

UT3_2.- LENGUAJES DE MARCAS EN ENTORNO WEB. CSS

BÁSICO

RESULTADOS DE APRENDIZAJE ASOCIADOS
<ol style="list-style-type: none">1. Reconoce las características de lenguajes de marcas analizando e interpretando fragmentos de código.2. Utiliza lenguajes de marcas para la transmisión de información a través de la Web analizando la estructura de los documentos e identificando sus elementos3. Establece mecanismos de validación para documentos XML utilizando métodos para definir su sintaxis y estructura.
CRITERIOS DE EVALUACIÓN
<ul style="list-style-type: none">- De RA1 – desde CEA hasta CEK- De RA2 – desde CEA hasta CEH- De RA3 – desde CEA hasta CEH

UT3_2.- LENGUAJES DE MARCAS EN ENTORNO WEB.

CSS BÁSICO

Índice de contenido

1.- Introducción.....	3
2.- Selectores básicos.....	3
2.1.- Selector universal.....	3
2.2.- Selector de tipo o etiqueta.....	4
2.3.- Selector de clase.....	6
2.4.- Selector ID.....	8
2.5.- Selectores de atributo.....	9
3.- Selectores descendentes.....	12
3.1.- Selector hijo.....	12
3.2.- Selector de hijo directo.....	14
4.- Selectores adyacentes.....	15
4.1.- Selector de hermano adyacente.....	15
4.2.- Selector de hermano general.....	17
4.3.- Aplicar el mismo estilo a varios selectores.....	17

1.- Introducción



Manual:

https://www.w3schools.com/csSref/css_selectors.php

Para crear diseños web profesionales, es imprescindible conocer y dominar los selectores de CSS. Como se vio en el capítulo anterior, una regla de CSS está formada por una parte llamada "selector" y otra parte llamada "declaración".



La declaración indica "qué hay que hacer" y el selector indica "a quién hay que hacérselo".

A un mismo elemento HTML se le pueden aplicar varias reglas CSS y cada regla CSS puede aplicarse a un número ilimitado de elementos.

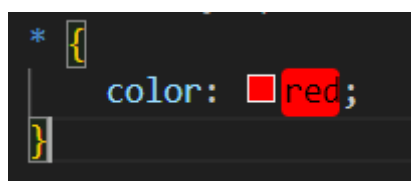
El estándar de CSS 2.1 incluye una docena de tipos diferentes de selectores, que permiten seleccionar de forma muy precisa elementos individuales o conjuntos de elementos dentro de una página web.

No obstante, la mayoría de las páginas de los sitios web se pueden diseñar utilizando solamente los cinco selectores básicos.

2.- Selectores básicos

2.1.- Selector universal

Se utiliza para seleccionar todos los elementos de la página. El siguiente ejemplo elimina el margen y el relleno de todos los elementos HTML (por ahora no es importante fijarse en la parte de la declaración de la regla CSS):



En este caso “todos” los elementos de nuestra página serán de color rojo.



El selector universal se indica mediante un asterisco (*). A pesar de su sencillez, no se utiliza habitualmente, ya que es difícil que un mismo estilo se pueda aplicar a todos los elementos de una página.

2.2.- Selector de tipo o etiqueta

Selecciona todos los elementos de la página cuya etiqueta HTML coincide con el valor del selector. El siguiente ejemplo selecciona todos los párrafos de la página:

```
p {
    color: #999;
    font-family: Verdana;
}
```

Para utilizar este selector, solamente es necesario indicar el nombre de una etiqueta HTML (sin los caracteres < y >) correspondiente a los elementos que se quieren seleccionar.

El siguiente ejemplo aplica diferentes estilos a los titulares y a los párrafos de una página HTML:

```
h1 {
    color: red;
}

h2 {
    color: blue;
}

p {
    color: black;
}
```

Si se quiere aplicar los mismos estilos a dos etiquetas diferentes, se pueden encadenar los selectores. En el siguiente ejemplo, los títulos de sección h1, h2 y h3 comparten los mismos estilos:

```
h1 {  
  color: #8A8E27;  
  font-weight: normal;  
  font-family: Arial, Helvetica, sans-serif;  
}
```

```
h2 {  
  color: #8A8E27;  
  font-weight: normal;  
  font-family: Arial, Helvetica, sans-serif;  
}
```

```
h3 {  
  color: #8A8E27;  
  font-weight: normal;  
  font-family: Arial, Helvetica, sans-serif;  
}
```

En este caso, CSS permite agrupar todas las reglas individuales en una sola regla con un selector múltiple. Para ello, se incluyen todos los selectores separados por una coma (,) y el resultado es que la siguiente regla CSS es equivalente a las tres reglas anteriores:

```
h1, h2, h3 {  
  color: #8A8E27;  
  font-weight: normal;  
  font-family: Arial, Helvetica, sans-serif;  
}
```



En las hojas de estilo complejas, es habitual agrupar las propiedades comunes de varios elementos en una única regla CSS y posteriormente definir las propiedades específicas de esos mismos elementos.

El siguiente ejemplo establece en primer lugar las propiedades comunes de los títulos de sección (color y tipo de letra) y a continuación, establece el tamaño de letra de cada uno de ellos:

```
h1, h2, h3 {  
  color: #8A8E27;  
  font-weight: normal;  
  font-family: Arial, Helvetica, sans-serif;  
}
```

```
h1 { font-size: 2em; }  
h2 { font-size: 1.5em; }  
h3 { font-size: 1.2em; }
```

2.3.- Selector de clase

Si se considera el siguiente código HTML de ejemplo:

```
<body>  
  <p>Lorem ipsum dolor sit amet...</p>  
  <p>Nunc sed lacus et est adipiscing accumsan...</p>  
  <p>Class aptent taciti sociosqu ad litora...</p>  
</body>
```



¿Cómo se pueden aplicar estilos CSS sólo al primer párrafo?



El selector universal (*) no se puede utilizar porque selecciona todos los elementos de la página. El selector de

tipo o etiqueta (p) tampoco se puede utilizar porque seleccionaría todos los párrafos. Por último, el selector descendente (body p) tampoco se puede utilizar porque todos los párrafos se encuentran en el mismo sitio.

Una de las soluciones más sencillas para aplicar estilos a un solo elemento de la página consiste en utilizar el atributo class de HTML sobre ese elemento para indicar directamente la regla CSS que se le debe aplicar:

```
<p class="destacado">Párrafo 1</p>
<p>Párrafo 2</p>
<p>Párrafo 3</p>
```

A continuación, se crea en el archivo CSS una nueva regla llamada destacado con todos los estilos que se van a aplicar al elemento. Para que el navegador no confunda este selector con los otros tipos de selectores, se prefija el valor del atributo class con un punto (.) tal y como muestra el siguiente ejemplo:

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

La salida sería la siguiente:

Párrafo 1

Párrafo 2

Párrafo 3

El selector **.destacado** se interpreta como "cualquier elemento de la página cuyo atributo class sea igual a destacado", por lo que solamente el primer párrafo cumple esa condición.



Este tipo de selectores se llaman selectores de clase y son los más utilizados.

La principal característica de este selector es que en una misma página HTML varios elementos diferentes pueden utilizar el mismo valor en el atributo class:

```
<p class="destacado">Párrafo 1</p>
<p>Párrafo 2
  <a href="https://www.google.es" class="destacado">Google</a>
</p>
<p>Párrafo 3
  <em class="destacado">Cursiva</em>
</p>
```

La salida en este caso será:

Párrafo 1

Párrafo 2 Google

Párrafo 3 *Cursiva*

Los selectores de clase son imprescindibles para diseñar páginas web complejas, ya que permiten disponer de una precisión total al seleccionar los elementos. Además, estos selectores permiten **reutilizar los mismos estilos para varios elementos diferentes**.

2.4.- Selector ID



Aunque podemos utilizar el selector id para dar estilos, no está aconsejado, ya que su uso se reserva para utilizarlo con JavaScript.

El selector de ID permite seleccionar un elemento de la página a través del valor de su atributo id.



Este tipo de selectores sólo seleccionan un elemento de la página porque el valor del atributo id no se puede repetir en dos elementos diferentes de una misma página.

La sintaxis de los selectores de ID es muy parecida a la de los selectores de clase, salvo que se utiliza el símbolo de la almohadilla (#) en vez del punto (.) como prefijo del nombre de la regla CSS:

❖ **index.html:**

```
<p>Párrafo 1</p>
<p id="destacado">Párrafo 2</p>
<p>Párrafo 3</p>
```

❖ **estilo.css:**

```
#destacado {
    color: red;
}
```

La salida será:

Párrafo 1

Párrafo 2

Párrafo 3

En el ejemplo anterior, el selector #destacado solamente selecciona el segundo párrafo (cuyo atributo id es igual a destacado).

2.5.- Selectores de atributo

El selector de atributo nos va a permitir seleccionar elemento por su atributo o por un valor que tome un atributo.

Existen diferentes tipos de selectores de atributos:

- ❖ **[nombre_atributo]**, selecciona los elementos que tienen establecido el atributo llamado nombre_atributo, independientemente de su valor.

```
[href]{  
    color: green;  
}
```

- ❖ **[nombre_atributo=valor]**, selecciona los elementos que tienen establecido un atributo llamado nombre_atributo con un valor igual a valor.

```
[href="#"]{  
    color: red;  
}
```

- ❖ **[nombre_atributo~=valor]**, selecciona los elementos que tienen establecido un atributo llamado nombre_atributo y al menos uno de los valores del atributo es valor.

```
[class~="verde"]{  
    color: green;  
}
```

- ❖ **[nombre_atributo^=valor]**, selecciona los elementos que tienen ese atributo y empiezan por ese valor

```
[class^="rojo"]{  
    color:purple;  
}
```

- ❖ **[nombre_atributo\$=valor]**, selecciona los elementos que tienen ese atributo y terminan por ese valor

```
[class$="rojo"]{  
    color:steelblue;
```

}

[nombre_atributo*=valor], selecciona los elementos que tienen ese atributo y contengan ese valor

```
[class*="rojo"]{
  color:brown;
}
```

Ejemplo:

❖ index.html

```
<p class="destacado">Párrafo 1</p>
<p>Párrafo 2 <a href="#" class="destacado">enlace</a></p>
<p>Párrafo 3 <em class="destacado">cursiva</em></p>
```

❖ estilo.css

```
.destacado {
  color: red;
}

[href="#"] {
  color: green;
}
```

La salida será la que sigue:

Párrafo 1

Párrafo 2 enlace

Párrafo 3 *cursiva*

3.- Selectores descendentes

Selecciona los elementos que se encuentran dentro de otros elementos.



Un elemento es descendiente de otro cuando se encuentra entre las etiquetas de apertura y de cierre del otro elemento.

3.1.- Selector hijo

El selector del siguiente ejemplo selecciona todos los elementos `` de la página que se encuentren dentro de un elemento `<p>`:

```
<p>
  ...
  <span>texto1</span>
  ...
  <a href="">...<span>texto2</span></a>
  ...
</p>
```

En el archivo .css, escribiríamos lo siguiente:

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

El selector `p span` selecciona tanto `texto1` como `texto2`.



El motivo es que, en el selector descendente, un elemento no tiene que ser descendiente directo del otro.

La única condición es que un elemento debe estar dentro de otro elemento, sin importar el nivel de profundidad en el que se encuentre.

Al resto de elementos de la página que no están dentro de un elemento <p>, no se les aplica la regla CSS anterior.

Los selectores descendentes permiten aumentar la precisión del selector de tipo o etiqueta. Así, utilizando el selector descendente es posible aplicar diferentes estilos a los elementos del mismo tipo. El siguiente ejemplo amplía el anterior y muestra de color azul todo el texto de los contenidos dentro de un <h1>:

p span { color: red; }
h1 span { color: blue; }

Con las reglas CSS anteriores:

- ✓ Los elementos que se encuentran dentro de un elemento <p> se muestran de color rojo.
- ✓ Los elementos que se encuentran dentro de un elemento <h1> se muestran de color azul.
- ✓ El resto de los elementos de la página, se muestran con el color por defecto aplicado por el navegador.

La sintaxis formal del selector descendente se muestra a continuación:

selector1 selector2 selector3 ... selectorN

Los selectores descendentes siempre están formados por dos o más selectores separados entre sí por espacios en blanco.



El último selector indica el elemento sobre el que se aplican los estilos y todos los selectores anteriores indican el lugar en el que se debe encontrar ese elemento.

En el siguiente ejemplo, el selector descendente se compone de cuatro selectores:

p a span em { text-decoration: underline; }



Los estilos de la regla anterior se aplican a los elementos de tipo `` que se encuentren dentro de elementos de tipo ``, que a su vez se encuentren dentro de elementos de tipo `<a>` que se encuentren dentro de elementos de tipo `<p>`.



No debe confundirse el selector descendente con la combinación de selectores:

/ El estilo se aplica a todos los elementos "p", "a", "span" y "em" */*
p, a, span, em { text-decoration: underline; }

/ El estilo se aplica solo a los elementos "em" que se encuentran dentro de "p a span" */*
p a span em { text-decoration: underline; }

3.2.- Selector de hijo directo

Selecciona todos los hijos que sean directos, es decir que no estén dentro de otras etiquetas. El estilo se aplica a todos los hijos

```
p > span {
  color: blue;
}
```

Se comporta igual, que el selector de hijo, con la única diferencia de que debe ser hijo directo, para el caso anterior sería válida la siguiente sintaxis HTML.

```
<p>Párrafo
  <span>Span</span>
  Más párrafo
</p>
```

Pero para el caso

```
div > a {
  color: red;
}
```

Con el siguiente HTML:

```
<div>
  <p><a href="">Enlace 1</a></p>
  <p><a href="">Enlace 2</a></p>
  <a href="">Enlace 3</a>
  <a href="">Enlace 4</a>
</div>
```

El resultado será que el «Enlace 3» y «Enlace 4» aparecerán en rojo, pues esos elementos <a> son hijos directos de <div>, mientras que el «Enlace 1» y «Enlace 2» son hijos directos de <p> y no serán seleccionados.

Enlace 1

Enlace 2

Enlace 3 Enlace 4

4.- Selectores adyacentes

El selector adyacente se emplea para **seleccionar elementos que en el código HTML de la página se encuentran justo a continuación de otros elementos.**

4.1.- Selector de hermano adyacente

Su sintaxis emplea el signo + para separar los dos elementos:

elemento1 + elemento2 { ... }

Si se considera el siguiente código HTML:

```
<h1>Título</h1>
<h2>Subtítulo 1</h2>
...
<h2>Subtítulo 2</h2>
...
```

En el código HTML podemos ver dos elementos <h2>, pero sólo uno de ellos se encuentra inmediatamente después del elemento <h1>. Si se quiere aplicar diferentes colores en función de esta circunstancia, el selector adyacente es el más adecuado:

```
h2 {
    color: green;
}

h1+h2 {
    color: red;
}
```

La salida será:

Título

Subtítulo 1

...

Subtítulo 2

...

Las reglas CSS anteriores hacen que todos los <h2> de la página se vean de color verde, salvo aquellos <h2> que se encuentran inmediatamente después de cualquier elemento <h1> y que se muestran de color rojo.

Técnicamente, los elementos que forman el selector adyacente deben cumplir las dos siguientes condiciones:

- ✓ elemento1 y elemento2 deben ser *elementos hermanos*, por lo que su elemento padre debe ser el mismo.
- ✓ elemento2 debe aparecer inmediatamente después de elemento1 en el código HTML de la página.

4.2.- Selector de hermano general

Selecciona todos los hermanos que esté por debajo de él.

```
p ~ span {
    color: red;
}
```

Si tenemos el siguiente código HTML

```
<div>
  <span>Este span no es rojo</span>
  <p>Aquí hay un párrafo.</p>
  <code>Aquí hay código</code>
  <span>Aquí hay un span. Es rojo porque comparten
    el mismo padre
  </span>
</div>
```

Se aplicará al último span ya que es hermano del p, no hace falta que sea hermano siguiente, es suficiente con que sean hermanos.

4.3.- Aplicar el mismo estilo a varios selectores

Para aplicar el mismo estilo a varios selectores utilizamos la coma para separar los selectores

Para el caso:

```
p,a {
  color: red;
  font-family: tahoma;
  border: 1px solid red;
}
```

Con el siguiente HTML

```
<p>Párrafo 1</p>
<p>Párrafo 2</p>
<a href="#">Enlace 1</a>
<a href="#">Enlace 2</a>
```

Estamos aplicando el mismo estilo tanto a los <p> como a los <a>.

ACTIVIDAD1 – UT3_2

- 1) A partir del código HTML y CSS que se muestra, añadir los selectores CSS que faltan para aplicar los estilos deseados. Cada regla CSS incluye un comentario en el que se explica los elementos a los que debe aplicarse.
- 2) Si el código HTML no validara, arréglalo.

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Ejercicio de selectores</title>
    <style type="text/css">
      /* Todos los elementos de la pagina */
```

```
{  
    font: 1em/1.3 Arial, Helvetica, sans-serif;  
}  
  
/* Todos los parrafos de la pagina */  
{  
    color: #555;  
}  
  
/* Todos los párrafos contenidos en #primero */  
{  
    color: #336699;  
}  
  
/* Todos los enlaces la pagina */  
{  
    color: #CC3300;  
}  
  
/* Los elementos "em" contenidos en #primero */  
{  
    background: #FFFFCC;  
    padding: .1em;  
}  
  
/* Todos los elementos "em" de clase "especial" en toda la  
página */  
{  
    background: #FFCC99;  
    border: 1px solid #FF9900;  
    padding: .1em;
```

```

    }

    /* Elementos "span" contenidos en .normal */
    {
        font-weight: bold;
    }
</style>
</head>
<body>
    <div id="primero">
        <p>Lorem ipsum dolor sit amet, <a href="#">consectetuer
adipiscing elit</a>. Praesent blandit nibh at felis. Sed
        nec diam in dolor vestibulum aliquet. Duis ullamcorper,
nisi non facilisis molestie, <em>lorem sem aliquam
        nulla</em>, id lacinia velit mi vestibulum
enim.</p>

    </div>

    <div class="normal">
        <p>Phasellus eu velit sed lorem sodales egestas. Ut
feugiat. <span><a href="#">Donec porttitor</a>, magna eu
        varius luctus,</span> metus massa tristique massa,
in imperdiet est velit vel magna. Phasellus erat.

        Duis risus. <a href="#">Maecenas dictum</a>, nibh vitae
pellentesque auctor, tellus velit consectetur
        tellus, tempor pretium felis tellus at metus.</p>

        <p>Cum sociis natoque <em class="especial">penatibus et
magnis</em> dis parturient montes, nascetur ridiculus
        mus. Proin aliquam convallis ante. Pellentesque
habitant morbi tristique senectus et netus et malesuada
        fames ac turpis egestas. Nunc aliquet. Sed eu metus.
Duis justo.</p>

```

```
<p>Donec facilisis blandit velit. Vestibulum nisi. Proin  
volutpat, <em class="especial">enim id iaculis  
congue</em>, orci justo ultrices tortor, <a  
href="#">quis lacinia eros libero in eros</a>. Sed malesuada  
dui vel quam. Integer at eros.</p>  
  
</div>  
  
</body>  
  
</html>
```