

# PSEUDO-CLASES Y PSEUDO-ELEMENTOS

## 1. Idea general

Una pseudo-clase describe un **estado** especial de un elemento (cómo está o dónde está en el árbol del DOM).

Un pseudo-elemento describe una **parte** concreta de ese elemento (primera letra, primera línea, contenido antes/después, marcador de lista, etc.).

---

## 2. Pseudo-clases en CSS

### 2.1. Conceptos básicos

- Sintaxis: selector:pseudo-clase { reglas }.
- Se usan para estados de interacción, estado de enlaces, validación de formularios, posición estructural, idiomas, etc.

#### Ejemplo sencillo:

```
css
a:hover {
    color: red;
}
```

### 2.2. Enlaces e interacción

- **:link:** enlace no visitado.
- **:visited:** enlace ya visitado.
- **:hover:** cuando el ratón está encima.
- **:active:** mientras se está haciendo clic.
- **:focus:** elemento con foco (clic o teclado).
- **:focus-visible:** foco visible sólo cuando viene del teclado (mejora accesibilidad).
- **:focus-within:** se aplica a un contenedor si algún descendiente tiene foco.
- **:any-link:** cualquier enlace con href (a o area).

#### Ejemplo:

```
css
a:link { color: blue; }
a:visited { color: purple; }
a:hover { text-decoration: underline; }
a:active { color: red; }
```

### **2.3. Formularios y validación**

- **:enabled / :disabled:** campos activados o desactivados.
- **:read-only / :read-write:** sólo lectura o editable.
- **:required / :optional:** con o sin atributo required.
- **:valid / :invalid:** el valor cumple o no las restricciones (tipo, min, max, pattern...).
- **:in-range / :out-of-range:** valores dentro o fuera del rango permitido.
- **:placeholder-shown:** cuando se muestra el placeholder.
- **:checked:** checkbox, radio u option marcado.
- **:indeterminate:** estado intermedio (por ejemplo, “seleccionar todo” parcialmente).
- **:user-valid / :user-invalid:** válidos/ inválidos, pero solo después de interacción del usuario.

#### **Ejemplo:**

css

```
input:valid {  
    border-color: green;  
}  
  
input:invalid {  
    border-color: red;  
}
```

### **2.4. Posición en el árbol (estructurales)**

- **:first-child / :last-child:** primer o último hijo de su padre.
- **:only-child:** único hijo del padre.
- **:nth-child(n) / :nth-last-child(n):** hijo en posición n, desde el principio o el final.
- **:first-of-type / :last-of-type:** primer o último elemento de su tipo entre sus hermanos.
- **:only-of-type:** único elemento de ese tipo entre sus hermanos.
- **:nth-of-type(n) / :nth-last-of-type(n):** como nth-child, pero filtrando por tipo de etiqueta.

#### **Ejemplo para filas alternas:**

css

```
tr:nth-child(even) { background: #f2f2f2; }  
tr:nth-child(odd) { background: white; }
```

## **2.5. Contenido, idioma, dirección y otros**

- **:empty:** elementos sin hijos ni texto.
- **:lang(xx):** elementos con un lang="xx" concreto (por ejemplo :lang(es)).
- **:dir(ltr) / :dir rtl:** según dirección del texto.
- **:root:** elemento raíz del documento (<html> en HTML), muy usado para variables CSS.
- **:defined:** elementos ya definidos (útil con Web Components).
- **:fullscreen:** elemento en pantalla completa.
- **:modal:** elemento en estado modal.
- **:popover-open:** elemento con popover visible.
- **:target:** elemento cuyo id coincide con el hash de la URL.
- **:scope:** referencia de contexto para selectores relativos.
- **:state():** estados personalizados en custom elements.

## **2.6. Pseudo-clases lógicas y relacionales**

- **:not(selector):** selecciona elementos que NO coinciden con el selector.
- **:is(selector-list):** agrupa varios selectores, simplificando el código.
- **:where(selector-list):** como :is, pero con especificidad 0 (ideal para estilos base).
- **:has(selector):** “mira hacia abajo”; selecciona un elemento si contiene algo que coincide con el selector.

### **Ejemplo con :has:**

```
css
.card:has(input:invalid) {
    border-color: red;
}
```

---

## **3. Pseudo-elementos en CSS**

### **3.1. Conceptos básicos**

- Sintaxis moderna: selector::pseudo-element { reglas }.
- Sirven para dar estilo a partes de un elemento que no son nodos HTML normales: primera letra, primera línea, marcador de lista, contenido generado, selección de texto, etc.

### **Ejemplo:**

```
css
q::before { content: "<"; }
q::after { content: ">"; }
```

### **3.2. Contenido generado: ::before y ::after**

- **::before:** inserta contenido antes del contenido real del elemento.
- **::after:** inserta contenido después del contenido real.
- Ambos requieren la propiedad content (aunque sea "").

#### **Ejemplo icono en enlaces externos:**

css

```
a.external::after {  
    content: "↗";  
    font-size: 0.8em;  
    margin-left: 0.25em;  
}
```

### **3.3. Tipografía: ::first-line y ::first-letter**

- **::first-line:** aplica estilo solo a la primera línea visual de un bloque.
- **::first-letter:** aplica estilo a la primera letra de la primera línea (capitulares).

#### **Ejemplo:**

css

```
p::first-letter {  
    font-size: 3em;  
    float: left;  
    margin-right: 0.1em;  
}
```

### **3.4. Formularios y selección de texto**

- **::placeholder:** estilo del texto placeholder de inputs y textareas.
- **::selection:** estilo del texto seleccionado por el usuario (color de fondo / texto).

#### **Ejemplo:**

css

```
::selection {  
    background: #ffeb3b;  
    color: #000;  
}
```

```
input::placeholder {  
    color: #aaa;  
    font-style: italic;  
}
```

### **3.5. Listas y controles especiales**

- **::marker:** marcador de listas (ul / ol), viñetas o números.
- **::file-selector-button:** botón de los inputs type="file".

#### **Ejemplos:**

css

```
li::marker {
```

```
  color: #f39c12;
```

```
  font-size: 1.2em;
```

```
}
```

```
input[type="file"]::file-selector-button {  
  background: #3498db;  
  color: white;  
  border: none;  
  padding: 0.4em 0.8em;  
  border-radius: 4px;  
  cursor: pointer;  
}
```

---

## **4. Resumen rápido para repasar**

1. **Pseudo-clases (:hover, :focus, :nth-child, :valid...):** estado, interacción del usuario o posición en el DOM.

2. **Pseudo-elementos (::before, ::after, ::first-letter, ::placeholder...):** partes concretas o contenido generado sin tocar el HTML.

### **3. Pseudo-clases clave:**

- **Interacción:** :hover, :active, :focus, :focus-visible.
- **Enlaces:** :link, :visited.
- **Formularios:** :required, :optional, :valid, :invalid, :checked, :disabled, :enabled, :placeholder-shown, :user-valid, :user-invalid.
- **Estructura:** :first-child, :last-child, :nth-child(), :first-of-type, :last-of-type, :only-child, :only-of-type.
- **Lógicas:** :not(), :is(), :where(), :has().

4. **Pseudo-elementos clave:** ::before, ::after, ::first-line, ::first-letter, ::placeholder, ::selection, ::marker, ::file-selector-button.