



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA – ICAI
Departamento de Sistemas Informáticos

Diseño y Creación de Páginas web Tema 6. Hojas de estilo CSS

Cristina Puente, Rafael Palacios
2009-2010



Formatos vs Estilos

- Los formatos aplicados a un elemento concreto sólo afectan a dicho elemento.
- Las hojas de estilos permiten centralizar definiciones de formatos.
- Las definiciones puede afectar a varios elementos de una página o a varias páginas.



Formatos HTML

- Algunos formatos HTML se introducen con etiquetas. Ejemplos:
 - Una palabra en `negrita` y otra en `<i>cursiva</i>`
 - `<h1><big>Título grande</big></h1>`
- HTML también admite ciertos formatos que afectan a todo el objeto. Ejemplos:
 - `<body bgcolor="black" text="white">`
 - `<p align="right">Párrafo alineado a la derecha</p>`

Propiedad="valor"



Formatos inline (tipo CSS)

- A partir de HTML 4 se permite definir estilos tipo CSS en cada elemento. Ejemplos:

Propiedad:valor;

– `<body style="background: black; color: white">`

– `<p style="color: fuchsia; text-indent: 2em">`Párrafo con texto en color fucsia y con sangrado de 2em en la primera línea. 2em equivale a dos veces la altura del tipo de letra.`</p>`

Párrafo con estilo por defecto.

Párrafo con texto en color fucsia y con sangrado de 2em en la primera línea. 2em equivale a dos veces la altura del tipo de letra.

Referencia HTML 4.01: <http://www.w3.org/TR/html401/>

Referencia CSS2: <http://www.w3.org/TR/REC-CSS2/>





ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA – ICAI

Estilos CSS



Estilos CSS

- Los navegadores modernos soportan estilos mediante el estándar CSS (Cascading Style Sheet)
- Los estilos se definen en un archivo .css que se carga junto con la página HTML
- Permite centralizar los estilos de todo el sitio web
- Si una página HTML redefine el estilo de un objeto, la nueva definición tiene prioridad frente a la hoja CSS



Estilos CSS

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
<HTML>
<HEAD>
  <TITLE>Página de ejemplo CSS</TITLE>
  <LINK href="mis_estilos.css" type="text/css" rel="stylesheet">
</HEAD>
<BODY>
  <P>Párrafo en el estilo por defecto definido en el fichero
  mis_estilos.css a la que hace referencia esta página html.</P>
</BODY>
```

HTML

Por defecto no hace falta definir formatos adicionales

Referencia a la definición de estilos



Estilos CSS: Definición de reglas

```
/* Ejemplo de definición de reglas */  
body {  
    margin-left: 10%;  
    margin-right: 10%;  
    color: blue; background: white;  
}  
  
P { text-indent: 2em; }
```

CSS

Estilo del body (aplica por defecto a toda la página)

Estilo del párrafo (aplica por defecto a todos los párrafos <p>)



Ventajas de la hoja de estilos

- Todas las definiciones de formato están centralizadas
- Las personas que desarrollan las páginas HTML sólo deben preocuparse del contenido, no del aspecto
- Cambiando la definición de los estilos, cambia el aspecto de todas las páginas que comparten el mismo CSS.



Estilos para elementos especiales

```
/* Ejemplo de definición de reglas */  
body {  
  margin-left: 10%;  
  margin-right: 10%;  
  color: blue; background: white;  
}  
  
P { text-indent: 2em; }  
  
P.especial {  
  color: fuchsia;  
  border: solid red;  
}
```

CSS

Estilo del body (aplica por defecto a toda la página)

Estilo del párrafo (aplica por defecto a todos los párrafos <p>)

Estilo del párrafo tipo especial

```
<P>Párrafo en el estilo por defecto definido</P>  
<P class= "especial">Párrafo en estilo especial</P>
```

HTML

Departamento de Sistemas Informáticos

Escuela Técnica Superior de Ingeniería – ICAI
Universidad Pontificia Comillas

100 años de
Ingeniería
ICAI
1908-2008



Varias páginas de estilos

- Se pueden definir varias páginas de estilo a la vez, además de las definiciones "inline"
- Cada nueva definición tiene prioridad sobre las anteriores.
- Se pueden definir hojas de estilo opcionales para que el navegador (usuario) pueda elegir:

```
<LINK rel="stylesheet" href="mis_estilos.css" type="text/css">  
<LINK rel="alternate stylesheet" href="grande.css" type="text/css" title="Letra grande">  
<LINK rel="alternate stylesheet" href="peque.css" type="text/css" title="Letra pequeña">
```

- También se pueden definir estilos en función del dispositivo:

```
media="screen"  
media="print"  
media="aural"  
media="handheld"
```





ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA – ICAI

Ejemplos de definiciones CSS2

Definiciones de tamaños

- Posibles definiciones de tamaños

- margin-left: 10%; ← Relativo a la anchura de la ventana
- margin-left: 25; ← 25 pixels. Método no recomendado
- margin-left: 2em; ← Relativo al tamaño de letra

- Definiciones de tamaños de letra










- font-size: 12pt; ← 12 puntos
- font-size: large; ← Definición relativa con factores 1.2
[xx-small | x-small | small | medium | large | x-large | xx-large]
- font-size: 120%; ← Definición relativa



Definiciones de colores

- Posibles definiciones de colores
 - color: purple;
 - color: #800080;
 - color: rgb(128, 0, 128);

Color names and sRGB values

 black = "#000000"	 green = "#008000"
 silver = "#C0C0C0"	 lime = "#00FF00"
 gray = "#808080"	 olive = "#808000"
 white = "#FFFFFF"	 yellow = "#FFFF00"
 maroon = "#800000"	 navy = "#000080"
 red = "#FF0000"	 blue = "#0000FF"
 purple = "#800080"	 teal = "#008080"
 fuchsia = "#FF00FF"	 aqua = "#00FFFF"



Definición de tipos de letra

font-family

- Valores Genéricos: serif | sans-serif | cursive | fantasy | monospace
- Ejemplo: font-family: "new century schoolbook", serif;

font-style

- Valores: normal | italic | oblique | inherit
- Ejemplo: font-style: italic;

font-variant

- Valores: normal | small-caps
- Ejemplo font-variant: normal;

font-weight

- Valores: normal | bold | bolder | lighter | 100 | 200 | 300 | 400...
- Ejemplo: font-weight: bold;

font-stretch

- Valores: normal | wider | narrower | ultra-condensed | extra-condensed | condensed | semi-condensed | semi-expanded | expanded | extra-expanded | ultra-expanded
- Ejemplo: font-stretch: condensed;

font-size

- Ejemplo: font-size: 1.2em;



Etiquetas DIV y SPAN

- DIV define una sección de la página web.
 - Por ejemplo vale para agrupar varios párrafos.
 - Es parecido a un estilo de párrafo en Word.
 - Se suele utilizar para agrupar elementos que se quieren formatear con el mismo estilo.
 - Permiten el uso de eventos y otros atributos especiales como z-index o visibility

Capa 1 por encima de capa 2:
<div style="position:relative; font-size:
50px; z-index:2;">CAPA 1</div>

<div style="position:relative; top:-50; left:
5; color:red; font-size:80px; z-index:
1">CAPA 2</div>

Capa 1 por encima de capa 2:

CAPA 1
CAPA 2



Etiquetas DIV y SPAN

- Algunos atributos de <DIV>:
 - Visibility, define si la capa se puede ver o no
 - Visible;
 - Hidden;
 - Nombre, id="capa1"
 - Z-index, posición de una capa con respecto a otras. Depende del número asignado.
 - Overflow: define cómo debe actuar el navegador cuando el contenido de una capa es mayor que la propia capa.
 - Visible
 - Hidden
 - Scroll



Etiquetas DIV y SPAN

- Algunos atributos de <DIV>:
 - Posicionamiento:

Atributo	Valores	Ejemplo
position	absolute, relative, static	position:relative;
top	%, pixels	top:-50px;
left	%, pixels	left:20px;
right	%, pixels	right:-20px;
bottom	%, pixels	bottom:0px;
height	%, pixels	height:50%
width	%, pixels	width:50%



Etiquetas DIV y SPAN

- SPAN es un estilo que aplica a unos caracteres
 - Equivale a etiquetas `` ``
 - Es parecido a un estilo de caracter en Word.
 - También se pueden poner unas encima de otras.

```
<SPAN STYLE="position:relative; background-color:#90EE90"> relative green parent relative  
elative green parent relative green parent  
relative green parent
```

```
<SPAN STYLE="position:absolute; top:10px;  
left:50px;background-color:#ADD8E6">  
Absolute blue child con top:10px y left:50px. </  
SPAN> </SPAN>
```

```
relative green parent relative green parent relative green  
parent r Absolute blue child with top:10px and  
left:50px.  
green páent r elative green parent r elative green páent
```



Ejemplo de DIV y SPAN

```
/* Definición de reglas */  
DIV.Abstract { text-align: justify }  
SPAN.programa {  
    font-family: Lucida Console, monospace;  
    font-size: 0.8em;  
}
```

CSS

```
<DIV class="Abstract">  
    <p>Párrafo primero...</P>  
    <p>Segundo párrafo...</p>  
</DIV>  
  
<h1>La variable <SPAN class="programa">int</SPAN> en lenguaje C</h1>
```

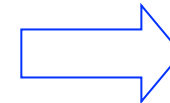
HTML



Formularios

- Un formulario HTML es una sección de un documento que contiene contenido normal, código, elementos especiales llamados *controles* (casillas de verificación (checkboxes), radiobotones (radio buttons), menús, etc).

```
<FORM action="http://algunsitio.com/prog/usuarioNuevo.php"
method="post">
  Nombre: <INPUT type="text" id="nombre"><BR>
  Apellido: <INPUT type="text" id="apellido"><BR>
  email: <INPUT type="text" id="email"><BR>
  <INPUT type="radio" name="sexo" value="Varón"> Varón<BR>
  <INPUT type="radio" name="sexo" value="Mujer"> Mujer<BR>
  <INPUT type="submit" value="Enviar"> <INPUT type="reset">
</FORM>
```



Nombre:
Apellido:
email:
 Varón
 Mujer



Formularios

– El elemento *form* actúa como contenedor de controles dentro de una página web.

– Atributos:

- Action: Programa que manejará el formulario completado y enviado (el atributo).
- Name: Nombre con el que se referirá al formulario en acciones posteriores. No es obligatorio pero conviene ponerlo!!!.
- Method: El método por el cual se enviarán los datos del usuario al servidor (el atributo). (Get por URL, Post por entrada estándar stdio)
- Accept-charset: Una codificación de caracteres que debe ser aceptada por el servidor para poder manejar este formulario



Formularios

- Métodos de envío de formularios:
 - get: Con el método HTTP "get", el conjunto de datos del formulario se agrega al URI especificado por el atributo action (con un signo de interrogación ("?") como separador) y este nuevo URI se envía al agente procesador.
 - post: Con el método HTTP "post", el conjunto de datos del formulario se incluye en el cuerpo del formulario y se envía al agente procesador.
 - El método "get" cuando no tiene efectos secundarios.
 - Si el servicio asociado con el procesamiento de un formulario causa efectos secundario (si el formulario modifica una base de datos), debería usarse el método "post".



Formularios

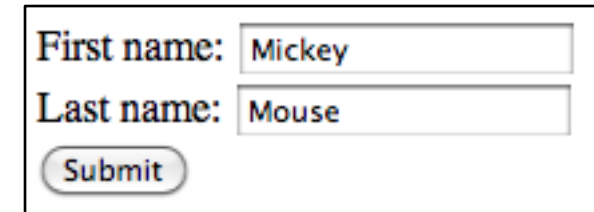
```
<html>
<head>
  <title>Ejemplo de PHP</title>
</head>
<body>
<H1>Ejemplo de procesamiento de formularios</H1>

<FORM ACTION="procesa2.php" METHOD="GET">
Introduzca su nombre:<INPUT TYPE="text" NAME="nombre"><BR>
Introduzca sus apellidos:<INPUT TYPE="text" NAME="apellidos"><BR>
<INPUT TYPE="submit" VALUE="Enviar">
</FORM>
</body>
</html>
```



Formularios

- Controles: text input



First name:
Last name:

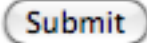
- La etiqueta `<input>` define la introducción de variables.
- Junto a esta etiqueta solemos encontrar los siguientes atributos:
 - `type=""` Indicará el tipo de variable a introducir.
 - `maxlength=""` Seguido de un valor que limitará el número máximo de caracteres a introducir en ese campo.
 - `size=""` Seguido de un valor que limitará el número de caracteres a mostrar en pantalla.
 - `value=""` Indica que no hay valor inicial del campo.
 - `name=""` Indicará el nombre que se asigna a un determinado campo.

http://www.w3schools.com/tags/tag_input.asp



Formularios

- Controles, botones

A rounded rectangular button with a light gray background and a thin border, containing the word "Submit" in a dark gray sans-serif font.

- Se definen mediante la tag `<input>` a la que le acompañan los atributos:
- `type`, Seguido de *submit* para enviar los datos del formulario y seguido de *reset* para borrar los datos que se han introducido, o *button* para ejecutar alguna acción.
- `value`, Indica el texto que incorporaran los botones. Normalmente, enviar y borrar
- `name`, Para identificar al botón

```
<input type="button" value="Cerrar esta ventana"  
onclick="window.close();">
```



Formularios

- Controles, checkbox

- El campo se elegirá marcando de entre varias opciones una casilla cuadrada. (Principal diferencia con radiobutton).
- Puede haber una opción seleccionada por defecto (checked)
- Se puede seleccionar más de una opción.

```
<INPUT TYPE="label" value="correcto">  
<INPUT TYPE="checkbox" name="c1" value= "1"  
onClick="if(this.checked == true){alert('verdadero!'); this.checked=false;}">  
<INPUT TYPE="label" value="falso">  
<INPUT TYPE="checkbox" name="c1" value= "0"  
onClick="if(this.checked == true){alert('falso!'); this.checked=false;}">
```

correcto falso



http://www.w3schools.com/tags/tag_input.asp



Formularios

- Controles, input especiales

- Tipos especiales de inputs:
- **Password** Indica que el campo a introducir será una palabra de paso. Mostrará asteriscos en lugar de letras escritas.
- **hidden** El visitante no puede modificar su valor ya que no está visible. Se manda siempre junto al atributo value= seguido de su valor entre comillas.
- Sus atributos son los mismos que para text.

```
<input type="password" name="clave" maxlength="5" size="6">
```



http://www.w3schools.com/tags/tag_input.asp



Formularios

- Controles, radio

- El campo se elegirá marcando de entre varias opciones una casilla circular.
- Sólo puede tener una casilla activa.

```
<input type="radio" name="g1" value="Leche"> Leche<br>
```

```
<input type="radio" name="g1" value="Mant" checked> Mantequilla<br>
```

```
<input type="radio" name="g1" value="Queso"> Queso
```

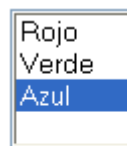
- Leche
- Mantequilla
- Queso

http://www.w3schools.com/tags/tag_input.asp



Formularios

- Controles, listas desplegables
 - Las etiquetas `<select>.....</select>` encierran los valores que podremos elegir a partir de una lista. Los atributos que acompañan a la etiqueta de apertura son:
 - `name=""` Indicará el nombre del campo de selección.
 - `Size=""` Indicará el número de opciones visibles. Si le asignamos 1, la selección se presentará como un menú desplegable. Si le asignamos un valor mayor se presentará como una lista con barra de desplazamiento.
 - `múltiple` indica si se pueden realizar múltiples selecciones.



Rojo
Verde
Azul



Rojo ▾

http://www.w3schools.com/tags/tag_select.asp

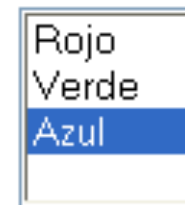


Formularios

- Controles, listas desplegables

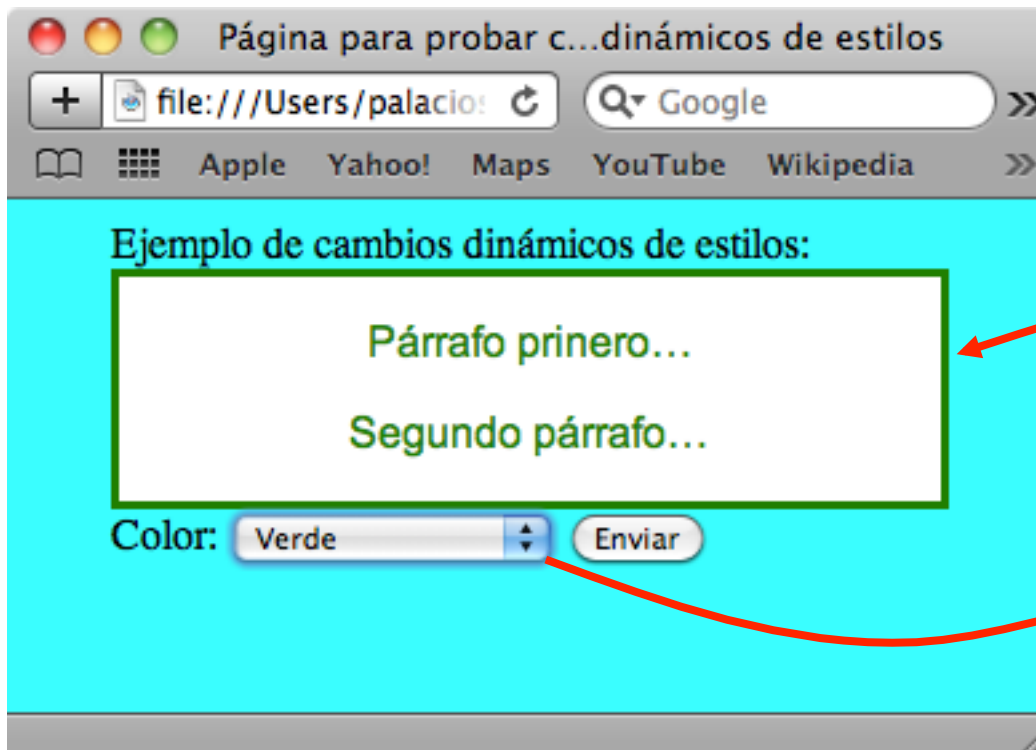
- Las diferentes opciones de la lista se indicarán mediante la etiqueta `<option>`
- Puede acompañarse del atributo *selected* para indicar cual es la opción que aparecerá por defecto.
- Si no lo especificamos, siempre será la primera de la lista

```
<SELECT NAME="Colores" MULTIPLE>
  <OPTION VALUE="r">Rojo</OPTION>
  <OPTION VALUE="g">Verde</OPTION>
  <OPTION VALUE="b">Azul</OPTION>
</SELECT>
<BR><BR>
<SELECT NAME="Colores" SIZE="1">
  <OPTION VALUE="r">Rojo</OPTION>
  <OPTION VALUE="g">Verde</OPTION>
  <OPTION VALUE="b">Azul</OPTION>
</SELECT>
```



Estilos dinámicos

- Una página web se puede comportar como una aplicación si unimos los estilos CSS, con formularios y con funciones JavaScript.



Cambio el estilo

OnChange



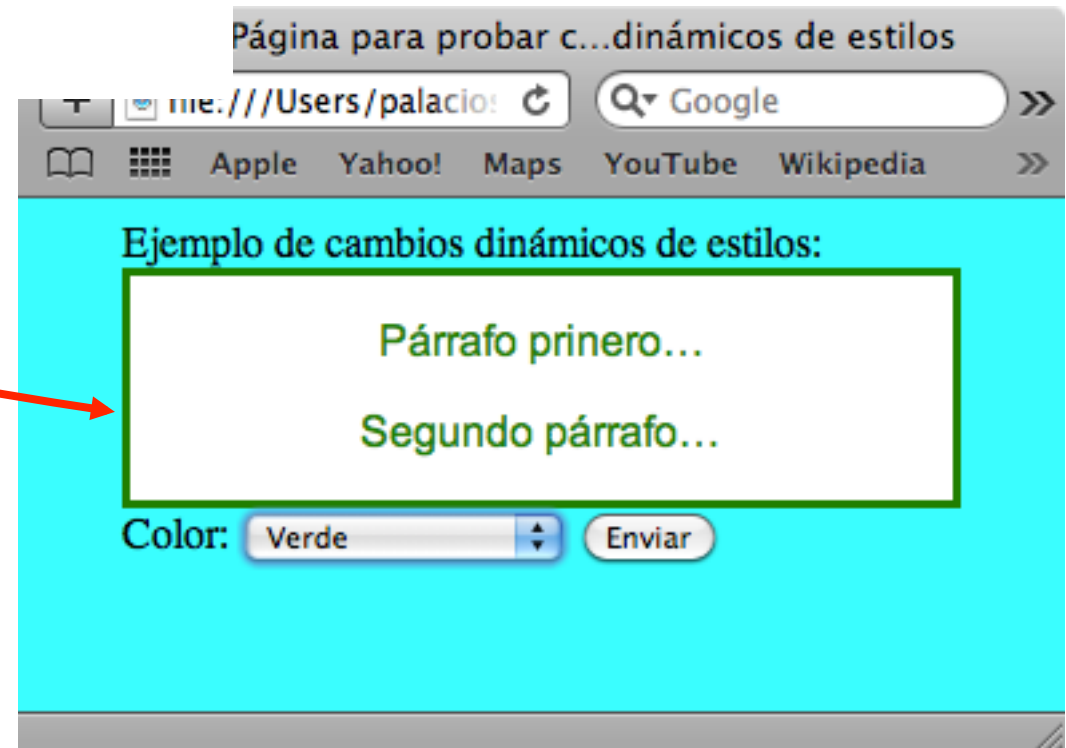
Estilos dinámicos

- Objeto div con estilo de tipo "cuadro" llamado "id_cuadro1"

```
<DIV class="cuadro" id="id_cuadro1">  
  <p>Párrafo printero...</P>  
  <p>Segundo párrafo...</p>  
</DIV>
```

mis_estilos.css

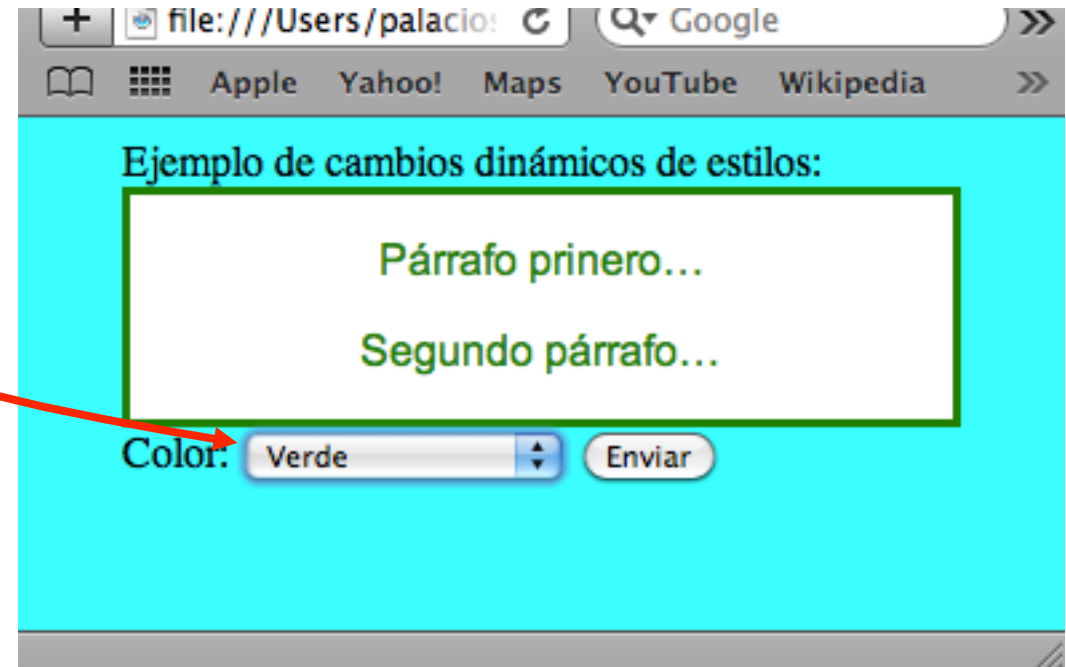
```
div.cuadro {  
  font-family:arial,Helvetica,sans-serif;  
  color:black;  
  background:white;  
  text-align:center;  
  border-style:solid;  
}
```



Estilos dinámicos

- Objeto `<select>` que llama a JavaScript para eventos de tipo "OnChange"

```
<SELECT NAME="colores" OnChange="CambiarColor(this.value,'id_cuadro1');">  
  <OPTION VALUE="">--Elegir color--</OPTION>  
  <OPTION VALUE="r">Rojo</OPTION>  
  <OPTION VALUE="g">Verde</OPTION>  
  <OPTION VALUE="b">Azul</OPTION>  
</SELECT>
```



Estilos dinámicos

- Código para crear el formulario

```
<FORM ACTION="http://www.iit.upcomillas.es/cgi-bin/test-cgi" METHOD="POST">
Color:
<SELECT NAME="colores" OnChange="CambiarColor(this.value,'id_cuadro1');">
  <OPTION VALUE="">--Elegir color--</OPTION>
  <OPTION VALUE="r">Rojo</OPTION>
  <OPTION VALUE="g">Verde</OPTION>
  <OPTION VALUE="b">Azul</OPTION>
</SELECT>
<INPUT TYPE="submit" VALUE="Enviar">
</FORM>
```



Estilos dinámicos

- Función JavaScript definida en <HEAD>

```
<HEAD>
<TITLE>Página para probar cambios dinámicos de estilos</TITLE>
<LINK href="mis_estilos.css" type="text/css" rel="stylesheet">

<SCRIPT type="text/javascript">
  function CambiarColor(color,nombre_objeto)
  {
    var objeto=document.getElementById(nombre_objeto);
    if (color=="r") objeto.style.color="red";
    if (color=="g") objeto.style.color="green";
    if (color=="b") objeto.style.color="blue";
  }
</SCRIPT>
</HEAD>
```



