**Pseudo-clases y Pseudo-elementos**

Las **Pseudoclases** **son palabras clave que representan el estado** en el que se puede encontrar un elemento HTML. En otras palabras, define el estilo de un elemento al momento de pasar el puntero.

La sintaxis de las pseudoclases es la siguiente:

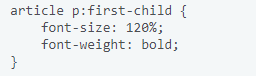


Como podemos ver las Pseudoclases comienzan con “:” luego de la palabra clave que tiene la pseudo clase. Estas tienen diferentes propiedades y vamos a visualizar algunas y que funciones tienen.

**First child**

Siempre seleccionará el primer elemento hijo del articulo seleccionado en el DOM. A continuación, visualizaremos una imagen en donde el selector es “p” y con la referencia de first-child estamos seleccionando al primer elemento hijo del dom de “p”.

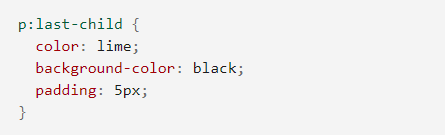
En el ejemplo podría escribirse “article:first-child” y la regla se aplicaría a cualquier elemento que sea primer hijo de un elemento “article”, no solo un párrafo primer hijo.



**H2:first-child{ text-decoration: underline;}** Sirve para agregar una línea de subrayado a todos los hijos <h2> de <article>.

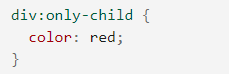
**Last Child**

La Pseudoclase:last-child representa el último elemento entre un grupos de elementos hermanos.  
A continuación, veremos un ejemplo en donde aplicaremos esta Pseudoclase a 3 hermanos “p” el cual estamos seleccionando el último de ellos para aplicarle los estilos.

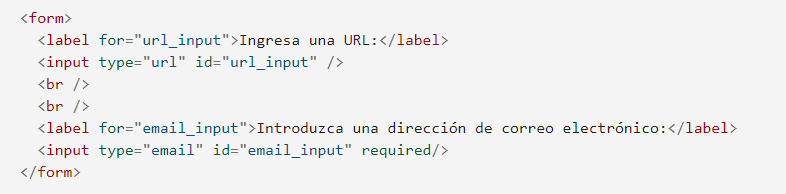


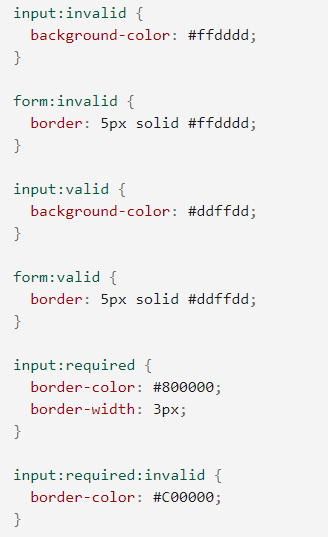
**Only Child**

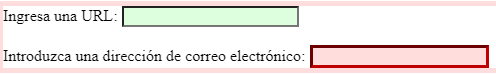
Esta pseudo clase hace **selección a un único hijo que no tenga hermanos.** En la siguiente imagen podemos ver un ejemplo en donde tenemos dos contenedores padres, en el primero con un **único div hijo** y el segundo **con varios elementos hijos**, esta regla aplicaría al primero por ser único hijo.



**Invalid**

Esta pseudo clase es útil para trabajar con **los formularios** que representa a cualquier elemento **input cuyos contenidos no se puedan validar**. Esta pseudo clase es útil para resaltar errores de campo para el usuario. En el siguiente ejemplo colorea los elementos en verde cuando son válidos y en rojo cuando no lo son. La sintaxis es como las demás pseudo clases “:invalid”.





**Hover**

La pseudo clase “:hover” es considerada dinámica, solo interviene si el usuario pasa el cursor sobre un elemento cuando esto ocurre hover aplica una acción de diseño.

**A:hover{ color: green;}** Cambiara el elemento a color green cuando el usuario pase el cursor por encima.

**Focus**

La pseudo clase “:focus” es considerada dinámica y solo interviene si el usuario selecciona el elemento con los controles del teclado.

**Input:focus{ width: 160px; height: 140px;}** Aumenta el tamaño de los input del formulario del contacto.

**Visited**

La pseudo clase **“:visited”** es considerada dinámica y generalmente se aplica a links ya que tomará el estilo luego de que el usuario haya hecho “click” en el elemento que tenga esta pseudo clase especificada.

**A[href^ = ’http://’]: visited{ color: red;}** Esto retornara que el link luego de visitar la página externa, quedará en color rojo.

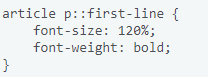
**Pseudoelementos**

Los **pseudoelementos** se comportan de manera similar. Sin embargo, actúan como si hubieras añadido un elemento HTML totalmente nuevo en el marcado, en lugar de haber aplicado una clase nueva a los elementos presentes. Los pseudoelementos comienzan con un doble signo de dos puntos “::”

Supongamos que se desea seleccionar la primera línea de un párrafo simplemente se puede delimitar con el elemento <span> y utilizar un selector de elementos. Sin embargo, esto fallará si el número de palabras que se delimitó resulta ser más largo o más corto que el ancho del elemento padre. Ya que normalmente no sabemos cuántas palabras caben en una línea porque esto cambia con el ancho de pantalla o con los cambios de tamaño de letra, no es posible hacer esto introduciendo solo HTML.

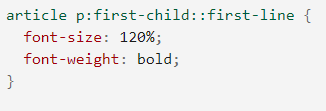
**First line**

Al ejemplo anterior podemos solucionarlo con el pseudo elemento “::first-name” ya que no importa si el número de palabras aumenta o disminuye, siempre se seleccionará la primer línea.



**Pseudoelementos y Pseudoclases**

Podemos hacer uso de ambos cuando sea necesario en el ejemplo anterior en el caso de tener 2 <p> podemos usar una pseudo clase para seleccionar el primer hijo del primer párrafo:



**Generar contenido con ::before y ::after**

Hay un par de pseudoelementos especiales que se utilizan junto con la propiedad de **content** para introducir contenido en el documento usando el CSS.

Los usos más habitualesson los iconos dentro de cadenas de texto los cuales se pueden cambiar la posición. En la siguiente imagen podemos ver un ejemplo en el que tenemos en el **content** una **imagen** en la que el pseudoelemento lo coloca luego del contenedor “box” por declararse “after”. Si el pseudoelemento es “before” lo colocaría antes de “box”.

