

Unidad 3

Definición de estilos en entornos web: CSS

CONTENIDOS

- 3.1. Introducción, evolución y estado actual de CSS**
- 3.2. Estructura y sintaxis de CSS**
- 3.3. Aplicación de CSS**
- 3.4. Prioridades y orden de CSS**
- 3.5. Valores y unidades de CSS**
- 3.6. Selectores CSS**
- 3.7. Propiedades CSS**

3.1. Introducción, evolución y estado actual de CSS

- CSS son las siglas de Cascading Style Sheets («hojas de estilo en cascada»), y es el lenguaje que se utiliza para definir el aspecto de las páginas HTML y XHTML.

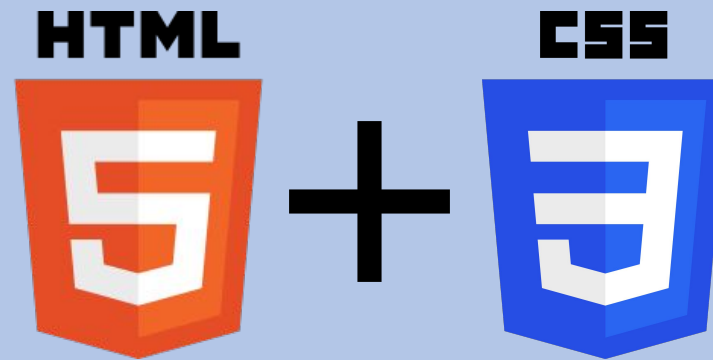


Figura 3.1. CSS proporciona estilo a las páginas HTML y XHTML.

- CSS permite modificar el aspecto de un mismo documento HTML

Las tecnologías básicas para crear contenidos Web son tres:

- HTML5
- CSS3
- JavaScript

Las tecnologías básicas para crear contenidos Web son tres:

HTML5
CSS3
JavaScript

Figura 3.2. Un mismo documento puede mostrarse de múltiples formas distintas gracias a CSS.

- Las versiones de CSS son gestionadas por el W3C

AÑO	VERSIÓN	OBSERVACIONES
1996	CSS1	El W3C ya no mantiene esta versión.
1998	CSS2	El W3C ya no mantiene esta versión.
2011	CSS2.1	Corrige errores de CSS2.
2012	CSS3	Esta versión está dividida en módulos, por lo que no existe una única fecha global de cambio de versión.

Figura 3.3. Versiones de CSS.

3.2. Estructura y sintaxis de CSS

- Los estilos CSS se aplican mediante reglas.
- Las reglas están compuestas por:
 - Selectores.
 - Declaraciones.

```
body {  
  
    text-align: center;  
    font-family: 'Roboto', sans-serif;  
    background-color: black;  
    color: white;  
  
}
```

Figura 3.4. Ejemplo de regla CSS.

3.3. Aplicación de CSS

- La aplicación de estilos CSS a un documento HTML se puede realizar de tres maneras:

- Como un documento CSS externo.

```
<link rel="stylesheet" href="ejemplo1_dark.css">
```

- Como una declaración CSS interna.

```
<style>
  ul {
    list-style: none;
  }
</style>
```

- Como una asignación inline:

```
<ul style="list-style: none;">
```



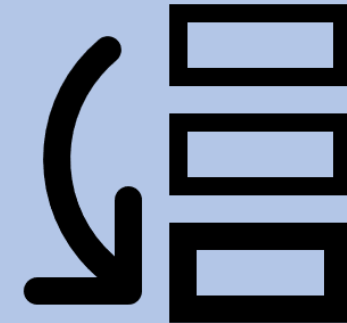
3.4. Prioridades y orden de CSS

- **CSS admite múltiples estilos aplicables a los mismos elementos.**

- **¿Cómo decidir cuál aplicar?**



- **Conceptos clave:**
 - **Origen e importancia.**
 - Estilo del navegador (menos prioridad).
 - Estilo del usuario.
 - Estilo del diseñador (más prioridad).
 - **Nivel de especificidad.**
 - Más específico \Rightarrow Mayor prioridad.
 - **Orden de aparición.**
 - Último estilo se impone a los anteriores.



- **Reglas:**

- La regla definida con el selector universal (*) se aplica a todos los elementos.
- Si una propiedad se modifica en un único punto, no hay conflicto y se aplica.
- Si una propiedad se modifica en el atributo style, se impone al intento de modificación de la misma propiedad en el elemento <style> o en una hoja CSS externa.
- Si una propiedad se modifica en el elemento <style>, se impone al intento de modificación de la misma propiedad en una hoja CSS externa.
- Si dos o más hojas CSS modifican la misma propiedad, se impone la hoja que se ha añadido en último lugar.
- Si dos reglas modifican la misma propiedad del mismo elemento, se impone aquella cuyo selector es más específico.
- En caso de igualdad en las prioridades se impone el último estilo aplicado.



- **Modificadores de la herencia de reglas:**
 - **inherit.** Activa la herencia, haciendo que la propiedad en la que se está asignando como valor herede la configuración del elemento padre.
 - **initial.** Al indicar este valor a una propiedad se consigue que tome el valor de la hoja de estilos por defecto para el elemento al que hace referencia.
 - **unset.** Restablece el valor de la propiedad a su valor natural.
 - **revert.** Restablece el valor de la propiedad al valor que habría tenido si no hubiese sufrido cambios desde ningún origen.
- La declaración **important!**

3.5. Valores y unidades en CSS

- **Tipos de datos admitidos en CSS:**

Tipo	Descripción
Entero	Número entero, ya sea positivo o negativo. Por ejemplo 10 ó -3.
Número	Un número decimal. Utiliza el punto como separador decimal. Si no hay decimales se omite el punto.
Dimensión	Es un número con una dimensión asociada: deg (grados), s (segundos) o px (píxeles) son algunos ejemplos.
Porcentaje	Una fracción de un total, normalmente referente a la dimensión del padre del elemento al que se está aplicando. Por ejemplo 60% en un font-size representa dicho porcentaje del tamaño del texto del elemento padre. Si, en cambio, se aplica un porcentaje a la propiedad width se aplica dicho porcentaje sobre el ancho del elemento contenedor.

Figura 3.5. Tipos de datos de CSS.

- **Unidades de longitud absolutas.**

Unidad	Nombre
px	Píxel
cm	Centímetro (96px/2,54)
mm	Milímetro
Q	Cuarto de milímetro
in	Pulgada (96px)
pt	Puntos (1/72 de in)
pc	Picas (1/16 de in)

Figura 3.6. Unidades de longitud absolutas de CSS.

- **Unidades de longitud relativas.**

Unidad	Nombre
em	Tamaño de letra del elemento padre.
ex	Altura de la fuente del elemento.
ch	Ancho del carácter “0” de la fuente del elemento.
rem	Tamaño de letra del elemento raíz.
lh	Altura de la línea del elemento.
vw	1% del ancho de la ventana gráfica.
vh	1% del alto de la ventana gráfica.
vmin	1% de la dimensión más pequeña de la ventana gráfica.
vmax	1% de la dimensión más grande de la ventana gráfica.

Figura 3.7. Unidades de longitud relativas de CSS.


- **Colores.**
 - Se pueden utilizar para asignar color a textos, fondos, bordes, sombras, etc.
 - Se pueden expresar de múltiples formas.
 - Por nombre: **red**.
 - Por valor RGB: **rgb(255,0,0)**.
 - Por valor hexadecimal: **#ff0000**.
 - Por valor HLS: **hsl(0, 100%, 50%)**.



Figura 3.8. Diversas alternativas para indicar el color ROJO.

- **Imágenes.**
 - **Admiten archivos y degradados de color (gradientes).**
 - **Los archivos:**
 - **Se referencian mediante la función url().**
 - **Pueden ser internos o ubicados en un servidor distinto.**

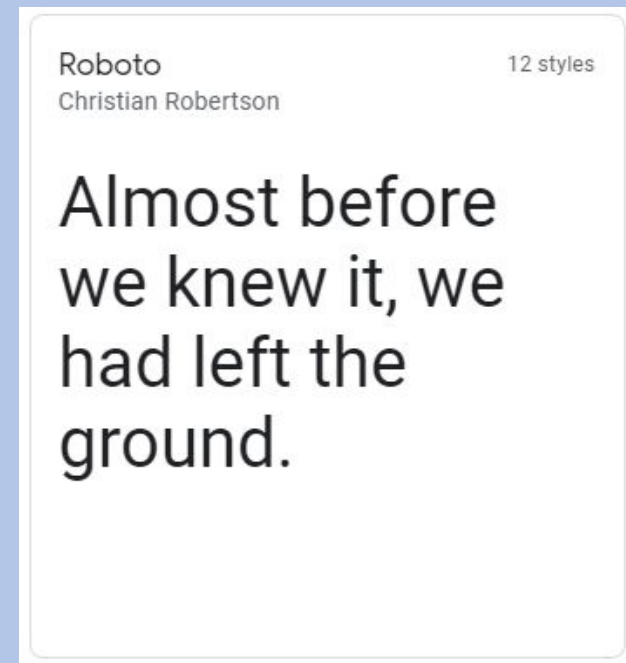
El diario *La Vanguardia* publicó la noticia en portada concediéndole la importancia que se merecía: "Los habitantes de la Tierra, en vela permanente, contemplan, a través de la Televisión, con asombro y emoción como un hombre llamado Neil Armstrong baja por una escalera y pone el pie en la Luna".



Los huertos ecológicos son cada vez más populares gracias a la calidad de sus productos.

- **Tipos de letras.**

- El diseño web permite utilizar múltiples tipos de letra pero estos tienen que estar disponibles en el dispositivo donde se utilizan.
- Existe un conjunto de tipos de letra seguros (siempre disponibles): Arial, Courier New, Georgia, Times New Roman, Trebuchet MS y Verdana.
- En CSS existen los tipos de letra genéricos: serif, sans-serif, monospace, cursive y fantasy. Se pueden utilizar como alternativas al tipo de letra elegido si no existe.



3.6. Selectores CSS

- Un selector determina sobre qué elementos aplicar un estilo.
- Selectores básicos:

Nombre	Selecciona	Sintaxis
Universal	Todos los elementos	*
De tipo	Todos los elementos de un tipo determinado	nombreElemento
De clase	Todos los elementos de una determinada clase	.nombreClase
De identificador	El elemento con el identificador indicado	#identificador
De atributo	Los elementos que tienen un determinado atributo con unas características específicas.	[atributo]

Figura 3.9. Selectores CSS básicos.

- **Agrupación de selectores:**

- Se pueden agrupar selectores para las mismas reglas se apliquen a distintos elementos. Sintaxis:

```
elemento, elemento, elemento { propiedad : valor }
```

- **Ejemplo:**

```
h1, h2, h3 {  
    color: blue;  
}
```

- **Combinadores:**

- Estos selectores tienen en cuenta la relación jerárquica entre los elementos.

Nombre	Combinador	Descripción
De hermanos	$A \sim B$	A y B son hermanos.
De hijos	$A > B$	B es hijo de A.
De hermanos adyacentes	$A + B$	A y B son hermanos y B está inmediatamente a continuación de A.
De descendientes	$A B$	B es descendiente de A, pero no necesariamente es hijo directo.

Figura 3.10. Combinadores CSS.

- **Pseudoclases:**

- Establecen un nuevo nivel de filtrado en los selectores en función del estado del elemento o elementos afectados.

- **Sintaxis:**

```
selector:pseudoclase { propiedad: valor; }
```

- **Algunas pseudoclases:**

Nombre	Estado del elemento o elementos proporcionados por el selector
:active	El elemento ha sido activado por el usuario.
:checked	Afecta a elementos <input> de tipo radio o checkbox o a elementos <option> cuando han sido marcados.
:default	Elemento de un formulario marcados como predeterminado.
:disabled	El elemento está deshabilitado.
:empty	El elemento no tiene hijos.

Figura 3.11. Ejemplo de *pseudoclases*. El listado completo es mucho más extenso.

- Ejemplo de aplicación de pseudoclases:
 - Mostrar de manera distinta las filas de una tabla en función de si están en posiciones pares o impares.

CONCEPTO	PRIMER SEMESTRE	SEGUNDO SEMESTRE	TOTAL
Alimentación	1500	1750	3250
Tecnología	1500	1750	3250
Deporte	1500	1750	3250
Calzado	1500	1750	3250
Viajes	1500	1750	3250
Papelería	1500	1750	3250
TOTAL	0000	0000	0000

Figura 3.12. Ejemplo de aplicación de *pseudoclases* a una tabla.

- ***Pseudoelementos:***

- Permiten especificar selectores que afecten a conjuntos más precisos de elementos.
- Sintaxis:

```
selector::pseudo-elemento { propiedad: valor; }
```

- ***Pseudoelementos más utilizados:***

Nombre	Descripción
::after	Permite añadir contenido con el atributo content después del elemento seleccionado.
::before	Permite añadir contenido con el atributo content antes del elemento seleccionado.
::first-letter	Primera letra del primer bloque de un elemento seleccionado.
::first-line	Primera línea del primer bloque de un elemento seleccionado.
::selection	Permite modificar el estilo de la selección realizada con el ratón.

Figura 3.13. Relación de *pseudoelementos*.

- **Pseudoelementos:**
 - Ejemplo: incrustar contenido al final del bloque y modificar el tamaño de la primera letra de un texto.

```
<p id="ElQuijote">
```

```
En un lugar de la Mancha, de cuyo nombre no quiero acordarme,  
no ha mucho tiempo que vivía un hidalgo
```

```
</p>
```

```
#ElQuijote:after {  
    content: "SEGUIR LEYENDO...";  
}  
#ElQuijote::first-letter {  
    font-size: 3em;  
}
```

En un lugar de la Mancha, de cuyo nombre no quiero acordarme, no ha mucho tiempo que vivía un hidalgo... Es, pues, de saber que este sobredicho hidalgo, los ratos que estaba ocioso —que eran los más del año—, SEGUIR LEYENDO...

Figura 3.14. Ejemplo de aplicación de *pseudoelementos* a un párrafo.

3.7. Propiedades CSS

- Las propiedades permiten modificar los aspectos de presentación de los elementos.
- Sintaxis de la propiedad:

Nombre-propiedad: valor-propiedad;

- El modelo de cajas:
 - Todos los elementos se encuentran ubicados en un contenedor rectangular denominado caja.
 - Propiedades básicas de las cajas:
 - Margen exterior: margin.
 - Margen interior: padding.
 - Borde: border.
 - Contorno: outline.
 - Ancho: width.
 - Alto: height.

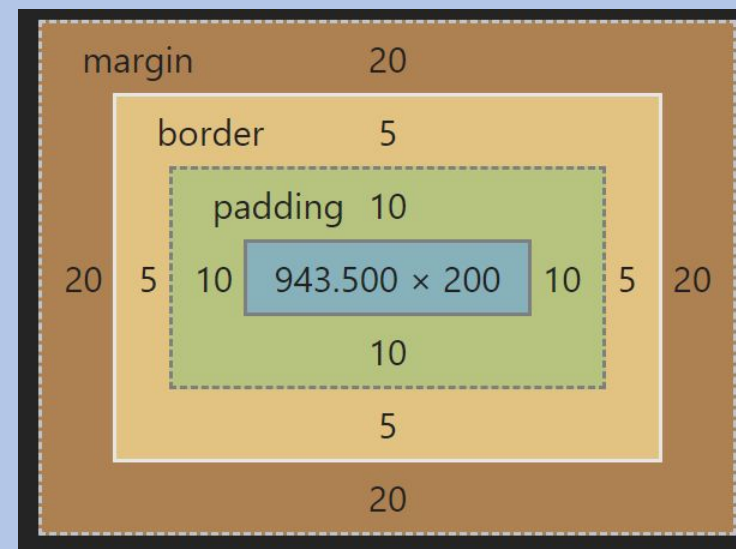


Figura 3.15. Modelo de cajas presentado por un navegador web para un documento HTML.

- **El modelo de cajas:**
 - **Cajas de tipo bloque:**
 - Después de la caja se realiza un salto de línea.
 - Suele ocupar el 100% del espacio del contenedor en el que se encuentra.
 - Se aplican los valores de las propiedades width y height.
 - Los márgenes, bordes y rellenos desplazan al resto de cajas.
 - **Cajas de tipo línea.**
 - No genera salto de línea.
 - Por defecto, ocupan el espacio que ocupa su contenido.
 - No se aplican los valores de las propiedades width y height.
 - Los márgenes, bordes y rellenos horizontales desplazan al resto de cajas en línea.
 - Los márgenes, bordes y rellenos verticales no desplazan al resto de cajas en línea.

- El modelo de cajas:
 - Cajas de tipo bloque:
 - Ejemplo: `<div>`
 - Cajas de tipo línea.
 - Ejemplo ``

```
<div>Caja de tipo bloque</div>  
<div>Caja de tipo <span>en línea</span> dentro de una caja  
de tipo bloque</div>
```



Caja de tipo bloque
Caja de tipo en línea dentro de una caja de tipo bloque

Figura 3.16. Cambio del comportamiento por defecto del modelo de cajas.

- **El modelo de cajas:**
 - **Se puede modificar el comportamiento por defecto con la propiedad `display`. Valores:**
 - ***inline*.** Hace que el elemento se comporte como elemento en línea.
 - ***block*.** Hace que el elemento se comporte como elemento de bloque.
 - ***flex*.** Hace que el elemento se comporte como elemento de bloque y de acuerdo al modelo *flexbox*.
 - ***grid*.** Hace que el elemento se comporte como elemento de bloque y de acuerdo al modelo de cuadrícula o malla.

- El modelo de cajas:
 - Ejemplo de uso de *display:flex*



Figura 3.17. *Flexbox* incrementa la flexibilidad en la maquetación de páginas HTML.

- El modelo de cajas:
 - Ejemplo de uso de *display:grid*

Primer contenedor	Segundo contenedor
Tercer contenedor	Cuarto contenedor
Quinto contenedor	Sexto contenedor

Figura 3.18. Ejemplo de distribución de tipo *grid*.

- Propiedad *float*.
 - Provoca que los elementos de HTML se posicionen en su contenedor según una serie de reglas.
 - Los principales valores son:
 - left: el elemento se “empuja” hacia la izquierda.
 - right: el elemento se “empuja” hacia la derecha.
 - none: el elemento se comporta según su comportamiento por defecto.
 - inherit: hereda el comportamiento de su padre jerárquico.

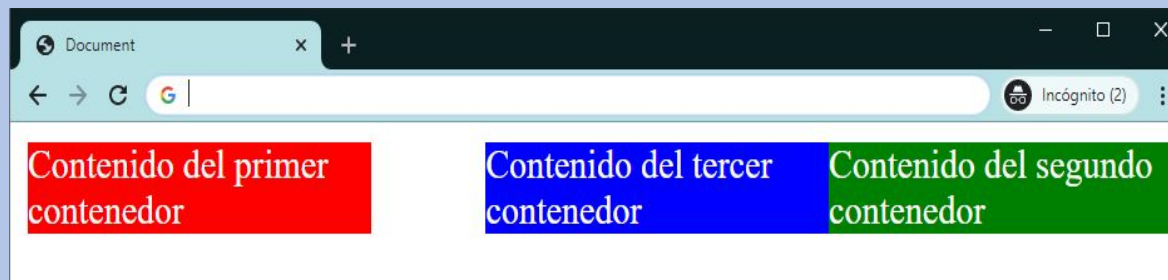


Figura 3.19. *float* “empuja” los bloques permitiendo distribuirlos por el espacio disponible.

- Propiedad *position*.
 - Determina como se posiciona un elemento respecto de su contenedor o del documento.
 - Valores:

Valor	Descripción
static	Es el valor por defecto. El elemento se posiciona siguiendo el flujo normal del documento.
relative	El elemento se posiciona de forma relativa a su posición por defecto.
absolute	El elemento se posiciona de manera relativa a su ancestro. Si no tiene ancestro se posiciona de manera relativa a la página completa.
fixed	El elemento se posiciona de manera relativa a la página completa.
sticky	Mediante las propiedades top, bottom, left y right se fijan unas posiciones límites relativas a su contenedor, de tal manera que no excederá de las mismas.

Figura 3.20. Con *position* se indica dónde ubicar un elemento con precisión.

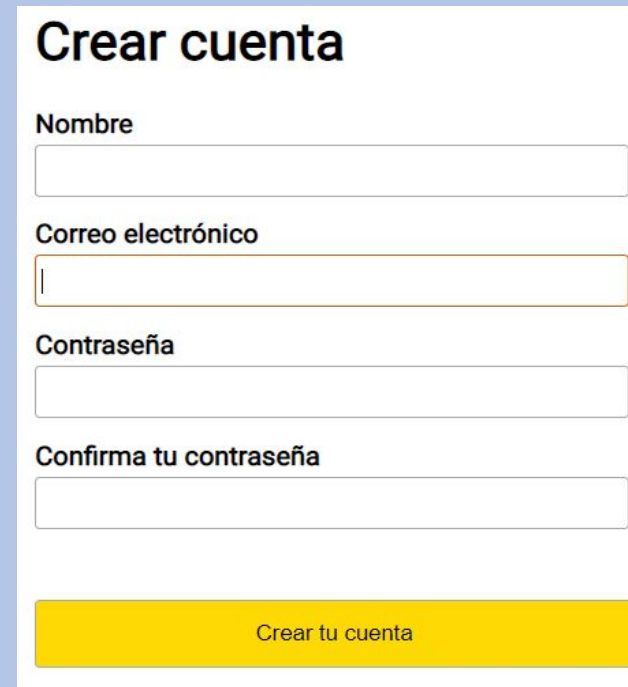
- Propiedad ***overflow***.
 - Determina como se comporta el contenido de un contenedor cuando excede del espacio que tiene disponible.

Valor	Descripción
visible	Es el valor por defecto. Provoca que el contenido desborde el espacio del contenedor consiguiendo así que sea visible por completo.
hidden	Oculto el contenido que desborda al contenedor.
scroll	Agrega unas barras de desplazamiento (scroll) permitiendo así consultar todo el contenido sin que este exceda de los límites del contenedor.
auto	Se comporta igual que el valor scroll pero sólo si el contenido excede del tamaño del contenedor.

Figura 3.21. ***overflow*** define el comportamiento del contenido de una caja cuando excede el espacio disponible.

- **Propiedades del texto.**
 - **Determinan como mostrar un texto.**
 - **Permiten determinar el color, la fuente de letra, el tamaño, el grosor del trazo, la alineación, etc.**
- **Propiedades de las tablas.**
 - **Determinan como mostrar las tablas y sus elementos.**
 - **Permiten determinar cómo han de presentarse los bordes, la alineación vertical de los elementos mostrados o la posición del título entre otros.**

- **Aplicación de propiedades a formularios.**
 - **Son propiedades generales, aplicables a cualquier elemento.**
 - **Existen técnicas de aplicación de las propiedades para generar formularios atractivos.**



Crear cuenta

Nombre

Correo electrónico

Contraseña

Confirma tu contraseña

[Crear tu cuenta](#)

Figura 3.22. Ejemplo de aplicación de estilo CSS a un formulario.

- **Propiedades de enlaces.**
 - **Permiten definir cómo han de mostrarse los enlaces en sus diferentes estados: activos, visitados, no visitados...**

Selector	Estado
a:link	Es el estado por defecto de un enlace.
a:visited	Se activa cuando el enlace ha sido activado. Suele estar deshabilitado en los navegadores modernos por razones de seguridad.
a:hover	Se activa cuando se pasa el ratón por encima del enlace.
a:focus	Se activa cuando el foco se posiciona en el enlace.
a:active	Se activa en el momento de la activación del enlace.

Figura 3.23. CSS permite determinar cómo han de mostrarse los enlaces.

- **Propiedades de fondos, bordes y contornos.**
 - **Permiten definir cómo han de mostrarse los fondos, bordes y contornos de los elementos.**
 - **Propiedad *background***
 - **Propiedad *border***
 - **Propiedad *outline***

- Diseño adaptativo con *media queries*.
 - Permiten declarar reglas variables que se aplicarán en función del dispositivo en el que se va a presentar la página (ancho de la pantalla, orientación del dispositivo, etc.)

```
@media (max-width:1000px) {  
    #parrafo {  
        color:springgreen;  
    }  
}  
@media (min-width:1001px) {  
    #parrafo {  
        color:rgb(171, 15, 199);  
    }  
}
```

Figura 3.24. Las *media queries* mejoran la adaptación de los contenidos a los diferentes dispositivos.