

:after & :before

Módulo 04

::after & ::before

::after , Este pseudo elemento hace referencia a que sucede con un elemento luego del mismo

::before, Hace referencia a un elemento con lo que sucede antes del mismo.

```
<style>
p::after { content: 'mi otro texto';}
</style>
```



content

Hemos incluido la propiedad content. Esta propiedad es obligatoria para trabajar ya que es la que me permite justamente luego incorporar elementos antes o después del elemento al cual le haya aplicado **::AFTER** o **::BEFORE**

Por otro lado, es interesante saber que también puedo aplicar imágenes. Por caso si quisiera que apareciera una imagen por ejemplo luego de un párrafo lo haría de la siguiente manera,

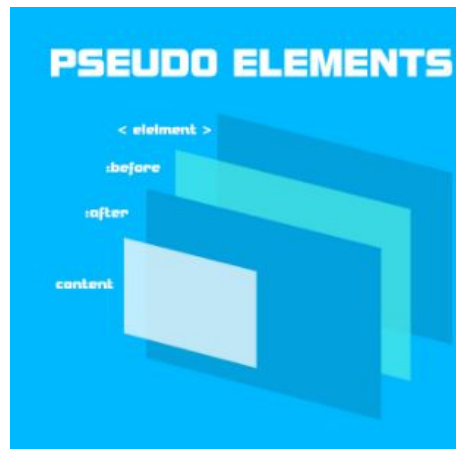
```
<style>
p::after { content: url(imagen.jpg);}
</style>
```



content

Recordemos que siempre que traemos una imagen como lo hicimos en la **propiedad background-image** como así también cuando hemos trabajado con import en el caso de traer un archivo , lo hacemos con el protocolo url.

En realidad, sería simple hablar sólo de estos ejemplos de **::AFTER** y **::BEFORE** pero la realidad es que tenemos mucho más por hacer con estos pseudo elementos.



Contadores

Para que lo podamos entender de una manera mucho más simple, en realidad, contamos con tres propiedades básicas para poder construir una especie de contador con estos pseudo elementos

- **counter-reset**

Esta propiedad me indica donde se inicializa, o donde arranca nuevamente el contador si por ejemplo se ubica en el body, eso claramente representa que el contador, será inicializado una sola vez porque el body está una sólo vez en nuestro documento, si por el contrario lo aplicara a un **h1**, cada vez que apareciera un **h1** el contador volvería a inicializarse.

- **counter-increment**

Es donde voy a incrementar el contador, es decir donde va a ir aumentado su número

- **counter**

Esto me permite llamar el contador según el nombre que le haya aplicado para poder identificarlo

Contadores

Esto me permite llamar el **contador** según el nombre que le haya aplicado para poder identificarlo.

Dependiendo el orden con el cual quiera mostrar los elementos será si primero pongo el **contenido de texto**, luego la imagen o el contador en el medio como en este caso.

```
<head>
<style>
body { counter-reset: miTitulo;}
h1 { counter-increment: miTitulo;}
h1::after { content: 'Titulo' counter(miTitulo) url(imagen.jpg);}
</style>
</head>

<body>

<h1></h1>
<h1></h1>

</body>
</html>
```

Contadores

En el **navegador** el **ejemplo anterior** se verá como en la imagen de la derecha,

También se puede profundizar sobre estos **pseudo elementos** y hacer referencia al valor de un atributo colocando como contenido también, veamos un ejemplo,

```
<head>
<style>
a::after {
  content: " (" attr(title) ")";
}
</style>
</head>
<body>

<p><a href="http://www.educacionit.com" title="vinculo">Educacion IT</a>.</p>

</body>
```

Titulo1★

Titulo2★

Contadores

De esta forma vemos cómo podemos generar estructuras repetitivas con **CSS** que son sumamente interesantes cuando no sabemos en principio cuántos elementos tendrá nuestro Sitio más adelante entonces preparamos el entorno para **múltiples escenarios**.

Muchas veces incluso **::AFTER** y **::BEFORE** terminan siendo herramientas para generar cortes en la estructura por ejemplo es común ver como a un contenido a su vez se le agrega la propiedad `display:block` o mismo `:inline` para cortar líneas por ejemplo de imágenes en una galería. Es decir en conclusión **::AFTER** y **::BEFORE** son mucho más que simples elementos para agregar imágenes o contenidos.

Veamos un ejemplo **con un vínculo**,

```
#encabezado a::before { content: '-'; }
```


Revisión

- Repase los **pseudo elementos** `::after` y `::before`
- Integre **content** y agregue **contenido e imágenes**
- Aplique todas las propiedades en el **proyecto integrador**
- Combine los **pseudo elementos** con las propiedades vistas hasta ahora
- Realice las preguntas necesarias al/la docente antes de continuar



¡Muchas gracias!

¡Sigamos trabajando!