❑ La saturación indica la pureza el color.
 - Se expresa en porcentajes.
 - ✓ 100% sería el color puro.
 - ✓ El 0% sería un tono de gris.
- ❑ La luminosidad indica el brillo del color.
 - Se expresa en porcentajes.
 - ✓ El 0% sería el color negro.
 - ✓ El 100% sería la máxima luminosidad.
 - Con una saturación del 100% sería el color puro.
- ❑ Con hsla se puede especificar la transparencia.
 - Un valor entre 0 (transparente) y 1 (opaco).



Color

Colores hsl y hsla (III)



```
div:first-of-type{
  margin:1em;width:50%; height: 10em;
  background:hsl(120, 75%, 75%);
}
div:nth-of-type(2){
  position:relative;
  left:4em;top: -8em;height: 10em;
  width:50%;
  background:hsla(0, 100%, 50%,0.5);
}
```

```
div:nth-of-type(3){
  position:relative;left:7em;top:-15em;
  height: 10em;width:50%;
  background:hsla(240, 100%, 50%,0.5);
  color:hsl(60,100%,50%);
}
...
<div></div>
<div></div>
<div></div>
```

Fondos y bordes

- ❑ El módulo todavía no es una recomendación, está en estado de “última llamada” antes de la recomendación:
- ❑ CSS Backgrounds and Borders Module Level (www.w3.org/TR/css3-background/)
- ❑ A los fondos, CSS 3 añade las propiedades:
 - `background-clip`
 - `background-origin`
 - `background-size`
- ❑ Modifica las propiedades:
 - `background-repeat` (añade los valores `space` y `round`).
 - `background-attachment` (añade el valor `local`).
 - `background` (permite incluir los valores para `background-clip`, `background-origin` y `background-size`).
- ❑ A los bordes añade:
 - Propiedades para esquinas redondeadas.
 - Propiedades para bordes con imágenes.

Fondos y bordes

Fondos

- ❑ La propiedad `background-clip` permite indicar el área dónde se aplicará el fondo.
 - Toma los valores:
 - ✓ `border-box` (opción por defecto). Rellena el elemento desde el área de borde.
 - ✓ `border-padding`. Rellena el elemento desde el área de padding.
 - ✓ `border-content`. Rellena el elemento desde el área de contenido.
- ❑ La propiedad `background-origin` indica el punto de origen de la imagen de fondo.
 - Toma los mismos valores que `background-clip`.
 - ✓ La opción por defecto es `border-padding`.
- ❑ La propiedad `background-size` especifica el tamaño de la imagen.
 - Toma los valores:
 - ✓ `cover`. Adapta la imagen al tamaño del contenedor.
 - ✓ `contain`. Da a la imagen el máximo tamaño posible sin variar su relación de aspecto.
 - ✓ También puede adoptar 1 o 2 medidas, porcentajes o `auto`.

Fondos y bordes

Fondos (II)

La imagen de fondo ocupa el área de borde `background-clip:border-box`



La imagen de fondo ocupa el área de padding `background-clip:padding-box`



El relleno amarillo ocupa el área de contenido `background-clip:content-box`

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Fondos y bordes

Fondos (IV)

La imagen se adapta a las medidas del contenedor `background-size:cover`



La imagen toma el tamaño mayor posible dentro del contenedor manteniendo la relación de aspecto `background-size:contain`



La imagen toma el tamaño de 100px de ancho por 50 de alto `background-size:100px 50px`



Fondos y bordes

Fondos (V)

- ❑ `background-repeat` establece la forma en como se repite la imagen de fondo.
 - En CSS3 añade los valores (sólo los admiten actualmente Opera e IE10):
 - ✓ `round`. Rellena el contenedor con la imagen, adaptando el tamaño para utilizar la imagen completa.
 - ✓ `space`. Adapta la imagen al contenedor añadiendo espacio en blanco.
- ❑ `background-attachment` establece la forma en como se fija la imagen en el fondo (si es fija respecto a la ventana o si se mueve al hacer scroll).
 - CSS3 añade el valor `local` a la propiedad (sólo en Opera e IE10).
 - ✓ Si el contenedor de la imagen tiene barras de desplazamiento, tanto la imagen como el contenido se desplazan al hacer scroll (con el valor `scroll`, sólo se desplaza el contenido).

Fondos y bordes

Fondos (VI)

La imagen se repite por el contenedor aunque tenga que modificar sus proporciones
background-repeat: round (sólo funciona en Opera e IE10)



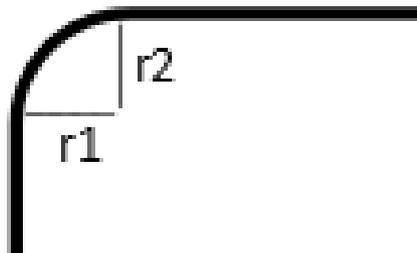
La imagen se repite por el contenedor ajustando la posición con espacios en blanco
background-repeat: round (sólo funciona en Opera e IE10)



Fondos y bordes

Bordes redondeados

- CSS3 admite bordes con esquinas redondeadas.
 - Las propiedades `border-top-left-radius`, `border-top-right-radius`, `border-bottom-right-radius`, `border-bottom-left-radius` especifican las propiedades de las cuatro esquinas.
 - ✓ Admiten 1 o 2 valores (medida o porcentajes).
 - El primer valor sería para el radio $r1$, el segundo para el radio $r2$.
 - Si sólo aparece un valor el radio sería el mismo.



Fondos y bordes

Bordes redondeados (II)

- La propiedad `border-radius` permite establecer las cuatro esquinas.
 - Admite de 1 a 4 valores.
 - ✓ 1 valor. Los mismos valores para ambos radios.
 - ✓ 4 valores. Esquina superior izquierda, superior derecha, inferior izquierda, inferior derecha para ambos radios.
 - ✓ 3 valores. Esquina superior izquierda, superior derecha y el mismo valor para las esquinas inferiores, para ambos radios.
 - ✓ 2 valores. El primer valor para las esquinas superiores y el segundo para las inferiores, para ambos radios.
 - Puede admitir otros cuatro valores separados por una barra.
 - ✓ En ese caso el primer grupo es para el radio r_1 y el segundo para el radio r_2 .

Fondos y bordes

Bordes redondeados (III)

```
border-radius:2em;
```

In hac habitasse platea dictumst. Curabitur at lacus ac velit ornare lobortis. Curabitur a felis in nunc fringilla tristique. Morbi mattis ullamcorper velit. Phasellus gravida semper nisi. Nullam vel sem. Pellentesque libero tortor, tincidunt et, tincidunt eget, semper nec, quam.

```
border-radius:3em 0;
```

Morbi nec metus. Phasellus blandit leo ut odio. Maecenas ullamcorper, dui et placerat feugiat, eros pede varius nisi, condimentum viverra felis nunc et lorem. Sed magna purus, fermentum eu, tincidunt eu, varius ut, felis.

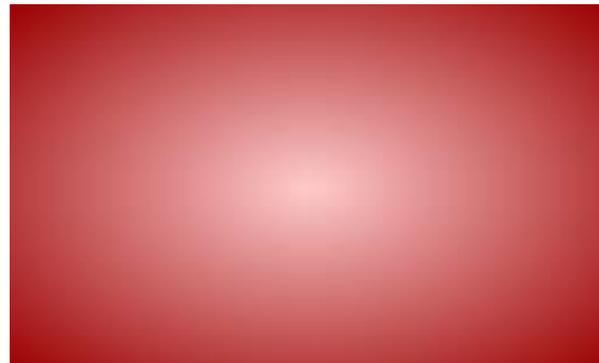
```
border-radius:25px / 50px;
```

Fusce convallis metus id felis luctus adipiscing. Pellentesque egestas, neque sit amet convallis pulvinar, justo nulla eleifend augue, ac auctor orci leo non est. Quisque id mi.

Fondos y bordes

Degradados

- ❑ La especificación de degradados es un borrador candidato a la versión final.
 - Está disponible en el módulo Image Values and Replaced Content Module Level 3 (www.w3.org/TR/css3-images/).
 - Se aplican como si fueran una imagen de fondo, como un valor de las propiedades `background` o `background-image`.
- ❑ Existen dos tipos de gradientes: el lineal y el radial.



Fondos y bordes

Degradados lineales

□ Gradientes lineales:

- `background: linear-gradient(color1, color2, ..., colorN)`
 - ✓ Hace degradado pasando desde `color1` hasta `colorN`.
 - El degradado se hace de forma continua.
 - Cada color puede llevar una medida para indicar en qué parte del bloque comienza el degradado.
 - `background: linear-gradient(gray 50%, silver)`
 - Hasta el 50% del bloque el primer color (gris) es homogéneo.
- Por defecto, el primer color ocupa la parte superior y el último la inferior.
 - ✓ Se puede añadir al comienzo:
 - Un ángulo (en unidades `deg`) que indica el ángulo del degradado (por defecto es 0).
 - Las palabras clave `to top`, `to bottom`, `to left`, `to right` para indicar la dirección del gradiente.

Fondos y bordes

Degradados lineales (II)

Gradiente lineal `background:linear-gradient(gray, silver,white)`



Gradiente lineal `background:linear-gradient(gray 50%, silver)`. El color gris no empieza a degradarse hasta el 50% del bloque



Gradiente lineal `background:linear-gradient(gray , silver 30%)`. El color plata no es continuo a partir del 30% del bloque



Fondos y bordes

Degradados lineales (III)

El degradado va de izquierda a derecha `background:linear-gradient(to right, gray , silver)`



La línea de degradado es de 45 grados `background:linear-gradient(45deg, gray 50%, silver)`



Fondos y bordes

Degradados radiales

- `background: radial-gradient(color1, ..., colorN)`
 - Hace un degradado desde *color1* hasta *colorN* pasando por todos los colores intermedios.
 - La forma del degradado se adaptaría a las dimensiones del bloque.
 - ✓ Un círculo si el ancho y el alto es igual, una elipse si son distintos.
 - El centro estaría en la posición central del bloque.
 - ✓ Añadiendo el argumento *at posición*, se puede determinar el centro del degradado.
 - *posición* puede ser un o dos medidas o la combinación de una o dos palabras clave de *top*, *bottom*, *left*, *right*, *center*.

Fondos y bordes

Degradados radiales (II)

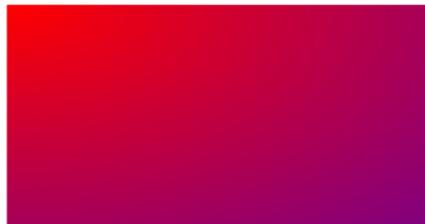
Degradado radial de rojo a púrpura. `background:radial-gradient(red , purple)`. La forma de degradado es elíptica



Degradado radial de rojo a púrpura. `background:radial-gradient(red , purple)`. La forma de degradado es circular



Degradado radial de rojo a púrpura, el centro está en la esquina superior izquierda. `background:radial-gradient(at top left, red , purple)`.



Fondos y bordes

Degradados radiales (III)

□ La forma y el tamaño del degradado se puede especificar:

- Mediante las palabras clave `circle` o `ellipse`.

- ✓ El tamaño se puede indicar mediante las palabras clave `closest-side` (el lado más cercano del centro del degradado), `farthest-side` (el lado más lejano), `closest-corner` (la esquina más cernana) o `farthest-corner` (la esquina más lejana).

- Estos argumentos también pueden aparecer sólo.

- Indicando las medidas:

- ✓ Dos medidas iguales para círculo, dos distintas para elipse.

- ✓ La medida será el tamaño final del degradado.

Fondos y bordes

Degradados radiales (IV)

Degradado radial de rojo a púrpura, `background:radial-gradient(8em 5em ,red , purple).`



Degradado radial desde una esquina a la opuesta, `background:radial-gradient(farthest-corner circle at top left,red, purple)`



Sombras

Sombras de caja

- ❑ El módulo todavía no es una recomendación.
 - Se encuentra dentro de CSS Backgrounds and Borders Module Level 3 (www.w3.org/TR/css3-background/)
- ❑ La propiedad `box-shadow` permite sombrear las cajas.
 - Admite de 2 a cuatro valores y un color.
 - ✓ Los dos primeros valores (obligatorios) marcan el desplazamiento horizontal y vertical de la sombra.
 - ✓ El tercer valor indica el tamaño del área de desenfoque.
 - ✓ El cuarto valor, si aparece, permite extender la sombra por todos los lados.
 - Opcionalmente puede llevar un primer parámetro `inset`.
 - ✓ La sombra se formaría dentro de la caja.

Sombras

Sombras de caja (II)

```
box-shadow: 0.5em 0.5em 0.3em grey;
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

```
La sombra se extiende má allá del desplazamiento box-shadow: 0.5em 0.5em 0.6em 0.8em grey;
```

Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim. Aliquam lorem ante, dapibus in, viverra quis, feugiat a, tellus. Phasellus viverra nulla ut metus varius laoreet. Quisque rutrum. Aenean imperdiet. Etiam ultricies nisi vel augue. Curabitur ullamcorper ultricies nisi. Nam eget dui.

```
Sombra interna box-shadow: inset 0em 0em 0.6em 0.6em hsl(220,100%,50%);
```

Etiam rhoncus. Maecenas tempus, tellus eget condimentum rhoncus, sem quam semper libero, sit amet adipiscing sem neque sed ipsum. Nam quam nunc, blandit vel, luctus pulvinar, hendrerit id, lorem. Maecenas nec odio et ante tincidunt tempus. Donec vitae sapien ut libero venenatis faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tincidunt. Duis leo. Sed fringilla mauris sit amet nibh. Donec sodales sagittis magna.

Sombras

Sombras de texto

- ❑ El módulo todavía es un borrador candidato a recomendación: CSS Text Decoration Module Level 3 (www.w3.org/TR/css-text-decor-3/)
 - La propiedad `text-shadow` funciona de forma similar a `box-shadow`.
 - Admite de 2 o tres valores y un color.
 - ✓ Los dos primeros valores (obligatorios) marcan el desplazamiento horizontal y vertical de la sombra.
 - ✓ El tercer valor, opcional, indica el tamaño del área de desenfoque.

```
text-shadow: 0.1em 0.1em 0.1em grey;
```

Curabitur ligula sapien

Sombras

Otros efectos con sombras de texto

- ❑ Se pueden poner varias sombras separándolas con comas.
 - En este efecto, cuanto más repitamos la sombra, mayor será el resplandor.



Tincidunt non

```
h1.resaltado {
  color:white;
  text-shadow:
    0em 0em 20px rgba(0,255,0,1),
    0em 0em 20px rgba(0,255,0,1),
    0em 0em 20px rgba(0,255,0,1);
}
```

- ❑ En este efecto,
 - Cada sombra se va desplazando abajo y a la derecha.
 - La luminosidad va disminuyendo.
 - Al final aparece una sombra negra y otra negra semitransparente.



Maecenas malesuada

```
h1.tresD {
  color: white;
  text-shadow:
    1px 1px 0px hsla(0,0%,80%,1),
    2px 2px 0px hsla(0,0%,60%,1),
    3px 3px 0px hsla(0,0%,50%,1),
    4px 4px 0px hsla(0,0%,40%,1),
    5px 5px 0px hsla(0,0%,0%,1),
    8px 8px 0px hsla(0,0%,0%,0.5);
}
```

Fuentes

@font-face

- ❑ Es una recomendación candidata:
 - CSS Fonts Module Level 3 (www.w3.org/TR/css-fonts-3/).
- ❑ Ligaduras de fuentes.
 - Fuentes que se cargan para una página determinada y que no necesitan estar instaladas en el ordenador.
 - Se cargan con la regla `@font-face`.
 - ✓ Estaba disponible en CSS2, pero hasta ahora no se ha extendido su uso.
 - Junto con los nuevos efectos de sombra, evita la necesidad de utilizar imágenes de texto.

Fuentes

@font-face (II)

- ❑ Los navegadores admiten los siguientes tipos de fuentes:
 - Fuentes true type (.ttf).
 - ✓ Desarrollado por Microsoft para los sistemas Windows.
 - Fuentes Open Type (evolución de las fuentes true type).
 - ✓ .otf. (Open Type Font). Formato base de las fuentes Open Type.
 - ✓ .eot. (Embedded Open Type). Formato compacto de las fuentes Open Type.
 - ✓ .woff (Web Open Font Format). Formato de fuente Open Type desarrollado por el W3C y optimizado para ser distribuido por un servidor.
 - Fuentes svg.
 - ✓ .svg. Scalar Vector Graphics.

Fuentes

@font-face (III)

- ❑ No todos los navegadores admiten todos los tipos de fuente (y más en versiones antiguas).
 - En caniuse.com/fontface aparece una tabla con el soporte de la regla `@font-face` y enlaces para ver la compatibilidad de fuentes con distintos navegadores.
 - ✓ En general, el formato `.woff` es el que más compatibilidad tiene con las versiones modernas de todos los navegadores de escritorio o móviles (con excepción de Opera mini).

	.eot	.ttf	.otf	.woff	.svg
IE	Versión 6.0			Versión 9.0	
Firefox		Versión 3.5	Versión 3.5	Versión 3.6	
Chrome		Versión 4.0	Versión 4.0	Versión 5.0	Versión 4.0
Opera		Versión 10.0	Versión 10.0	Versión 11.1	Versión 9.0
Safari		Versión 3.1	Versión 3.1	Versión 5.1	Versión 3.2

Fuentes

@font-face (III)

- ❑ Habría que poner una regla `@font-face` por cada tipo de letra a importar.
 - En tipografía, el tipo de letra serían las variaciones (itálica, negrita, condensada, etc.) que hay sobre una familia de fuentes.
- ❑ Sintaxis de la regla `@font-face`.

```
@font-face{
    font-family: nombreFuente;
    url: url [format(nombreFormato)]...;
    [font-style: normal | italic | oblique;]
    [font-weight: normal | bold | 100 | 200 |
        300 | 400 | 500 | 600 | 700 | 800 | 900 ;]
    [font-stretch: normal | ultra-condensed |
        extra-condensed | condensed |
        semi-condensed | semi-expanded | expanded |
        extra-expanded | ultra-expanded ;]
}
```

Fuentes

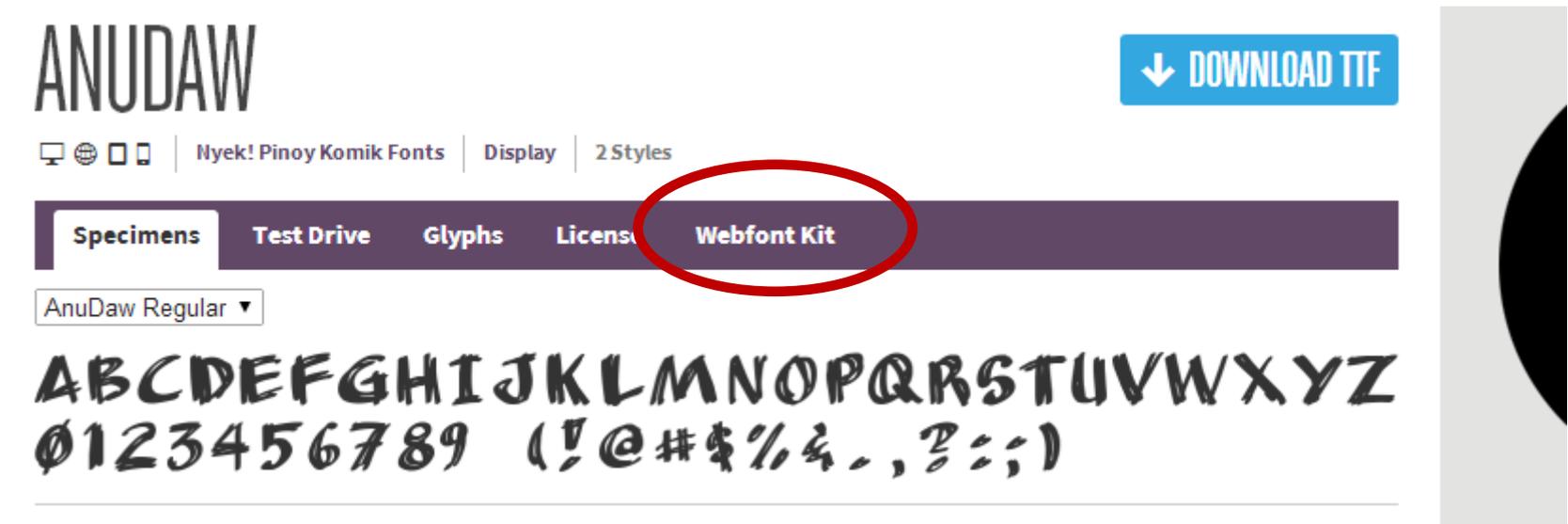
@font-face (IV)

- ❑ `font-family` sirve para especificar el nombre de la fuente que se va a dar dentro de las reglas CSS de la página.
- ❑ `url` indica dónde está almacenada la fuente y cómo se llama el archivo que la contiene.
 - Algunas tipografías pueden contener varios tipos de letra.
 - ✓ En ese caso, el `url` puede estar acompañado de `?#id`, dónde `id` sería el identificador de la fuente que se quiere cargar.
 - El `url` puede estar acompañado de una indicación del formato.
 - ✓ *nombreFormato* puede tomar los valores `"woff"`, `"truetype"`, `"opentype"`, `"embedded-opentype"` o `"svg"`.
- ❑ Se pueden añadir especificaciones sobre el estilo de la fuente, su peso o su nivel de condensado o expandido.
 - Si existen múltiples reglas `@font-face` permite indicar a qué estilo de letra corresponde esa regla.

Fuentes

@font-face (V)

- ❑ Para descargar los distintos formatos de fuentes existen algunos sitios web como Fontsquirrel (www.fontsquirrel.com/).
 - No todas las fuentes son gratuitas.
 - Para las fuentes gratuitas, permite descargarlas y generar un archivo zip con todos los estilos de la fuente, las reglas @font-face y un archivo html de prueba.



The screenshot shows the Fontsquirrel website interface for the 'ANUDAW' font. At the top left, the font name 'ANUDAW' is displayed in a large, bold, black font. To the right of the font name is a blue button with a white download icon and the text 'DOWNLOAD TTF'. Below the font name, there are several icons (desktop, globe, mobile) and the text 'Nyek! Pinoy Komik Fonts | Display | 2 Styles'. A dark purple navigation bar contains several tabs: 'Specimens', 'Test Drive', 'Glyphs', 'License', and 'Webfont Kit'. The 'Webfont Kit' tab is circled in red. Below the navigation bar, there is a dropdown menu showing 'AnuDaw Regular'. The main content area displays the font's alphabet and symbols in a large, bold, black font: 'A B C D E F G H I J K L M N O P Q R S T U V W X Y Z' on the first line, and '0 1 2 3 4 5 6 7 8 9 (! @ # \$ % & . , ? : ;)' on the second line.

Fuentes

@font-face (VI)

- ❑ Una vez establecida la regla, se puede utilizar el nombre que se ha puesto en font-family para indicar el estilo en una regla CSS.

```
@font-face{
  font-family: 'anudawregular';
  src: url(fonts/ANUDRG__-webfont.eot);
  src: url(fonts/ANUDRG__-webfont.eot?#iefix) format('embedded-opentype'),
       url(fonts/ANUDRG__-webfont.woff) format('woff'),
       url(fonts/ANUDRG__-webfont.ttf) format('truetype'),
       url(fonts/ANUDRG__-webfont.svg#anudawregular) format('svg');
  font-weight: normal;
  font-style: normal;
}

h1 {
  font-family: anudawregular;
}
```

LOREM IPSUM

Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque

Fuentes

@font-face (VII)

```
@font-face {font-family: 'anudaw';
  src: url(fonts/ANUDRG__-webfont.eot);
  src: url(fonts/ANUDRG__-webfont.eot?#iefix) format('embedded-opentype'),
  url(fonts/ANUDRG__-webfont.woff) format('woff'),
  url(fonts/ANUDRG__-webfont.ttf) format('truetype'),
  url(fonts/ANUDRG__-webfont.svg#anudawregular) format('svg');
  font-weight: normal;
  font-style: normal;}

@font-face{font-family: anudaw;
  src: url('fonts/ANUDI___-webfont.eot');
  src: url(fonts/ANUDI___-webfont.eot?#iefix) format('embedded-opentype'),
  url(fonts/ANUDI___-webfont.woff) format('woff'),
  url(fonts/ANUDI___-webfont.ttf) format('truetype'),
  url(fonts/ANUDI___-webfont.svg#anudawregular) format('svg');
  font-weight: normal;
  font-style: italic;}

h1 {font-family: anudaw;}
h2 {font-family: anudaw; font-style:italic;}
```

LOREM IPSUM

CONSECTETUER ADIPISCING ELIT

Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et

Fuentes

@font-face (VIII)

□ En el ejemplo anterior...

- El nombre de la familia y las url pueden ir con comillas o sin ellas (si se ponen se pueden utilizar espacios en blanco).
- El formato se puede poner o no.
- Puede haber varios orígenes de fuentes en un solo url, separados por comas.
- Las dos fuentes se han llamado igual (anudaw).
 - ✓ Una es para letra recta (`font-style:normal`).
 - ✓ Otra es para letra itálica (`font-style:italic`).
 - ✓ En la regla CSS, al indicar que el estilo de `h2` es en itálica, utilizará la segunda regla.

Columnas

- ❑ CSS3 trae propiedades para formatear el texto en columnas.
 - Se trata de un borrador candidato a recomendación:
 - ✓ CSS Multi-column Layout Module (www.w3.org/TR/css3-multicol/).
- ❑ Para organizar un bloque en columnas se puede optar por establecer alguna de estas propiedades:
 - Propiedad `column-count`.
 - ✓ Calcula automáticamente el ancho de columna.
 - Propiedad `column-width`.
 - ✓ Divide el ancho en tantas columnas como sea necesario.
 - ✓ El ancho puede ser algo mayor o menor para ajustarlas al ancho del contenedor.
- ❑ La propiedad `columns` permite establecer el ancho y el número de columnas.
 - Utiliza un algoritmo para intentar ajustar, en la medida de lo posible, ambos valores.

Columnas (II)

□ Espaciado entre columnas.

- El espaciado entre columnas por defecto es de aproximadamente 1em (depende del agente de usuario).
- La propiedad `column-gap` permite establecer un espaciado distinto.

Columnas (III)

- ❑ Se puede establecer un corondel (línea entre columnas).
 - `column-rule-color`. Color del corondel.
 - `column-style`. Estilo del corondel.
 - ✓ Puede tomar los valores:
 - `none`, ningún borde.
 - `hidden`, igual a `none`.
 - `dotted`, el borde es una serie de puntos.
 - `dashed`, el borde es una serie de pequeños segmentos de línea.
 - `solid`, el borde es un único segmento de línea.
 - `double`, el borde son dos líneas sólidas.
 - `groove`, el borde luce como si estuviese tallado en la página.
 - `ridge`, el borde parece que estuviera sobresaliendo de la página.
 - `inset`, el borde hace que toda la caja luzca como si estuviera empotrada en la página.
 - `outset`, el borde hace que toda la caja parezca sobresalir de la página.
 - `column-rule-width`. Ancho de la columna.
 - `column-rule`. Establece el color, estilo y ancho de la columna.

Columnas (IV)

- ❑ Las columnas, actualmente, sólo están disponibles en IE y Opera.
- ❑ CSS permite extensiones específicas de los vendedores.
 - En general utilizan las mismas propiedades con los prefijos:
 - ✓ `-moz-` para Firefox (developer.mozilla.org/en-US/docs/Web/CSS/Reference/Mozilla_Extensions).
 - ✓ `-webkit-` para Chrome y Safari (css-infos.net/properties/webkit).
 - ✓ `-o-` o `-xv-` para Opera, por versiones se puede ver en (www.opera.com/docs/specs/productspecs/).
 - ✓ `-ms-` o `-msO-` para IE (msdn.microsoft.com/en-us/library/ie/ms531209%28v=vs.85%29.aspx).
 - Si un navegador no soporta alguna característica, habría que incluir sus propiedades específicas.
 - ✓ Siempre es conveniente acabar con la propiedad estándar.

Columnas (V)

```
div:first-of-type{
  -moz-column-count: 3;
  -moz-column-gap: 3em;
  -moz-column-rule: red solid thin;
  -webkit-column-count: 3;
  -webkit-column-gap: 3em;
  -webkit-column-rule: red solid thin;
  column-count: 3;
  column-gap: 3em;
  column-rule: red solid thin;
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

ipsum. Nam quam nunc, blandit vel, luctus pulvinar, hendrerit id, lorem. Maecenas nec odio et ante tincidunt tempus. Donec vitae sapien ut libero venenatis faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tincidunt. Duis leo. Sed fringilla mauris sit amet nibh. Donec sodales

Vestibulum volutpat pretium libero. Cras id dui. Aenean ut eros et nisl sagittis vestibulum. Nullam nulla eros, ultricies sit amet, nonummy id, imperdiet feugiat, pede. Sed lectus. Donec mollis hendrerit risus. Phasellus nec sem in justo pellentesque facilisis. Etiam imperdiet imperdiet orci.

Transformaciones, transiciones, animaciones

- ❑ Las transformaciones permiten modificar el aspecto de un elemento:
- ❑ Está en estado de borrador en CSS Transforms Module Level 1 (www.w3.org/TR/css-transforms-1/).
- ❑ No las soportan los navegadores basados en webkit (Chrome y Safari).
 - Es necesario utilizar el modificador `-webkit-`.

Transformaciones, transiciones, animaciones (II)

□ Las transformaciones que se admiten son:

- Rotación, desplazamiento, escala y distorsión.
- La propiedad `transform` permite indicar el tipo o tipos de transformación.

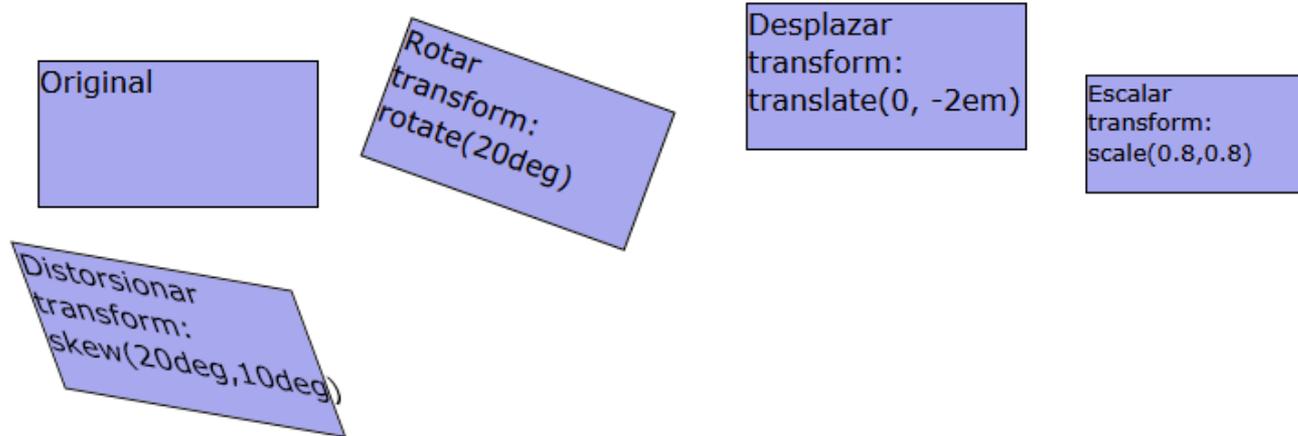
✓ Permite los valores:

- `rotate(ángulo)`. El *ángulo* se expresa en grados.
 - Por ejemplo, `rotate(45deg)` gira el elemento 45 grados.
- `translate(x[, y])`. *x* e *y* expresan el desplazamiento en unidades de longitud o porcentajes.
 - Por ejemplo, `translate(2em, 20%)`, desplaza el elemento en dos caracteres a la derecha y un 20% de su tamaño hacia abajo.
- `scale(x[, y])`. *x* e *y* son dos números que expresan la escala de ampliación.
 - Por ejemplo, `scale(0.5, 2)`, escala el elemento a la mitad de su ancho y el doble de su alto.
- `screw(ánguloX, ánguloY)`, gira el elemento en los ejes *x* e *y*.
 - Por ejemplo, `screw(10deg, 45deg)` gira el elemento 10 grados en el eje *x* y 45 en el eje *Y*.

✓ Se pueden aplicar varias transformaciones a un mismo elemento.

- `transform: rotate(45deg) scale(0.8);`

Transformaciones, transiciones, animaciones (III)



```
#transformaciones2D div:nth-of-type(2){  
  -webkit-transform: rotate(20deg);  
  transform: rotate(20deg);  
}  
  
#transformaciones2D div:nth-of-type(3){  
  -webkit-transform: translate(0, -2em);  
  transform: translate(1em, -2em);  
}
```

```
#transformaciones2D div:nth-of-type(4){  
  -webkit-transform: scale(0.8, 0.8);  
  transform: scale(0.8, 0.8);  
}  
  
#transformaciones2D div:nth-of-type(5){  
  -webkit-transform: skew(20deg, 10deg);  
  transform: skew(20deg, 10deg);  
}
```

Transformaciones, transiciones, animaciones (IV)

- ❑ Las transiciones permiten cambiar algunas propiedades CSS a lo largo de un intervalo de tiempo.
- ❑ Las transiciones están en estado de borrador en CSS Transitions.
 - www.w3.org/TR/css3-transitions/
- ❑ En las versiones modernas de navegadores no las soporta correctamente Safari.
 - Es necesario utilizar el modificador `-webkit-`.
 - ✓ Algunas todavía funcionan de forma extraña con el modificador.

Transformaciones, transiciones, animaciones (V)

- No todas las propiedades se pueden animar.
 - La lista de propiedades que se pueden cambiar se pueden encontrar en www.w3.org/TR/2013/WD-css3-transitions-20131119/#animatable-properties.
 - La propiedad `transform` puede incluir:
 - ✓ La propiedad a transformar.
 - ✓ Duración de la transformación
 - ✓ Una función de tiempo de la transformación que indica la curva de velocidad en que se produce la transformación:
 - `ease`. Comienza lento, luego rápido y termina lento. Opción por defecto.
 - `linear`. Movimiento uniforme.
 - `ease-in`. Comienzo lento.
 - `ease-out`. Final lento.
 - `ease-in-out`. Comienzo y final lento.

Transformaciones, transiciones, animaciones (VI)

□ Ejemplos:

- El elemento se desdibuja, pasando el color de fondo a transparente al pasar el cursor por encima.

```
#transiciones div:nth-of-type(1){
  -webkit-transition: background 2s, border 2s;
  transition: background 2s, border 2s;
}
#transiciones div:nth-of-type(1):hover{
  background: transparent;
  border: none;
}
```

- El elemento aumenta su tamaño y el color de fondo pasa a azul al pasar el cursor por encima.

```
#transiciones div:nth-of-type(2){
  -webkit-transition: width 2s, background 2s;
  transition: width 2s linear, background 2s ease-in;
}
#transiciones div:nth-of-type(2):hover{
  width: 50%;
  background: blue;
}
```

Transformaciones, transiciones, animaciones (VII)

□ Ejemplo:

- Al hacer clic sobre el elemento, gira y se desplaza a la derecha. Al hacer clic de nuevo vuelve a su posición también girando.

```
#transiciones div:nth-of-type(4) {
  width: 10em;
  height:10em;
  -webkit-transition: transform 5s, margin-left 5s;
  transition: transform 5s, margin-left 5s;
}
...
<div id="rueda" onclick="rodar(this);"></div>
<script>
  function rodar(elem) {
    if(elem.style.marginLeft!='40em') {
      elem.style.marginLeft='40em';
      elem.style.WebkitTransform='rotate(360deg)';
      elem.style.transform='rotate(360deg)';}
    else {
      elem.style.marginLeft='2em';
      elem.style.WebkitTransform='rotate(0deg)';
      elem.style.transform='rotate(0deg)';}
  }
</script>
```

Transformaciones, transiciones, animaciones (VIII)

- ❑ La especificación sobre animaciones también está en estado de borrador en CSS Animations.
 - www.w3.org/TR/css3-animations/
- ❑ Sólo funcionan correctamente en IE y Opera.
 - Es necesario utilizar el modificador `-webkit-` y el modificador `-moz-` para Firefox.
- ❑ Para hacer la animación:
 - Hay que especificar en el elemento los parámetros de la animación.
 - Hay que crear el “fotograma base” con el comportamiento del elemento a lo largo del tiempo.

Transformaciones, transiciones, animaciones (IX)

- ❑ Algunos de los parámetros de la animación serían:
 - Nombre de la animación.
 - ✓ `animation-name` hace referencia la regla `keyframe` que especifica la animación.
 - Duración de la animación.
 - ✓ `animation-duration` hace referencia al tiempo que va a durar la animación.
 - Número de ciclos.
 - ✓ `animation-iteration-count` indica el número de veces que se repite la animación.
 - Dirección de la animación en cada ciclo.
 - ✓ `normal`. La animación se hace en sentido normal.
 - ✓ `reverse`. La animación se hace en sentido inverso.
 - ✓ `alternate`. La animación se hace en sentido normal los ciclos impares y en sentido inverso los impares.
 - ✓ `alternate-reverse`. La animación se hace en sentido inverso los ciclos impares y en sentido normal los impares.

Transformaciones, transiciones, animaciones (X)

□ Además de especificar las características de la animación, hay que especificar los fotogramas de la animación mediante la regla `@keyframe`.

- La regla incluye el estado (valores de las propiedades) del elemento a animar en algunos puntos clave.
- Al menos hay que indicar el estado inicial y el final.

```
@keyframe nombreAnimación {  
  from { propiedades estado inicial}  
  to {propiedades estado final}  
}
```

- También se pueden indicar más pasos:

```
@keyframe nombreAnimación {  
  0%{ propiedades estado inicial}  
  25%{propiedades al 25% del tiempo de animación}  
  50%{propiedades al 50% del tiempo de animación}  
  75%{propiedades al 75% del tiempo de animación}  
  100% {propiedades estado final}  
}
```

Transformaciones, transiciones, animaciones (XI)

```
#animaciones div:nth-of-type(1){
  background: url(radios.png) center;
  -webkit-animation-name: rotar;
  -moz-animation-name: rotar;
  animation-name: rotar;
  -webkit-animation-duration: 10s;
  -moz-animation-duration: 10s;
  animation-duration: 10s;
  -webkit-animation-direction:alternate;
  -moz-animation-direction:alternate;
  animation-direction:alternate;
  -webkit-animation-iteration-count: 2;
  -moz-animation-iteration-count: 2;
  animation-iteration-count: 2;
  -webkit-transition: transform 10s;
  -moz-transition: transform 10s;
  transition: transform 10s;
}
```

```
@-webkit-keyframes rotar{
  from {left:0px;}
  to {left:500px;
      -webkit-transform: rotate(360deg);
    }
}

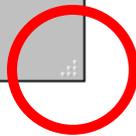
@-moz-keyframes rotar{
  from {left:0px;}
  to {left:500px;
      -moz-transform: rotate(360deg);
    }
}

@keyframes rotar{
  from {left:0px;}
  to {left:500px;
      transform: rotate(360deg);
    }
}
```

Cajas dimensionables

- ❑ La propiedad `resize` está en CSS Basic User Interface Module Level 3 (borrador).
 - www.w3.org/TR/css3-ui/#pseudo-validity
 - No las soporta ni IE ni Opera.
 - En Chrome y Safari, el tamaño mínimo de la caja lo especifican `width` y `height`.
- ❑ Para utilizar el atributo, la propiedad `overflow` debe tomar un valor distinto de `visible`:
 - `hidden`, `scroll` o `auto`.
- ❑ `resize` puede tomar los valores:
 - `none`.
 - `both`. Permite redimensionar alto y ancho de la caja.
 - `horizontal`. Permite redimensionar la caja horizontalmente.
 - `vertical`. Permite redimensionar la caja verticalmente.
- ❑ Por ejemplo:
 - `p{overflow:auto;resize:both}`

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.



Nuevo modelo de caja: box-sizing

□ En el modelo de caja estándar de CSS las propiedades `height` y `width` determinan el alto y el ancho del área de contenido.

- Al ancho total de la caja habría que sumar el ancho del padding, el ancho del borde y el ancho del margen.
- Por ejemplo

```
p {  
    width: 100px;  
    padding: 10px;  
    margin: 10 px;  
    border: solid black 1px;  
}
```

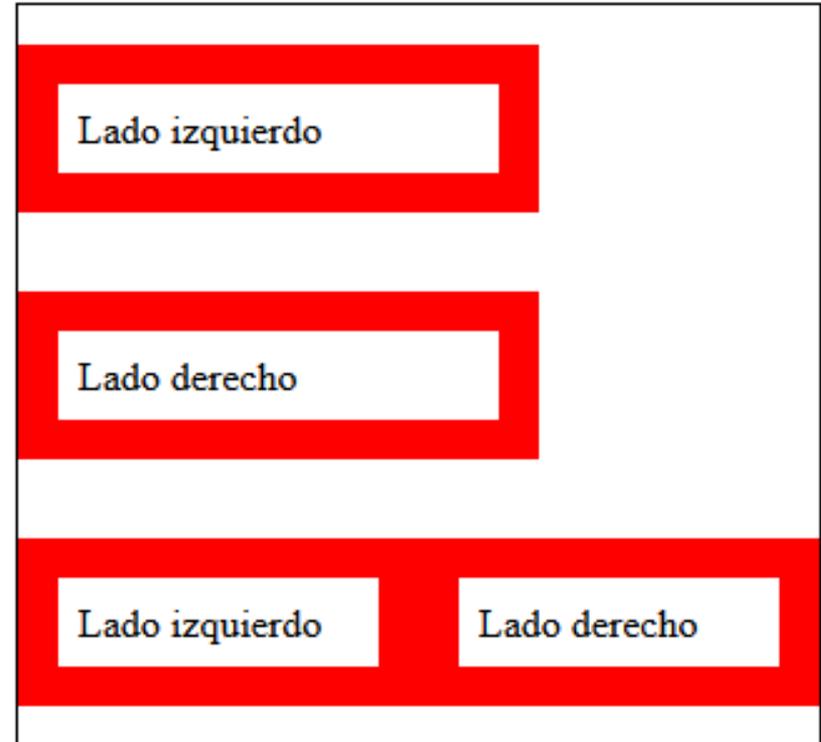
✓ Tendría un ancho de 142 pixel.

Nuevo modelo de caja: box-sizing (II)

- ❑ La propiedad `box-sizing` modifica este comportamiento.
- ❑ También está dentro de CSS Basic User Interface Module Level 3.
 - Puede tomar los valores:
 - ✓ `content-box`. Valor por omisión. La caja se comporta como en el modelo de caja tradicional.
 - ✓ `border-box`. Las medidas hacen referencia a las medidas del contenido, el padding y el borde.
 - A la propiedad `width` se le resta el ancho del borde y del padding.
 - ✓ `padding-box`. Las medidas hacen referencia a las medidas del contenido y el padding.
 - A la propiedad `width` se le resta el ancho del padding.
 - El W3C no asegura que se mantenga esta propiedad.

Nuevo modelo de caja: box-sizing (III)

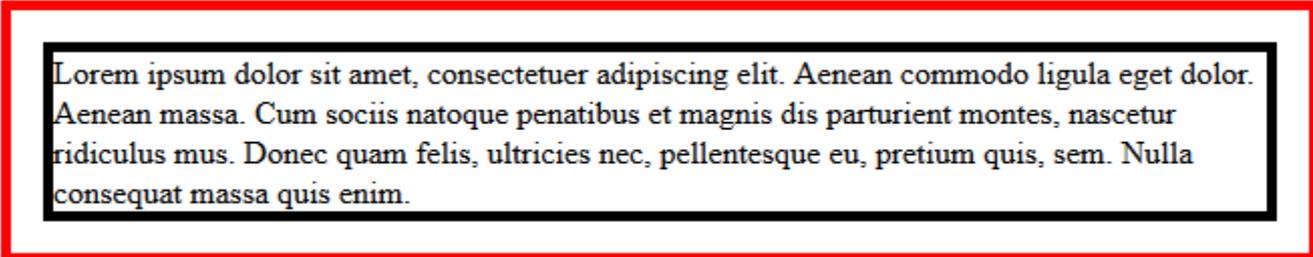
```
div.box{width:20em;
  border:thin black solid;}
.box p:nth-of-type(1), .box p:nth-of-type(2){
  width:50%;
  padding:0.5em;
  border:1em solid red;
  float:left;}
.box p:nth-of-type(3), .box p:nth-of-type(4){
  -moz-box-sizing:border-box;
  box-sizing:border-box;
  width:50%;
  padding:0.5em;
  border:1em solid red;
  float:left;}
...
<div class="box">
  <p>Lado izquierdo</p>
  <p>Lado derecho</p>
  <p>Lado izquierdo</p>
  <p>Lado derecho</p>
  <div></div>
</div>
```



Contornos

- ❑ La propiedad `outline` de CCS 2.1 permite trazar contornos alrededor de un elemento.
 - El formato es el mismo que el de la propiedad `border`.
 - El contorno se dibuja fuera del borde.
 - El contorno no computa en las dimensiones del elemento.
- ❑ CSS3 incluye una propiedad `outline-offset` que permite separar el contorno del borde.
 - También está dentro de CSS Basic User Interface Module Level 3.
 - No funciona en IE.
- ❑ Por ejemplo:

```
p.conreborde{  
  margin:2em;  
  border: black solid thick;  
  outline-offset: 1em;  
  outline: red solid thick;  
}
```



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Desbordamiento del texto

- ❑ Cuando el texto no cabe en una caja, la renderización depende de la propiedad `overflow`.
 - Puede desbordarse, recortarse o mostrar barras de desplazamiento.
- ❑ La propiedad `text-overflow` permite añadir nuevos comportamientos con los valores:
 - `clip`. Corta el texto. Igual que `overflow:hidden`.
 - `ellipsis`. Añade puntos suspensivos al texto antes de cortarlo.
 - `cadena`. Añade `cadena` al texto antes de cortarlo.
 - ✓ Sólo funciona en Firefox.
- ❑ La propiedad `text-overflow` también aparece en CSS Basic User Interface Module Level 3.
- ❑ Cuando una palabra no cabe en el contenedor, la palabra desborda la caja.
 - La propiedad `word-wrap` con el valor `break-word` hace que la palabra se corte.
 - Aparece en CSS Text Module Level 3 como último borrador.
 - ✓ www.w3.org/TR/css3-text/

Desbordamiento del texto (II)

Propiedad word-space: nowrap (para que ocupe una línea); text-overflow:clip

In hac habitasse platea dictumst. Cura

Propiedad word-space: nowrap (para que ocupe una línea); text-overflow:ellipsis

In hac habitasse platea dictumst. ...

Propiedad word-space: nowrap (para que ocupe una línea); text-overflow:" (Continua...)"

In hac habitasse platea (continua...)

Sin word-wrap

Supercalifragilisticoespialidoso

word-wrap: break-word;

Supercalifragilisticoespi
alidoso

Modelo de cajas flexible

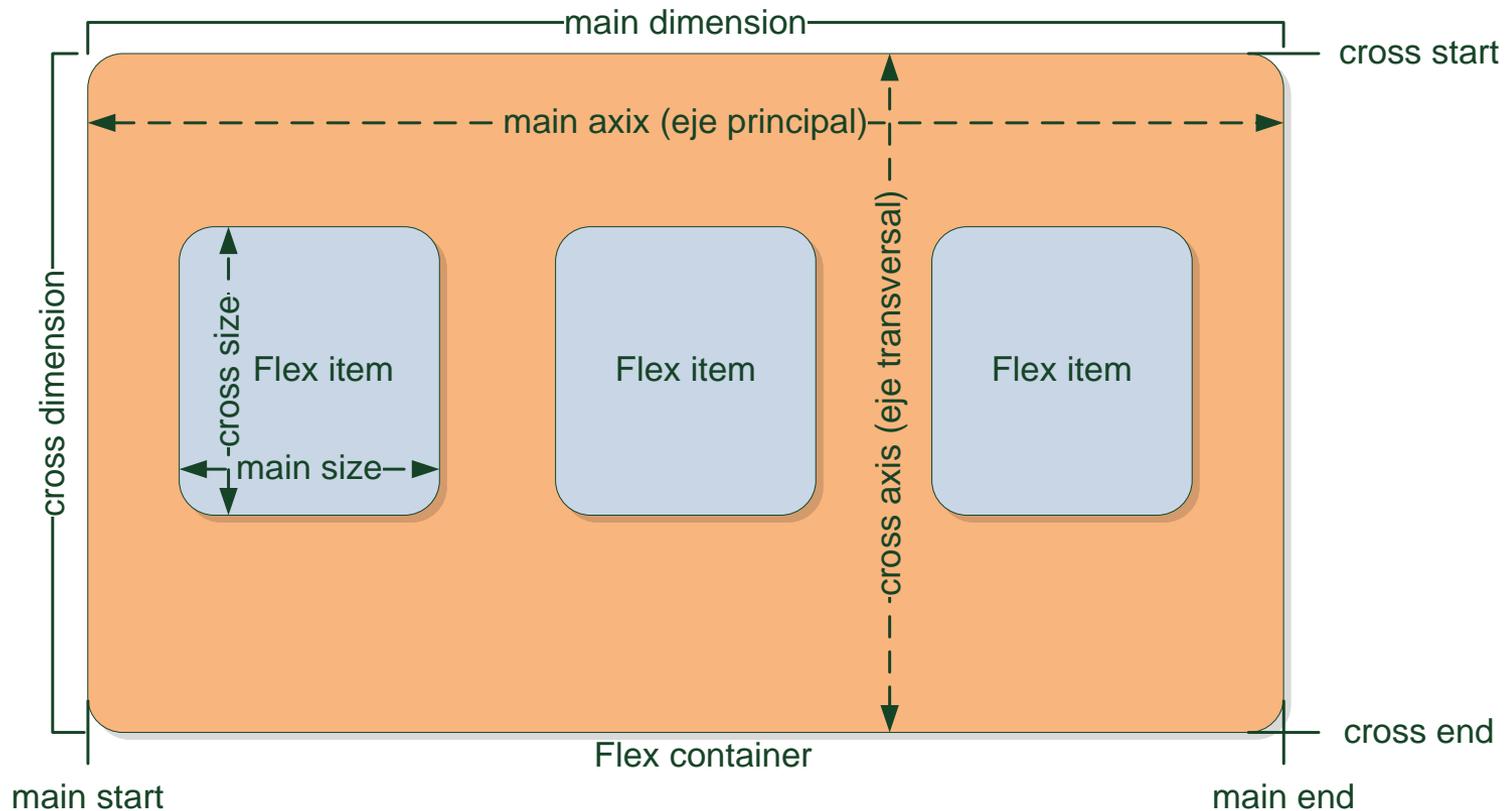
- ❑ El modelo de cajas flexible permite la crear un layout flexible de la página sin necesidad de utilizar las propiedades `width`, `float`, etc.
- ❑ Está recogido en el borrador candidato a recomendación CSS Flexible Box Layout Module.
 - www.w3.org/TR/css3-flexbox/
- ❑ Puede ser un sustituto de los framework CSS basados en retículas.
 - El problema es que sólo es soportado por las últimas versiones de los navegadores.
 - ✓ Las versiones de Safari, Opera o Chrome para iOS no la soportan.
- ❑ Se va a tratar la última sintaxis de las propiedades.
 - La sintaxis cambió en 2011.

Modelo de cajas flexible (II)

- ❑ La propiedad `display` tiene ahora dos nuevos valores:
 - `flex`. Genera una caja flexible como elemento de bloque.
 - `inline-flex`. Genera una caja flexible como elemento en línea.
- ❑ Ambos generan un contenedor *flexible*.
 - Sus dimensiones se adaptan al tamaño de la ventana.
- ❑ Los elementos hijos serán lo que se llama *flex items* que se adaptarán al contenedor padre.
- ❑ En los elementos con la propiedad `display` a `block` o `inline`, el posicionamiento se calcula a partir de las direcciones del flujo de los elementos de bloque o en línea.

Modelo de cajas flexible (III)

- En el modelo de cajas flexible, el posicionamiento se calcula a partir de las direcciones del flujo flexible.



Modelo de cajas flexible (IV)

- ❑ El eje principal es el eje a partir del cual se van a colocar los flex items.
 - Su tamaño lo fija la main dimension.
 - ✓ Los elementos que agrupa arrancan en el lado main start y terminan en el lado main end.
- ❑ El eje transversal es perpendicular al principal.
 - Su tamaño lo fija la cross dimension.
 - ✓ Los elementos que agrupa arrancan en el lado cross start y terminan en el lado cross end.
- ❑ El tamaño del flex item en el sentido del eje principal es la main size
 - La propiedad main size puede corresponder tanto a las propiedades `height` o `width` (según sea el sentido del eje principal).
- ❑ El tamaño del flex item en el sentido del eje transversal es la cross size
 - La propiedad cross size puede corresponder tanto a las propiedades `height` o `width`.

Modelo de cajas flexible (V)

- ❑ Para crear una caja flexible hay que crear un contenedor con la propiedad `display` a `flex` o `inline-flex`.
 - Crean un contenedor en el que la propiedad `float` de los hijos no interfieren en el posicionamiento y los márgenes del contenedor no se mezclan con los del contenido.
- ❑ Dentro del contenedor se situarán sus elementos flexibles.
 - Cada hijo será un elemento flexible.
 - En los elementos flexibles, las propiedades `min-width` y `min-height` admiten un valor `auto`.
 - ✓ El valor `auto` asegura que el elemento sea lo suficientemente largo como para que encaje su contenido (actualmente sólo funciona bien en Chrome y Opera).

Modelo de cajas flexible (VI)

□ Dirección y ajuste.

- La propiedad `flex-direction` del contenedor establece el sentido en que se colocan los elementos flexibles.
 - ✓ Puede tomar los valores:
 - `row` (valor por defecto). Los elementos se disponen en línea.
 - `column`. Los elementos se disponen en columna.
 - `row-reverse`. Los elementos se disponen en línea, pero en sentido inverso.
 - `column-reverse`. Los elementos se disponen en columna, pero en sentido inverso.
- La propiedad `flex-wrap` del contenedor establece el ajuste de los elementos.
 - ✓ Puede tomar los valores:
 - `nowrap` (valor por defecto). No se realiza el ajuste. Si los elementos no caben, sobrepasan la caja del contenedor.
 - `wrap`. Si los elementos no caben se ajustan en varias filas (o columnas).
 - `wrap-reverse`. Los elementos hacen ajuste, pero comienzan a ajustarse de bajo a arriba.

Modelo de cajas flexible (VII)

El contenedor tiene flex-direction:row

Flex item 1

Flex item 2

Flex item 3

El contenedor tiene flex-direction:row-reverse

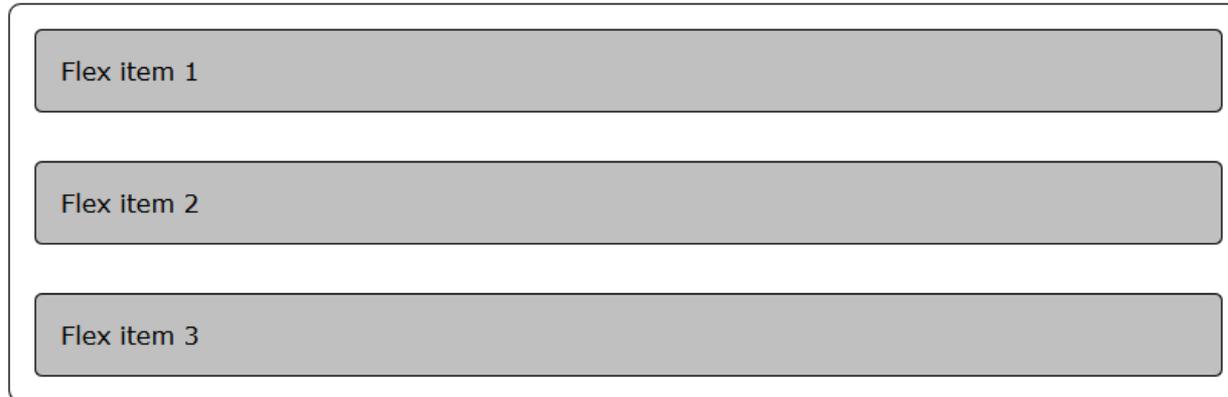
Flex item 3

Flex item 2

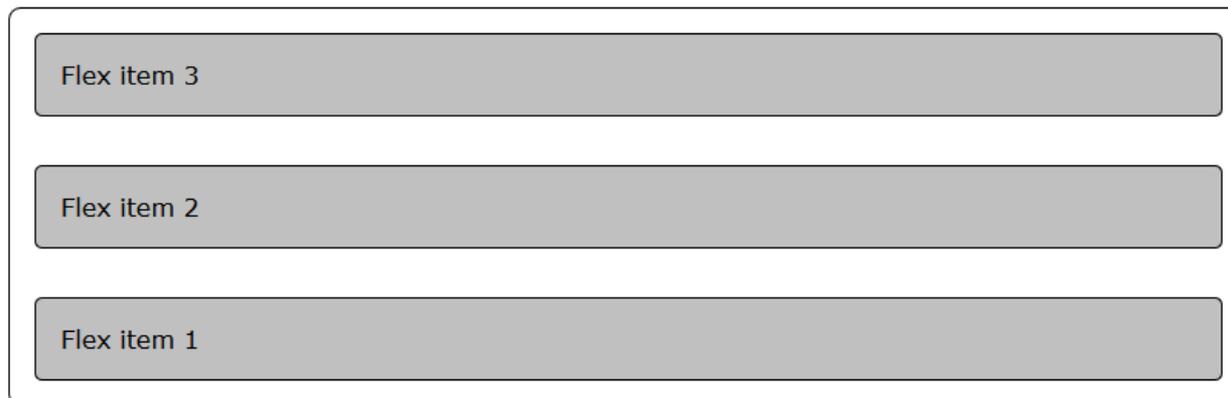
Flex item 1

Modelo de cajas flexible (VIII)

El contenedor tiene flex-direction:column

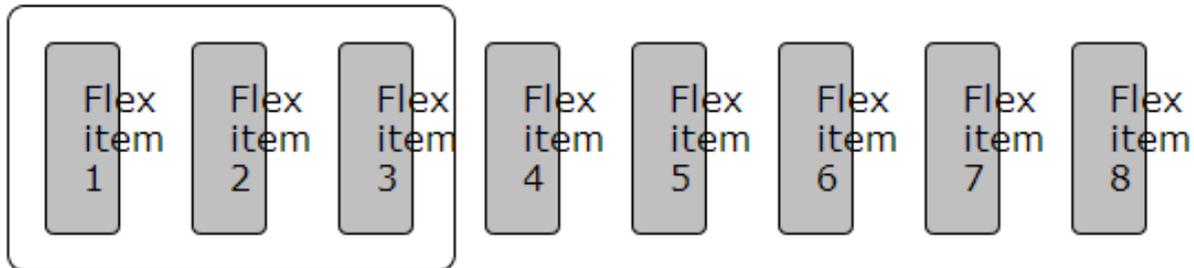


El contenedor tiene flex-direction:column-reverse

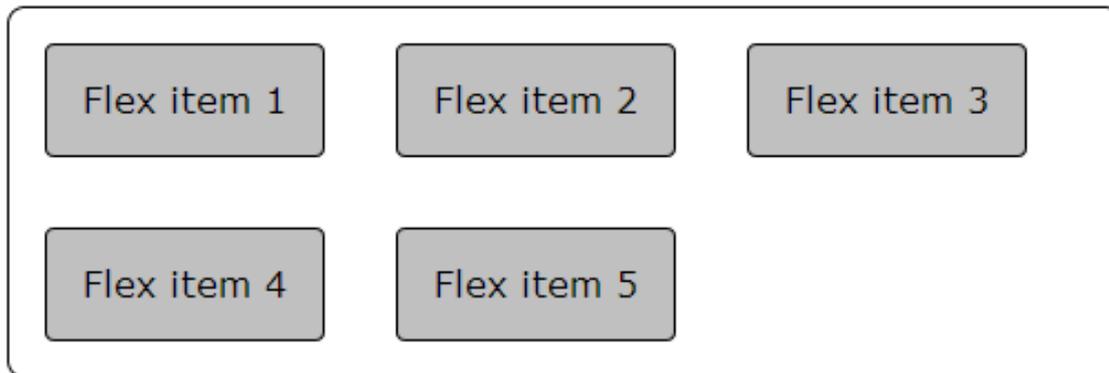


Modelo de cajas flexible (IX)

El contenedor tiene flex-wrap: nowrap

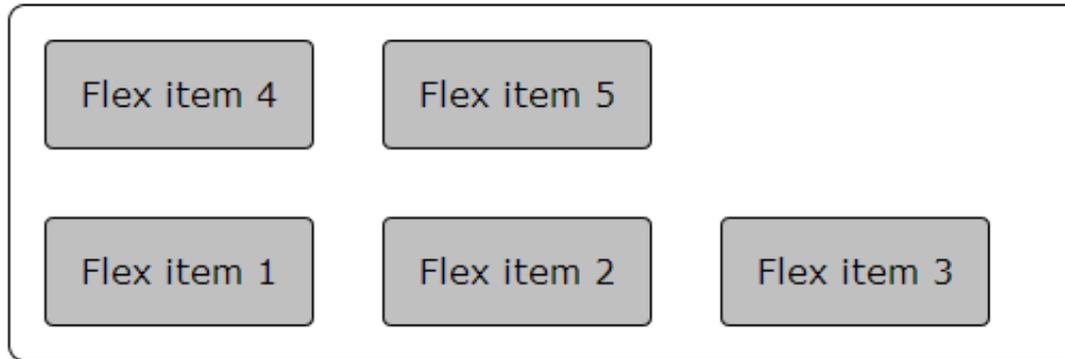


El contenedor tiene flex-wrap: wrap

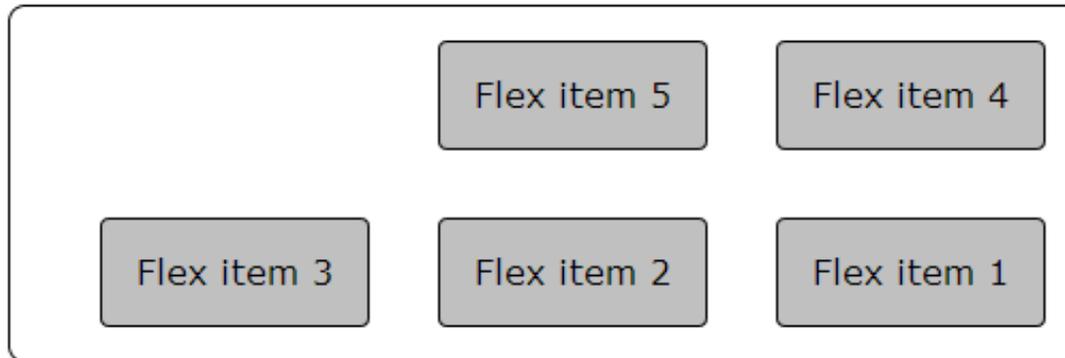


Modelo de cajas flexible (X)

El contenedor tiene flex-wrap: wrap-reverse



El contenedor tiene flex-wrap: wrap-reverse y flex-direction: row-reverse



Modelo de cajas flexible (XI)

- ❑ El orden en el que aparecen los elementos es igual al orden del código (en uno u otro sentido).
- ❑ La propiedad `order` permite cambiar el orden.
 - Sólo afecta a los medios visuales.
 - ✓ No afecta a la salida de, por ejemplo, un lector de pantalla.
 - ✓ Toma un valor entero.
 - Un valor negativo hace que “retroceda” de su posición actual.
 - Un valor 0 hace que se quede dónde le corresponde.

Se ha cambiado la propiedad `order` de los flex item

Flex item 2
order: -1

Flex item 1
order: 2

Flex item 4
order: 3

Flex item 5
order: 4

Flex item 3
order: 5

Modelo de cajas flexible

Flexibilidad de los ítems

□ La flexibilidad (el factor de aumento o disminución) de los flex items la controla la propiedad flex de dichos elementos.

● Tiene tres argumentos opcionales:

- ✓ *flex*: *factorAumento* *factorDisminución* *tamañoInicial*
 - *factorAumento* indica la proporción respecto a los otros elementos en que crecerá cuando sobre espacio en el contenedor.
 - También lo controla la propiedad `flex-grow`.
 - *factorDisminución* indica la proporción respecto a los otros elementos en que disminuirá cuando falte espacio en el contenedor.
 - También lo controla la propiedad `flex-shrink`.
 - *tamañoInicial* indica el tamaño inicial del elemento antes de distribuir el espacio.
 - También lo controla la propiedad `flex-basis`.

Modelo de cajas flexible

Flexibilidad de los ítems (II)

Si el contenedor tiene 800px de ancho y cada elemento tiene un ancho inicial de 100px (más 2 píxel del borde), el espacio sobrante serán 494 píxel.

Si todos los elementos tienen la propiedad `flex-grow` a 1, cada ítem ocupará: $494/3 + 100 = \sim 264\text{px}$.

Si todos los ítem 1 y 3 tienen la propiedad `flex-grow` a 1 y el ítem 2 a 2, el espacio sobrante se dividirá entre 4.

- El ítem 1 tendrá $(494/4) * 1 + 100 = \sim 223,5\text{px}$.
- El ítem 2 tendrá $(494/4) * 2 + 100 = \sim 347\text{px}$.
- El ítem 3 tendrá $(494/4) * 1 + 100 = \sim 223,5\text{px}$.

El contenedor tiene 800px

Flex item 1
`flex: 1 1 100px;`

Flex item 2
`flex: 2 1 100px;`

Flex item 3
`flex: 1 1 100px;`

Modelo de cajas flexible

Flexibilidad de los ítems (III)

Si el contenedor tiene 800px de ancho y cada elemento tiene un ancho inicial de 400px (más 2 píxel del borde), faltarán 406 píxel para situar los elementos.

Si todos los elementos tienen la propiedad `flex-shrink` a 1, cada ítem ocupará: $400 - (406/3) = \sim 264,4\text{px}$.

Si todos los ítem 1 y 3 tienen la propiedad `flex-shrink` a 1 y el ítem 2 a 2, el espacio sobrante se dividirá entre 4.

- El ítem 1 tendrá $400 - (406/4) * 1 = \sim 298,5\text{px}$.
- El ítem 2 tendrá $400 - (406/4) * 2 = \sim 197\text{px}$.
- El ítem 3 tendrá $400 - (406/4) * 1 = \sim 298,5\text{px}$.

El contenedor tiene 800px

Flex item 1

flex: 1 1 400px;

Flex item 2

flex: 1 2 400px;

Flex item 3

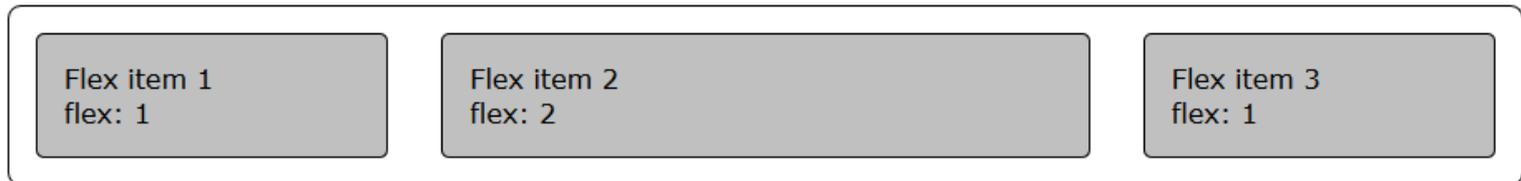
flex: 1 1 400px;

Modelo de cajas flexible

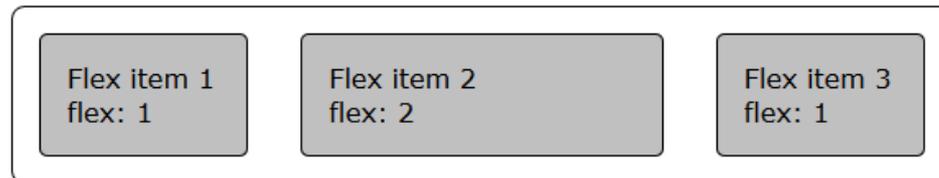
Flexibilidad de los ítems (IV)

- Un único valor numérico en la propiedad `flex` indica el factor de aumento del elemento.
 - Sería lo mismo que incluir sólo la propiedad `flex-grow`.

Divide el contenedor en 4 partes. Flex item 2 siempre ocupará el doble



Divide el contenedor en 4 partes. Flex item 2 siempre ocupará el doble



Modelo de cajas flexible

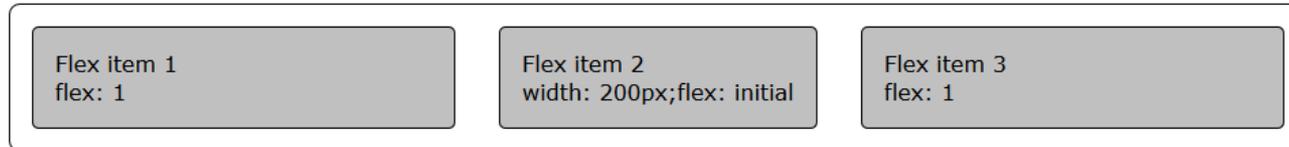
Flexibilidad de los ítems (V)

- La propiedad `flex-basis` puede tomar los valores:
 - Medida o porcentaje.
 - `initial`. Trata de mantener el valor inicial de la caja si hay espacio suficiente.
 - ✓ Si no se ha establecido una propiedad `width`, mantiene el ancho del contenido.
 - `auto`. Mantiene el elemento flexible, aumentado o disminuyendo su ancho según sobra espacio en el contenedor.
 - `none`. Es similar a `initial`, pero mantiene el tamaño si hay desbordamiento.

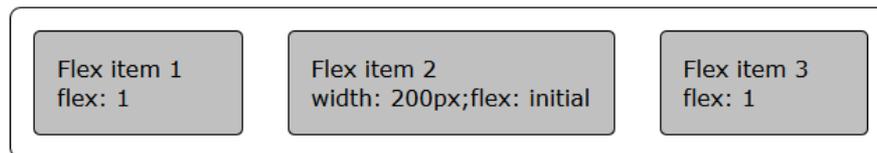
Modelo de cajas flexible

Flexibilidad de los ítems (V)

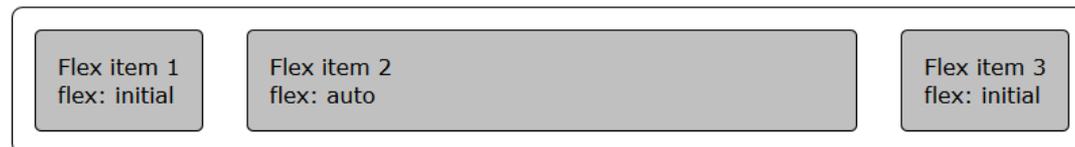
El elemento 2 trata de mantener su tamaño de 200px



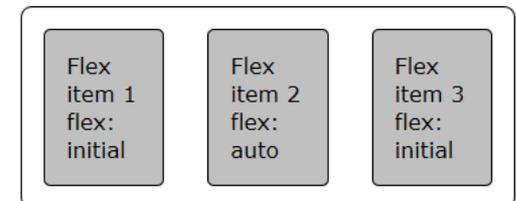
El elemento 2 trata de mantener su tamaño de 200px



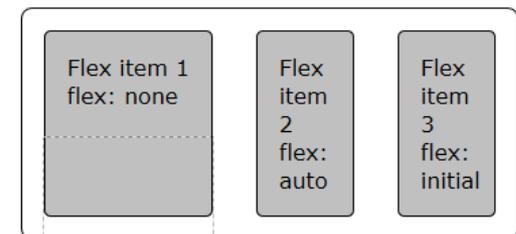
El elemento 2 modifica su tamaño, los otros tienen el ancho del contenido



El elemento 2 modifica su tamaño, los otros tienen el ancho del contenido



El elemento 2 modifica su tamaño, el elemento 1 mantiene el ancho del contenido



Modelo de cajas flexible

Alineación

- ❑ Una vez el contenedor flexible ha calculado la posición y el tamaño de los elementos se hace la alineación.
- ❑ Alineación con la propiedad `margin`.
 - El valor `auto` en el eje principal separa los elementos en varios bloques.

[Inicio](#) | [Productos](#) | [Carrito](#)

[Ayuda](#)

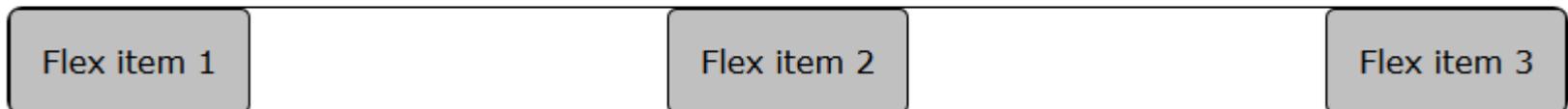
```
nav ul{
  ...
  display:flex;}
nav>ul>li#ayuda{margin-left:auto;}
<nav>
  <ul>
    <li><a href="#1">Inicio</a></li>
    <li><a href="#2">Productos</a></li>
    <li><a href="#3">Carrito</a></li>
    <li id="ayuda"><a href="#4">Ayuda</a></li>
  </ul>
</nav>
```

Modelo de cajas flexible

Alineación en el eje principal

- ❑ La propiedad `justify-content` distribuye los elemento a lo largo del eje principal.
- ❑ Puede tomar los valores:
 - `flex-start`. Alinea los elementos flexibles al comienzo del eje principal.
 - `flex-end`. Alinea los elementos flexibles al final del eje principal.
 - `center`. Centra los elementos en el eje principal.
 - `space-between`. Alinea los elementos a lo largo del eje principal. El primero y el último se alinean en los extremos del eje, dejando el espacio sobrante entre ellos.
 - `space-around`. Alinea los elementos a lo largo del eje principal, dejando el espacio sobrante alrededor de ellos.

El contenedor tiene la propiedad `justify-content: space-between`

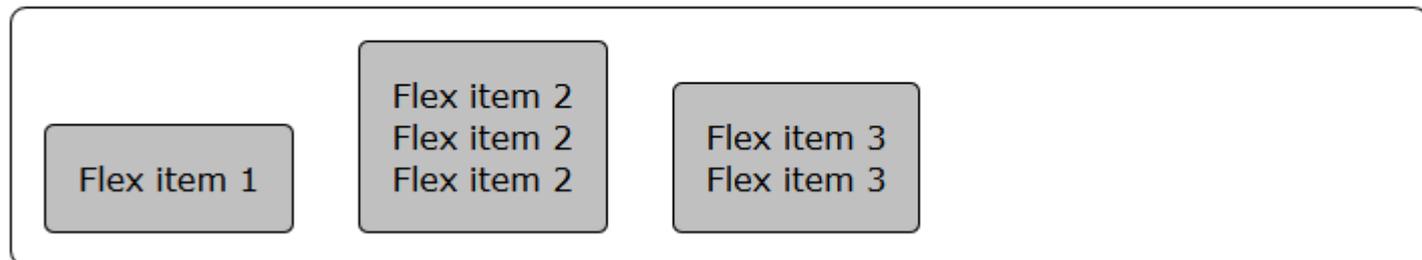


Modelo de cajas flexible

Alineación en el eje transversal

- ❑ La propiedad `align-items` permite alinear los elementos en el eje transversal.
 - Admite los valores:
 - ✓ `flex-start`. Alinea los elementos flexibles al comienzo del eje trasversal.
 - ✓ `flex-end`. Alinea los elementos flexibles al final del eje transversal.
 - ✓ `center`. Centra los elementos en el eje transversal.
 - ✓ `baseline`. Alinea los elementos a partir de la altura de los caracteres de la primera línea respecto al eje transversal.
 - ✓ `stretch`. Alarga los elementos para todos ocupen la totalidad del eje transversal. Es la opción por omisión.

El contenedor tiene la propiedad `align-items: flex-end`

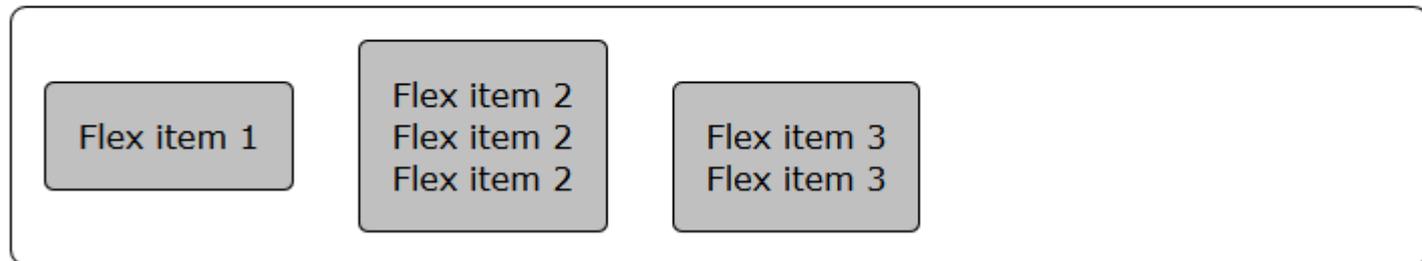


Modelo de cajas flexible

Alineación en el eje transversal (II)

- ❑ La propiedad `align-items` establece la alineación de todos los flex items de un contenedor.
- ❑ La propiedad `align-self` de un flex item permite romper esa alineación para un elemento concreto.
 - Admite los mismos valores, más el valor `auto` que establece el mismo valor que el `align-items` del padre o `stretch` si no está establecida.

El contenedor tiene la propiedad `align-items: flex-end`; el primer elemento tiene `align-self: center`

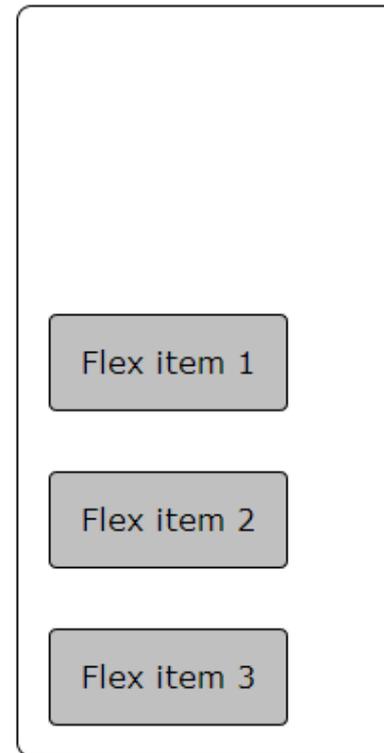


Modelo de cajas flexible

Alineación en el eje transversal (III)

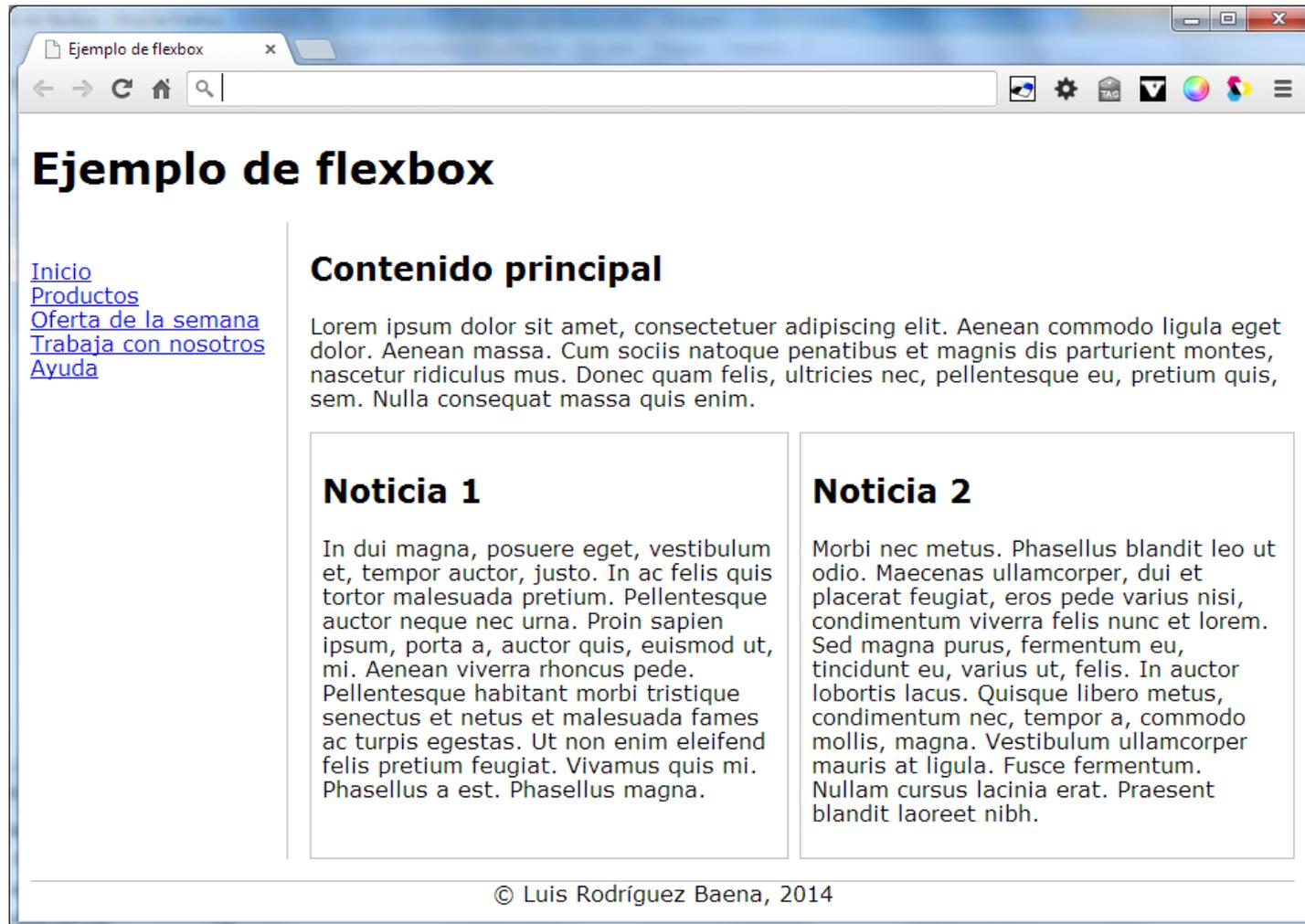
- ❑ La propiedad `align-content` hace la misma función que `justify-content`, pero en el eje vertical.
 - Sólo tiene efecto si los flex item ocupan más de una línea.
 - Toma los mismos valores que `justify-content` (`flex-start`, `flex-end`, `center`, `space-between`, `space-around`) más el valor `stretch`.
 - ✓ Con `stretch` los elementos ocupan todo el contenedor.

El contenedor tiene la propiedad `flex-wrap: wrap` y `align-content: start-end` (sólo funciona en IE y Chrome)



Modelo de cajas flexible

Ejemplo



Modelo de cajas flexible

Ejemplo (II)

```
<header>
  ...
</header>
<div id="central">
  <nav>
    <ul>
      <li><a href="/">Inicio</a></li>
    </ul>
  </nav>
  <main>
    ...
    <article>
      <section>
        ...
      </section>
      <section>
        ...
      </section>
    </article>
  </main>
</div>
<footer>
  &copy; Luis Rodr&iacute;guez Baena, 2014
</footer>
```

Modelo de cajas flexible

Ejemplo (III)

```
#central{
  display: flex;
  align-items:baseline;
  min-width: 30em;
}

nav>ul{
  padding-left:0;
  flex: 1 initial;
}

nav>ul li{
  list-style-type:none;
}

main{
  flex: 4;
  margin-left: 1em;
  padding-left: 1em;
  border-left: solid silver thin;
}
```

```
main article{
  display:flex;
}

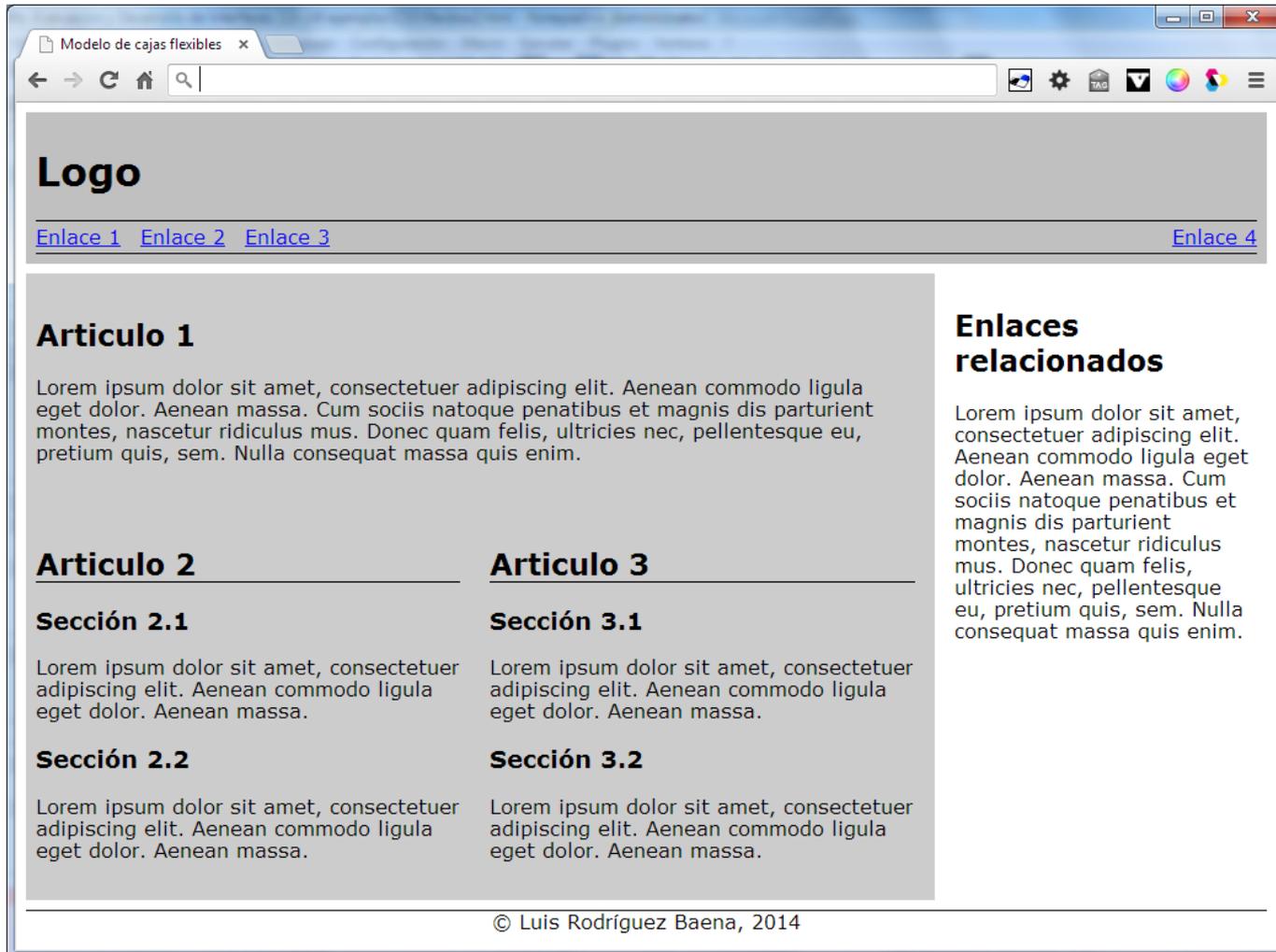
main article section {
  border: silver solid thin;
  padding:0.5em;
}

main article section+section {
  margin-left:0.5em;
}

footer {
  margin-top:1em;
  text-align: center;
  border-top: silver solid thin;
}
```

Modelo de cajas flexible

Ejemplo (IV)



Compatibilidad con los navegadores

- ❑ El problema de CSS3 es que muchas propiedades no las soportan los navegadores antiguos y algunas tampoco las últimas versiones de los navegadores.
- ❑ Esto se puede solucionar utilizando las extensiones propietarias.
 - Para garantizar la compatibilidad de alguna propiedad se pueden usar las propiedades con prefijo `-moz-` o prefijo `-webkit-`.
 - En estos casos, conviene acabar la regla con la propiedad normal.
 - Para garantizar que las columnas funcionen en un rango mayor de versiones...

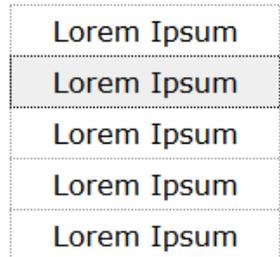
```
div {  
  -moz-column-count: 3;  
  -moz-column-gap: 3em;  
  -moz-column-rule: red solid thin;  
  -webkit-column-count: 3;  
  -webkit-column-gap: 3em;  
  -webkit-column-rule: red solid thin;  
  column-count: 3;  
  column-gap: 3em;  
  column-rule: red solid thin;  
}
```

Compatibilidad con los navegadores (II)

- ❑ En general, se trata de que la funcionalidad básica esté disponible en todos los navegadores, e ir incluyendo mejoras para versiones más modernas.
 - Ejemplo, un menú lateral, básico:

```
ul {
  border-bottom: thin dotted #999;
  list-style: none;
  margin: 1em;
  padding-left: 0;
  width: 20%;
}
li {
  background-color: #fff;
  border: thin dotted #999;
  border-bottom-width: 0;
  font: 1em Verdana, Arial, sans-serif;
}
```

```
li a {
  color: black;
  display: block;
  height: 100%;
  padding: 0.25em 0;
  text-align: center;
  text-decoration: none;
}
```



Compatibilidad con los navegadores (III)

- ❑ A partir de IE7 se admiten los nuevos selectores.
 - Se puede añadir...

```
body > ul { border-width: 0; }

ul > li {
  border: thin solid #fff;
  border-width: thin 0 0 0;
}

li > a {
  background-color: #666;
  color: white;
  font-weight: bold;
}

li:first-child a { color: yellow; }

li > a:hover { background-color: #999; }
```

Lorem Ipsum

Compatibilidad con los navegadores (IV)

- ❑ Navegadores más modernos admiten opacidad...
 - Se puede añadir...

```
li {  
  opacity: 0.9;  
}  
  
li:hover {  
  opacity: 1;  
}
```

Lorem Ipsum

Lorem Ipsum

Lorem Ipsum

Lorem Ipsum

Lorem Ipsum

Compatibilidad con los navegadores (V)

❑ Y otros sombra en el texto...

```
li a:hover {  
    text-shadow: 2px 2px 4px #333;  
}
```

Lorem Ipsum

Lorem Ipsum

Lorem Ipsum

Lorem Ipsum

Lorem Ipsum

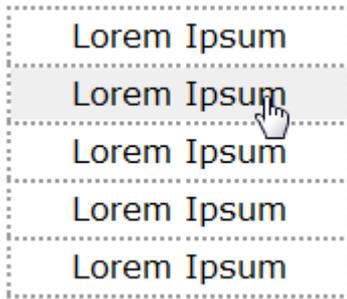
Compatibilidad con los navegadores (VI)

❑ Y si admiten los selectores de CSS3...

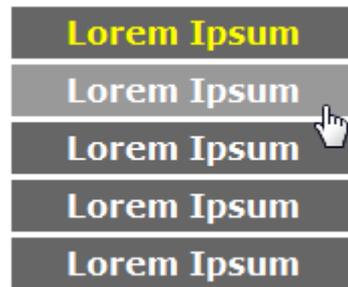
```
li:nth-child(odd) a {  
    background-color: #333;  
}  
li:nth-child(n) a:hover {  
    background-color: #aaa;  
    color: #000;  
}  
li:first-child > a:hover {  
    color: yellow;  
}
```

Lorem Ipsum

Compatibilidad con los navegadores (VII)



Con IE 6



Con IE 7



Con IE 9



Con IE 10

Compatibilidad con los navegadores

Modernizr

- ❑ Modernizr ofrece utilidades para controlar navegadores que no ofrecen CSS3.
 - Es necesario generar un script que contenga las propiedades que quiero controlar.
 1. En modernizr.com/download/ hay que seleccionar las características que queremos controlar (por ejemplo CSS Transitions).
 2. Con el botón "Generate" se genera el script sólo con las características seleccionadas.
 3. Con el botón "Download" se crea un archivo Javascript personalizado.
 4. En la página html se crea una línea que llame al script (con el URL adecuado) justo después de las especificaciones de estilo.

```
<script src="modernizr.js"></script>
```

Compatibilidad con los navegadores

Modernizr (II)

- ❑ Se pueden crear scripts para ver si el navegador soporta esas características:

```
<script>
  if (Modernizr.csstransitions){
    /* Qué hacer si el navegador admite transform */
  }
  else{
    /* Qué hacer si el navegador NO admite transform */
  }
</script>
```

Compatibilidad con los navegadores

Modernizr (III)

- ❑ El script de Modernizr también crea unas clases de forma dinámica en nuestra página html para las propiedades CSS que se quieren controlar.
 - Si el navegador soporta la propiedad, en la etiqueta html, se habría generado...

```
<html class=" js csstransitions" lang="es">
```
 - Se puede utilizar esa clase para escribir reglas para los navegadores que admitan la propiedad.
 - ✓ Si el navegador no la admite, no existirá la clase `csstransitions`.
 - La regla no tendrá efecto.
 - ✓ Si el navegador la admite, la regla tendrá efecto.

Compatibilidad con los navegadores

Modernizr (IV)

```
#menunavegacion {
  width: 25%;
  float: left;
}
#menunavegacion ul{
  list-style:none;
  padding-left: 0;
}
#menunavegacion a{
  width:75%;
  padding:0.5em;
  background: hsl(240,70%,90%);
  border: solid blue thin;
  color: black;
  text-decoration: none;
  display: block;
}
```

```
#menunavegacion a:hover, #menunavegacion
a:active, #menunavegacion a:visited,
#menunavegacion a:focus{
  text-decoration: underline;
}

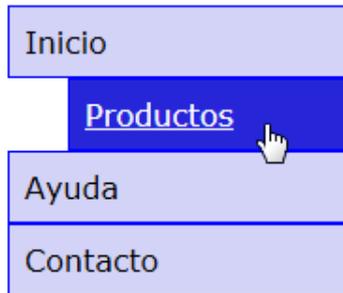
.csstransitions #menunavegacion a{
  transition: margin-left 2s,color 0s,
  background 2s;
}

.csstransitions #menunavegacion a:hover,
#menunavegacion a:active, #menunavegacion
a:visited, #menunavegacion a:focus{
  margin-left:2em;
  background: hsl(240,70%,50%);
  color: white;
}
```

Compatibilidad con los navegadores

Modernizr (V)

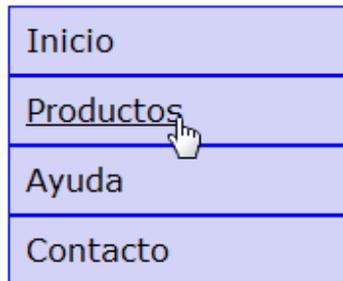
❑ Con Firefox 27.0.1.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt

❑ Con Firefox 3.0.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis