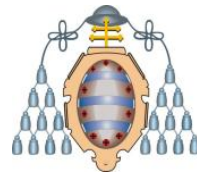


# Hojas de estilo en cascada CSS



Jose Emilio Labra Gayo  
Departamento de Informática  
Universidad de Oviedo



# Evolución

## CSS1 (recomendación 1996, fin 2008)

Propiedades de fuentes y colores, alineación de texto  
Tablas, márgenes, bordes, rellenos y posicionamiento

## CSS2 (recomendación 1998, fin 2008)

Capas (z-index), posicionamiento (relativo, absoluto, fijo)  
Texto bidireccional, tipos de medios

## CSS 2.1 (candidata 2004, recomendación 2011)

## CSS3 (en desarrollo desde 1998)

Nuevos selectores, múltiples columnas, fuentes incorporadas  
Transformaciones, transiciones y animaciones  
Diferentes perfiles: Print, TV, Mobile, etc.  
División en módulos  
Selectores, cajas, bordes, transformaciones, multicolumna, ...



# Objetivos CSS

Separar estructura (HTML) - Presentación (CSS)

**Hojas de Estilo:** Define lo que es presentación

Colores, Tipos de fuentes, márgenes, posiciones,...

**Cascada:** Preferencias incrementales

Diferentes prioridades: Navegador, Usuario, Diseñador

**Ventaja:** Reutilización

Agrupar declaraciones de estilo de varios documentos HTML

Pueden aplicarse para HTML, SVG, etc.

**Diferentes tipos de medios**

Visuales, auditivos (aural), interactivos, etc.



# Lenguaje CSS

Objetivo: sencillez

Reglas @

```
@import "impresion.css" print
```

Declaraciones

```
h2 {  
  color: blue;  
  background-color: yellow;  
}
```

Comentarios

```
/*  
  . . .  
*/
```



# Reglas @

## Añaden funcionalidad a CSS

`@charset` define el tipo de caracteres

`@import` importa una hoja de estilos externa

`@media` tipos de medio objetivo de unas declaraciones

`@font-face` tipos de fuente

`@page` tipo de página

`@keyframes` pasos de una animación



# Conjunto de caracteres

@charset define el tipo de caracteres

```
@charset "ISO-8859-1";
```

Pueden utilizarse caracteres unicode mediante \

Ejemplo, \3B1\3B2 =  $\alpha\beta$



# Importación de hojas de estilo

Permite importar otras hojas de estilo

Puede añadirse una condición sobre el tipo de medio

```
@import "http://example.com/base.css" ;  
@import "impresion.css" print ;  
@import "presenta.css" projection, tv ;
```



# Consultas de medios

Especifican tipos de medios.

Pueden ser identificadores (print, tv,...) o expresiones

Se utilizan en 3 lugares:

```
<link rel="stylesheet" href="impreso.css" media="print">
```

```
@import "impresion.css" print
```

```
@media print {  
  . . .definiciones de impresión . . .  
}  
@media screen and (min-width: 600px) {  
  . . .definiciones para pantalla . . .  
}
```





# Consultas de medios

## Tipos de medios:

`all, screen, print, projection, tv, handheld, speech, ...`

## Expresiones

Operadores: `and, not, only`

Características:

`(min-,max-) width, height, device-width, device-height`

`(min-, max-) aspect-ratio, device-aspect-ratio: fracción`

`orientation: portrait, landscape`

`(min-,max-) color: 0, no bits por color`

`(min-,max-) color-index: tamaño tabla de colores`

`(min-, max-) monochrome: no tabla`

`(min-,max-) resolution: no dpi/no dpcm`

`scan: progressive, interlace`

`grid: 0 (bitmap), 1 (grid)`



# Declaraciones CSS

Un fichero CSS está formado por una serie de reglas

Cada regla:

```
selector {  
  propiedad1: valor1;  
  propiedad2: valor2;  
  ...  
}
```

Ejemplo:

```
h2 {  
  color: blue;  
}
```



# Agrupación de reglas

Se pueden agrupar varias reglas

```
h2 {  
  color: blue;  
}
```

```
h2 {  
  background-color: yellow;  
}
```

=

```
h2 {  
  color: blue;  
  background-color: yellow;  
}
```



# Agrupación de selectores

Varios selectores separados por coma

```
h1 {  
  color: blue;  
  background-color: yellow;  
}  
  
h2 {  
  color: blue;  
  background-color: yellow;  
}
```

=

```
h1, h2 {  
  color: blue;  
  background-color: yellow;  
}
```



# Selección de elementos

Los nombres de elementos se seleccionan directamente

```
h1, h2 {  
  color: blue;  
  background-color: yellow;  
}  
  
p {  
  color: green;  
  background-color: white;  
}
```



# Selección mediante clases

Mediante `.clase` se seleccionan los elementos de clase

```
.verso {  
  margin-left: 3em;  
}
```

```
...  
<article id="golondrinas" class="poema">  
<h1>Poemario</h1>  
<p>A continuación se recitan varios poemas:</p>  
<h2>Primer poema</h2>  
<p class="verso">Volverán las oscuras golondrinas</p>  
<p class="verso">en tu balcón sus nidos a colgar,</p>  
...  
<article id="primavera" class="poema">  
...  
...
```



# Selección por identificador

Mediante `#id` se selecciona el elemento con identificador `id`

```
#golondrinas {  
  color: red;  
}
```

```
...  
<article id="golondrinas" class="poema">  
  <h1>Poemario</h1>  
  <p>A continuación se recitan varios poemas:</p>  
  <h2>Primer poema</h2>  
  <p class="verso">Volverán las oscuras golondrinas</p>  
  <p class="verso">en tu balcón sus nidos a colgar,</p>  
  ...  
<article id="primavera" class="poema">  
  ...
```



# Selección combinada

Se pueden combinar los anteriores

```
article.poema {  
  margin-left: 3em;  
}
```

```
. . .  
<article id="golondrinas" class="poema">  
<h1>Poemario</h1>  
<p>A continuación se recitan varios poemas:</p>  
<h2>Primer poema</h2>  
<p class="verso">Volverán las oscuras golondrinas</p>  
<p class="verso">en tu balcón sus nidos a colgar,</p>  
. . .  
<article id="primavera" class="poema">  
. . .
```





# Modelo de procesamiento CSS

- 1.- Crear árbol DOM
- 2.- Asociar reglas de estilo con elementos del árbol  
Anotar elementos con valores asignados a las propiedades
- 3.- Generar estructura de formateo del árbol anotado  
Cada elemento = una caja  
Anidamiento de cajas
- 4.- Visualización de la estructura de formateo  
Diferentes medios: Pantalla, impresora, lector, ...



# Herencia

Muchas propiedades son heredadas por subelementos

Pueden ser eliminadas ó modificadas (aumentar tamaño...)

Las reglas pueden obtenerse de fuentes diferente

Agente de usuario, Usuario, Autor

Pueden aparecer reglas contradictorias

Ejemplo: **Navegador**: 10px, **Autor**: 12px, **Usuario**: 16px

Algoritmo de asignación de prioridades

Agente usuario < Usuario < Autor < Autor (!) < Usuario (!)



# Selectores

Selector	Descripción	Ejemplo	Nivel
*	Todos los elementos	<code>* { border: none; }</code>	1
<code>.clase</code>	Elementos de clase	<code>.intro { ... }</code>	1
<code>#id</code>	Elementos con identificador <code>id</code>	<code>#id23 { ... }</code>	1
<code>E</code>	Elemento <code>E</code>	<code>p { ... }</code>	1
<code>E, F</code>	Elementos <code>E</code> y <code>F</code>	<code>h1, h2 { ... }</code>	1
<code>E F</code>	Elementos <code>F</code> dentro de <code>E</code>	<code>div p { ... }</code>	1
<code>E &gt; F</code>	Elementos <code>F</code> cuyo padre es <code>E</code>	<code>div &gt; p { ... }</code>	2
<code>E + F</code>	Elementos <code>F</code> precedidos de <code>E</code>	<code>h1 + p { ... }</code>	2
<code>E ~ F</code>	Elementos <code>F</code> a continuación de <code>E</code>	<code>h1 ~ p { ... }</code>	3



# Selectores con atributos

Selector	Descripción	Ejemplo	Nivel
<code>E[atr]</code>	Elementos <code>E</code> con atributo <code>atr</code>	<code>img[alt]</code>	2
<code>E[atr='v']</code>	Elementos <code>E</code> cuyo atributo <code>atr</code> tenga el valor <code>v</code>	<code>img[src='foto.jpg']</code>	2
<code>E[atr~='v']</code>	Elementos <code>E</code> cuyo atributo <code>atr</code> contenga la palabra <code>v</code>	<code>a[rel~='copyright']</code>	2
<code>E[atr^='v']</code>	Atributo <code>atr</code> comienza por <code>v</code>	<code>object[type^='image/']</code>	3
<code>E[atr\$='v']</code>	Atributo <code>atr</code> finaliza con <code>v</code>	<code>a[href\$='.html']</code>	3
<code>E[atr*='v']</code>	Atributo <code>atr</code> contiene <code>v</code>	<code>a[href*='uniovi']</code>	3
<code>E[atr ='v']</code>	Atributo <code>atr</code> empieza por <code>v</code> ó <code>v</code> seguida de guión y más cosas	<code>a[hreflang ='en']</code>	2



# Pseudo-classes

## Información que no está en el árbol DOM

Normalmente, depende de la interacción con el usuario

Se utiliza ':'

Selector	Descripción	Nivel
<code>E:link</code>	Enlaces no visitados	1
<code>E:visited</code>	Enlaces visitados	1
<code>E:active</code>	E está activado	1
<code>E:hover</code>	E está bajo el cursor	1
<code>E:focus</code>	E recibe el foco	1



# Pseudo-clases

Selector	Descripción	Nivel
<code>E:first-child</code>	E que son el primer hijo	2
<code>E:last-child</code>	E que son el último hijo	3
<code>E:only-child</code>	E que son hijos únicos	3
<code>E:first-of-type</code>	E que son el primero de ese tipo	3
<code>E:last-of-type</code>	E que son los últimos de ese tipo	3
<code>E:only-of-type</code>	E que son los únicos de ese tipo	3



# Pseudo-classes

Selector	Descripción	Nivel
<code>E:nth-child(exp)</code>	n-ésimo hijo	3
<code>E:nth-last-child(exp)</code>	n-ésimo hijo (contando desde el final)	3
<code>E:nth-of-type(exp)</code>	n-ésimo hijo de tipo E	3
<code>E:nth-last-of-type(exp)</code>	n-ésimo hijo de tipo E (contando desde el final)	3

exp puede ser `even` (par), `odd` (impar) ó `a+n` donde `a` y `b` son n<sup>o</sup>s

```
tr:nth-of-type(2n+1) { ... } = tr:nth-of-type(odd) { ... }  
tr:nth-of-type(2n+0) { ... } = tr:nth-of-type(even) { ... }
```



# Pseudo-classes

Selector	Descripción	Nivel
<code>E:lang(id)</code>	Elemento E con el idioma <code>id</code>	2
<code>:root</code>	Elemento raíz	3
<code>E:empty</code>	Elemento E sin hijos	3
<code>E:enabled</code>	Elemento E habilitado	3
<code>E:disabled</code>	Elemento E deshabilitado	3
<code>E:checked</code>	Elemento E chequeado	3
<code>E:not(s)</code>	Elemento E que no encaja con <code>s</code>	3





# Pseudo-elementos

Abstracciones de elementos que no están en el árbol.

Ejemplo: primer letra, primer párrafo, texto anterior, ...

Se utiliza '::' aunque también se admite ':'

Selector	Descripción	Nivel
E::first-letter	Primer letra	1
E::first-line	Primer línea	1
E:before	Contenido anterior	2
E:after	Contenido posterior	2



# Propiedades

## Diferentes tipos:

Formato de texto (`color`, `text-align`, ...)

Fuentes (`font-style`, `font-family`, ...)

Bordes, márgenes y relleno (`border`, `margin`, `padding`, ...)

Colores y fondo (`color`, `background-color`, ...)

Posicionamiento y dimensiones (`position`, `width`, `height`, ...)

## Propiedades abreviadas

```
margin-top: 2%;  
margin-right: 20%;  
margin-bottom: 5%;  
margin-left: 10%;
```

=

```
margin: 2% 20% 5% 10%;
```



# Características experimentales

## Características en desarrollo de las implementaciones

Empiezan por `_` y `-` seguido del identificador de la implementación

Para poder utilizar la característica sin prefijo, la implementación debe pasar todos los tests

Ejemplo: `-webkit-box-shadow`

Recomendación: utilizar nombre con prefijo y sin él

```
article.poema {  
  -webkit-box-shadow: 3px 3px 5px 6px grey;  
  -moz-box-shadow: 3px 3px 5px 6px grey;  
  box-shadow: 3px 3px 5px 6px grey;  
}
```

Problema: mantenimiento de CSS

- `-webkit`: Safari & Chrome
- `-moz`: Mozilla Firefox
- `-o`: Opera
- `-ms`: Microsoft IE
- ...



# Valores de propiedades

## Valores absolutos:

```
column-count: 3;  
color: green;
```

## URIs:

```
background-image: url("fondo.png");
```

## Unidades (distancias o tamaños)

```
margin-left: 2cm ;
```

## Colores

```
color: rgb(255,255,0)
```

## Combinaciones

```
border: 5px solid red;
```



# Unidades en CSS

Unidad	Definición
%	Porcentaje
em	Depende del tipo de fuente
px	Pixel
cm, mm, in	Centímetros, milímetros, pulgadas (in)
pt, pc	Punto (1/72 pulgadas), Pica (12 puntos)
deg	Grados ( <i>degrees</i> )



# Colores en CSS

## Mediante el nombre del color

black, silver, gray, white,  
maroon, red, purple, fuchsia,  
green, lima, olive, yellow,  
navy, blue, teal, aqua

## Componentes rgb

3 valores entre 0 y 255  
`rgb(255,214,0)`

## Valores hexadecimales

`#f00`

`#ff0000`

0 = 0, 125 = 7D, 255 = FF...



# Colores en CSS3

## HSL: Hue, Saturation, Lightness

**Hue:** ángulo del círculo de color

Valor entre 0-360

0(rojo), 120 (azul), 240 (verde), 360 (vuelve a ser rojo)

**Saturation:** porcentaje (0% = gris, 100% saturación total)

**Lightness:** porcentaje (0% negro, 50% normal, 100% blanco)

Ejemplo: `hsl(120,90%,50%)`

## Canal Alfa (Opacidad)

Valor entre 0.0 (completamente transparente) 1.0 (opaco)

`rgba(100%, 50%, 0%, 0.5)`

`hsla(120,90%,50%,0.4)`



# Formato de texto

Propiedad	Definición	Valores posibles
<b>color</b>	Color de escritura	Color
<b>text-align</b>	Alineación del texto	left, right, center, justify
<b>vertical-align</b>	Alineación vertical	baseline, top, middle, botttom, %
<b>text-indent</b>	Sangría	medida ó %
<b>text-decoration</b>	Decoración del texto	none, underline, overline, line-through, blink
<b>text-transform</b>	Transformación del texto	none, capitalize, uppercase, lowercase





# Formato de texto

Propiedad	Definición	Valores posibles
<b>text-shadow</b>	Sombreado del texto	Horizontal, Vertical, borrosidad, color Ejemplo: 5px 10px 7px gray
<b>text-stroke</b> <b>text-stroke-width</b> <b>text-stroke-color</b>	Pluma de escritura del texto	Tamaño, color Ejemplo: 5px green
<b>text-wrap</b>	Cómo dividir palabras grandes	normal, none, avoid
<b>hyphens</b>	Permitir guiones	none, manual, auto
<b>text-align-last</b>	Alineación de última línea	auto, start, end, left, right, center, justify
<b>text-decoration-style</b>	Estilo de línea de decoración	solid, double, dotted, dashed, wavy
<b>text-decoration-color</b>	Color de línea de decoración	Color



# Fuentes

Propiedad	Definición	Valores posibles
<b>font-family</b>	Familia del tipo de letra	Nombres de fuentes separados por comas Familias genéricas: serif, sans-serif, cursive, fantasy, monospace Ejemplo: font-family: 'Times New Roman', Times, Serif;
<b>font-size</b>	Tamaño del tipo de letra	xx-small, x-small, small, medium, large, x-large, xx-large, larger, smaller medida ó %
<b>font-weight</b>	Peso de la fuente	normal, bolder, lighter nº
<b>font-style</b>	Estilo del tipo de letra	normal, bold, italic



# Fuentes definidas por el usuario



@font-face permite definir fuentes propias

Se define una familia, junto con las URIs donde se puede obtener

```
@font-face {
  font-family: 'miFuente' ;
  src: url('fuentes/miFuente.eot?#ie') format('eot'),
       url('fuentes/miFuente.woff') format('woff'),
       url('fuentes/miFuente.ttf') format('truetype') ;
  font-weight: bold;
  font-style: normal;
}

h1 {
  font-family: 'miFuente', sans-serif;
  text-shadow: 4px 4px 4px #aaa;
}
```



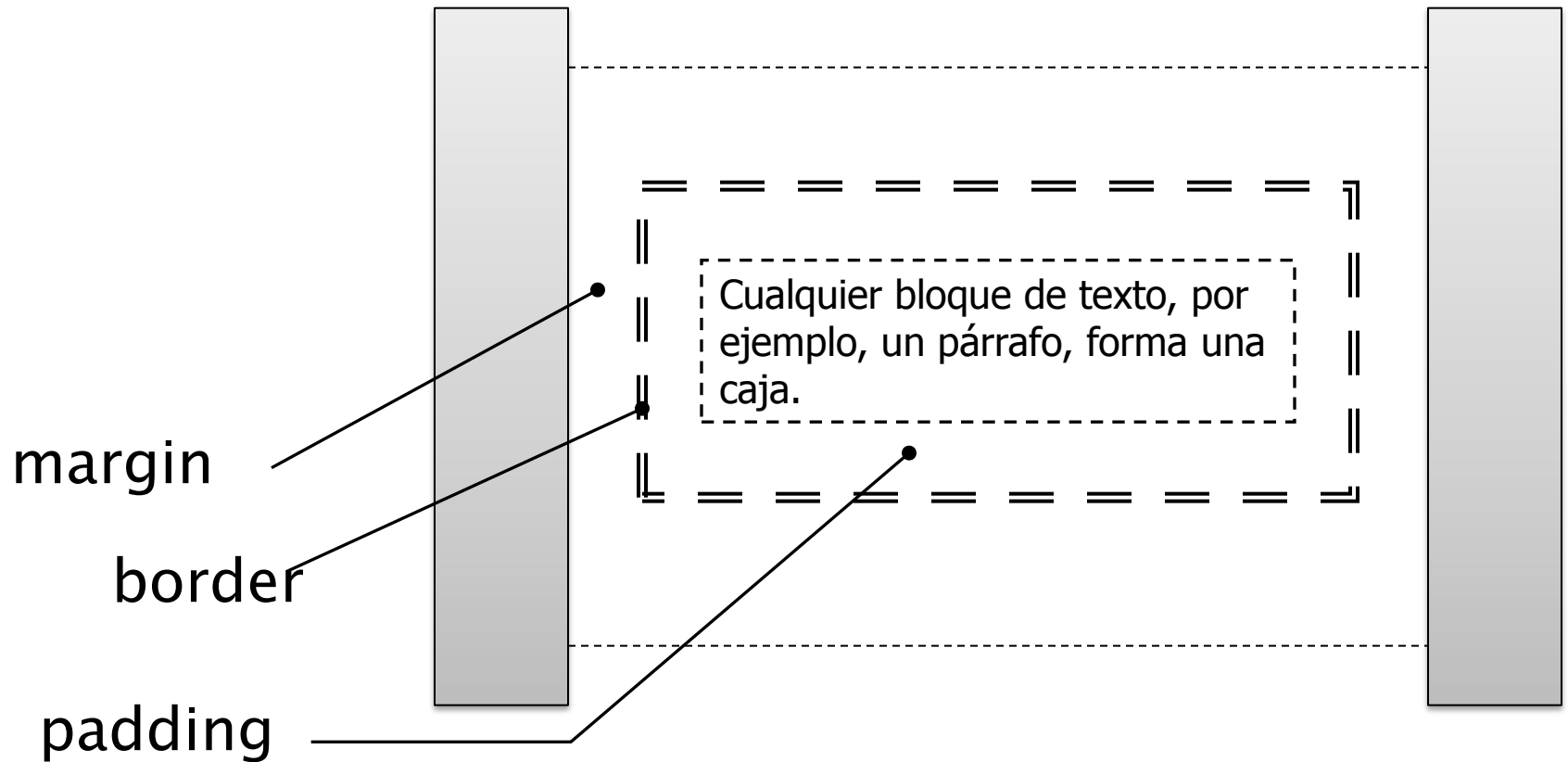
# Tipos de letra

Formatos para tipos de letra:

Formato	Nombre	Autor	Soporte
<b>TTF</b>	TrueType	Apple	IE9, FF, Webkit, Opera
<b>OTF</b>	OpenType	Adobe, Microsoft	IE9, FF, Webkit, Opera
<b>EOT</b>	Embedded OpenType	Microsoft	Sólo IE
<b>SVG</b>	Scalable Vector Graphics	W3c	Webkit, Opera
<b>WOFF</b>	Web Open Font Format	W3c	IE9, FF, Webkit, Opera



# Modelo de cajas

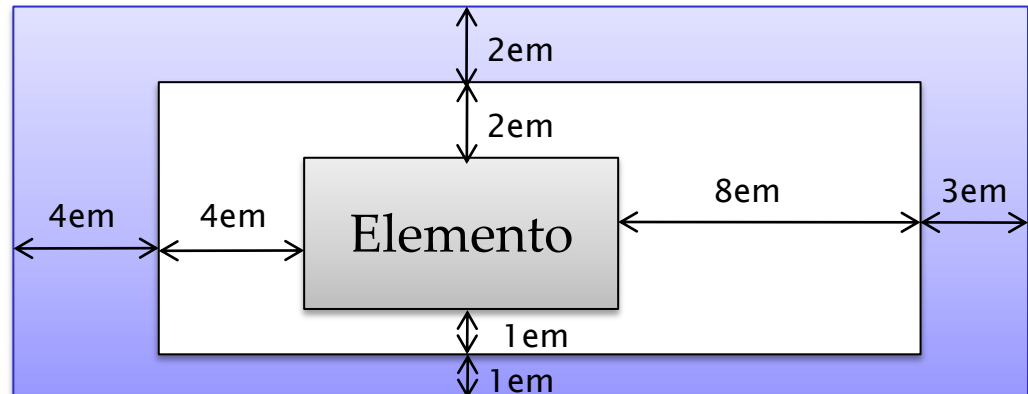




# Modelo de cajas

Propiedad	Definición	Valores posibles
<code>margin-left</code> , <code>margin-top</code> <code>margin-right</code> , <code>margin-bottom</code> <code>margin</code>	Margen	Medida o %
<code>padding-left</code> , <code>padding-top</code> <code>padding-right</code> , <code>padding-bottom</code> <code>padding</code>	Relleno	Medida o %

```
margin: 2em 3em 1em 4em;  
padding: 2em 8em 1em 4em;
```





# Bordes

Propiedad	Definición	Valores posibles
<b>border</b> <b>border-top</b> <b>border-bottom</b> <b>border-left</b> <b>border-right</b>	Define el borde	Combinada. Ejemplo: <code>1px solid red</code>
<b>border-width</b>	Grosor del borde	Medida
<b>border-style</b>	Estilo del borde	<code>none, hidden, dotted, dashed, solid, double, groove, ridge, inset, outset</code>
<b>border-color</b>	Color del borde	Color

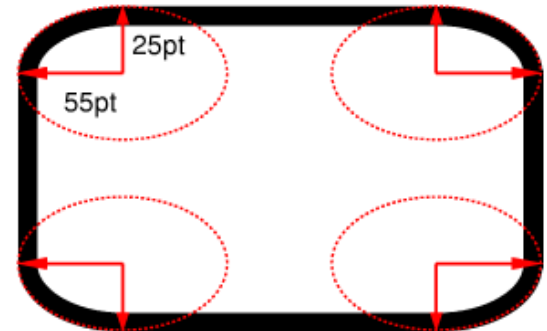


# Bordes redondeados

Propiedad	Definición	Valores posibles
<code>border-radius</code> <code>border-top-left-radius</code> <code>border-top-right-radius</code> <code>border-bottom-left-radius</code> <code>border-bottom-right-radius</code>	Bordes redondeados	Medida

```
border-top-left-radius: 55pt 25pt;
```

```
• • •
```

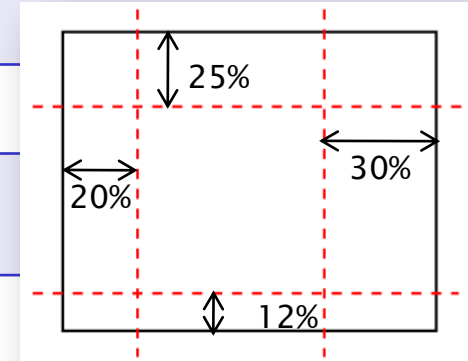






# Bordes con imágenes

Propiedad	Definición	Valores posibles
<b>border-image</b>	Utilizar una imagen como borde	Combinado de: source, slice, width, outset, repeat
<b>border-image-source</b>	Fuente de la imagen	URI
<b>border-image-slice</b>	Tamaño de los trozos de la imagen	25% 30% 12% 20%
<b>border-image-width</b>	Ancho de la imagen	%, medida ó auto
<b>border-image-repeat</b>	Control de repetición	stretch, repeat, round, space
<b>border-image-outset</b>	Cantidad que sobrepasa el área	Nº ó longitud





# Cajas sombreadas

Propiedad	Definición	Valores posibles
<b>box-shadow</b>	Sombra de una caja	color, desplazamiento horizontal, desplazamiento vertical, borrosidad, <code>inset</code>  Ejemplos: <code>10px 10px rgba(0,0,0,0.4)</code> <code>10px 10px 10px black inset</code>
<b>box-decoration-break</b>	Qué hacer al romper una caja	<code>clone</code> , <code>slice</code>



# Cajas sombreadas

<code>border:5px solid blue; background-color:orange;</code>	<code>border-radius:20px</code>	<code>border-radius:0</code>
<code>box-shadow: 10px 10px rgba(0,0,0,0.4);</code>		
<code>box-shadow: 10px 10px rgba(0,0,0,0.4) inset;</code>		
<code>box-shadow: 10px 10px 0 10px; rgba(0,0,0,0.4);</code>		
<code>box-shadow: 10px 10px 0 10px rgba(0,0,0,0.4) inset;</code>		



# Posicionamiento

Facilitar la maquetación de páginas Web

Propiedad	Definición	Valores posibles
<b>position</b>	Tipo de posición	<code>static</code> , <code>absolute</code> , <code>relative</code> , <code>fixed</code>
<b>top</b> , <b>bottom</b> , <b>left</b> , <b>right</b>	Posición	Medida o porcentaje
<b>float</b>	Posición flotante	<code>left</code> , <code>right</code> , <code>none</code> , <code>inherit</code>
<b>clear</b>	Eliminar elementos flotantes	<code>left</code> , <code>right</code> , <code>none</code> , <code>both</code> , <code>inherit</code>
<b>z-index</b>	Capa	<code>nº</code> , <code>auto</code> , <code>inherit</code>



# Dimensiones

Propiedad	Definición	Valores posibles
<b>width</b>	Alcho de la caja	Medida, porcentaje ó <code>inherit</code>
<b>height</b>	Alto de la caja	Medida, porcentaje ó <code>inherit</code>
<b>min-width</b> <b>min-height</b>	Ancho/alto mínimo permitido	Medida, porcentaje ó <code>inherit</code>
<b>max-width</b> <b>max-height</b>	Ancho/alto máximo permitido	Medida, porcentaje ó <code>inherit</code>



# Visualización

Propiedad	Definición	Valores posibles
<b>display</b>	Método de disposición	<code>none, block, inline, inherit</code>
<b>visibility</b>	Método de visualización	<code>visible, hidden, collapse, inherit</code>



# Rotaciones

No soportada por navegadores actuales

Propiedad	Definición	Valores posibles
<b>rotation</b>	Rotación	Ángulo. Ej. 45deg
<b>rotation-point</b>	Punto de rotación	Posición. Ej. 50% 50%



# Columnas múltiples

Propiedad	Definición	Valores posibles
<code>column-count</code>	Nº columnas	Nº de columnas
<code>column-width</code>	Ancho de columnas	Longitud
<code>column-gap</code>	Espacio entre columnas	Espacio entre columnas
<code>columns</code>	Definición de columnas	Propiedad combinada (width, count)
<code>column-rule</code> <code>column-rule-width</code> <code>column-rule-style</code> <code>column-rule-color</code>	Borde entre columnas	Ejemplo: <code>3px outset #ff00ff;</code>
<code>column-span</code>	Ocupar o no columnas	<code>none</code> , <code>all</code>
<code>column-fill</code>	Balancear contenido entre columnas	<code>auto</code> , <code>balance</code>





# Rupturas entre columnas



Propiedad	Definición	Valores posibles
<b>break-before</b>	Control de ruptura de columnas antes de la caja	auto, always, avoid, left, right, page, column, avoid-page, avoid-column
<b>break-after</b>	Ruptura de columnas después de la caja	auto, always, avoid, left, right, page, column, avoid-page, avoid-column
<b>break-inside</b>	Ruptura de columnas dentro de la caja	auto, always, avoid, left, right, page, column, avoid-page, avoid-column



# Fondos

Propiedad	Definición	Valores posibles
<b>background</b>	Imagen del borde	Valor combinado
<b>background-color</b>	Color del fondo	Color ó transparent
<b>background-attachment</b>	Desplazamiento respecto al documento	<code>fixed</code> , <code>local</code> , <code>scroll</code>
<b>background-position</b>	Posición del fondo	medidas o porcentajes
<b>background-repeat</b>	Controlar repetición del fondo	<code>repeat-x</code> , <code>repeat-y</code> , <code>repeat</code> , <code>space</code> , <code>round</code> , <code>no-repeat</code>
<b>background-image</b>	Imagen de fondo	URI Degradado en CSS3



# Fondos degradados

`background-image` admite fondos mediante degradados

Función	Definición	Valores posibles
<code>linear-gradient</code>	Degradado lineal	<code>linear-gradient(ángulo, colores de parada)</code> Ejemplos: <code>linear-gradient(45deg, white, black)</code> <code>linear-gradient(yellow, blue 20%, green)</code> <code>linear-gradient(to top right, red, blue)</code>
<code>gradient</code>	Degradado radial	<code>radial-gradient(posición, colores de parada)</code> Ejemplo: <code>radial-gradient(yellow, green);</code> <code>radial-gradient(red, yellow, green);</code> <code>radial-gradient(20px 30px at 20px 30px, red, green);</code>

También pueden repetirse mediante:  
`repeating-linear-gradient,`  
`repeating-radial-gradient`



# Nuevas propiedades de fondos



Propiedad	Definición	Valores posibles
<b>background-clip</b>	Área de corte del fondo	content-box, border-box, padding-box
<b>background-origin</b>	Especifica desde dónde se calcula la posición relativa del fondo	content-box, border-box, padding-box
<b>background-size</b>	Tamaño del fondo	ancho alto   cover   contain Ejemplo: background-size: 50% 50%; background-size: cover;



# Listas

## Estilo de listas

Propiedad	Definición	Valores posibles
<b>list-style</b>	Área de corte del fondo	Combinado: tipo, posición, imagen
<b>list-style-type</b>	Tipo de lista	Cadena, Estilo contador, <i>inline</i> , <i>none</i>
<b>list-style-position</b>	Posición del marcador	<i>inside</i> , <i>hanging</i> , <i>outside</i>
<b>list-style-image</b>	Imagen del marcador	Imagen, <i>none</i>

En CSS3 se define un pseudoelemento `::marker`



# Tablas

Propiedad	Definición	Valores posibles
<b>caption-side</b>	Posición de la leyenda	top, bottom, inherit
<b>table-layout</b>	Algoritmo presentación tabla	auto, fixed, inherit
<b>border-collapse</b>	Control de bordes	collapse, separate
<b>border-spacing</b>	Espaciado entre bordes	Longitud



# Contenido generado

Propiedad `content` indica contenido a insertar

Aplicable a `:before`, `:after`

El valor puede ser una cadena

```
aside.Nota:before {  
  content: "NOTA: ";  
}
```

También se puede hacer referencia a un atributo

```
@media print {  
  a:after {  
    content: "(" attr[href] ")" ;  
  }  
}
```



# Contadores

Es posible definir contadores

Permiten automatizar tareas de conteo

`counter-reset`: Inicializa un contador

`counter-increment`: Incrementa el contador

`counter(contador)`: Obtiene valor de un contador

```
body {  
    counter-reset: contadorPs;  
}  
p:before {  
    content: "Párrafo: " counter(contadorPs) ". ";  
    counter-increment: contadorPs;  
}
```





# Transformaciones, transiciones y animaciones



# Introducción

**Transformación:** Modificar posición/tamaño de una caja

Ejemplo: rotar, trasladar, aumentar/disminuir, etc.

**Transición:** cómo debe cambiar el valor de una propiedad

Instante temporal, tiempo de la transición, etc.

Se lanzan como reacciones ante eventos (ej. :hover)

Animaciones sencillas entre dos estados

**Animación:** animaciones avanzadas

Control sobre cuándo y cómo efectuar las animaciones

Iteraciones, retardos, etc.



# Transformaciones

Modifican tamaño/posición/forma de un elemento

Ejemplo:

```
#logo:hover {  
  transform: rotate(27deg) translate(10px);  
}
```



# Transformaciones 2D y 3D

Propiedad	Definición	Valores posibles
<b>transform</b>	Transformación	Combinada: origen, style none, rotate, rotateX, rotateY, rotateZ, rotate3d, matrix, matrix3d, scale, scaleX, scaleY, scale3d, skew, skewX, skewY, perspective
<b>transform-origin</b>	Origen de transformación	auto, fixed, inherit
<b>transform-style</b>	Estilo de transformación	flat, preserve3d
<b>perspective</b>	Perspectiva	Longitud
<b>perspective-origin</b>	Origen de la perspectiva	Longitud, %, left, center, right, top, bottom
<b>backface-visibility</b>	Determina si es visible la parte de atrás de un elemento	visible, hidden



# Transiciones

Especifican cómo cambiar el valor de una propiedad

Permiten realizar cambios suaves

Sólo pueden aplicarse a ciertas propiedades

Ejemplo:

```
#logo {  
  transition: transform 0.3s ease-out;  
}  
  
#logo:hover {  
  transform: rotate(27deg);  
}
```



# Transiciones

Cambiar propiedades gradualmente (durante un tiempo)

Propiedad	Definición	Valores posibles
<b>transition</b>	Propiedad abreviada	Propiedad, Duración, Función de transición, delay
<b>transition-property</b>	Propiedad a cambiar	Propiedad, none, all
<b>transition-duration</b>	Duración transición	Tiempo, ej. 2s
<b>transition-timing-function</b>	Función de transición	ease, linear, ease-in, ease-in-out, step-start, step-end, steps(n <sup>º</sup> s), cubic-bezier(4 n <sup>º</sup> s)
<b>transition-delay</b>	Tiempo de comienzo	Tiempo



# Funciones de transición

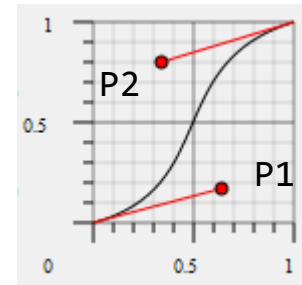
2 formas:

Por pasos (steps)

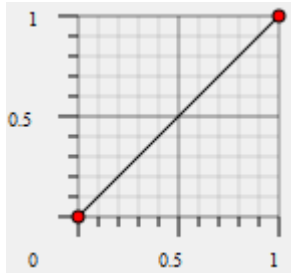
Continuas (a partir de una curva Bezier)

`cubic-bezier(x1,y1,x2,y2)`

Valores predefinidos:

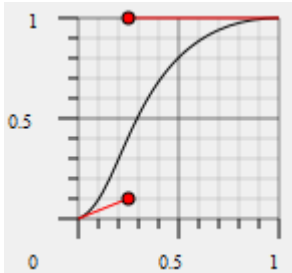


linear



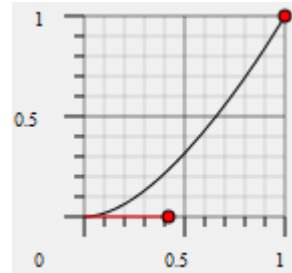
P1 (0,0)  
P2 (1,1)

ease



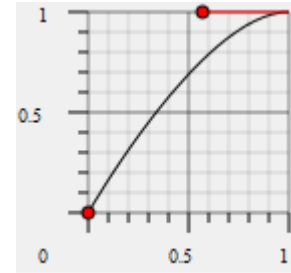
P1 (0.25,0.1)  
P2 (0.25,1)

ease-in



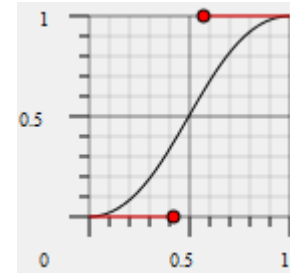
P1 (0.42,0)  
P2 (1,1)

ease-out



P1 (0,0)  
P2 (0.58,1)

ease-in-out



P1 (0.42,0)  
P2 (0.58,1)



# Animaciones

Las animaciones permiten controlar progreso completo

Transiciones sólo permitían declarar estado inicial y final

`@keyframes` declara caminos de animaciones

Se indica el instante de tiempo del camino

`animation-*` define propiedades sobre animaciones

```
@keyframes movimiento {
  from { left: 10px; top: 10px; }
  25% { left: 100px; top: 10px; }
  50% { left: 100px; top: 100px; }
  75% { left: 10px; top: 100px; }
  to { left: 10px; top: 10px; }
}

div { animation-name: movimiento ;
      animation-duration: 5s;
      animation-iteration-count: 10;
}
```





# Keyframes

@keyframes permite definir caminos de animaciones

Propiedad	Definición	Valores
from	Inicio	Valores que deben tener las propiedades al comienzo
porcentaje	Punto intermedio	Valores que deben tener las propiedades en ese instante
to	Final	Valores que deben tener las propiedades al final



# Animaciones

Propiedad	Definición	Valores posibles
<b>animation</b>	Combinada	
<b>animation-name</b>	Nombre de la animación	Identificador definido en <a href="#">@keyframes</a>
<b>animation-duration</b>	Duración	Tiempo
<b>animation-timing-function</b>	Función de tiempo	ease, linear, ease-in, ease-in-out, step-start, step-end, steps(nºs), cubic-bezier(4 nºs)
<b>animation-delay</b>	Retardo	Tiempo
<b>animation-iteration-count</b>	Nº de veces	Nº, infinite
<b>animation-direction</b>	Dirección	normal, reverse, alternate, alternate-reverse
<b>animation-play-state</b>	Estado	running, paused
<b>animation-fill-mode</b>	Valores fuera de la animación	none, forward, backwards, both



# Maquetado



# Maquetado

## Varios estándares en desarrollo:

Grid (propuesta de Microsoft)

`grid-columns`, `grid-rows`

Model de cajas flexibles

`box-flex`, `box-align`, ...

Template Layout

`display`, `position`, `::slot`

Regions

`flow-into`, `flow-from`



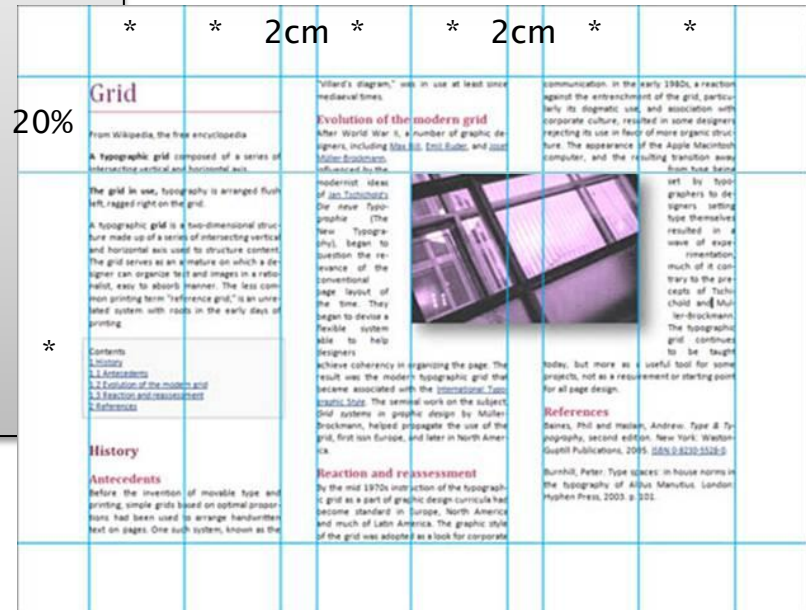
# Grid Layout

Propuesto originalmente por Microsoft

Permite especificar columnas y filas de una rejilla

Es posible hacer referencia a la columna con `gr`

```
body {  
  grid-columns: * * (2cm * *)[2];  
  grid-rows: 20% *;  
  columns: 3;  
  column-gap: 2cm; }  
  
img {  
  float: page top left;  
  float-offset: 4gr 1gr;  
}
```





# Cajas flexibles

Nuevo valor para display, flexbox

Propiedades:

flex, flex-pack, flex-direction, flex-order,...

```
<style>
ul { display: flexbox; flex-pack: distribute; }
. . .
</style>
<ul>
  <li><button><img src='new.svg' alt="New"></button></li>
  <li><button><img src='upload.svg' alt="Upload"></button></li>
  <li><button><img src='save.svg' alt="Save"></button></li>
  <li><button><img src='trash.svg' alt="trash"></button></li>
</ul>
```



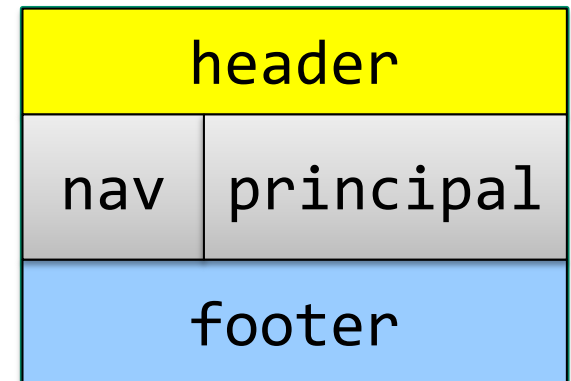


# Plantillas de Layout

Permite maquetados avanzados

No soportada por navegadores (librería JQuery)

```
body {  
  display: "a a a"  
         "b c c"  
         "d d d";  
}  
header { position: a; }  
footer { position: d; }  
#nav { position: d; }  
#principal { position: c; }  
  
::slot(a) { background: yellow; }  
::slot(d) { background: cyan; }
```





*Fin de la Presentación*

