



Universidad Autónoma de Baja California

Aplicaciones web

Unidad 3: Programación del lado del cliente (Front-End)

Introducción a CSS

Profesor: MC. Itzel Barriba Cázares

CSS

- ▶ CSS (Cascading Style Sheets) permite **definir el diseño y la apariencia** de las paginas web mediante reglas de estilo aplicadas a los elementos HTML. Facilita la personalización del diseño al **permitir modificar colores, fuentes, márgenes, disposiciones de columnas, fondos y otros efectos visuales**.
- ▶ Su objetivo principal es agilizar y optimizar el desarrollo de interfaces web, separando la estructura del contenido de su presentación.

CSS

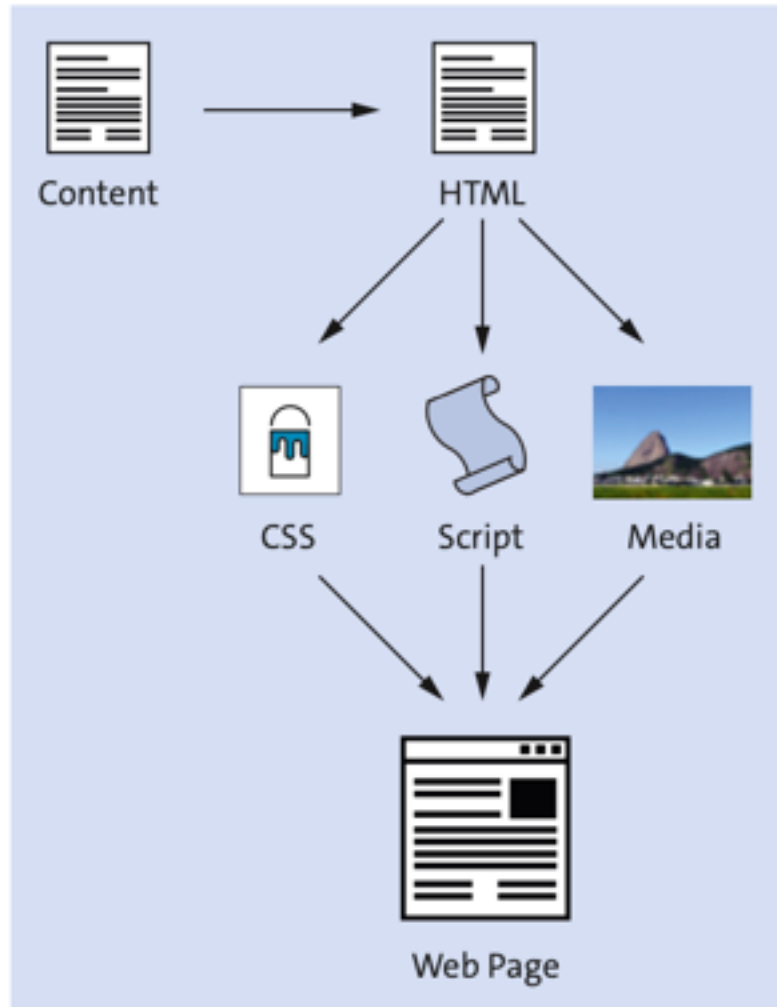


Imagen: Composición básica de los componentes de un sitio web sencillo

Hoja de estilo CSS Interna

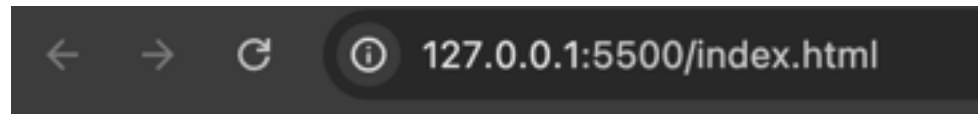
- **Interno:** Una forma de agregar estilo a una página web es **insertando las declaraciones requeridas en el encabezado** entre las etiquetas <head> y </head>.

```
<!DOCTYPE html>
<html>
<head>
  <title>App web.</title>
  <style>
    h1 { color:red; font-size:3em; font-family:Arial; }
  </style>
</head>
<body>
  <h1>Hola que tal!</h1>
</body>
</html>
```

Hoja de estilo CSS Interna

- **Interno:** Otra forma de agregar estilo es usando el atributo style en el elemento HTML para aplicar directamente en línea.

```
<!DOCTYPE html>
<html>
<head>
  <title>App web.</title>
</head>
<body>
  <h1 style="color: red; font-size: 15px;">Hola que tal!</h1>
</body>
</html>
```



Hola que tal!

Hoja de estilo CSS Externa

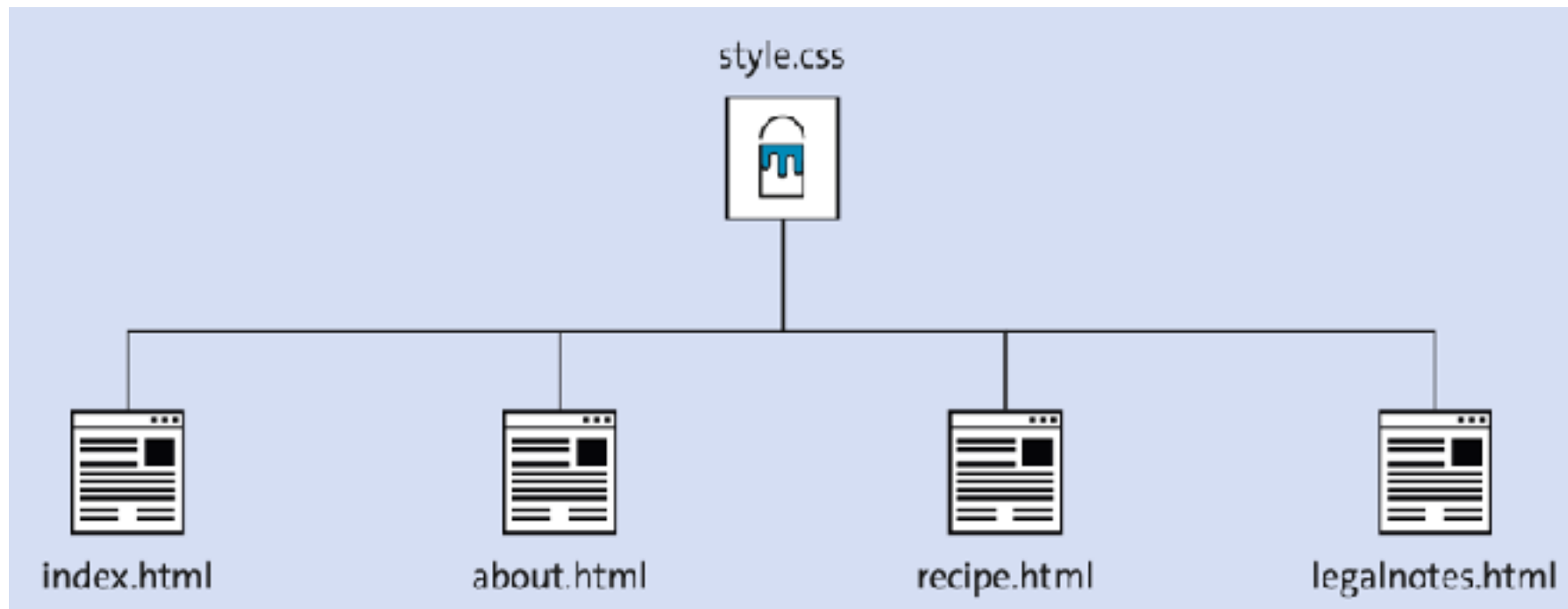
- ▶ **Externo:** Tener las hojas de estilo en un **archivo separado con extensión css** y luego **importar la hoja de estilo**.

```
<link rel='stylesheet' type='text/css' href='css/styles.css'>
```

- ▶ Esta declaración le dice al navegador que busque la hoja de estilo con el nombre “styles.css”.

Hoja de estilo CSS Externa

- Se recomienda el uso de **hojas de estilo externas** para mantener la separación entre contenido y presentación, lo que **mejora la mantenibilidad y la escalabilidad** de los proyectos web.



Regla CSS

- Una regla en CSS, es una instrucción que define el estilo de uno o varios elementos HTML. Cada regla consta de un **selector** y **declaraciones**.

```
Selector header {  
    background: #add8e6;  
    padding: 2px;  
    text-align: center;  
}
```

Propiedades Valores

Regla CSS

- Todos los cambios en las reglas deben realizarse dentro de los símbolos { y } que siguen al selector.

El selector de CSS

► Propiedades

- Especifica una característica CSS (ejemplo, color, fuente, alineación) que desea cambiar para el elemento HTML seleccionado con el selector.

► Valores

- Especifica el valor de la función CSS utilizada. Depende de la función CSS que se esta utilizando.

El selector de CSS

► Sintaxis:

```
Selector {  
    CSSProperty1: Value1;  
    CSSProperty2: Value2;  
    ...  
}  
Selector_01, Selector_02, Selector_03 {  
    CSSProperty1: Value1;  
    CSSProperty2: Value2;  
    ...  
}
```

- Puedes agrupar varios selectores independientes para las mismas declaraciones si los separa con comas.

Selectores en CSS

- ▶ **Selector simple:** incluye el selector de tipo, el selector universal (*), el selector de clase (.class), el selector de ID (#id), el selector de atributo y varias pseudoclases.
- ▶ **Combinadores:** son dos selectores concatenados por un signo mayor que (E > F; combinador hijo), el signo más (E+F; combinador hermano adyacente), un signo tilde (E~ F; combinador hermano general) o un espacio (E F; combinador descendiente)

Ejemplo de un selector

- Consta de dos partes: al menos una propiedad y un valor.

```
h1 {  
    font-family: "Arial";  
    color: red;  
    text-align: center;  
}  
h2, h3 {  
    font-family: "Courier";  
    color: blue;  
}
```

Ejemplo de selectores

- También es posible **aplicar una regla a varios elementos HTML** separando los elementos HTML individuales con comas:

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

Selector simple

► Direccionamiento de elementos HTML mediante el selector tipo

- El **selector tipo**, a veces también denominado selector de elemento HTML, es el selector mas simple. Un selector de este tipo disecciona los elementos HTML directamente con el nombre del elemento.

```
p {  
    text-align: justify;  
}
```

- **Selectores descendientes:** permiten aplicar estilos a elementos contenidos dentro de otros elementos.

```
p b {  
    color: red;  
}
```

- Los selectores descendientes pueden continuar anidando indefinidamente.

```
ul li b {  
    color: blue;  
}
```

Selector simple

► Direccionamiento de elementos HTML mediante el selector tipo

- El **selector tipo**, a veces también denominado selector de elemento HTML, es el selector mas simple. Un selector de este tipo disecciona los elementos HTML directamente con el nombre del elemento.

```
header, nav, footer {  
    text-align: center;  
    border: 1px solid black;  
    margin-top: 5px;  
}  
  
h1,  
  abbr { color: gray; }  
  
p { border: 1px dotted gray; }
```

- **Selectores descendientes:** permiten aplicar estilos a elementos contenidos dentro de otros elementos.

Selector simple

- ▶ De acuerdo con las reglas CSS, los elementos HTML individuales se seleccionan utilizando el selector de tipo correspondiente y se formatean de acuerdo con las declaraciones de la regla.

Header
Navigation

Type selectors

Such a type selector addresses the HTML elements directly via the element names.

This rule will be applied to all elements of the same type in the HTML document. With ... ,

Footer

Ejemplo:

- Considera el siguiente ejemplo:

```
h1 {  
  font-family: Arial, Helvetica,  
  sans-serif;  
  font-size: 160%;  
  color: blue;  
}  
h2 {  
  font-family: Arial, Helvetica,  
  sans-serif; font-size: 140%;  
  color: blue;  
}  
h3 {  
  font-family: Arial, Helvetica,  
  sans-serif; font-size: 120%;  
  color: blue;  
}
```

- Puedes acortar estas reglas CSS agrupándolas:

```
h1, h2, h3 {  
  font-family: Arial, Helvetica, sans-  
  serif;  
  color: blue;  
}  
h1 { font-size: 160%;}  
h2 { font-size: 140%;}  
h3 { font-size: 120%;}
```

Uso de comentarios para CSS

- ▶ Puedes introducir un comentario mediante `/*` y cerrarlo con `*/`. Todo lo que esté entre medio (incluidos los saltos de línea) serán ignorados por el navegador web.

```
/* This is a CSS comment */
```

- ▶ O puedes extender un comentario a muchas líneas, como esta:

```
/*  
A Multi  
line  
comment */
```

Clases

- ▶ **Clase:** es un selector que permite aplicar estilos a uno o varios elementos HTML, los cuales van a pertenecen a una clase en particular.
- ▶ En HTML, se utiliza el atributo “**class**” seguido por el nombre de la clase.
- ▶ **Sintaxis HTML:**

```
<p class="note">A paragraph text ...</p>
```

- ▶ En CSS se define con un punto (.) seguido del **nombre de la clase** de la siguiente manera:
- ▶ **Sintaxis CSS:**

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

Selector de Clases

- Puede utilizar varias clases a la vez en elementos HTML separando las clases individuales con un espacio, por ejemplo:
- Ejemplo CSS:

```
/* Font family for all p elements, irrespective of the class.1*/  
p { font-family: Verdana, Arial; }  
/* Style for a note*/  
.note {  
  margin-left: 50px;  
  border-left: 10px solid green;  
  padding-left: 5px;  
}  
/* Style for a warning*/  
.warning {  
  border-left: 10px solid red;  
  border-top: 2px solid red;  
  border-right: 10px solid red;  
  border-bottom: 2px solid red;  
  text-align: center;  
}  
/* Font size to 140%; background color to gray*/  
.headfoot {  
  font-size: 140%; background: #f5f5f5;  
}  
/* Font size to 130%*/  
.bigfont { font-size: 130%; }
```

Selector de Clases

- ▶ Aquí hay otro documento HTML simple que utiliza estos selectores de clase y demuestra cómo usarlos:
- ▶ Ejemplo HTML:

```
<body>
  <header class="headfoot">Header</header>
  <nav class="bigfont">Navigation</nav>
  <main>
    <h1>class selector</h1>
    <p>The p element without a class.</p>
    <p class="note">The p element with the class <code>note</code></p>
    <p class="note warning">The p element with the classes <code>note warning</code></p>
    <p class="warning">The p element with the class <code>warning</code></p>
    <p class="note bigfont">The p element with the classes <code>note bigfont</code></p>
  </main>
  <footer class="headfoot">Footer</footer>
</body>
</html>
```

- ▶ Ahora puede dirigirse al elementos HTML con el nombre de **class** note en CSS a través del selector de clases colocando un punto (.) delante del nombre de la clase, de la siguiente manera:

Selector de Clases

► Resultados:

Header
Navigation

class selector

The p element without a class.

■ The p element with the class note

The p element with the classes note warning

The p element with the class warning

■ The p element with the classes note bigfont

Footer

Selector de Clases

- ▶ En los navegadores modernos, puede hacer que los elementos HTML utilicen mas de una clase separando los nombres de las clases con espacios:

```
<span class='class1 class2 class3'>
```

- ▶ Puede limitar el alcance de acción de una clase especificando los tipos de elementos a los que se debe aplicar.

```
p.main {  
  text-indent:30px;  
}
```

Selector Identificador

- ▶ En CSS un ID, es un selector que permite aplicar estilos a un único elemento HTML de manera específica. A diferencia de las clases, los ID son siempre elementos únicos en un documento HTML, por lo que **solo se pueden asignar este ID a un elemento** de todo el documento HTML.
- ▶ **Sintaxis HTML:**

```
<div id="header">Header</div>
```

- ▶ Para crear una regla CSS se utiliza el símbolo #, seguido del ID de la siguiente manera:
- ▶ **Sintaxis CSS:**

```
#header { ... }
```

- ▶ El uso de los marcadores HTML se utilizan para permitir a los lectores saltar a partes específicas de una página web.

¿Cuando utilizar el selector de clases en lugar del selector id?

- ▶ El selector de clases es el selector con el punto (.), mientras que el selector de ID es el selector con la tecla gato o hash (#).
- ▶ Es preferible utilizar selectores de clase para propiedades o grupos específicos, como notas, advertencias o mensajes de error.
- ▶ Puedes utilizar los selectores de ID para escribir áreas principales únicas de una pagina web, como #headarea, #mainarea, o #footarea.

Uso de nombre de clase y de ID

- ▶ Los nombres deben tener significado y no debe elegir ningún nombre que refleje el formato.
- ▶ Un nombre con significado le ayudara a comprender el significado más rápidamente durante una revisión posterior y le resultara más fácil asignar este nombre de clase o de ID a un elemento HTML al diseñar las paginas web.

Selector universal

- ▶ Se utiliza para seleccionar todos los elementos de un documento a la vez.
- ▶ El selector universal debe escribirse con un (*).

```
*{  
  margin: 5px;  
  padding: 3px;  
  border: 1px dotted black;  
  text-align: center;  
}
```

- ▶ El selector universal se puede utilizar no solo para seleccionar todos los elementos, sino también para seleccionar todos los elementos dentro de un elemento.

Selector universal

- ▶ Es poco probable que uses el selector * solo, pero como parte de una regla compuesta puede ser poderoso.

```
#boxout * p {  
  border:1px solid green;  
}
```

- ▶ 1. Busca el objeto con el ID de boxout
- ▶ 2. Busca todos los subelementos del objeto devuelto en el paso 1.
- ▶ 3. Encuentra todos los p subelementos de los objetos devueltos en el paso 2 y, debido a que este es el selector final del grupo, encuentre también todos los p subelementos y subsubelementos (y así sucesivamente) de los objetos devueltos en el paso. 2.
- ▶ 4. Aplique los estilos dentro de los caracteres {y} a los objetos devueltos en el paso 3. El resultado neto de esto es que el borde verde se aplica solo a los párrafos que son nietos (o bisnietos, etc.) del elemento principal. .

DIVs

- ▶ El elemento div le **permite agrupar un conjunto de elementos en un cuadro a nivel de bloque**.
- ▶ El elemento `<div>` se puede anidar dentro de otros elementos `<div>` u otros elementos HTML para crear diseños complejos o agrupaciones de contenido,

Diferencias entre div y span

- ▶ Los elementos `<div>` y `` son tipos de contenedores pero con algunas cualidades diferentes.
- ▶ Un **div tiene un ancho infinito** (al menos hasta el borde del navegador).
- ▶ Un elemento span, **es tan ancho como el texto que contiene.**

```
<html>
<head>
  <title>Div and span example</title>
  <style>
    div, span { border :1px solid black; }
    div      { background-color:yellow; }
    span     { background-color:cyan; }
  </style>
</head>
<body>
  <div>This text is within a div tag</div>
  This isn't.
  <div>And this is again.</div><br>
  <span>This text is inside a span tag.</span>
  This isn't.
  <span>And this is again.</span><br><br>
  <div>This is a larger amount of text in a div that wraps around
  to the next line of the browser</div><br>
  <span>This is a larger amount of text in a span that wraps
  around
  to the next line of the browser</span>
</body>
</html>
```

Diferencias entre div y span

This text is within a div tag
This isn't.
And this is again.

This text is inside a span tag. This isn't. And this is again.

This is a larger amount of text in a div that wraps around to the next line of the browser

This is a larger amount of text in a span that wraps around to the next line of the browser

```
<html>
<head>
  <title>Div and span example</title>
  <style>
    div, span { border: 1px solid black; }
    div       { background-color: yellow; }
    span      { background-color: cyan; }
  </style>
</head>
<body>
  <div>This text is within a div tag</div>
  This isn't.
  <div>And this is again.</div><br>
  <span>This text is inside a span tag.</span>
  This isn't.
  <span>And this is again.</span><br><br>
  <div>This is a larger amount of text in a div that wraps around
  to the next line of the browser</div><br>
  <span>This is a larger amount of text in a span that wraps
  around
  to the next line of the browser</span>
</body>
</html>
```

pseudoclasses

- ▶ Las pseudoclasses en CSS son palabras clases añadidas a los selectores para aplicar estilos a elementos en estados específicos.
- ▶ En HTML, hay elementos a los que no se puede acceder mediante selectores comunes. Entre ellos se incluyen, por ejemplo: los elementos sobre los que se encuentra el apuntador del mouse o un hipervínculo que ya se ha visitado o no.
- ▶ **Sintaxis:**
 - ▶ pseudoclasses:link y :visited
- ▶ Permiten seleccionar y marcar de forma especial los vínculos no visitados y visitados, respectivamente.

Pseudoclases para hipervinculos visitados y no visitados

- Las pseudoclases y los pseudoelementos están separados por un carácter : (dos puntos).

```
.navegacion-principal a:hover {  
    background-color: #ffc107;  
}  
a:link { color: red; }  
a:visited { color: green; }
```

- Estos estados pueden estar relacionados con la interacción del usuario con el documento como estados en los enlaces (:hover, :active, :visited), el estado de un checkbox (:checked) o incluso la posición de un elemento (:first-child :last-child).

Seudoclasas para hipervinculos visitados y no visitados

- Las pseudoclasas y los pseudoelementos están separados por un carácter : (dos puntos).

```
input { background-color: lightgray; }  
input:focus { background-color: white; }  
input:hover { box-shadow: 0 0 3px blue; }  
input:placeholder-shown { color: white; }  
li { background-color: lightgray; }  
li:hover { background-color: snow; }  
li:active { background-color: gray; }  
a:link { text-decoration: none; color: blue; }  
a:hover { font-weight: bold; }  
a:active { color: red; }
```

- Es importante mantener el orden de los selectores de pseudoclasas, es decir, :link, :visited, :hover, :focus y :active (LVHFA); de lo contrario, :visited sobrescribirá la pseudoclase :hover.

Ejemplo

```
<!DOCTYPE html>
<html>
  <head>
    <title>Pseudo-classes</title>
    <style>
      a:link      { color:blue; }
      a:visited   { color:gray; }
      a:hover     { color:red; }
      a:active    { color:purple; }
      *:focus    { background:yellow; }
    </style>
  </head>
  <body>
    <a href='http://google.com'>Link to Google'</a><br>
    <a href='nowhere'>Link to nowhere'</a><br>
    <input type='text'>
  </body>
</html>
```

Selector child

- ▶ Las pseudoclasas estructurales en CSS permiten seleccionar elementos secundarios específicos: Estas incluyendo:
 - ▶ **:first-child**: selecciona el primer hijo de un elemento padre
 - ▶ **:last-child**: selecciona el último hijo de un elemento padre.
 - ▶ **:nth-child()** selecciona el n-ésimo hijo de un elemento padre, permitiéndolos los valores enteros, cálculos o palabras clave como odd(impares) y even(pares).
 - ▶ **:nth-last-child(n)** similar a nth-child(n), pero cuenta desde el final.
 - ▶ **:only-child** selecciona elementos que son el único hijo de su padre.
- ▶ Un elemento secundario (o elemento descendiente) es un elemento que se encuentra debajo de un elemento determinado cuando se observa el árbol del DOM.

Pseudoclases estructurales para elementos secundarios específicos

- ▶ Los selectores estructurales `:first-of-type`, `:last-of-type`, `:nth-of-type()`, `:nth-last-of-type()` y `:only-of-type` se aplican únicamente a elementos secundarios específicos.
- ▶ Esto contrasta con los selectores `:...child`, que se pueden aplicar a todos los elementos secundarios.

Ejemplo

- A continuación, se muestra un ejemplo simple que no podría haber implementado con :first-child o :last-child:

```
<body>
  <header>Header</header>
  <article>
    <h1>Article 1</h1>
    <p>Text for article</p>
  </article>
  <article>
    <h1>Article 2</h1>
    <p>Text for article</p>
  </article>
  <article>
    <h1>Article 3</h1>
    <p>Text for article</p>
  </article>
  <footer>Footer</footer>
</body>
```



- CSS

```
article:first-of-type { border: 2px solid black; }
article:last-of-type { border: 2px dotted gray; }
```

Combinadores: Concatenación de los selectores

- Un combinador es un carácter que conecta dos selectores, donde el primer selector define una condición y el segundo es el objetivo si se cumple dicha condición.

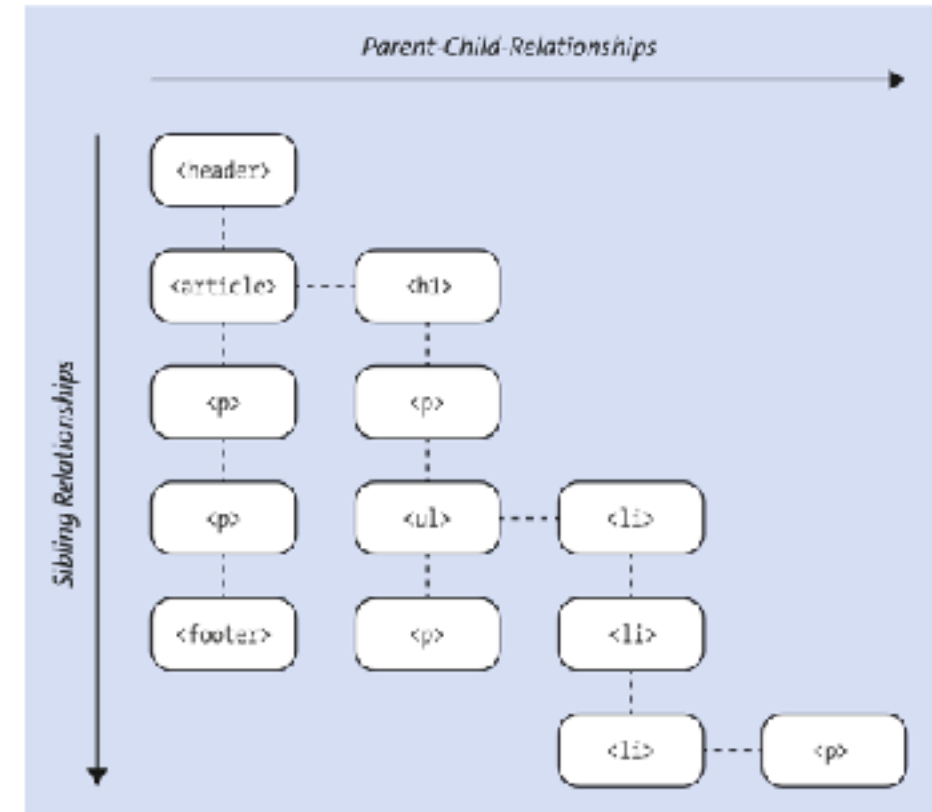
Combinator	Name	Meaning
E F	Descendant selector (<i>descendant combinator</i>)	F gets selected if it's a descendant of an E element.
E > F	Child selector (<i>child combinator</i>)	F gets selected only if it's a direct descendant of an E element.
E + F	Adjacent sibling selector (<i>adjacent sibling combinator</i>)	F gets selected only if it occurs directly after E (in the same parent element).
E ~ F	General sibling selector (<i>general sibling combinator</i>)	F gets selected only if it occurs after E (in the same parent element).

Combinadores: Concatenación de los selectores

- El combinador descendiente (E F) conecta dos selectores con un espacio y selecciona todos los elementos secundarios y los elementos secundarios, incluidos los descendientes de otros secundarios.

```
article p { background: lightblue; }
```

- Especifica que todos los elementos p ubicados dentro de un elemento article se seleccionaran y se les asignará un color de fondo azul claro



Combinadores: Concatenación de los selectores

Header

Article 1

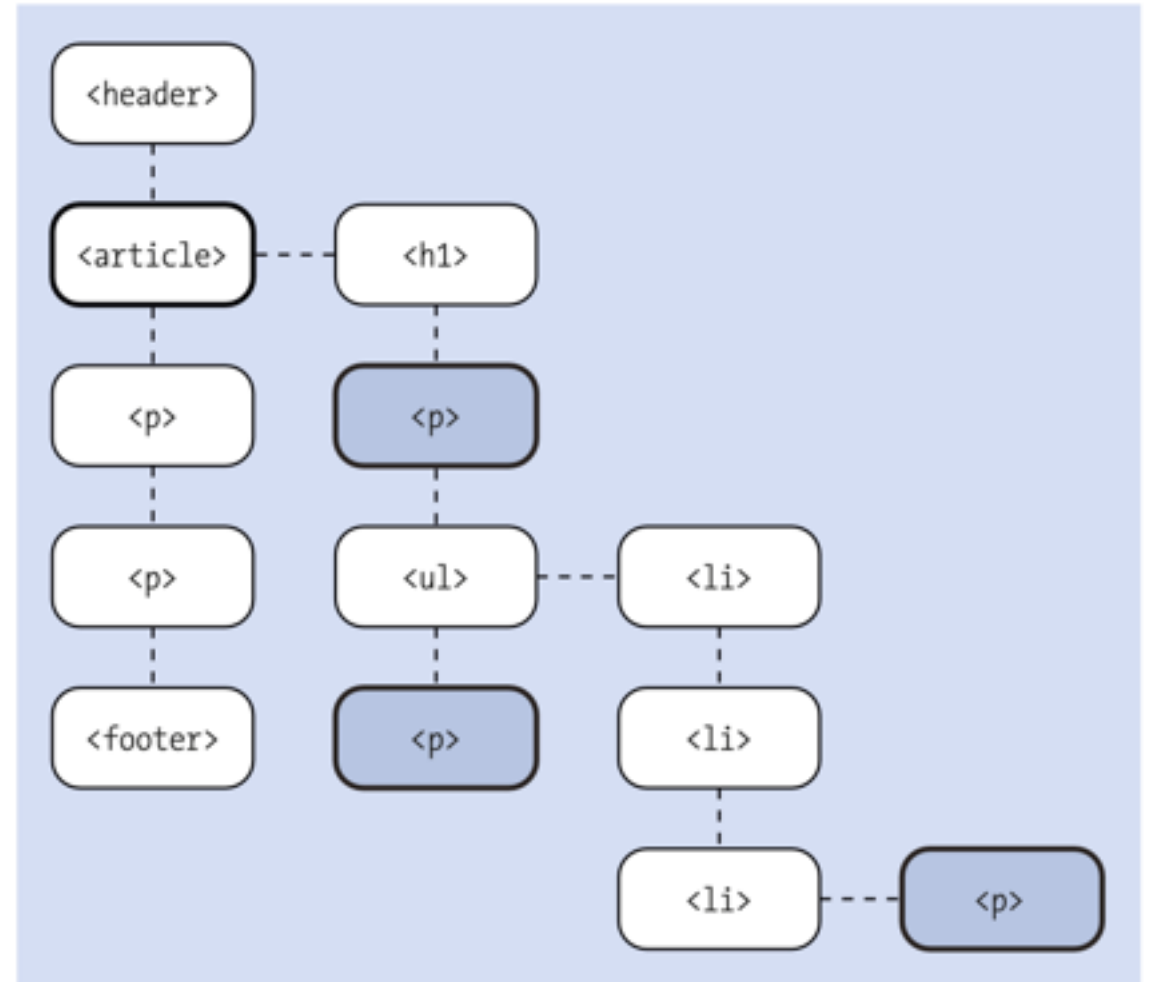
1. Paragraph text for article

- List item 1
- List item 2
- A paragraph text in the list item

2. Paragraph text for article

1. Paragraph text after the article
2. Paragraph text after the article

Foster

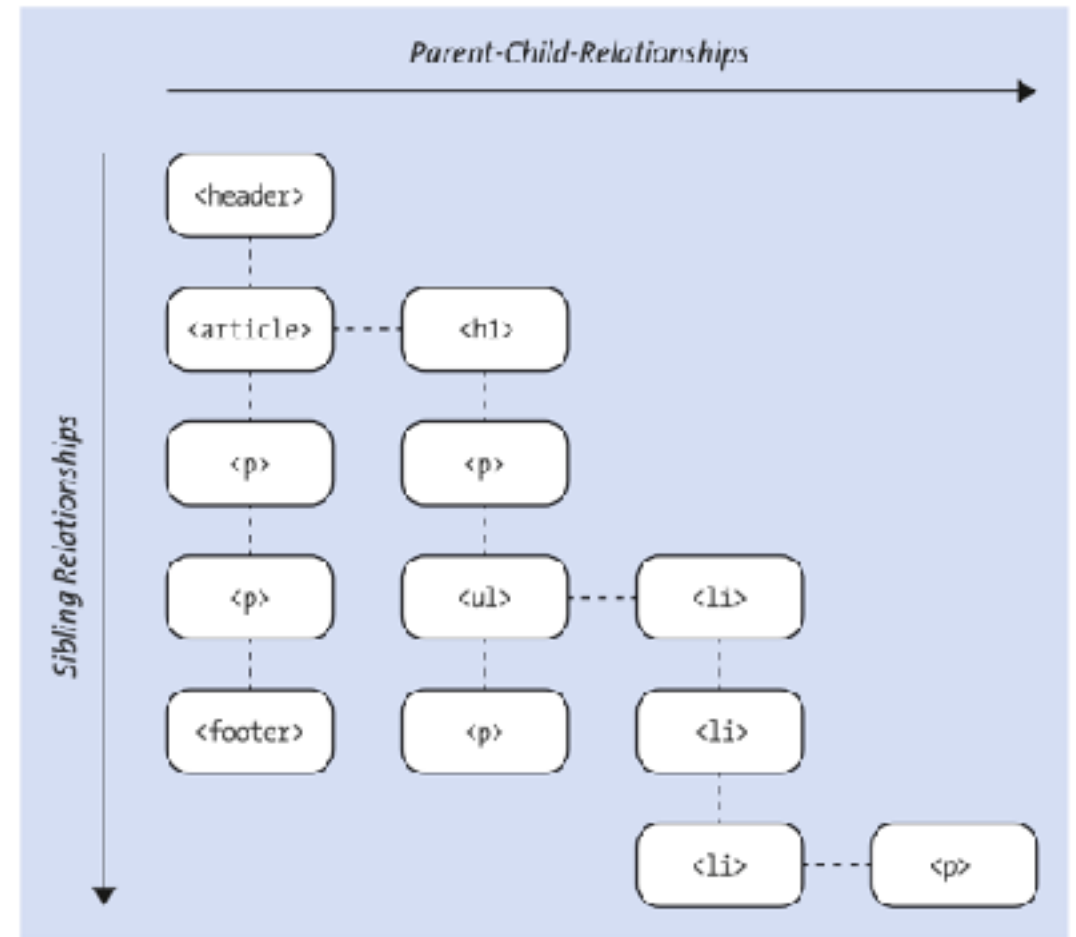


Combinador child

- El combinador hijo $E1 > E2$ (child) selecciona los elementos que son descendientes directos del elemento principal, representados por el corchete ($>$).

```
article > p { background: lightblue; }
```

- De esta manera, puede seleccionar todos los elementos `p` que son descendientes directos del `article` y aplicarles un estilo de fondo azul



Combinador child

Header

Article 1

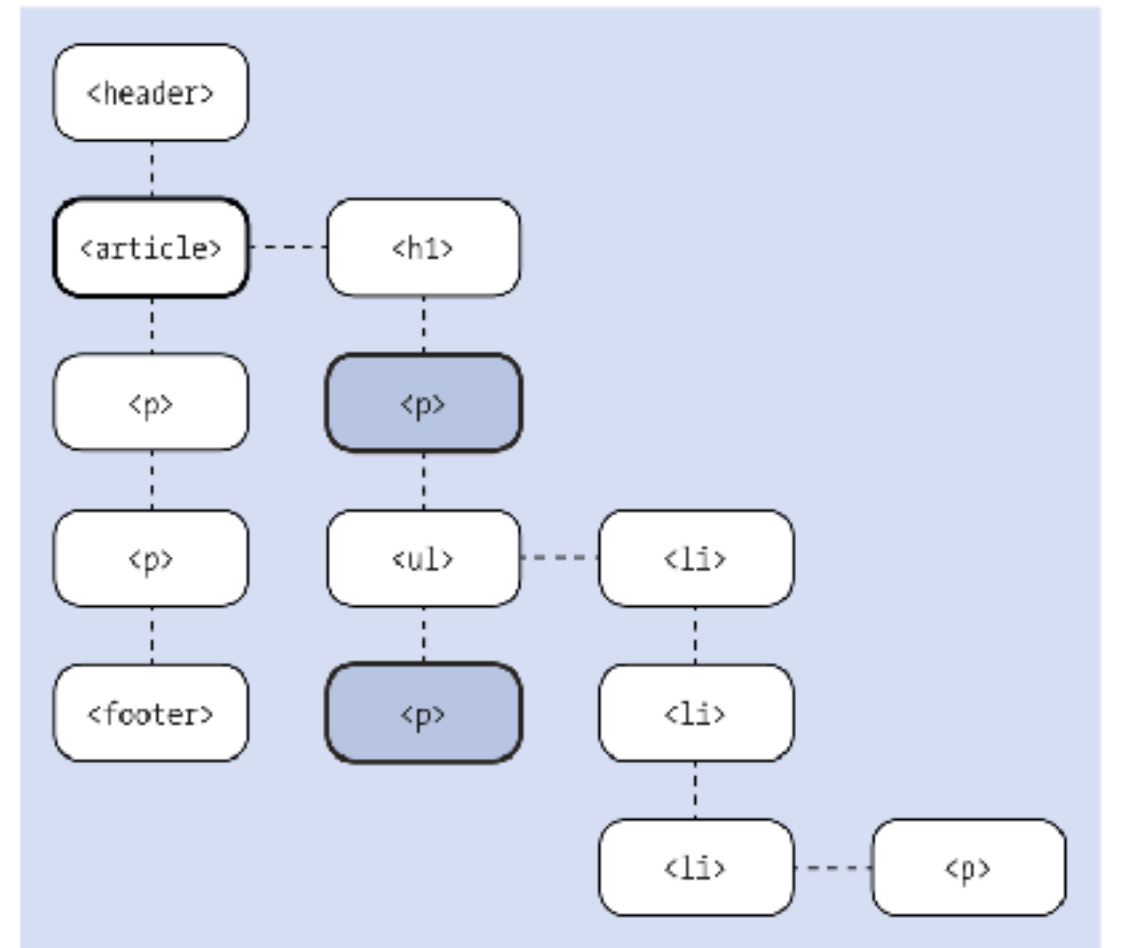
1. Paragraph text for article

- List item 1
- List item 2
- A paragraph text in the list item

2. Paragraph text for article

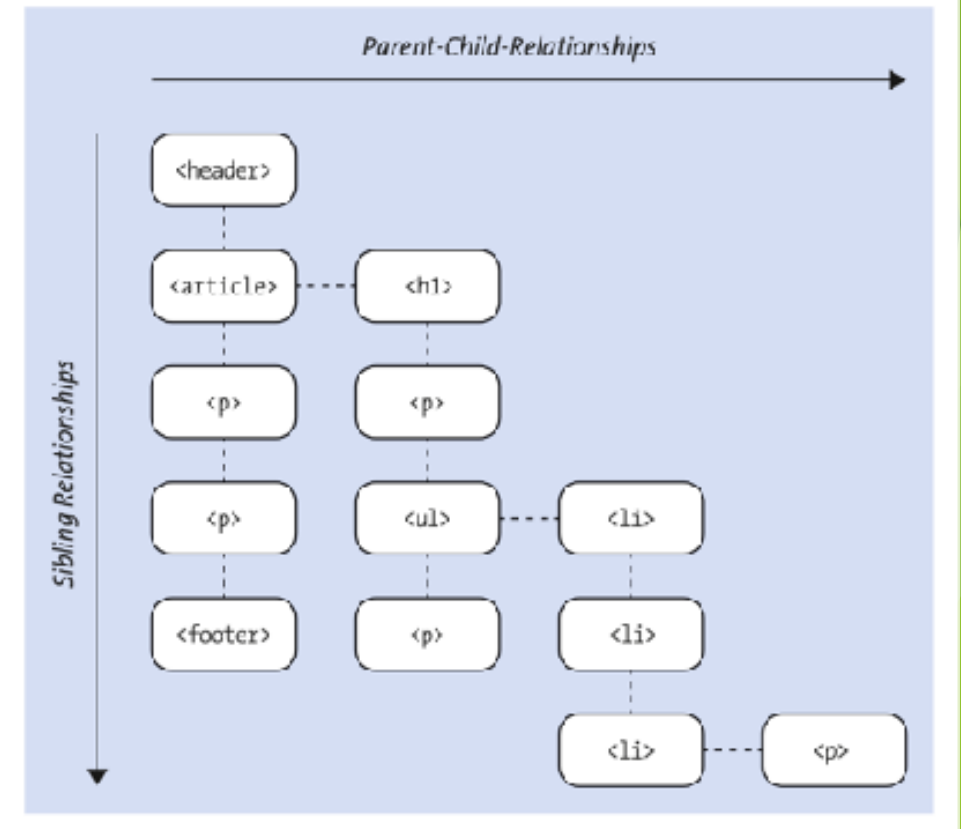
1. Paragraph text after the article
2. Paragraph text after the article

Footer



El combinador de hermanos adyacentes (E1+E2)

- ▶ El combinador de hermanos adyacentes (E1 + E2) conecta dos selectores mediante el signo + y selecciona solo el primer elemento que sigue inmediatamente después del primero en el mismo nivel (hermanos con el mismo padre).
- ▶ `article + p { background: lightblue; }`
- ▶ Aquí, solo se selecciona el elemento p, que es un descendiente del elemento article



El combinador de hermanos adyacentes (E1+E2)

Header

Article 1

1. Paragraph text for article

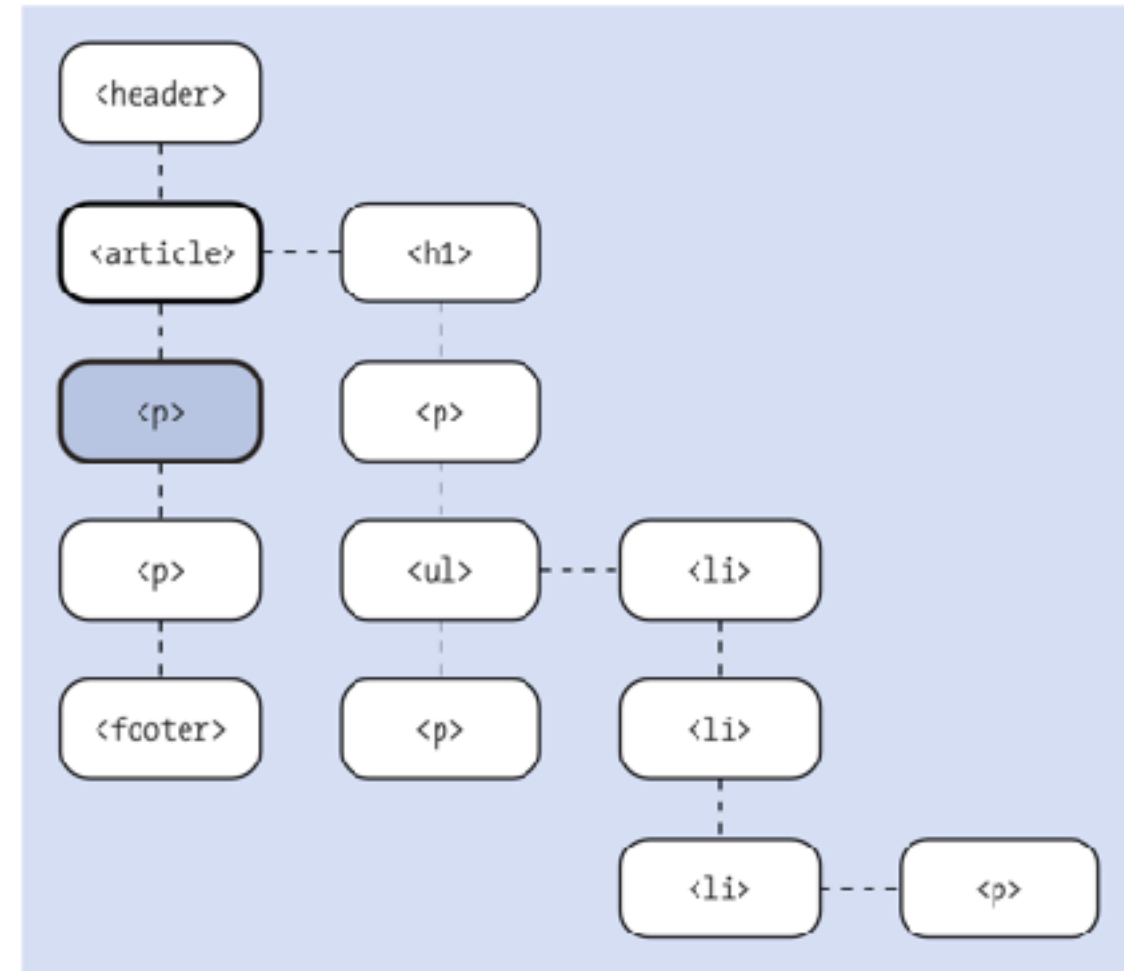
- List item 1
- List item 2
- A paragraph text in the list item

2. Paragraph text for article

1. Paragraph text after the article

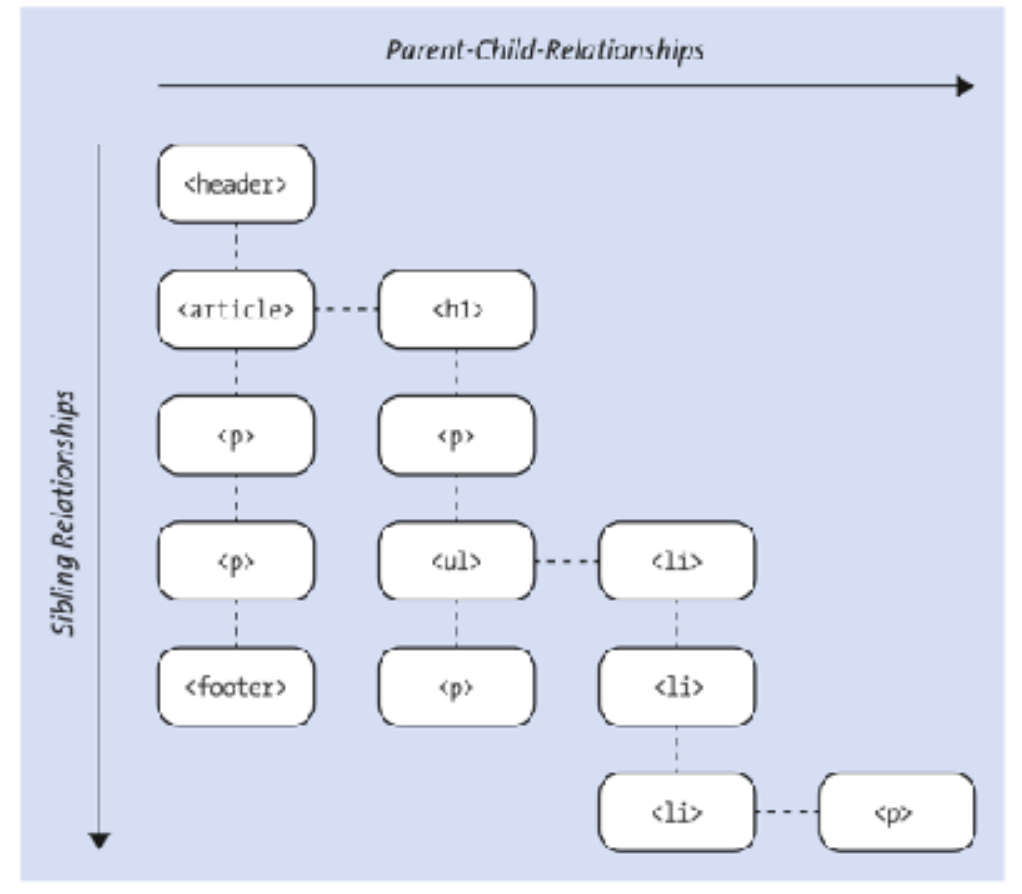
2. Paragraph text after the article

Footer



Combinador general de hermanos

- ▶ El combinador general de hermanos (E1~E2) conecta dos selectores con el carácter tilde (~). Permite seleccionar elementos que son hermanos en el mismo nivel (comparten el mismo padre) pero no es necesario que estén directamente adyacentes; pueden tener otros elementos entre ellos.
- ▶ `article ~ p { background: lightblue; }`
- ▶ Esto selecciona todos los elementos p que siguen a un elemento article y están en el mismo nivel en el árbol del documento.



Combinador general de hermanos

Header

Article 1

1. Paragraph text for article

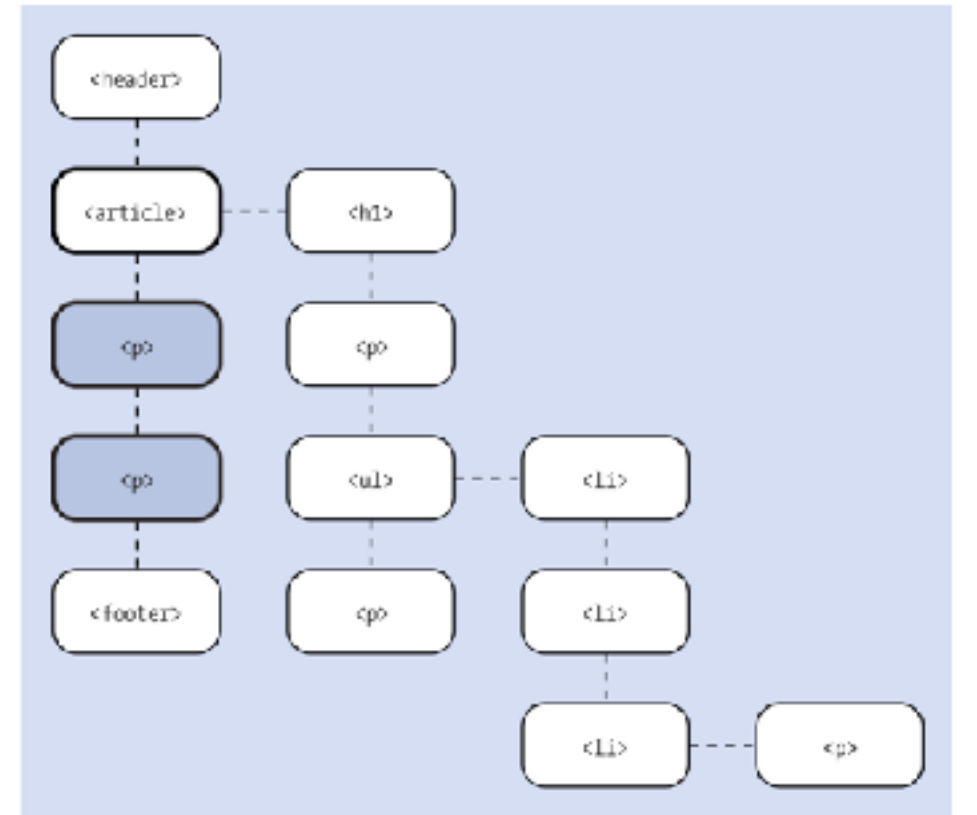
- List item 1
- List item 2
- A paragraph text in the list item

2. Paragraph text for article

1. Paragraph text after the article

2. Paragraph text after the article

Footer



Selector por grupo

- Usando CSS puedes aplicar una regla a más de un elemento, clase o cualquier otro tipo de selector al mismo tiempo separando los selectores con comas.

```
p, #idname, .classname {  
    border-bottom: 1px dotted orange;  
}
```


Herencia

- ▶ CSS permite definir estilos centralizados para varias características (como color, fuente y tamaño de fuente), aplicándolos a múltiples elementos sin tener que asignar las mismas propiedades repetidamente.
- ▶ En un documento HTML, la estructura sigue una jerarquía de padre e hijo, formándonos un árbol. Los elementos hijo heredan muchas de las propiedades de estilo de sus elementos padre, gracias a esta relación de jerarquía.