codecademy

Learn CSS: The Box Model

CSS Margin Collapse

CSS *margin collapse* occurs when the top and bottom margins of blocks are combined into a single margin equal to the largest individual block margin.
Margin collapse only occurs with vertical margins, not for horizontal margins.

```
/* The vertical margins will collapse to 30
pixels
instead of adding to 50 pixels. */
.block-one {
  margin: 20px;
}

.block-two {
  margin: 30px;
}
```

CSS `auto` keyword

The value `auto` can be used with the property `margin` to horizontally center an element within its container. The `margin` property will take the width of the element and will split the rest of the space equally between the left and right margins.

```
div {
  margin: auto;
}
```

Dealing with `overflow`

If content is too large for its container, the CSS `overflow` property will determine how the browser handles the problem. By default, it will be set to `visible` and the content will take up extra space. It can also be set to `hidden`, or to `scroll`, which will make the overflowing content accessible via scroll bars within the original container.

```
small-block {
  overflow: scroll;
}
```

Height and Width Maximums/Minimums

The CSS `min-width` and `min-height` properties can be used to set a minimum width and minimum height of an element's box. CSS `max-width` and `max-height` properties can be used to set maximum widths and heights for element boxes.

```
/* Any element with class "column" will be at
most 200 pixels wide, despite the width property
value of 500 pixels. */

.column {
  max-width: 200px;
  width: 500px;
}
```

The `visibility` Property

The CSS `visibility` property is used to render `hidden` objects invisible to the user, without removing them from the page. This

```
.invisible-elements {
```

ensures that the page structure and organization remain unchanged.

```
  visibility: hidden;
}
```

The property `box-sizing` of CSS *box model*

The CSS *box model* is a box that wraps around an HTML element and controls the design and layout. The property `box-sizing` controls which aspect of the box is determined by the `height` and `width` properties. The default value of this property is `content-box`, which renders the actual size of the element including the content box; but not the paddings and borders. The value `border-box`, on the other hand, renders the actual size of an element including the content box, paddings, and borders.

```
.container {
  box-sizing: border-box;
}
```

CSS `box-sizing: border-box`

The value `border-box` of the `box-sizing` property for an element corresponds directly to the element's total rendered size, including padding and border with the `height` and `width` properties.
The default value of the `border-box` property is `content-box`.
The value `border-box` is recommended when it is necessary to resize the `padding` and `border` but not just the content. For instance, the value `border-box` calculates an element's `height` as follows: `height` = content `height` + `padding` + `border`.

```
#box-example {
  box-sizing: border-box;
}
```