

# 3.1 ELABORACIÓN DE HOJAS DE ESTILO

...

## 1. HOJAS DE ESTILO EN LA CONSTRUCCIÓN DE PÁGINAS WEB

# Estilos CSS: Cascade Style Sheet

- Permiten la modificación del aspecto visual del contenido de una página web.
- Permiten la adaptación de una página web al dispositivo desde el que se está accediendo.

# Estilos CSS: Cascade Style Sheet

## Definición de estilos:

- En línea:  
Dentro de la etiqueta HTML

`<p style="color:red;">Esto es un párrafo en rojo</p>`

## Desventajas:

- Repetición de código CSS donde se necesite.
- Mezcla de códigos HTML y CSS.

# Estilos CSS: Cascade Style Sheet

## Definición de estilos:

- En línea:  
Dentro de la etiqueta HTML
- Incrustados:  
Dentro del documento HTML

```
...<head>
    <style>
        p {
            color: red;
        }
    ...
</style>
...
</head>...
```

## Desventajas:

- Repetición de código CSS para distintos docs.
- Mezcla de códigos HTML y CSS.
- Aumento longitud de archivo.

# Estilos CSS: Cascade Style Sheet

## Definición de estilos:

- En línea:  
Dentro de la etiqueta HTML
- Incrustados:  
Dentro del documento HTML
- Enlazados/Importados:  
En un fichero independiente

```
...<head>  
    <link rel="stylesheet" href="css/styles.css">  
    ...  
</head>...
```

```
styles.css:  
p {  
    color: red;  
}
```

- Separación de la información/contenido, de su definición de aspecto.

# Estilos CSS: Cascade Style Sheet

## Definición de estilos:

- En línea:  
Dentro de la etiqueta HTML
- Incrustados:  
Dentro del documento HTML
- Enlazados/**Importados**:  
En un fichero independiente

```
...<head>
    <style>
        @import
url('http://otroSitioWeb/css/estilo.css')
    </style>
</head>...
```

```
styles.css:
p {
    color: blue;
}
```

## Desventajas:

- Fichero y definiciones fuera de nuestro control

# Adecuación de hojas de estilo

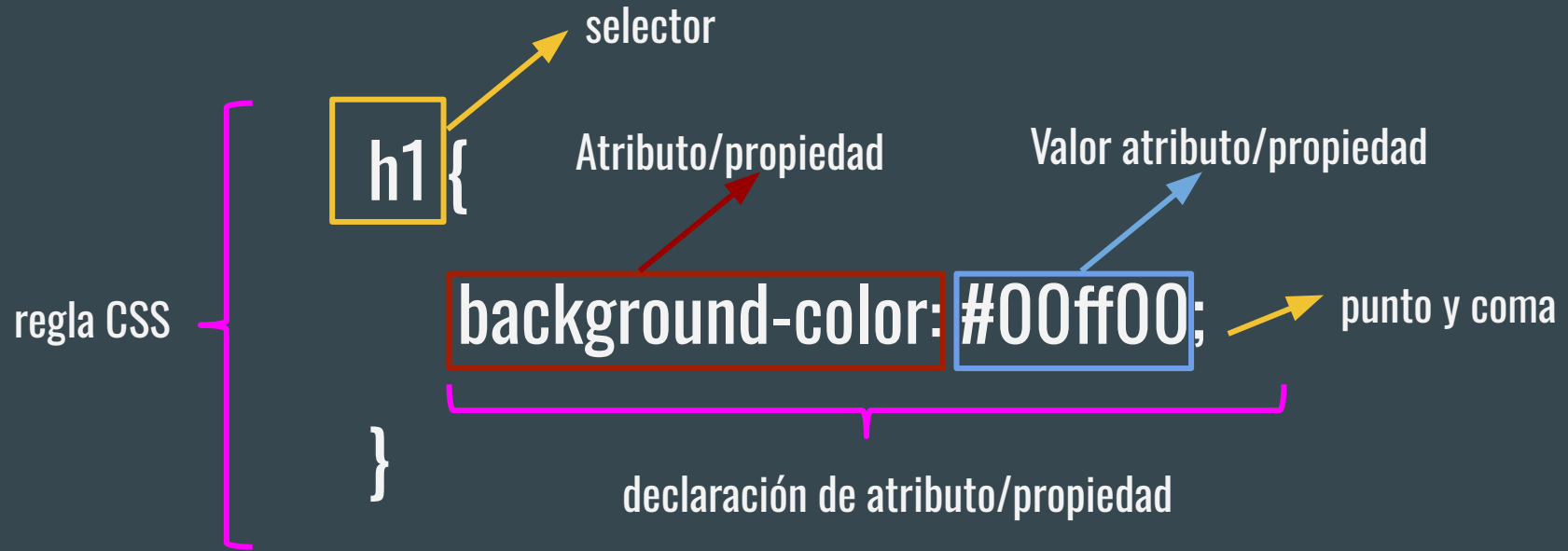
- Intente utilizar el menor número de hojas de estilo en cada web.
- Intente utilizar hojas de estilo enlazadas, evitando utilizar estilos incrustados y en línea.
- Para indicar el tamaño de la letra, utilice “em”, en lugar de “píxeles”. (em es una unidad relativa, respecto al tamaño de letra del elemento).
- Si tenemos varias hojas de estilo, evitaremos las normas duplicadas y los conflictos.
- Utilizaremos medidas relativas. Evitaremos la utilización de medidas absolutas, salvo que sea imprescindible.
- Siempre que sea posible, utilizaremos un texto alternativo. Por ejemplo, en el caso de las imágenes, utilizaremos el atributo “alt”, para incluir un texto alternativo a la imagen. Estos textos serán leídos por los lectores de pantalla para personas invidentes o con visión reducida.
- Intente utilizar un tipo de letra genérico.
- No utilice elementos que no estén marcados como estándares por la W3C, como por ejemplo Blink o Marquee.
- Evite representar el texto con imágenes. Utilice las hojas de estilo para conseguir textos atractivos.

# Adecuación de hojas de estilo (II)

- Indique los colores con números en lugar de con letra. Por ejemplo el rojo sería #f13d1a en hexadecimal, frente a utilizar un nombre en inglés (red).
- Intente crear páginas web con colores que faciliten su lectura. Evite transmitir información solo con colores.
- Maquete el contenido de su web mediante hojas de estilo. Nunca utilice tablas para maquetar, ya que un lector de pantalla detectará la información como el contenido de una tabla sin serlo.
- Verifique que la información de su página puede leerse sin problemas si se desactiva la hoja de estilo.
- Realice un desarrollo responsive de su web, es decir, desarrolle distintas hojas de estilo que adapten su página a las características de los distintos dispositivos en los que pueda ser visualizada.



# Estructura y sintaxis de una regla CSS



# Sintaxis del color en CSS

- Nombre del color (en inglés): background-color: green;
- Valor componentes RGBA (0-255): background-color: rgba(0, 255, 0, .5);

Diagram illustrating the syntax of the CSS `background-color: rgba(0, 255, 0, .5);` property, showing the components of the RGBA color value:

- Valor componente rojo (R): 0
- Valor componente verde (G): 255
- Valor componente azul (B): 0
- Valor componente transparencia (A): .5

# Sintaxis del color en CSS

- Nombre del color (en inglés): background-color: green;
- Valor componentes RGBA (0-255): background-color: rgba(0, 255, 0, .5);
- Valor componentes RGB en hexadecimal (00-ff 256 valores) sin componente transparencia:

Valor componente rojo (R)      Valor componente verde (G)

background-color: #00ff00;

Valor componente azul (B)

# Sintaxis del color en CSS

- Nombre del color (en inglés): background-color: green;
- Valor componentes RGBA (0-255): background-color: rgba(0, 255, 0, .5);
- Valor componentes RGB en hexadecimal (00-ff 256 valores) sin componente transparencia:
- Versión reducida, sólo para ciertos colores:

Valor componente rojo (R)      Valor componente verde (G)

background-color: #0f0;

Valor componente azul (B)

# Selectores CSS: Básicos

- Selector universal: \*

Selecciona todos los elementos del documento

```
* {  
    margin: 0;  
    padding: 0;  
}
```

# Selectores CSS: Básicos

- Selector universal: `*`
- Selector etiqueta: `h1`

Selecciona todos los elementos con esa etiqueta

```
h1 {  
    color: #0000ff;  
}
```

# Selectores CSS: Básicos

- Selector universal: `*`
- Selector etiqueta: `h1`
- Selector de clase: `.miclase`

Selecciona todos los **elementos** en cuyo atributo class se incluya el nombre de la clase:

```
.texto-azul {  
    color: #0000ff;  
  
}
```

`<p class="texto-azul"> párrafo en azul </p>`

`<p class="texto-azul negrita col-xs-6">Este párrafo tb es en azul </p>`

# Selectores CSS: Básicos

- Selector universal: `*`
- Selector etiqueta: `h1`
- Selector de clase: `.miclase`
- Selector de ID: `#menu`

Selecciona todos los **elementos** con atributo id igual a ese nombre:

```
#menu {  
    color: #0000ff;  
  
}
```

```
<ul id="menu">
```

```
    <li>...</li>
```

```
</ul>
```



# Selectores CSS: Concatenación

- Selector universal: `*`
- Selector etiqueta: `h1`
- Selector de clase: `.miclase`
- Selector de ID: `#menu`

`<p>hola mundo!</p>`

`<p id="miclase">hola mundo!</p>`

`<p class="miclase">hola mundo!</p>`

`<h1 class="miclase">hola mundo!</h1>`

`<h1 id="menu" class="miclase">hola mundo!</h1>`

`<h1 id="menu" class="miclase tuclase">hola mundo!`

## Concatenación de selectores

```
p.miclase {  
    color: #0000ff;  
}
```

```
h1#menu.miclase {  
    background-color:  
#56ff26;  
}
```

¿qué líneas se verán afectadas?

# Selectores CSS: Concatenación

- Selector universal: \*
- Selector etiqueta: *h1*
- Selector de clase: *.miclase*
- Selector de ID: *#menu*

<p>hola mundo!</p>

<p id="miclase">hola mundo!</p>

<p class="miclase">hola mundo!</p>

<h1 class="miclase">hola mundo!</h1>

<h1 id="menu" class="miclase">hola mundo!</h1>

<h1 id="menu" class="miclase tuclase">hola mundo!

## Concatenación de selectores

```
p.miclase {  
    color: #0000ff;  
}
```

```
h1#menu.miclase {  
    background-color: #56ff26;  
}
```

# Selectores CSS: Concatenación

- Selector universal: \*
- Selector etiqueta: *h1*
- Selector de clase: *.miclase*
- Selector de ID: *#menu*

```
<p class="mi-clase">hola mundo!</p>
```

```
<h1 class="mi-otra-clase">hola mundo!</h1>
```

## Concatenación de selectores

### Selectores separados por comas

```
.mi-clase, .mi-otra-clase {  
    color: #0000ff;  
}
```

# Selectores CSS: Concatenación

- Selector universal: `*`
- Selector etiqueta: `h1`
- Selector de clase: `.miclase`
- Selector de ID: `#menu`

Los estilos definidos para IDs tienen prioridad sobre los definidos sobre una clase.

Concatenación de selectores

Selectores separados por comas

Prioridad de ID sobre Clases

# Ejercicio 1

Usar selectores de clase, ID, y combinados

Usar al menos estos tipos de selectores:

- universal
- al menos un ID
- al menos una clase
- una combinación de etiqueta+id o clase
- una mismo estilo para varias clases, ids, y etiquetas

---

# Selectores CSS: De contexto

Son selectores en función de la posición entre ellos.

- `sel1 sel2 ... selN`

```
div p span {  
  font-style:italic;  
  color: #ff0000;  
}
```

*(ejemplo1.html)*

Se leería de forma inversa a la escrita: `sel1 sel2 ... selN`  


Selecciona el elemento `selN` que está contenido dentro de un elemento `selN-1`, ... que está dentro de un `sel2` que está dentro de un `sel1`

```
<div>
```

```
  <p> Lorem ipsum dolor sit amet, consectetur  
adipisicing elit, sed do eiusmod tempor <span>incidunt  
</span> ut labore et dolore</p>
```

```
</div>
```

```
<p> Lorem ipsum dolor sit amet, consectetur adipisicing  
elit, sed do eiusmod tempor <span>incidunt </span> ut  
labore et dolore</p>
```

# Selectores CSS: De contexto

Son selectores en función de la posición entre ellos.

- `sel1 sel2 ... selN`
- `sel1+sel2`

```
div+span {  
    background-color: #fab111;  
}
```

*(ejemplo2.html)*

Selecciona el elemento `sel2` definido **inmediatamente a continuación** del elemento `sel1`

```
<div>  
    <h1>Título de ejemplo</h1>  
    <span> A este NO afectaría</span>  
</div>  
    <span> A este SI afectaría</span>  
<p><span>A este NO afectaría</span></p>
```

# Selectores CSS: De contexto

Son selectores en función de la posición entre ellos.

- `sel1 sel2 ... selN`
- `sel1+sel2`
- `sel1>sel2`

```
div>span {  
    background-color: #9812ff;  
}
```

(*ejemplo3.html*)

Selecciona los elementos `sel2` contenidos dentro de un elemento `sel1`.

Se suele decir que `sel1` es el elemento *padre*, o contenedor, de `sel2`

```
<div>  
    <h1>Título de ejemplo</h1>  
    <span> A este SI afectaría</span>  
</div>  
<span> A este NO afectaría</span>  
<div>  
    <p><span>A este NO afectaría</span></p>  
</div>
```



# Selectores CSS: De contexto

Son selectores en función de la posición entre ellos.

- `sel1 sel2 ... selN`
- `sel1+sel2`
- `sel1>sel2`
- `sel1~sel2`

```
p ~ ul {  
    background: #992233;  
}
```

*(ejemplo4.html)*

~ -> Alt Gr + 4

Afecta a los elementos de tipo `selector2` que están definidos a raíz de la primera definición de `selector1`.

```
<h1>Carta</h1>  
<ul>  
    <li>Cazón en adobo</li>  
    <li>Chipirones fritos</li>  
</ul>  
<p>Carnes</p>  
<ul>  
    <li>Carrillada</li>  
    <li>Ternera</li>  
</ul>
```

# Ejercicio 2

Usar selectores de contexto

Usar al menos estos tipos de selectores:

- padre hijo
- selector1+selector2
- selector1>selector2
- selector1~selector2

---



*That's all Folks!*

*...or not?*