
UD02 - Conceptes bàsics

Índex

- CSS - Conceptes Bàsics
 - Nomenclatura
 - Regles
 - Valors Especials:
 - Selectors
 - Selectors Bàsics
 - Combinadors (Combinators)
 - Selectors de Pseudo-Classe (Pseudo-class Selectors)
 - Selectors de Pseudo-classe Genèrics
 - Selectors de Pseudo-classe Específics per a Formularis
 - Selectors de Pseudo-element
 - `::before` `i` `::after`
 - `::first-letter` `i` `::first-line`
 - `::selection`
 - Selectors d'Atribut
 - Selectors d'Atribut *Case-insensitive*
 - Cascada (WIP)
 - Origen (WIP)
 - Especificitat (WIP)
 - Herència (WIP)
 - Variables
 - Aritmètica CSS3
 - Unitats
 - Longituds absolutes
 - Longituds relatives
 - Càlculs (WIP)
 - Regles "AT" (@)
 - Referències
- Javascript - jQuery
 - Referències

CSS - Conceptes Bàsics

CSS - Cascade Style Sheets



Nivells d'estandardització:

1. Editor's Draft (ED)
2. Working Draft (WD)
3. Transition – Last Call Working Draft (LCWD)
4. Candidate Recommendation (CR)
5. Transition – Proposed Recommendations (PR)
6. Recommendation (REC)

Font: <https://css-tricks.com/css-standards-process/>

Nomenclatura

- **Propietat:** Designa un paràmetre o característica que desitjam ajustar a l'element o elements seleccionats.

- **Valor:** El valor que li volem assignar a una *propietat*.

- **Declaració:** És el conjunt d'una *propietat* i el seu *valor*, separades per dos punts (":").

- **Bloc de Declaracions:** És una llista d'una o més *declaracions*, separades per punt i coma (";"), i envoltades en claus ("{" ... "}").



- **Selector:** És una expressió que ens permet seleccionar un o varis elements del DOM als que volem aplicar un *bloc de declaracions*.
- **Regla:** Es el conjunt format per un *selector* i un *bloc de declaracions*.
 - També pot tenir varis selectors separats per comes (",").
- **"Multipropietats" (shorthand property):**
 - Ex: `font = font-style + font-weight + font-size + line-height + font-family`.
 - Ex: `margin = margin-top + margin-right + margin-bottom + margin-left`.
 - Acostumen guardar coherència:
 - Ex: Top - Right - Bottom - Left ("TRouBLe" / Des d'amunt i en sentit de les agulles del rellotge).
 - Valors per defecte "simètrics"
 - En cas de dubte, consultar documentació.
 - **Efecte col·lateral:** SEMPRES estableixen totes les propietats (encara que s'ometin).

Regles

Valors Especials:

- *Initial*: És el valor per defecte de l'atribut a un determinat tipus d'element.
- *Auto*: Significa que el navegador pot ajustar-lo automàticament en funció de certes premisses preestablertes.

Selectors

Selectors Bàsics

- tagname
- .class
- #id
- *

Combinadors (Combinators)

Nom	Sintaxi	Exemple
Compound (composició)	(res) (a) (b)	h1.page-header
Descendant (descendent)	(espai) (a) (b)	header h2
Child (descendent directe)	(>) (a) > (b)	li>ul
Adjacent Sibling (posterior)	(+) (a) + (b)	th+tr
Following Sibling (el següent)	(~) (a) ~ (b)	tr.today~tr

Selectors de Pseudo-Classe (Pseudo-class Selectors)

Els selectors de pseudo-classe ens permeten adreçar elements per alguna característica extrínseca a ells com ara la ubicació en relació a altres elements o determinats estats transitoris com ara si ténen el focus del teclat o si el cursor del ratolí es troba a sobre d'ells.

La llista completa de selectors de pseudo-classe es pot trobar al MDN sota:

- <https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes>.

Selectors de Pseudo-classe Genèrics

:first-child	:first-of-type
:last-child	:last-of-type
:only-child	:only-of-type
:nth-child(an+b)	:nth-of-type(an+b)
:nth-last-child(an+b)	:nth-last-of-type(an+b)
:not(<selector>)	:root
:focus	:hover
:empty	":blank" (proposta)

Notes:

- :root selecciona el tag html en documents HTML.
- :empty no pot contenir ni un espai en blanc o salt de línia.
- ":blank" de moment només és una proposta: no suportat per cap navegador a dia d'avui.

Selectors de Pseudo-classe Específics per a Formularis

Alguns selectors de pseudo-classe es refereixen específicament a característiques de camps de formularis (tags <input>, <textarea>, <button>, <select> i/o <option>, segons el cas):



:enabled	:disabled
:valid	:invalid
:required	:optional
:checked	



Avís: Alguns d'ells han estat introduïts o redefinits recentment pel que no funcionaran com s'espera en alguns navegadors.

Haurem de consultar cadascun a <http://caniuse.com> per a saber exactament com estan de suportats als distints navegadors.

Selectors de Pseudo-element

Els pseudo-elements, a diferència de les pseudo-classes, no adrecen elements (tags) realment existents al document, sinó que *virtualment* en creen de nous en un lloc determinat o sobre una part determinada de l'element seleccionat.

Per exemple, `p::first-letter` ens permetria aplicar un estil distint **únicament** a la primera lletra de cada paràgraf com si aquesta estigués continguda en el seu propi *subtag*.

Es distingeixen de les pseudo-classes per anar prefixats per dos caràcters de "dos punts" (: :) en comptes d'un sol com els pseudo-elements.



Avís: Ens podem trobar però casos en que s'hagin especificat amb només ':' ja que així fou com s'especificaren originalment i per ara tots els navegadors els reconeixen també. Nosaltres **sempre** els escriurem com a `::after` i no `:after`, per exemple. I si detectam codi antic o algun cas en que, per error, hem fet servir la versió incorrecta, la rectificarem.

`::before` i `::after`

Creen un pseudo-element que passa a ser el primer o darrer fill, respectivament, de l'element seleccionat.

- Es poden utilitzar per pre/su-fixar texts o imatges.
- És **obligatori** definir la propietat `content`, encara que sigui amb un espai en blanc, ja que **del contrari no apareixeran**.

`::first-letter` i `::first-line`

Permeten adreçar, respectivament, la primera lletra o la primera línia de text, d'un element.

`::selection`

Permet adreçar text (i sub-elements) que estiguin seleccionats per l'usuari.

Només permet unes poques propietats. Incloues `color`, `background-color`, `cursor`, and `text-decoration`.

Selectors d'Atribut

Els selectors d'atribut permeten seleccionar elements en funció dels seus atributs HTML i ténen la mateixa especificitat que els selectors de classe (0,1,0).

- `[attr]`: Existència de l'atribut (l'elements tenen l'atribut especificat, independentment del seu valor).
 - Exemple: `input[disabled]`
- `[attr="value"]`: Concordància de valor.
 - Exemple: `input[type="button"]`
- `[attr^="value"]`: Concordància inicial (que el valor "comença per...")
 - Exemple: `a[href^= "https"]`
- `[attr$="value"]`: Concordància final (que el valor "acaba en...")
 - Exemple: `a[href$= ".pdf"]`
- `[attr*="value"]`: Concordància parcial (l'atribut conté el text especificat).
 - Exemple: `*[class*="sprite-"]`.
- `[attr~="value"]`: Concordància d'element en llista (d'elements separats per espai)
 - Exemple: `div[class="menu"]` (en aquest cas equivaldria a `div.menu`, però no està limitat a l'atribut "class").
- `[attr|="value"]` Igualtat o concordància de prefix (prefix = fins al primer "-").
 - Exemple: `div[lang|"es"]` (concordaria amb `lang="es"`, `lang="es-ca"`, etc..)

Selectors d'Atribut *Case-insensitive*

Els selectors d'atribut *tradicionalment* distingeixen entre majúscules i minúscules, pel que, per exemple, '

//// TODO ////

All of the previous attribute selectors are case-sensitive. The selectors level 4 specification introduces a case-insensitive modifier that can be added to any attribute selector. To do this, add an `i` before the closing bracket. Example: `input[value="search" i]`.

Many browsers don't yet support this. Those that don't will ignore it. For this reason, if you use case-insensitive modifiers, be sure to add a fallback to a regular case-sensitive version.

Cascada (WIP)

1. Origen del Full d'Estil
2. Especificitat del Selector
3. Ordre al Codi Font

Origen (WIP)



Avís: (Browser Styles) WIP...

- [normalize.css](#).

Especificitat (WIP)

Selector	Nº d'IDs	Nº de Classes	Nº de Tags	Notation
html body header h1	0	0	4	0,0,4
section header.section-header h1	0	1	3	0,1,3
.page-header .title	0	2	0	0,2,0
#sidebar	1	0	0	1,0,0

Més: <https://developer.mozilla.org/en-US/docs/Web/CSS/Specificity>.

Herència (WIP)

Variables

Una variable CSS es defineix mitjançant una declaració a la qual, el nom de la propietat comença per dos guions ("--").

Exemple:

```
--home_background_color: cyan;
```

Funcionen com una declaració normal i son heretades pels elements descendents amb la única diferència que no tenen cap efecte per si mateixes.

Per llegir-les fem servir la funció `var()`.

Exemple:

```
background-color: var(--home_background_color);
```

Aritmètica CSS3

Unitats

Font: https://www.w3schools.com/cssref/css_units.asp

Longituds absolutes

Unitat	Descripció
cm	centimeters
mm	millimeters
in	inches (1in = 96px = 2.54cm)
px *	pixels (1px = 1/96th of 1in)
pt	points (1pt = 1/72 of 1in)
pc	picas (1pc = 12 pt)

🔑 Els 'px' son en realitat una unitat relativa i no es corresponen amb els pixels físics del dispositiu sino que van en funció de la definició d'aquest amb la finalitat que les mesures en 'px' siguin el més semblants possibles en el món real independentment del dispositiu de visualització.

Longituds relatives

Unitat	Descripció
em	Relative to the font-size of the element (2em means 2 times the size of the current font)
ex	Relative to the x-height of the current font (rarely used)
ch	Relative to width of the "0" (zero)
rem	Relative to font-size of the root element
vw	Relative to 1% of the width of the viewport*
vh	Relative to 1% of the height of the viewport*
vmin	Relative to 1% of viewport's* smaller dimension
vmax	Relative to 1% of viewport's* larger dimension
%	Relative to the parent element

Càlculs (WIP)

(funció calc())

Regles "AT" (@)

Referències

- Media print:
 - Designing For Print With CSS: <https://www.smashingmagazine.com/2015/01/designing-for-print-with-css/>
 - Maquetació avançada d'impressió
 - (Desgraciadament cap navegador suporta massa més que establir els marges).

Javascript - jQuery.

Referències