

Web Development Bootcamp: Assignment 1

Name: Theresa Kahale

I. Relative vs Absolute Positioning in CSS:

Absolute Positioning: allows the developer to insert the element exactly where he wants. The element is positioned with respect to the closest/first positioned (not default, element we clearly specify its position) ancestor/parent. Positions the element out of the flow of the rest of the document and other parts will disregard it. It requires the coordinates of the element (top, bottom, right, left). If there is no positioned parent, the default is the body itself.

Relative Positioning: All components present in this division positioned relatively have their coordinates relative to it. When adding coordinates properties, the element will be placed relative to its original/normal position (itself) without affecting the layout or the elements around it. For example, if I specify a left of 20px with a position: relative, the element will move horizontally to the left direction 20px away from where it was.

II. CSS box sizing: Margin vs Padding:

Padding: Adds space between the element inside of its borders (division). The added space will never go outside the border of the element. Either we specify the four properties: *padding-top*, *padding-right*, *padding-bottom*, *padding-left* or directly using *padding* with many parameters. Negative values are not allowed basically because the element is not allowed to go outside its border. If the width of the element is specified, then the padding adds size to it.

- If **one** value inputted: add space inside the element evenly in the four directions.

Margin: Adds space outside the element. Used to create gaps between elements (positive margin) or to overlap elements (negative margin). We determine the space surrounding the element, in all four directions. Either we specify the four properties: *margin-top*, *margin-right*, *margin-bottom*, *margin-left* or directly using *margin* with many parameters.

- If **one** value inputted: add space outside the elements evenly in the four directions. If we want to horizontally center the element in its division (container), then we can input one value: **auto**. The element will have its width as specified, and the remaining space horizontally is equally divided between the left and right margins.
- If **two** values inputted: correspond to top/bottom, right/left.
- If **three** values inputted: correspond to top, right/left, bottom.
- If **four** values inputted: correspond to top, right, bottom, left.

For both sizing:

- If a **length** is inputted: specifies a fixed value of the margin/padding.
- If a **percentage** is inputted: partition of the width of the containing element
- If **auto** is inputted: browser specifies the margin/padding
- If **inherit** is inputted: the child will use the margin/padding specifications of its parent.