



Introducción a CSS

Irene López

CONTENIDO

01

CONCEPTOS BÁSICOS

02

ESPECIFICIDAD

03

HERENCIA

04

CASCADA

05

MODELO DE CAJA

HERENCIA

La herencia en CSS es el mecanismo mediante el cual determinadas propiedades de un elemento padre se transmiten a sus hijos.

Sin embargo, no todas las propiedades son heredadas.

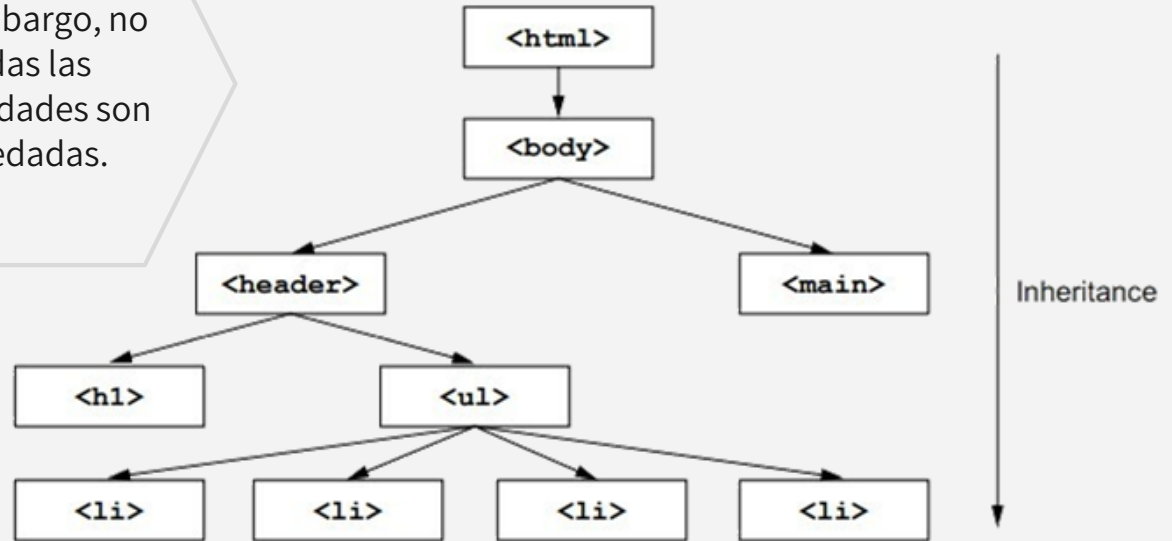


Tabla de especificaciones : <https://www.w3.org/TR/css-2010/#properties>

Control de la herencia

inherit

Establece que el valor de la propiedad que se aplica a un elemento determinado sea exactamente igual al del elemento padre.

initial

Establece que el valor de la propiedad que se aplica a un elemento seleccionado tenga el mismo valor que esté establecido para esa propiedad en la hoja de estilo por defecto del navegador.

unset

Restablece la propiedad a su valor natural, lo que significa que si la propiedad se hereda de forma natural, actúa como inherit, y en caso contrario como initial.

```
.color {  
  color: ■ red;  
}  
  
.color-inherit{  
  color: inherit;  
}  
  
.color-initial{  
  color: initial;  
}  
  
.color-unset{  
  color: unset;  
}
```

html

```
<div class="color">  
  <p> <a href="#">Link</a> por default</p>  
  <p> <a href="#" class="color-inherit">Link</a> por inherit</p>  
  <p> <a href="#" class="color-initial">Link</a> por initial</p>  
  <p> <a href="#" class="color-unset">Link</a> por unset</p>  
</div>
```

Link por default

Link por inherit

Link por initial

Link por unset

COMBINADORES

Combinador de descendientes

El combinador (espacio) selecciona los elementos que son descendientes del primer elemento.

Sintaxis:

```
selector1 selector2 { estilos  
}
```

Combinador de hijo directo

El combinador > selecciona los elementos que son hijos directos del primer elemento.

Sintaxis:

```
selector_padre > selector_hijo { estilos }
```

COMBINADORES

Combinador de hermanos adyacentes

El combinador + selecciona hermanos adyacentes. Es decir, sólo seleccionará un elemento especificado que esté inmediatamente después de otro elemento especificado.

Sintaxis:

```
elemento_anterior + elemento_afectado {  
    estilos  
}
```

Combinador de hermanos generales

El combinador ~ selecciona hermanos. Esto quiere decir que el segundo elemento sigue al primero (no necesariamente de forma inmediata) y ambos comparten el mismo elemento padre.

Sintaxis:

```
elemento ~ elemento { estilos  
}
```

CASCADA

La cascada es el algoritmo por el cual el navegador decide qué estilos CSS aplicar a un elemento

El orden de las reglas importa en CSS

ORDEN DE IMPORTANCIA:

1. Posición y orden de aparición
2. Especificidad
3. Importancia





Posición y orden de aparición

Orden de las reglas:

Cuando dos reglas tienen selectores con la misma especificidad, se aplica la que aparece en último lugar en el CSS.

CSS

```
.rojo{  
  color:  red;  
}  
  
.azul{  
  color:  blue;  
}
```

html

```
<body>  
  <h1 class="rojo azul">Título</h1>  
</body>
```

Posición y orden de aparición

Orden de las hojas de estilo:

Son las formas de incluir un archivo CSS en un documento HTML.

Orden de aplicación:





1. Externa
2. Inserción en secciones style
3. Inserción de estilos inline

Posición y orden de aparición

La posición de las declaraciones:


Si se especifican dos valores para la misma propiedad se aplica el que aparece en último lugar.

```
.color{  
  color:  red;  
  color:  blue;  
}
```

Especificidad

La especificidad es el modo que tiene el navegador de decidir qué regla se aplica si diversas reglas tienen selectores diferentes pero podrían aplicarse a un mismo elemento.

Orden de aplicación:

- 
1. etiqueta HTML
 2. clases | pseudo-clases | atributos
 3. id

Los combinadores (+, > y ~), el selector universal (*) y la pseudoclase `:not()` no tienen ningún efecto sobre la especificidad.

Propiedad !important

Se utiliza para convertir una propiedad y un valor particular en el elemento más específico, de modo que se invalidan las reglas normales de la cascada.

```
h1{  
  color: violet !important;  
}  
  
h1{  
  color: orange;  
}
```

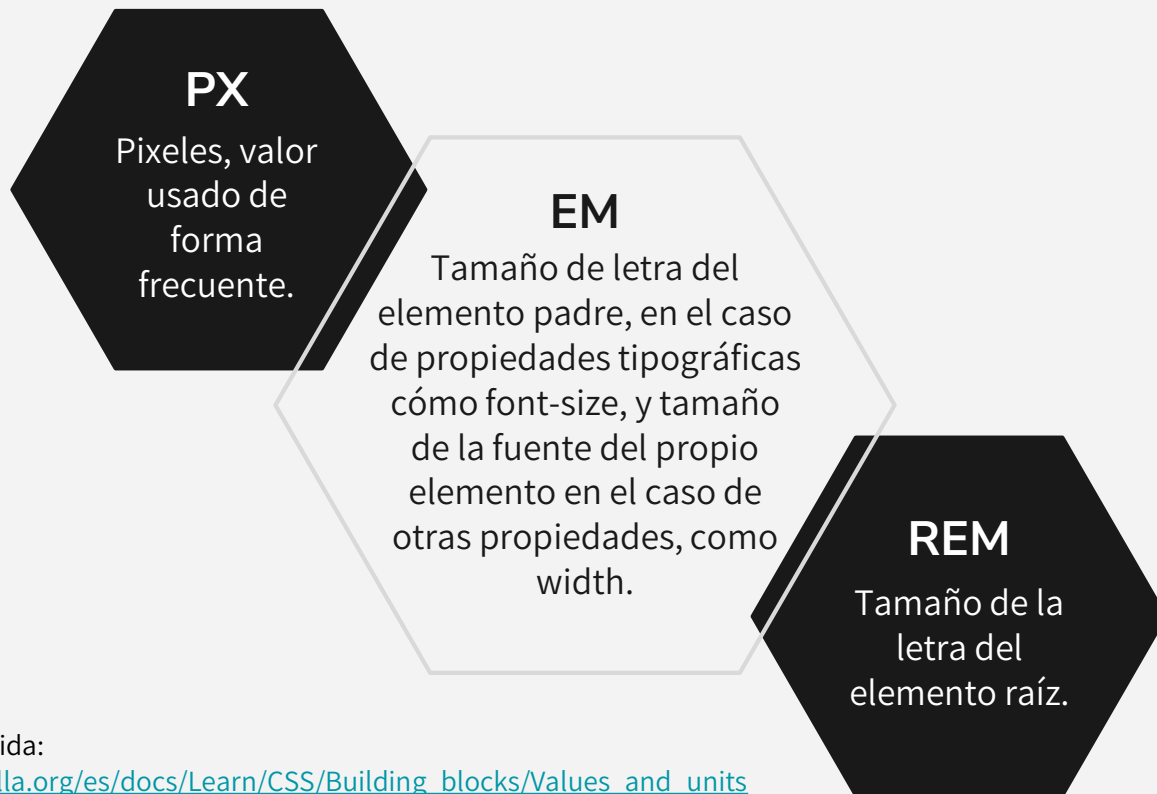
Nota: La única manera de anular la declaración !important sería incluir otra declaración !important en una declaración con la *misma especificidad* que aparezca más adelante en el orden del código fuente, o con una especificidad superior.

MODELO DE CAJA

Su flujo y posicionamiento

```
top: 0;  
left: 0;  
bottom: 0;  
width: 65%;  
margin-left: 12.5vw;  
margin-top: 100px;  
margin-bottom: 100px;  
margin-right: 40px;
```

UNIDADES DE MEDIDA EN CSS



Otras unidades de medida:

https://developer.mozilla.org/es/docs/Learn/CSS/Building_blocks/Values_and_units

HACK

```
html {  
  font-size: 62.5%;  
}  
  
body {  
  font-size: 16px;  
}
```

1rem = 10px