

# Fundamentos de Aplicaciones y Servicios Telemáticos

2º Grado en Ingeniería de Tecnologías de Telecomunicación

Departamento de Ingeniería Telemática



## Tema 01

### Programación Web Estática CSS

# CSS: Contenido

---

- ❑ Introducción: Motivación, Etapas CSS
- ❑ Modelo de cajas
- ❑ Reglas CSS
  - Sintaxis
  - Selectores
  - Formas de aplicarlas
    - ❑ Superposición de estilos (cascading)
  - Propiedades/Valores
    - ❑ Propiedades básicas para el Modelo de cajas
    - ❑ Propiedades de Posicionamiento
  - Tendencias
  - Resumen y referencias

# CSS: Introducción

---

## □ Motivación:

- **HTML**: diseñado para organizar el contenido, sin formato.

```
<p>Parrafo</p>  
<table><tr><td>(1,1)</td>...</table>
```

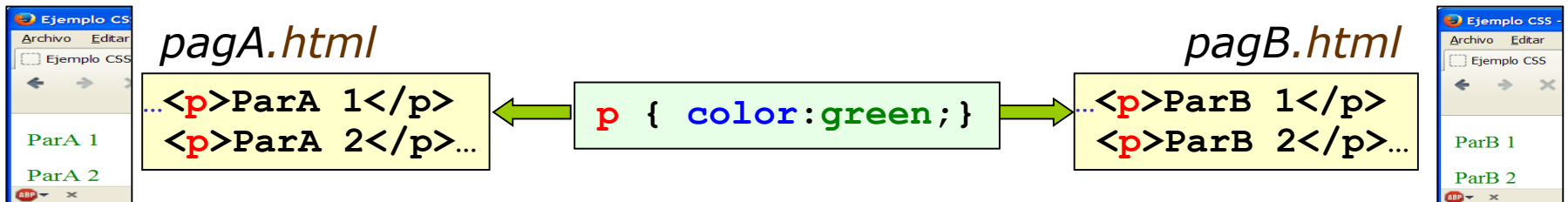
- **HTML 3.2**: incluyó atributos de formato (HTML 4.01 heredó muchos):

```
<p align="right" >Parrafo</p>  
<table bgcolor="green" border="1"><tr><td>(1,1)</td>...</table>  
<p align="right" >Parrafo2</p>
```

- **Problema**: documentos "confusos" y costosos en tiempo desarrollo.
  - Formato independientemente de cada elemento (etiqueta).
  - Solución: no usar atributos de estilo en HTML, sino CSS.

# CSS: Objetivo

- ❑ Objetivo CSS: lenguaje
  - Para describir estilos de presentación
    - ❑ Aspecto
    - ❑ **Posicionamiento** (más potente, controla más aspectos del formato)
  - Utilizable desde múltiples lenguajes de etiquetas (HTML, XHTML, ...).
  - Capaz de aplicar mismo estilo a varias etiquetas y de distintos ficheros



# CSS: Ventajas

---

- ❑ Ventajas de separar **contenidos** (html) y **presentación** (css):
  - Documentos homogéneos (todos los párrafos igual, ...)
  - Cambios de estilo centralizados en el CSS.
  - Visualizar un documento en dispositivos diferentes (basta un CSS para cada uno).
- ❑ Estándar W3C CSS: es **único**, dividido en etapas de desarrollo
  - CSS 1, 2, 3 – No son versiones, son niveles.
  - CSS nivel 2 se basa en la recomendación CSS2.1
  - A partir del nivel 3 y siguientes es modular
  - Por eso en los ficheros CSS no se indica ninguna versión (no hay equivalente al DOCTYPE de los ficheros HTML).
- ❑ CSS: se basa en:
  - Representación de cada elemento (etiqueta): **Modelo de cajas**.
  - Aplicar un estilo de presentación a cada elemento: **Reglas**.

# CSS: Modelo de cajas

- Cada elemento (`html`, `body`, párrafo, lista, tabla, celda tabla,...) consta de una "caja" constituida (del centro hacia fuera) por:

Area de contenido ( <code>content</code> )	Contenido en sí del elemento: texto de un párrafo, imagen, lista, tabla, celda tabla, etc
Relleno ( <code>padding</code> )	Espacio libre entre contenido y borde
Borde ( <code>border</code> )	Línea que encierra relleno/contenido
Margen ( <code>margin</code> )	Espacio libre entre esta caja y las adyacentes
Línea externa ( <code>outline</code> )	Línea adicional tras el borde, superponiéndose al margen (NO afecta a su tamaño)

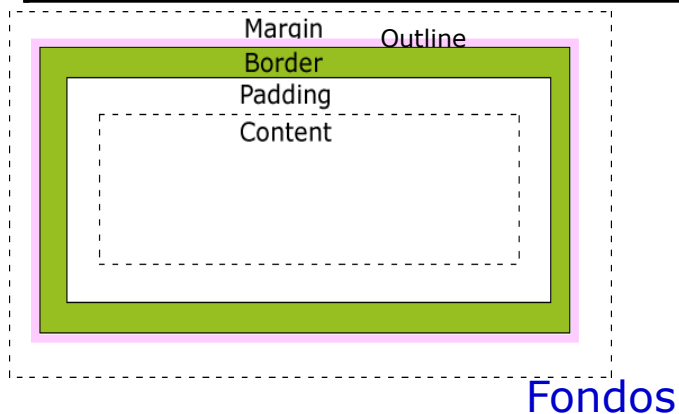
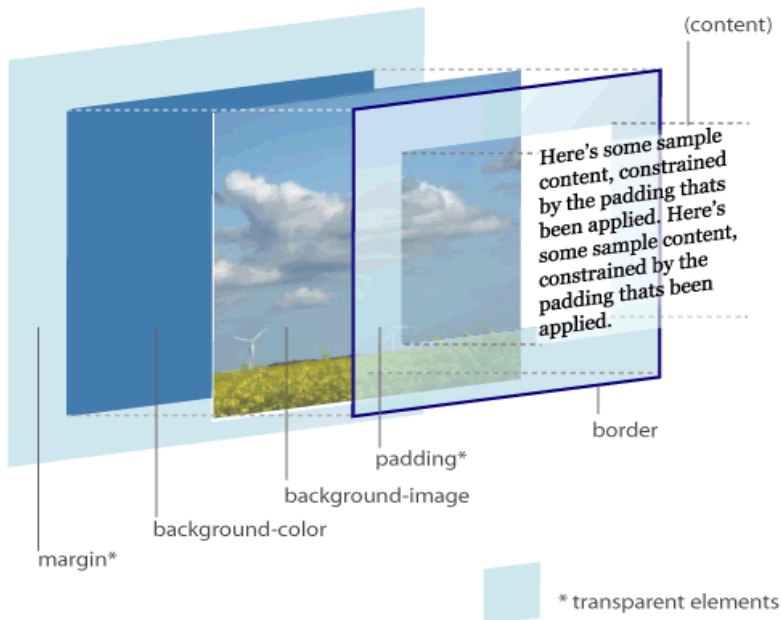


Imagen de fondo	Tras el contenido y el relleno
Color de fondo	Tras la imagen de fondo

# CSS: Tamaño del Modelo de cajas

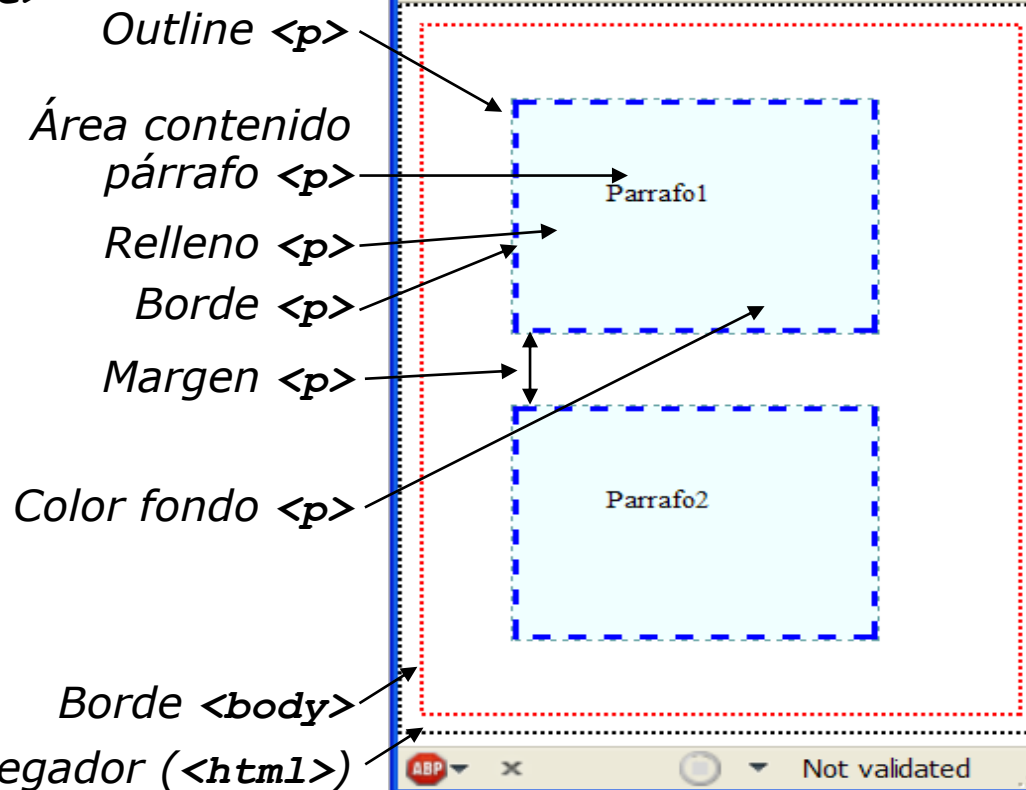
- Tamaño Caja =  
Márgen +  
Borde +  
Relleno +  
Contenido

## THE CSS BOX MODEL HIERARCHY



# CSS: Modelo de cajas, Ejemplo

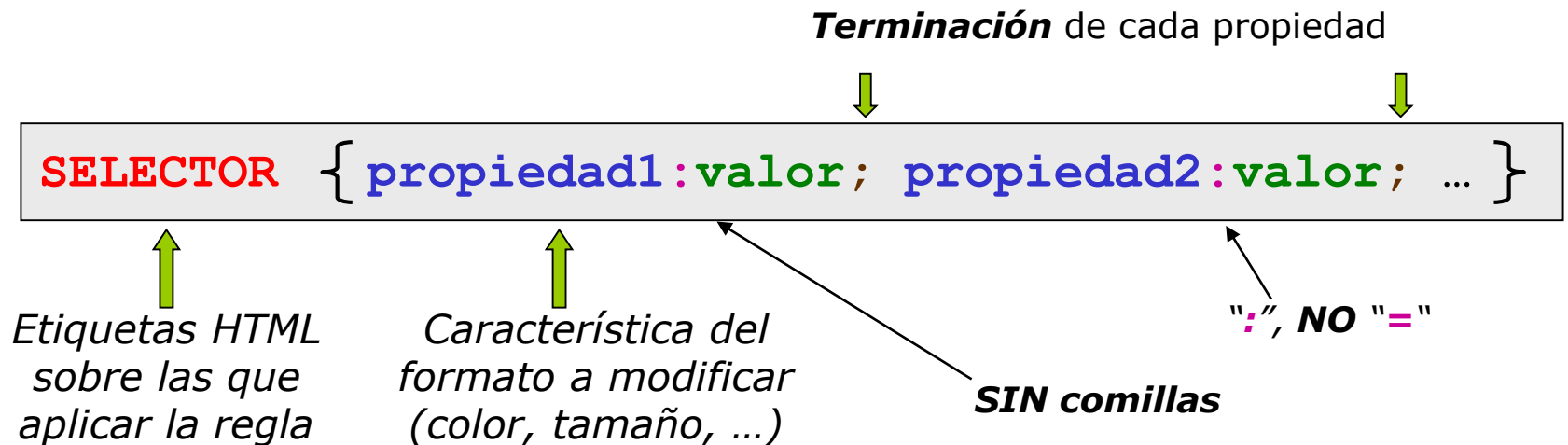
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
    "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
  <title>Modelo de cajas</title>
  <style type="text/css">
    ...xxx...
  </style>
</head>
<body>
<body>
  <p>Parrafo1</p>
  <p>Parrafo2</p>
</body>
</html>
```





# CSS: Reglas. Sintaxis

- Regla CSS: estilo para uno/varios elementos (HTML, ...)
  - Case **IN**sensitive. Puede escribirse en varias líneas y con espacios, incluso entre una propiedad y su valor



# CSS: Reglas. Ejemplos

- Además, CSS soporta comentarios (el navegador los ignora):

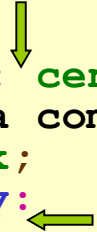
```
/* Comentario */
```

- Ejemplos:

*Reglas CSS*

```
h3 { color:red; font-size:8pt; }
```

```
p
{
  text-align:center;
  /*This is a comment*/
  color:black;
  font-family:arial;
}
```



# CSS: Reglas. Selectores y ejemplos

Tipos de Selectores simples	Elementos (HTML, ...) afectados	Ejemplo
<b>*</b>	Todos (incluidos <code>&lt;html&gt;</code> y <code>&lt;body&gt;</code> )	<b>*</b>
<b>etiqueta</b>	Con esta etiqueta	<b>p</b>
<b>[atributo]</b>	Con este atributo	<b>[href]</b>
<div> <div>Con/Sin espacios</div> <div> <div>[atributo op valor]</div> <div>op</div> </div> </div>	Idem y que su valor sea igual (=), contenga palabra (~=) ← "palabra": separada por espacios o empiece por ( =) ← Valor completo o prefijo seguido de quión "-"	<b>[href="web.html"]</b>
<b>#ID</b>	(Identificador) Con el atributo "id=ID"	<b>#AAAA</b>
<b>.clase</b>	(Clase) Con el atributo "class=clase"	<b>.c1</b>
<b>:característica</b>	(Pseudo-clase) Con esta característica: ":visited" (enlace visitado), ":hover" (ratón encima del enlace), ...	<b>:visited</b>

```

...
<body>
  <p class="c1">Enlaces:
    <a href="web.html">Marca</a>
    <p id="AAAA">Parrafo 2</p>
    <a href="index.html">Marca</a>
  </body></html>

```

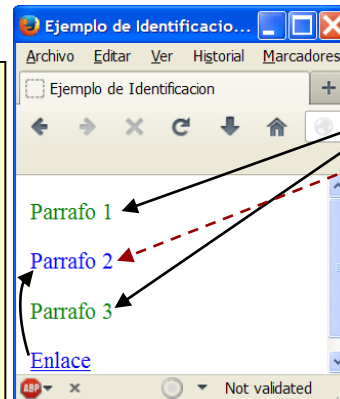
pagina.html

## HTML: Etiquetas en <body>, Identificador y Clase

- ❑ **Asociar** un **ID único** a **una** etiqueta:
  - Atributo "id" (global, cualquier etiqueta lo soporta).
  - Uso: referenciar **a esa etiqueta** para aplicarle algo. Ej:
    - ❑ Saltar (<a>), un Estilo CSS, procesado ECMAScript.
- ❑ **Asociar** un **nombre común** (clase) a **varias** etiquetas:
  - Atributo "class" (global).
  - Uso: referenciar **a esas etiquetas** para aplicarles algo. Ej:
    - ❑ Un Estilo CSS, procesado ECMAScript.
- ❑ Nombres ID/clase: el primer carácter no puede ser numérico.

*pagina.html*

```
<body>
  <p class="GA">Parrafo 1</p>
  <p id="Ident">Parrafo 2</p>
  <p class="GA">Parrafo 3</p>
  <a href="#Ident">Enlace</a>
</body></html>
```



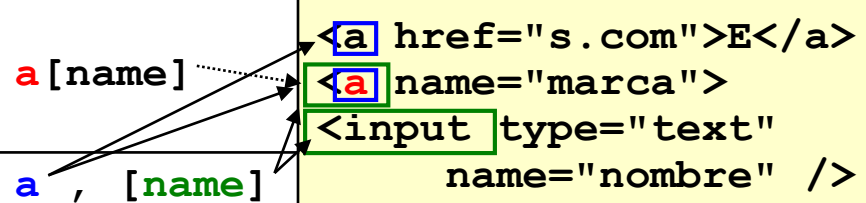
*estilo.css*

```
.GA { color:green;}
#Ident { color: blue;}
```

# CSS: Reglas. Combinación de Selectores. Operador lógico

## □ Combinación de Selectores:

- Notación: "S1, S2" selectores simples, o compuestos (por varios selectores con los operadores indicados).

Operador lógico	Elementos (HTML, ...) afectados	Ejemplo	
S1S2...	AND: S1 y S2 ...	a[name]	 <pre>&lt;a href="s.com"&gt;E&lt;/a&gt; &lt;a name="marca"&gt; &lt;input type="text"       name="nombre" /&gt;</pre>
S1 , S2	OR: S1 o S2	a , [name]	

# CSS: Reglas. Combinación de Selectores. Operador de parentesco

Operador de parentesco	Elementos (HTML, ...) <b>afectados</b>	Ejemplo	
$s1 \ s2$	<b>Descendiente</b> : S2 dentro ( <i>descendiente</i> -hijo, nieto, ...-) de S1	$div \ em$	
$s1 \ > \ s2$	<b>Hijo</b> : S2 es <i>hijo</i> (directo) de S1	$div \ > \ em$	
$s1 \ + \ s2$	<b>Adyacente</b> : S2 ubicado justo tras un ( <i>siguiente hermano de</i> ) S1	$div \ + \ p$	
$s1 \ \sim \ s2$	<b>Mismo nivel</b> : S2 ubicado al mismo nivel ( <i>cualquier hermano de</i> ) que S1	$div \ \sim \ p$	

\* **Selectores se leen *de derecha a izquierda***. Ej:

`div p a[href]`, `p` ← Todos "`<p>`"; y "`<a>`'s con href" dentro de `<p>` y éste dentro de `<div>`

# CSS: Reglas. Formas de aplicarlas

---

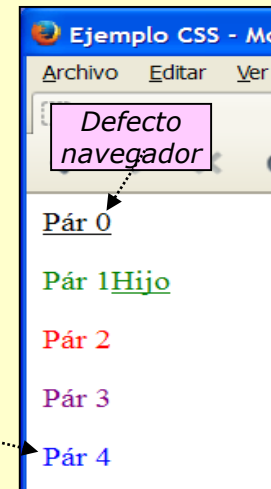
- 3 mecanismos (pueden usarse a la vez) permiten aplicar:
  - Inline (atributo `style`): estilo sobre elemento concreto. SIN Selector.
  - Interno (en `<head>`): Un mismo estilo sobre varios elementos.
  - Externo (en fichero CSS): Idem, y de distintos ficheros. Este es el más aconsejable.

# CSS: Reglas. Aplicación inline

- ① Inline (atributo `style`): estilo sobre elemento concreto. SIN Selector

*pagina.html*

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
    "http://www.w3.org/TR/html4/strict.dtd">
<html><head>
  <title>Ejemplo CSS</title>
  <link rel="stylesheet"
        type="text/css"
        href="estilo.css" />
  <style type="text/css">-----
    p.C      { color:red; }
    body #A  { color:purple; }
  </style>
</head>
<body> <ins>                Pár 0</ins>
  <p>Pár 1<ins>Hijo</ins>  </p>
  <p class="C">                Pár 2</p>
  <p id="A" class="C">Pár 3</p>
  ① → <p style="color:blue;">Pár 4</p>
</body></html>
```





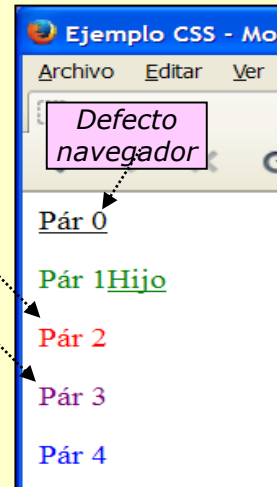
# CSS: Reglas. Aplicación interna

- ② Interno (en <head>): Un mismo estilo sobre varios elementos.

*pagina.html*

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
    "http://www.w3.org/TR/html4/strict.dtd">
<html><head>
  <title>Ejemplo CSS</title>
  <link rel="stylesheet"
        type="text/css"
        href="estilo.css" />
  ② <style type="text/css">
      p.C      { color:red; }
      body #A { color:purple; }
  </style>
</head>
<body>
  <ins>
    Padre
    Hijo
    <p>Pár 1<ins>Hijo</ins> </p>
    <p class="C">Pár 2</p>
    <p id="A" class="C">Pár 3</p>
    <p style="color:blue;">Pár 4</p>
  </ins>
</body></html>
  
```



# CSS: Reglas. Aplicación externa

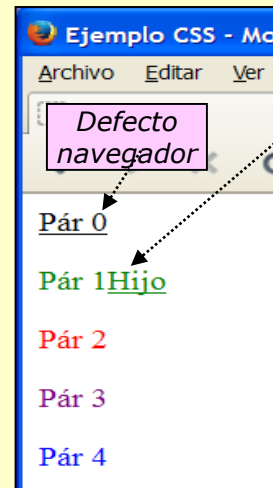
- ③ Externo (en fichero CSS): Un mismo estilo sobre varios elementos, y de distintos ficheros.

*pagina.html*

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
    "http://www.w3.org/TR/html4/strict.dtd">
<html><head>
  <title>Ejemplo CSS</title>
  ③ <link rel="stylesheet"
        type="text/css"
        href="estilo.css" />
  <style type="text/css">-----
    p.C { color:red; }
    body #A { color:purple; }
  </style>
</head>
<body> <ins>
  <p>Pár 1<ins>Hijo</ins> </p>
  <p class="C">Pár 2</p>
  <p id="A" class="C">Pár 3</p>
  <p style="color:blue;">Pár 4</p>
</body></html>
```

Descripción MIME de un contenido CSS (diferencia de otros como "text/xslt")

Padre Hijo



Extensión habitual  
(puede ser cualquiera)

Sin versión

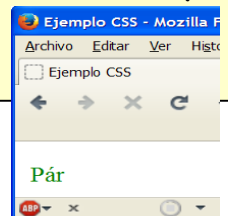
*estilo.css*

```
p { color:green;
    align:left; }
```

\* Un fichero CSS contiene sólo un conjunto de reglas

*pag2.html*

```
...<link...
  href="estilo.css" />
...
<p>Pár</p>
```



# CSS: Reglas. Propiedades Posicionamiento

---

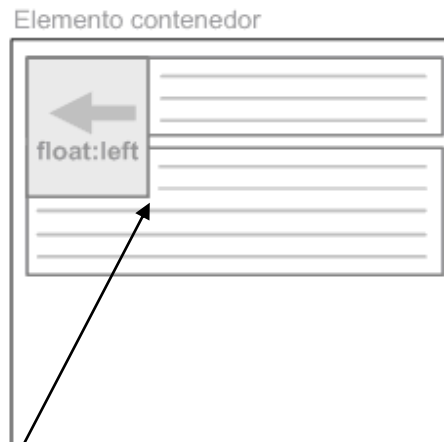
- ❑ Cambiar la posición de un elemento: dos formas:
  - A) Desplazar respecto a un área de referencia:
    - `"position"` (debe estar antes): determina el área de referencia.
  - B) Flotar hasta el borde del contenedor (o hasta otro flotante).
    - `"float"`: borde horizontal hasta el que flotar.
    - `"clear"`: lado/s (horizontal) del elemento (de bloque) en los que no puede haber un elemento flotante.

# CSS: Reglas. Posicionamiento float

Normal (Sin desplazamientos)



"Imagen" {float:left}



"Imagen" {float:left}  
"Parrafo2" {clear:left}



Los elementos que se encuentran alrededor del flotante adaptan sus contenidos para fluir alrededor.

# CSS: Reglas. Posicionamiento

▣ Muy útil (sustituto antiguos "frames"): menús, superposiciones, ...

```

...<body>                                pagina.html
<div id="contenedor">
  <div id="cabecera">
    <h1>FAST ..</h1>
  </div>
  <div id="menu">
    <p>Item 1</p>
    <p>Item 2</p> ...
  </div>
  <div id="contenido">
    <h2> Titular </h2>
    <p> This spec.....</p>
  </div>
  <div id="pie">
    <p> pie de página ...</p>
  </div>
</div>
</body></html>

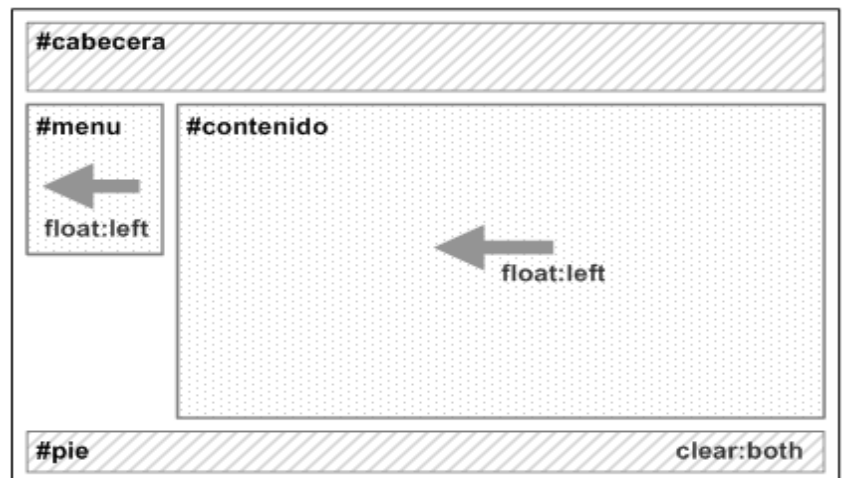
```

```

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

```

#contenedor



# Tendencias

---

- ❑ Uso de frameworks:
  - Bootstrap
    - ❑ Tutorial: <http://www.w3schools.com/bootstrap>
  - W3.CSS
    - ❑ Sin librerías añadidas
    - ❑ Tutorial: <http://www.w3schools.com/w3css/>
- ❑ Ventajas:
  - Comodidad
  - Diseño adaptativo:
    - ❑ RWD, Responsive Web Design
    - ❑ mobile-first web site

# Resumen y Referencias

---

- Lenguaje CSS: Para describir estilos de presentación
  - Aspecto
  - Posicionamiento
- Referencias CSS

- *Estándar CSS:* <https://www.w3.org/TR/CSS/>
- *Base 2.1:* <https://www.w3.org/TR/CSS2/>
- *Uso práctico:* <http://www.w3schools.com/css/>
  - *Lista propiedades:* <http://www.w3schools.com/cssref/>
  - *Lista selectores:* [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp)