

Web Design & Programming

Cascading Style Sheets (CSS) – Part 2

Xavier Belanger

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

<http://creativecommons.org/licenses/by-sa/4.0/>



You are free to:

- **Share** — copy and redistribute the material in any medium or format
- **Adapt** — remix, transform, and build upon the material for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

- **Attribution** — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- **ShareAlike** — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
- **No additional restrictions** — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Viewport

- The viewport is the usable, visible area in a web browser.
- For a desktop web browser, that's the inside of the web browser window (excluding menus and such).
- For a mobile web browser, the full size of the screen is used most of the time (fullscreen mode).

Viewport Meta Declaration

- That rule will define how the content should be displayed.
- The current best practice is to not define anything strict, and to let the end-user adjust the display as needed.
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`

Units

- Two types of units can be used with CSS:
 - absolute (predefined, used in the real world)
 - relative (calculated from another value)

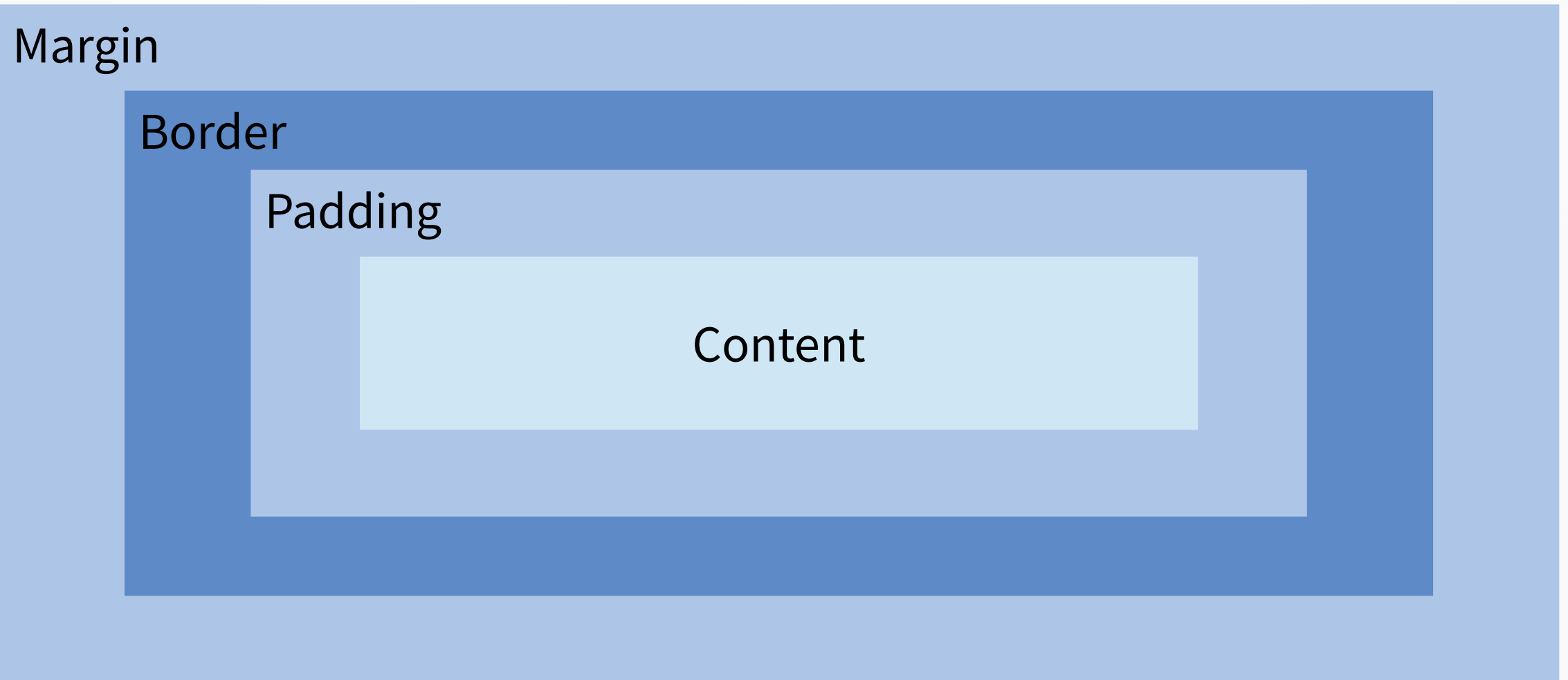
Absolute Units

- *in*: inch
- *px*: pixel, 1/96 of an inch
- *mm, cm*: millimeter, centimeter
- *q*: 1/40 of a centimeter
- *pt, pc*: point, pica, used for printed media

Relative Units

- *em*: font size of the current element
- *rem*: font size of the root element (16 pixels by default)
- *ex*: "height of an x character"
- *vh*: 1/100 of the viewport height
- *vw*: 1/100 of the viewport width
- *vmin*, *vmax*: minimum and maximum viewport values

The CSS Box Model



Borders

- *border-style*: none, solid, dotted, dashed, ...
- *border-width*: thin, medium, thick or a value
- *border-color*: color
- *border-radius*: value
- *border-top-**, *border-bottom-**, *border-right-**, *border-left-**

Box Shadows

- *Box-shadow:*
 - horizontal offset
 - vertical offset
 - blur distance
 - spread distance
 - color
- `box-shadow: 3px 3px 5px 10px #808080;`

Positioning

- You can position an element using the position CSS attribute.
- Five values are possible: static, relative, absolute, fixed and sticky.
- You can combine this with top, bottom, right and left values.

Positioning Values

- *static*: normal positioning, no changes
- *relative*: offset from the original position
- *absolute*: based from the viewport or the containing element
- *fixed*: based from the viewport and doesn't move
- *sticky*: relatively positioned, until scrolling where it will become fixed

Transformations

- CSS rules that can be used to rotate, translate, scale or skew an element.
- Multiples transformations can be applied to the same element.
- You can define the origin for the transformation (center by default) with transform-origin.
- `transform-origin: center bottom;`

Rotate

- transform: rotate(90deg);
- transform: rotate(-90deg);

Translate

- `transform: translateX(100px);`
- `transform: translateY(20px);`
- `transform: translate(100px, 20px);`

Scale

- `transform: scaleX(1.5);`
- `transform: scaleY(2);`
- `transform: scale(1.5, 2);`
- You can also use *scale3d* with 3 values (x, y, z)

Skew

- `transform: skewX(10deg);`
- `transform: skewY(20deg);`
- `transform: skew(10deg, 20 deg);`

Gradients

- linear-gradient, radial-gradient, conic-gradient
- repeating-linear-gradient,
- repeating-radial-gradient
- `background-image: linear-gradient 15deg, black, white);`