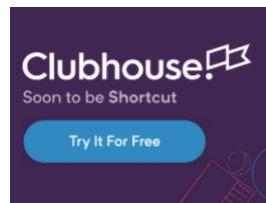


Search for a property

- border-bottom-width border-left-width border-right-width border-top-width border-width box-sizing height line-height margin-bottom margin-left margin-right margin-top margin max-height max-width min-height min-width



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

# Box model in CSS

The CSS properties that allow you to define the size and behavior of the HTML element

Share this page



New!

My 44-page ebook "CSS in 44 minutes" is out! 😊

Get it now →



Microsoft Azure — Quickly and accurately transcribe audio to text with Azure AI.

AD

## # border-bottom-width

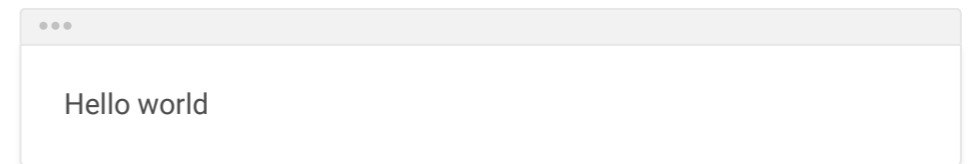
In collection: **Box-Model** Permalink Share MDN

Like `border-width`, but for the **bottom** border only.

```
border-bottom-width: 0;
```

default

Removes the bottom border.



```
border-bottom-width: 4px;
```

You can use **pixel** values.



## # border-left-width

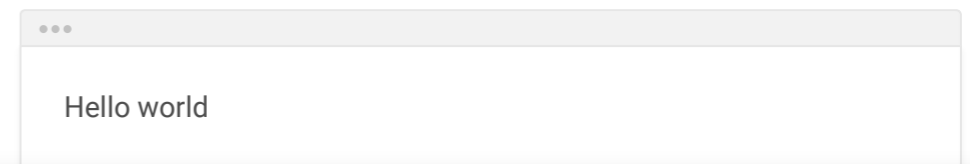
In collection: **Box-Model** Permalink Share MDN

Like `border-width`, but for the **left** border only.

```
border-left-width: 0;
```

default

Removes the left border.



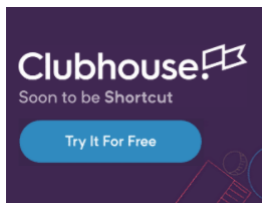
A free visual guide to CSS Created by @jgthms

Share

Star 4,261

Search for a property

- [border-bottom-width](#)
- [border-left-width](#)
- [border-right-width](#)
- [border-top-width](#)
- [border-width](#)
- [box-sizing](#)
- [height](#)
- [line-height](#)
- [margin-bottom](#)
- [margin-left](#)
- [margin-right](#)
- [margin-top](#)
- [margin](#)
- [max-height](#)
- [max-width](#)
- [min-height](#)
- [min-width](#)



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

You can use **pixel** values.



## # border-right-width

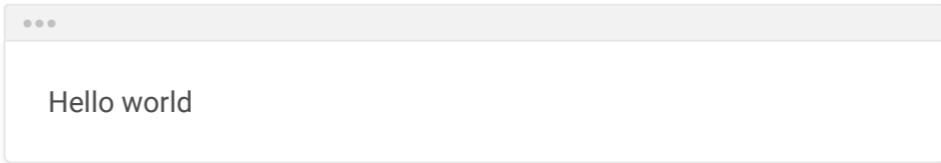
In collection: **Box-Model** [Permalink](#) [Share](#) [MDN](#)

Like `border-width`, but for the **right** border only.

```
border-right-width: 0;
```

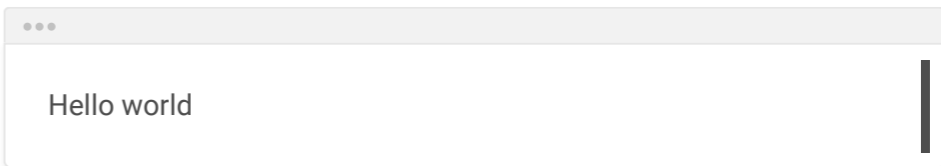
default

Removes the right border.



```
border-right-width: 4px;
```

You can use **pixel** values.



## # border-top-width

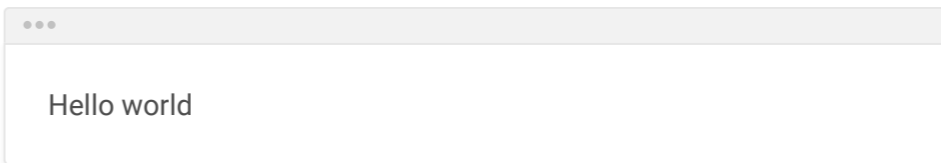
In collection: **Box-Model** [Permalink](#) [Share](#) [MDN](#)

Like `border-width`, but for the **top** border only.

```
border-top-width: 0;
```

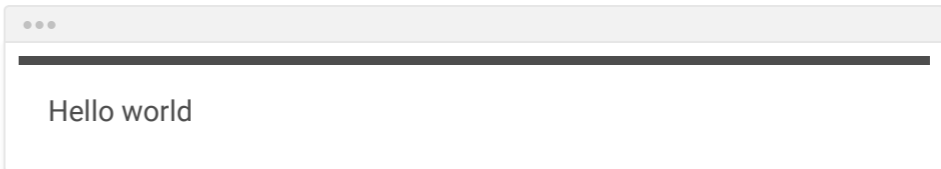
default

Removes the top border.



```
border-top-width: 4px;
```

You can use **pixel** values.



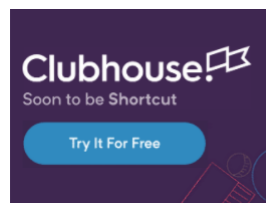
## # border-width

In collection: **Box-Model** [Permalink](#) [Share](#) [MDN](#)

A free visual guide to CSS Created by @jgthms

Share

Star 4,261



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

`border-width: 1px;`

Defines the width of **all** borders to 1px.

`border-width: 2px 0;`

Defines the top and bottom borders to 2px, the left and right to 0.



## # box-sizing

Defines how the width and height of the element are calculated: whether they include the *padding* and *borders* or not.

`box-sizing: content-box;`

default

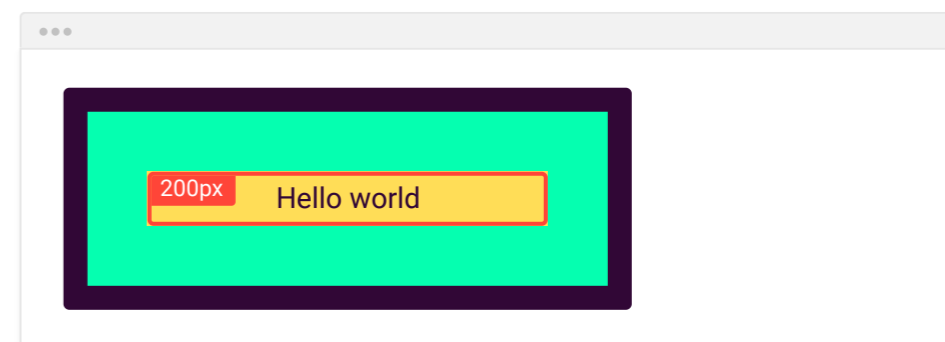
The **width** and **height** of the element only apply to the **content** of the element.

For example, this element has

- `border-width: 12px`
- `padding: 30px`
- `width: 200px`

The *full width* is  $24px + 60px + 200px = 284px$ .

The content has the defined width. The box accomodates for those dimensions.



`box-sizing: border-box;`

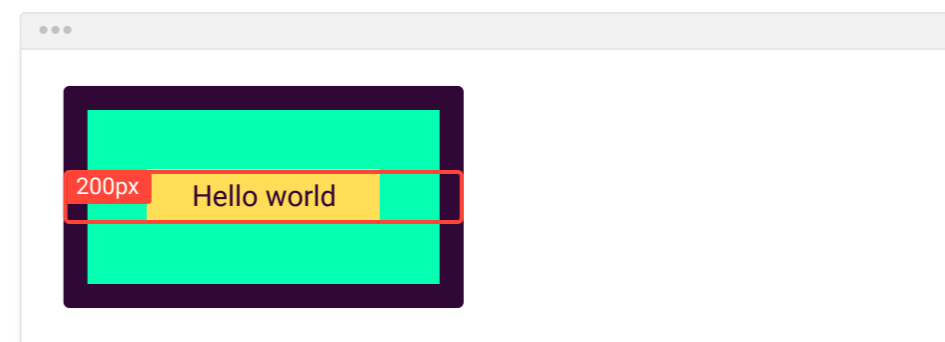
The **width** and **height** of the element apply to all parts of the element: the **content**, the **padding** and the **borders**.

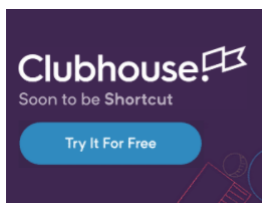
For example, this element has

- `border-width: 12px`
- `padding: 30px`
- `width: 200px`

The *full width* is **200px**, no matter what.

The box has the defined width. The content accomodates for those dimensions, and ends up being  $200px - 60px - 24px = 116px$ .





Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

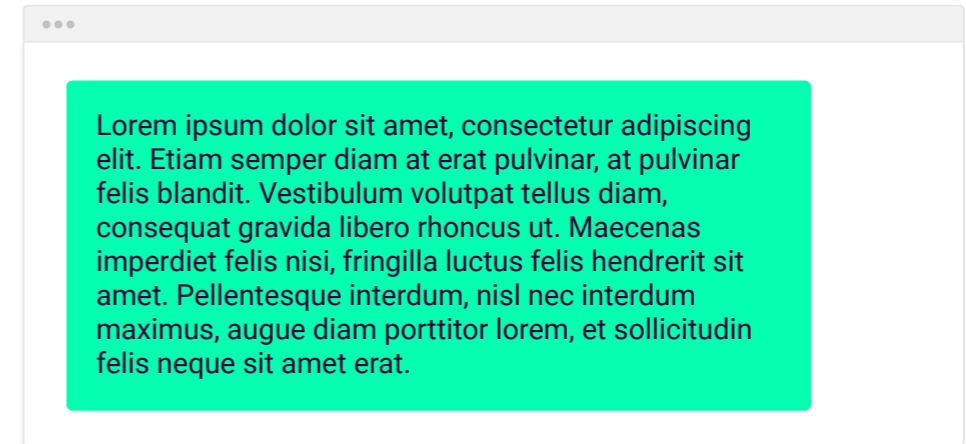
## # height

Defines the height of the element.

```
height: auto;
```

default

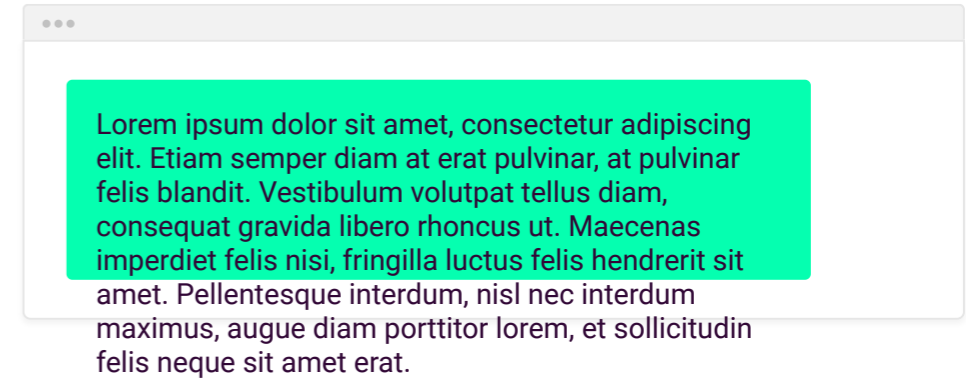
The element will **automatically** adjust its height to allow its content to be displayed correctly.



```
height: 100px;
```

You can use **numeric values** like **pixels, (r)em, percentages...**

If the content does not fit within the specified height, it will **overflow**. How the container will handle this overflowing content is defined by the `overflow` property.



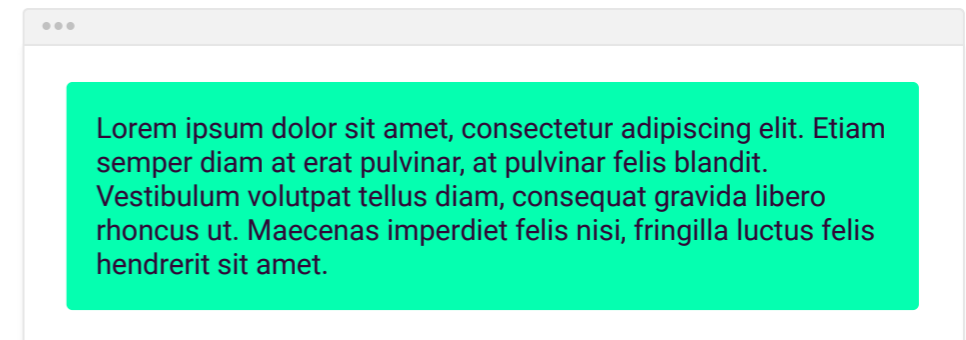
## # line-height

Defines the height of a single line of text.

```
line-height: normal;
```

default

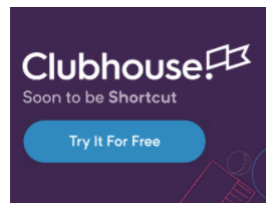
Reverts to the **default** value of the browser.



```
line-height: 1.6;
```

recommended





Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

```
line-height: 30px;
```

You can use **pixel** values.

```
line-height: 0.8em;
```

You can use **em** values: like with unitless values, the line height will be relative to the font size.

semper diam at erat pulvinar, at pulvinar felis blandit.

Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut. Maecenas imperdiet felis nisi, fringilla luctus felis hendrerit sit amet.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit.

Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut. Maecenas imperdiet felis nisi, fringilla luctus felis hendrerit sit amet.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut. Maecenas imperdiet felis nisi, fringilla luctus felis hendrerit sit amet.

## # margin-bottom

In collection: **Box-Model** [Permalink](#) [Share](#) [MDN](#)

Defines the space **outside** the element, on the **bottom** side.

```
margin-bottom: 0;
```

default

Removes any margin at the bottom.

First item

Target

Third item



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

You can use **pixel** values.

```
margin-bottom: 2em;
```

You can use **(r)em** values.

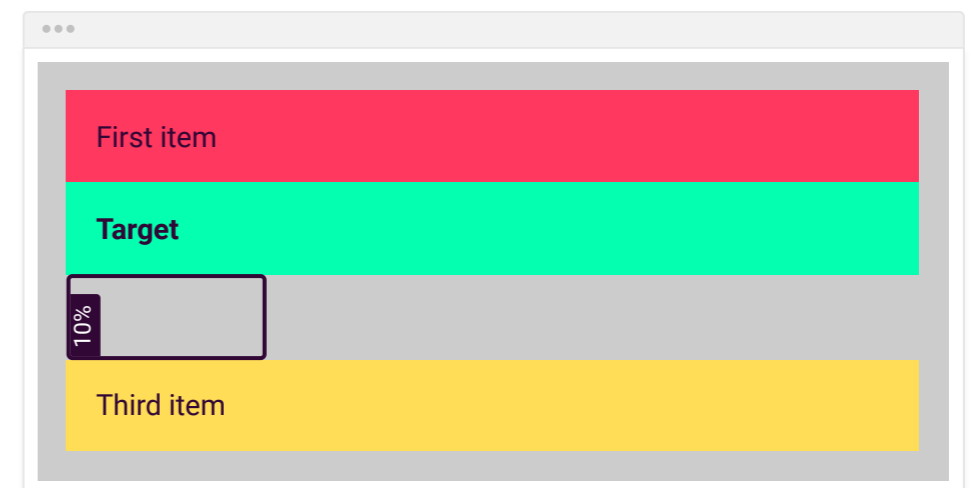
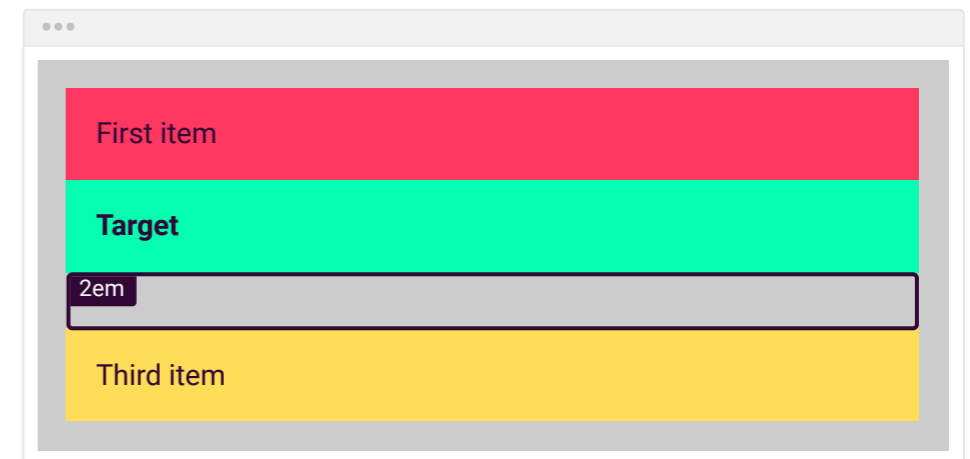
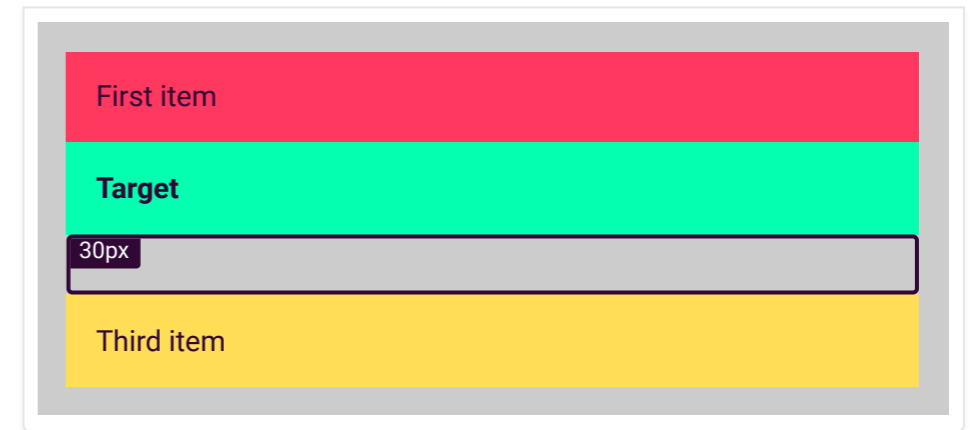
The value is **relative** to the font size:

- **em**: relative to the *element's* current font size
- **rem**: relative to `<html>` the *root element's* font size

```
margin-bottom: 10%;
```

You can use **percentage** values.

The percentage is based on the *width* of the *container*.



## # margin-left

Defines the space **outside** the element, on the **left** side.

In collection: [Box-Model](#) [Permalink](#) [Share](#) [MDN](#)



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

Remove any margin on the left

```
margin-left: 50px;
```

You can use **pixel** values.

```
margin-left: 7em;
```

You can use **(r)em** values.

The value is **relative** to the font size:

- **em**: relative to the *element's* current font size
- **rem**: relative to `<html>` the *root element's* font size

```
margin-left: 30%;
```

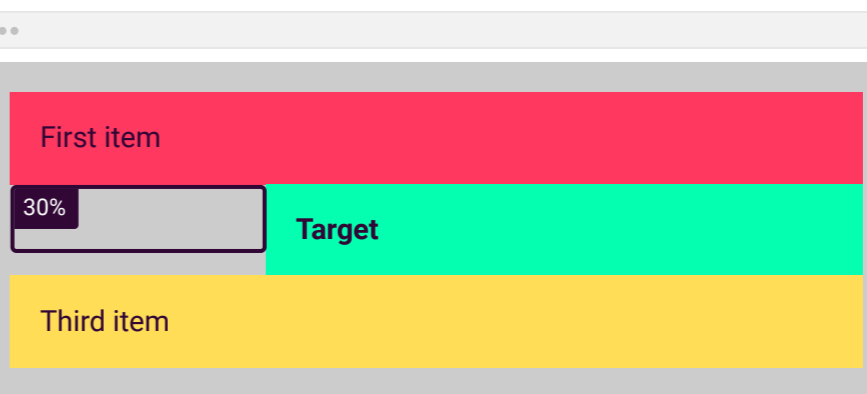
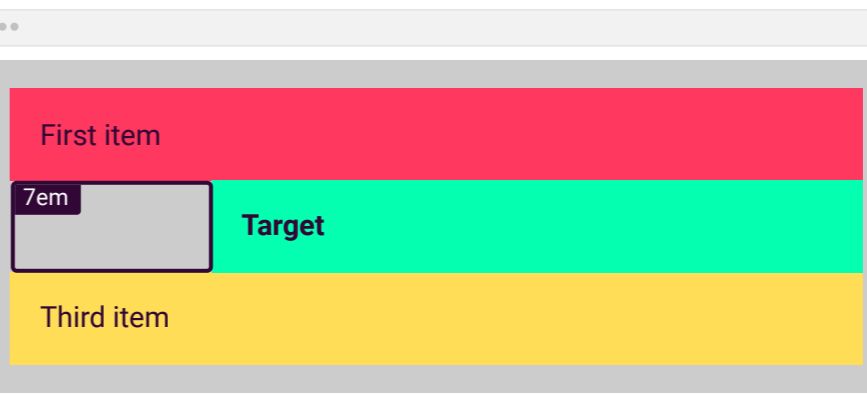
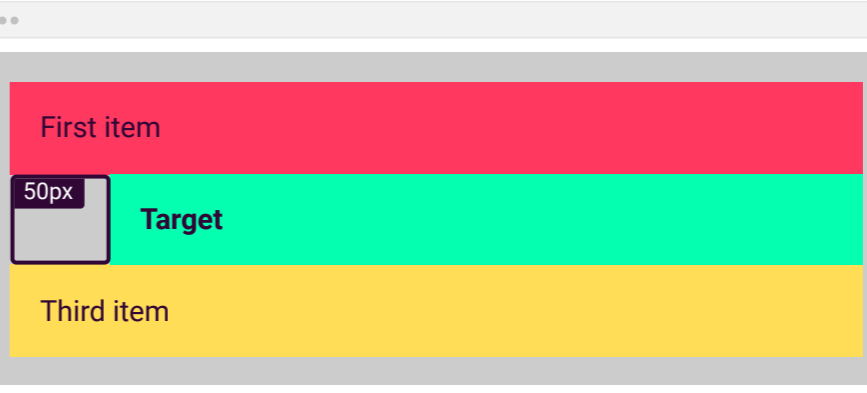
You can use **percentage** values.

The percentage is based on the *width* of the *container*.

```
margin-left: auto;
```

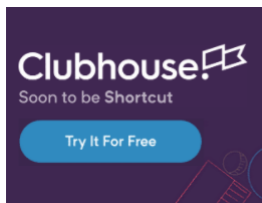
The **auto** keyword will give the left side a share of the **remaining space**.

When combined with `margin-right: auto`, it will **center** the element, if a fixed width is defined.



Search for a property

- border-bottom-width
- border-left-width
- border-right-width
- border-top-width
- border-width
- box-sizing
- height
- line-height
- margin-bottom
- margin-left
- margin-right
- margin-top
- margin
- max-height
- max-width
- min-height
- min-width



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

## # margin-right

Defines the space **outside** the element, on the **right** side.

```
margin-right: 0;
```

Removes any margin on the right.

default

```
margin-right: 50px;
```

You can use **pixel** values.

```
margin-right: 7em;
```

You can use **(r)em** values.

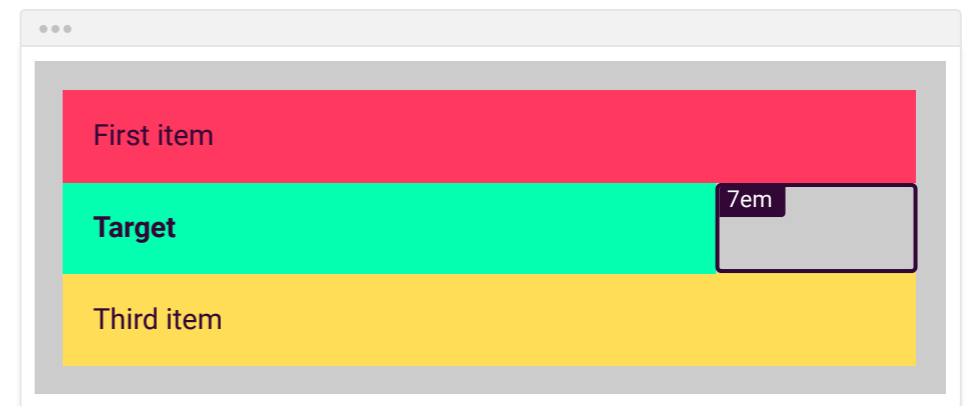
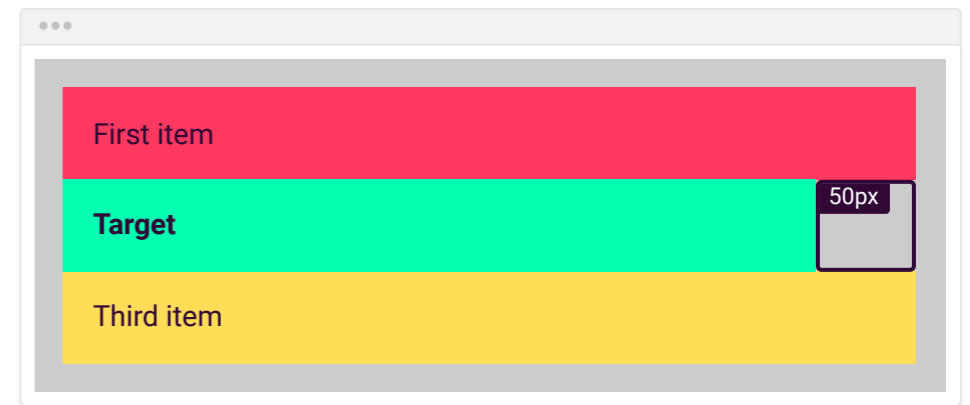
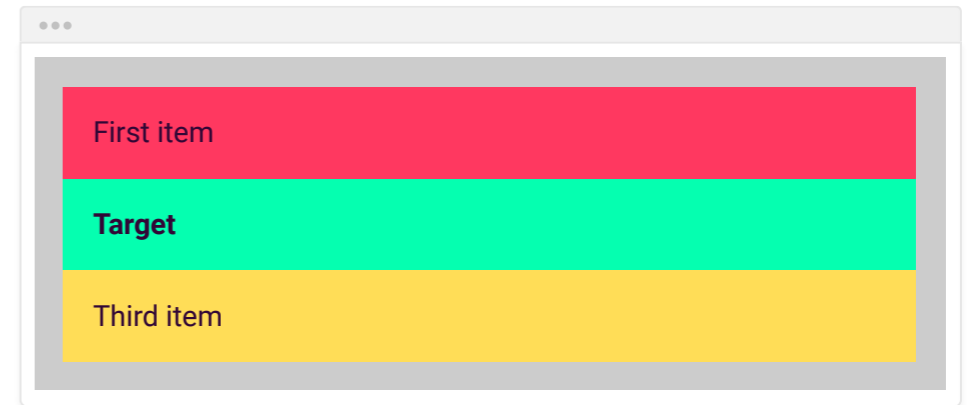
The value is **relative** to the font size:

- em**: relative to the *element's* current font size
- rem**: relative to `<html>` the **root element's** font size

```
margin-right: 30%;
```



In collection: **Box-Model** Permalink Share MDN





Search for a property

- [border-bottom-width](#)
- [border-left-width](#)
- [border-right-width](#)
- [border-top-width](#)
- [border-width](#)
- [box-sizing](#)
- [height](#)
- [line-height](#)
- [margin-bottom](#)
- [margin-left](#)
- [margin-right](#)
- [margin-top](#)
- [margin](#)
- [max-height](#)
- [max-width](#)
- [min-height](#)
- [min-width](#)



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

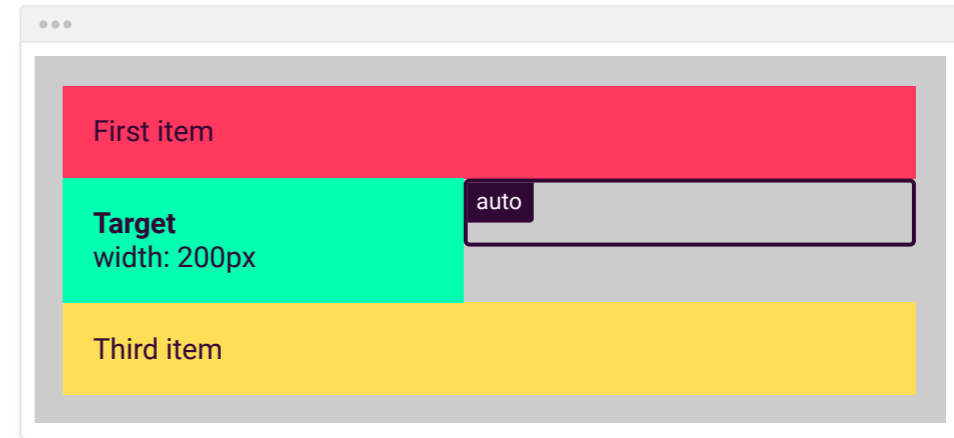
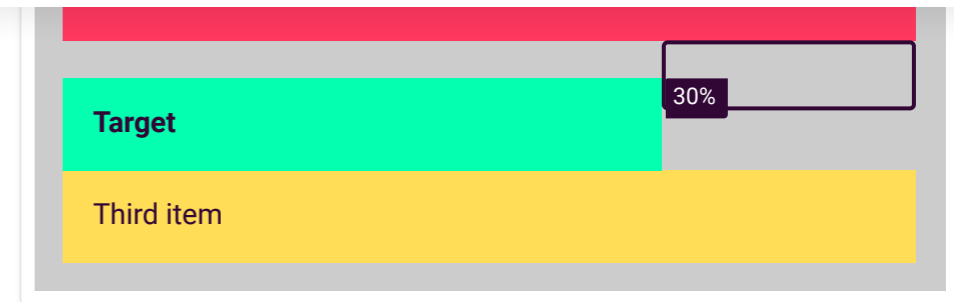
ads via Carbon

The percentage is based on the *width* of the *container*.

```
margin-right: auto;
```

The **auto** keyword will give the right side a share of the **remaining space**.

When combined with `margin-left: auto`, it will **center** the element, if a fixed width is defined.



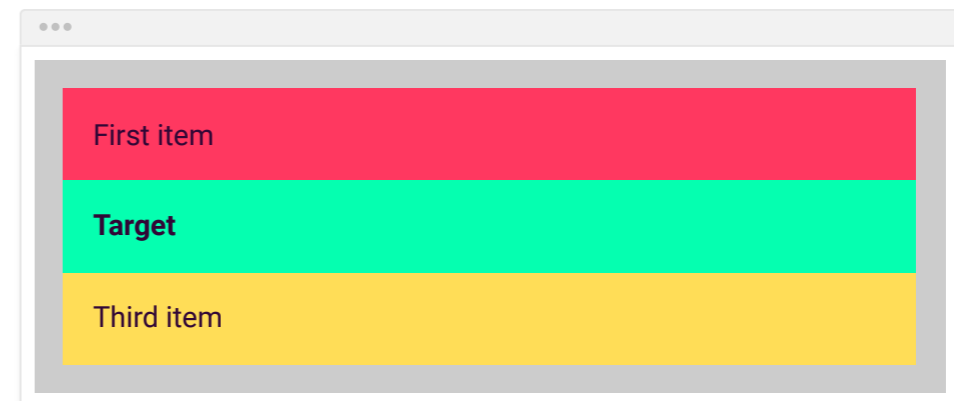
## # margin-top

Defines the space **outside** the element, on the **top** side.

```
margin-top: 0;
```

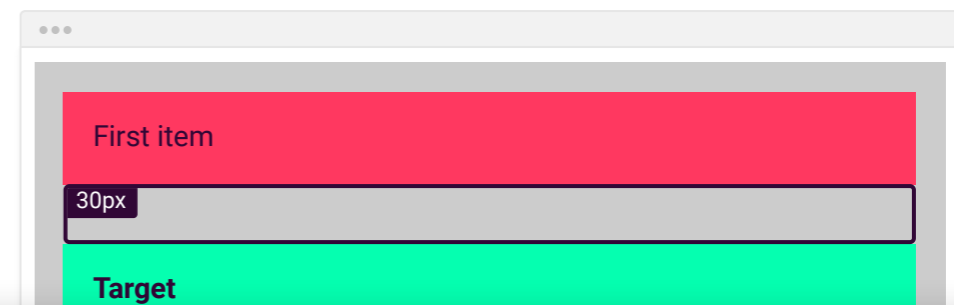
Removes any margin at the top.

default



```
margin-top: 30px;
```

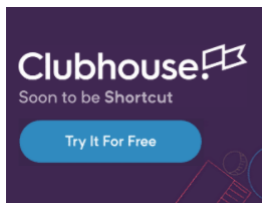
You can use **pixel** values.



In collection: [Box-Model](#) [Permalink](#) [Share](#) [MDN](#)

Search for a property

- border-bottom-width
- border-left-width
- border-right-width
- border-top-width
- border-width
- box-sizing
- height
- line-height
- margin-bottom
- margin-left
- margin-right
- margin-top
- margin
- max-height
- max-width
- min-height
- min-width



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

```
margin-top: 2em;
```

You can use **(r)em** values.

The value is **relative** to the font size:

- **em**: relative to the *element's* current font size
- **rem**: relative to `<html>` the *root element's* font size

```
margin-top: 10%;
```

You can use **percentage** values.

The percentage is based on the *width of the container*.

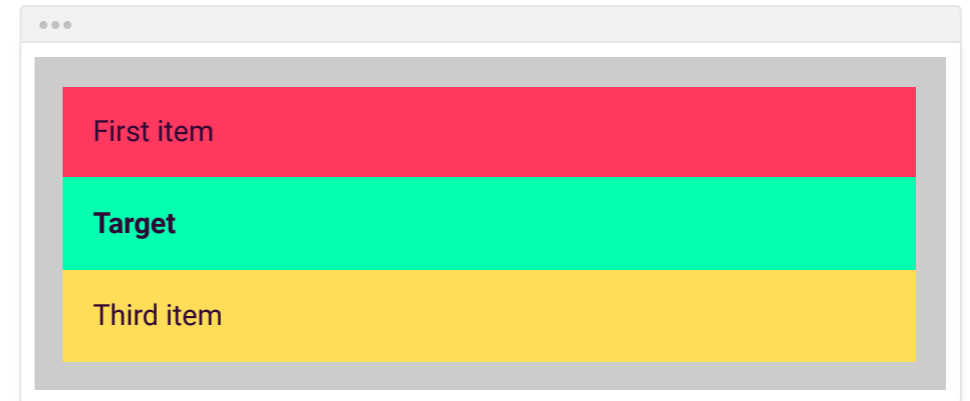
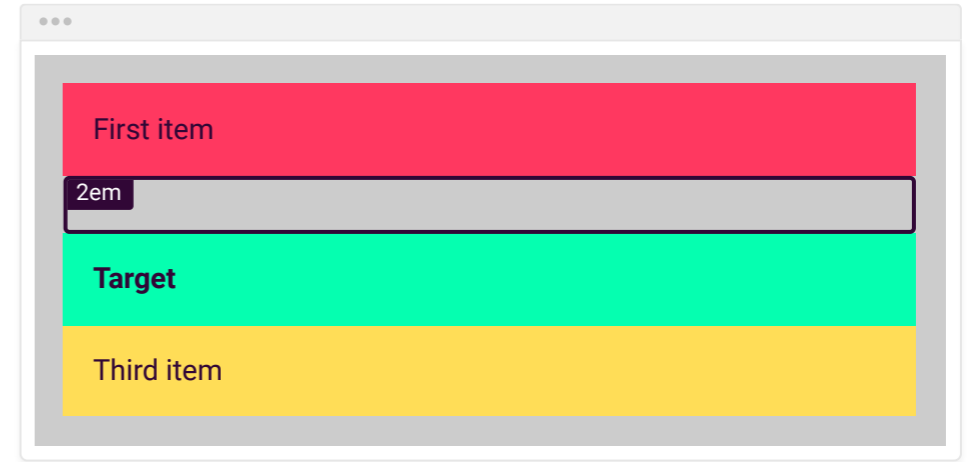
## # margin

Shorthand property for `margin-top` `margin-right` `margin-bottom` and `margin-left`.

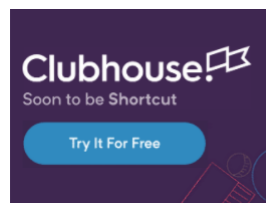
```
margin: 0;
```

Removes all margins.

default



In collection: **Box-Model** [Permalink](#) [Share](#) [MDN](#)



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

When using **1** value, the margin is set for **all 4 sides**.

```
margin: 30px 60px;
```

When using **2** values:

- the **first** value is for **top/bottom**
- the **second** value is for **right/left**

To remember the **order** think about the values you **haven't defined**.

If you enter **2** values (top/right), you *omit* setting bottom and left. Because bottom is the vertical counterpart of top, it will use top's value. And because left is the horizontal counterpart of right, it will use right's value.

```
margin: 30px 60px 45px;
```

When using **3** values:

- the **first** value is for **top**
- the **second** value is for **right/left**
- the **third** value is for **bottom**

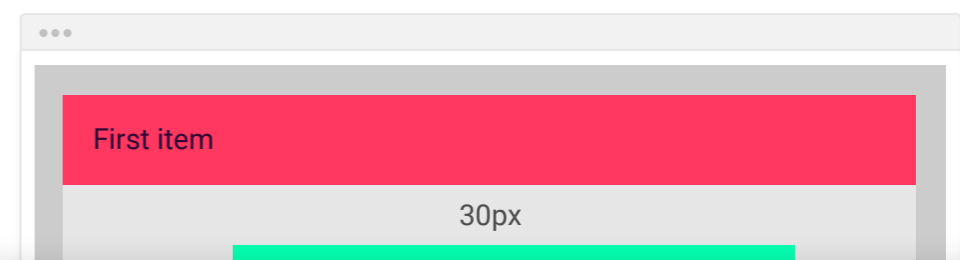
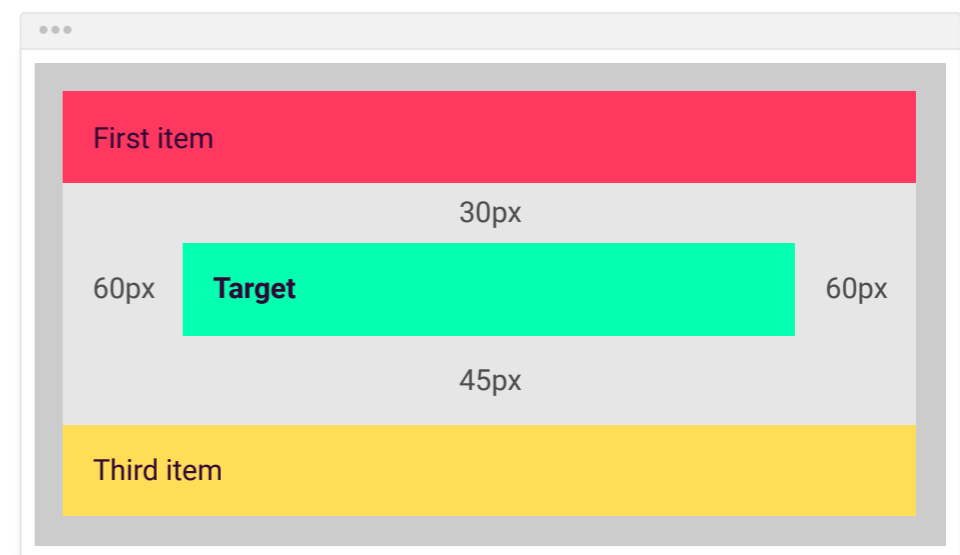
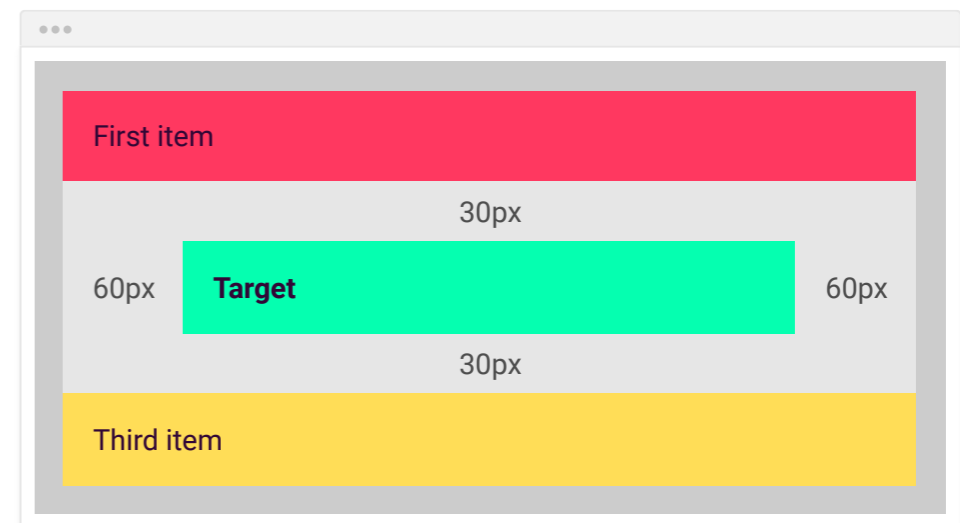
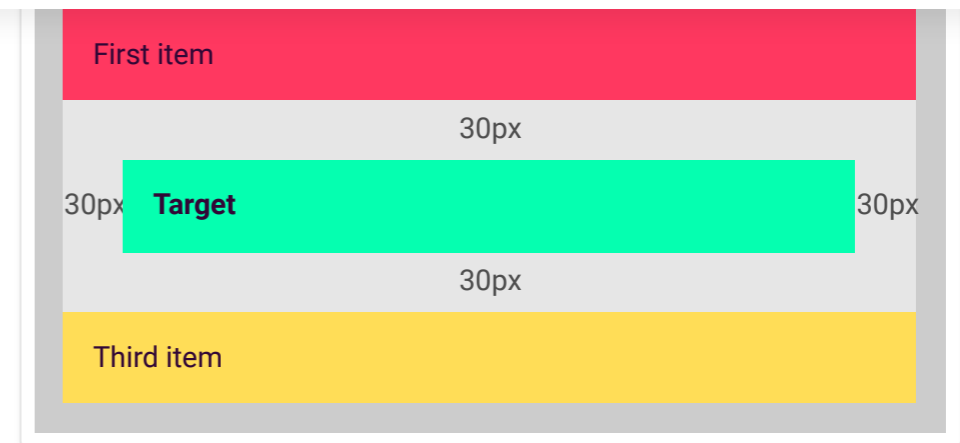
To remember the **order** think about the values you **haven't defined**.

If you enter **3** values (top/right/bottom), you omit setting left. As right's counterpart, it will use its value.

```
margin: 30px 60px 45px 85px;
```

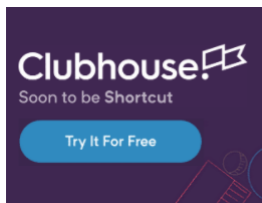
When using **4** values:

- the **first** value is for **top**
- the **second** value is for **right**



Search for a property

- border-bottom-width
- border-left-width
- border-right-width
- border-top-width
- border-width
- box-sizing
- height
- line-height
- margin-bottom
- margin-left
- margin-right
- margin-top
- margin
- max-height
- max-width
- min-height
- min-width



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

To remember the **order**, start at the **top** and go **clockwise**.



## # max-height

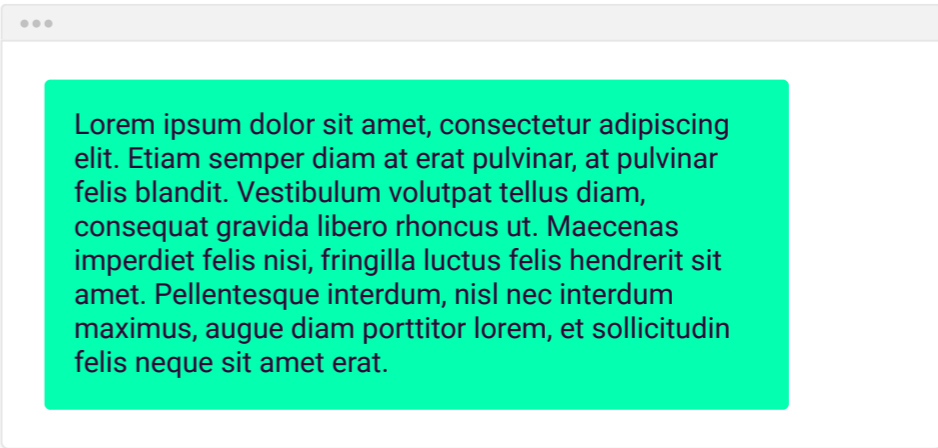
In collection: **Box-Model** Permalink Share MDN

Defines the maximum height the element can be.

```
max-height: none;
```

default

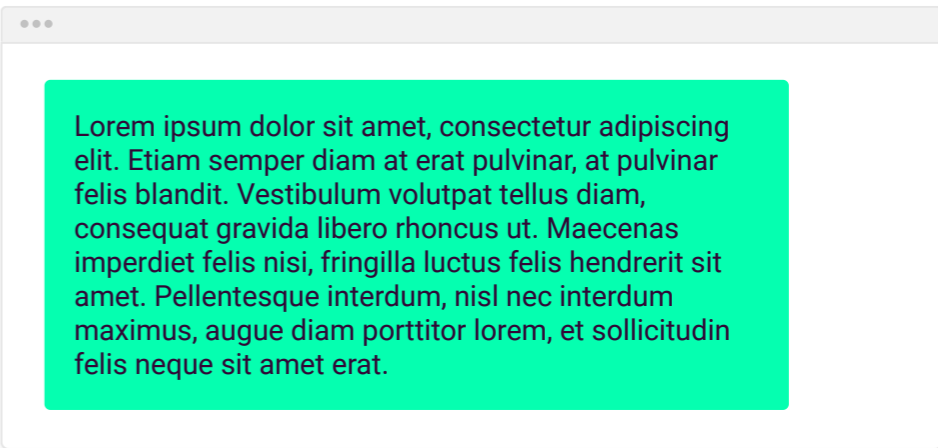
The element has **no limit** in terms of height.



```
max-height: 2000px;
```

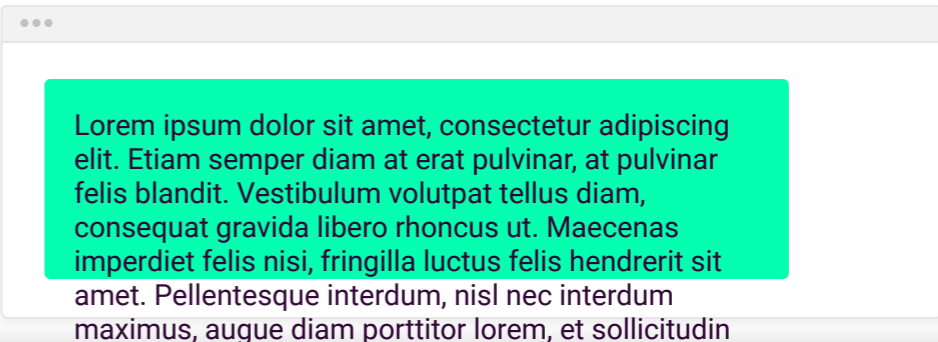
You can use **numeric values** like **pixels**, **(r)em**, **percentages**...

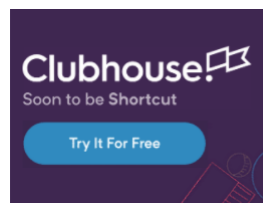
If the *maximum* height is **larger** than the element's *actual* height, the max height has **no effect**.



```
max-height: 100px;
```

If the content does not fit within the maximum height, it will **overflow**. How the container will handle this overflowing content is defined by the `overflow` property.





Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

## # max-width

Defines the maximum width the element can be.

```
max-width: none;
```

The element has **no limit** in terms of width.

default

```
max-width: 2000px;
```

You can use **numeric values** like **pixels, (r)em, percentages...**

If the *maximum* width is **larger** than the element's *actual* width, the max width has **no effect**.

```
max-width: 150px;
```

If the content does not fit within the maximum width, it will automatically change the **height** of the element to accomodate for the wrapping of the content.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut. Maecenas imperdiet felis nisi, fringilla luctus felis hendrerit sit amet. Pellentesque interdum, nisl nec interdum maximus, augue diam porttitor lorem, et sollicitudin felis neque sit amet erat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut. Maecenas imperdiet felis nisi, fringilla luctus felis hendrerit sit amet. Pellentesque interdum, nisl nec interdum maximus, augue diam porttitor lorem, et sollicitudin felis neque sit amet erat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut. Maecenas imperdiet felis nisi, fringilla luctus felis hendrerit sit amet. Pellentesque

Search for a property

- border-bottom-width
- border-left-width
- border-right-width
- border-top-width
- border-width
- box-sizing
- height
- line-height
- margin-bottom
- margin-left
- margin-right
- margin-top
- margin
- max-height
- max-width
- min-height
- min-width



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

## # min-height

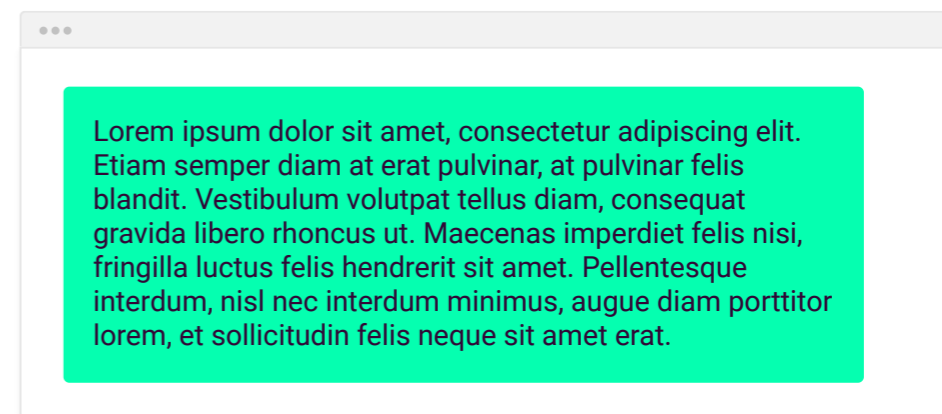
In collection: **Box-Model** Permalink Share MDN

Defines the minimum height of the element.

```
min-height: 0;
```

default

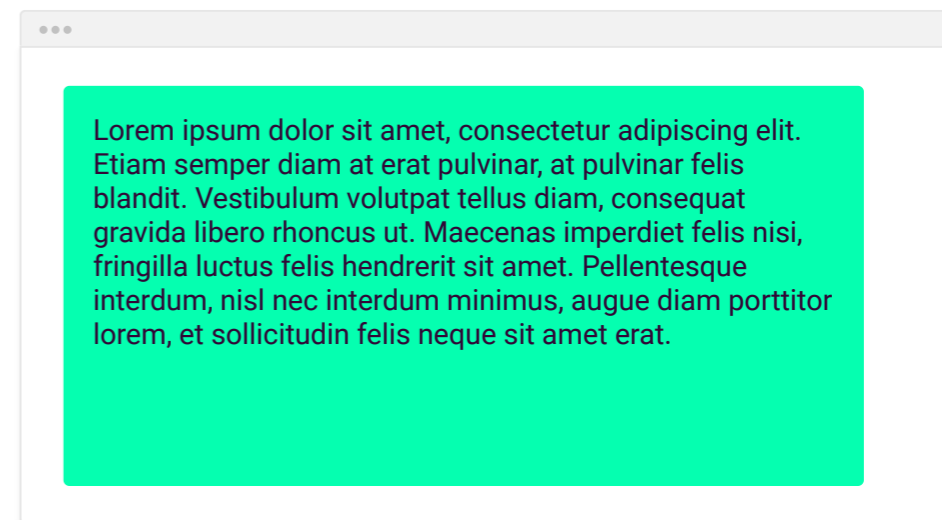
The element has **no minimum** height.



```
min-height: 200px;
```

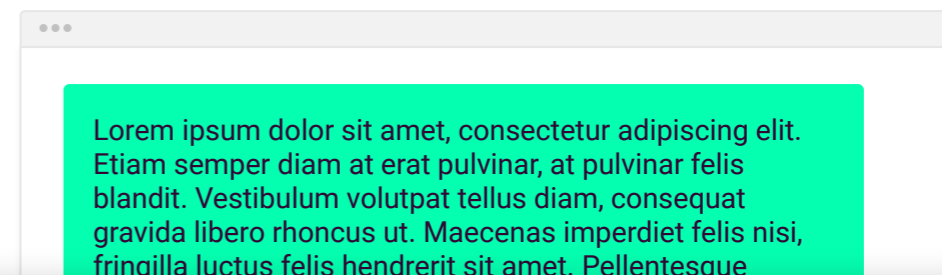
You can use **numeric values** like **pixels, (r)em, percentages...**

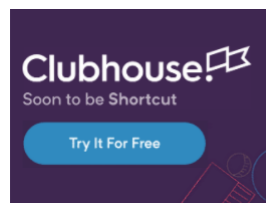
If the *minimum* height is **larger** than the element's *actual* height, the min height will be applied.



```
min-height: 5px;
```

If the *minimum* height is **smaller** than the element's *actual* height, the min height has **no effect**.





Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

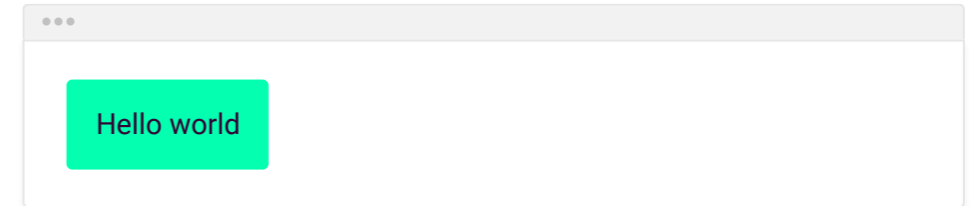
## # min-width

Defines the minimum width of the element.

```
min-width: 0;
```

default

The element has **no minimum** width.



```
min-width: 300px;
```

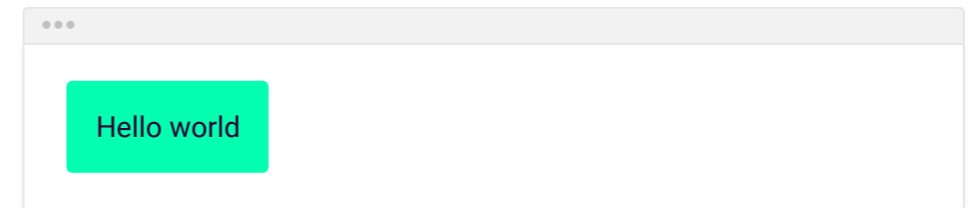
You can use **numeric values** like **pixels, (r)em, percentages...**

If the *minimum* width is **larger** than the element's *actual* width, the min width will be applied.



```
min-width: 5px;
```

If the *minimum* width is **smaller** than the element's *actual* width, the min width has **no effect**.



In collection: [Box-Model](#) [Permalink](#) [Share](#) [MDN](#)

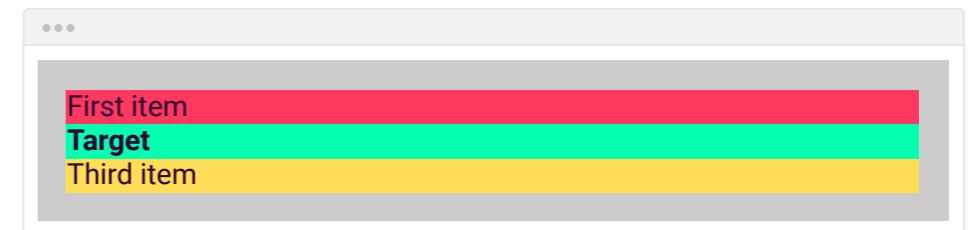
## # padding-bottom

Defines the space **inside** the element, on the **bottom** side.

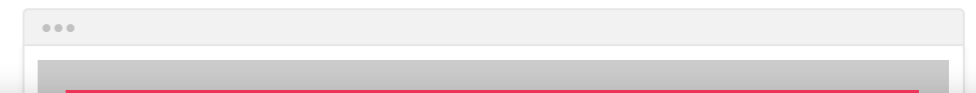
```
padding-bottom: 0;
```

default

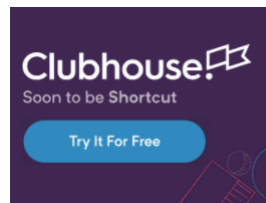
Removes any padding on the bottom.



```
padding-bottom: 50px;
```



In collection: [Box-Model](#) [Permalink](#) [Share](#) [MDN](#)



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

```
padding-bottom: 7em;
```

You can use **(r)em** values.

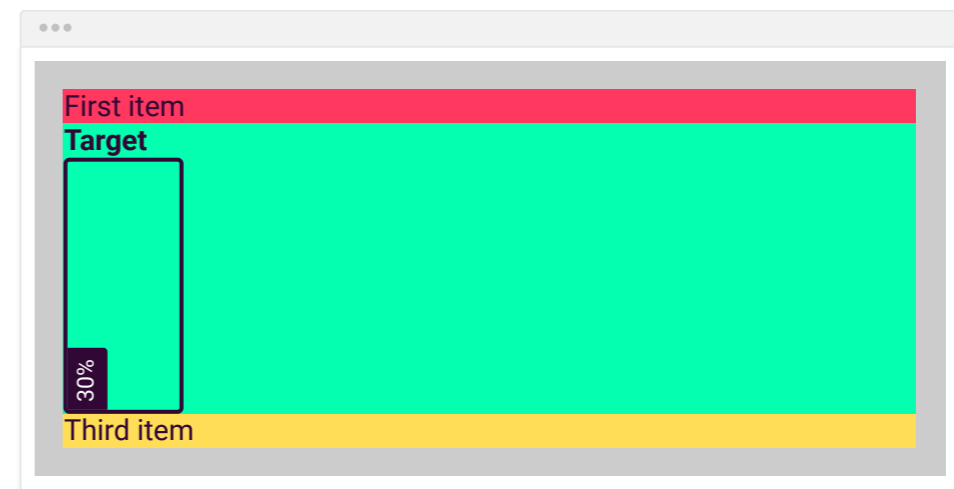
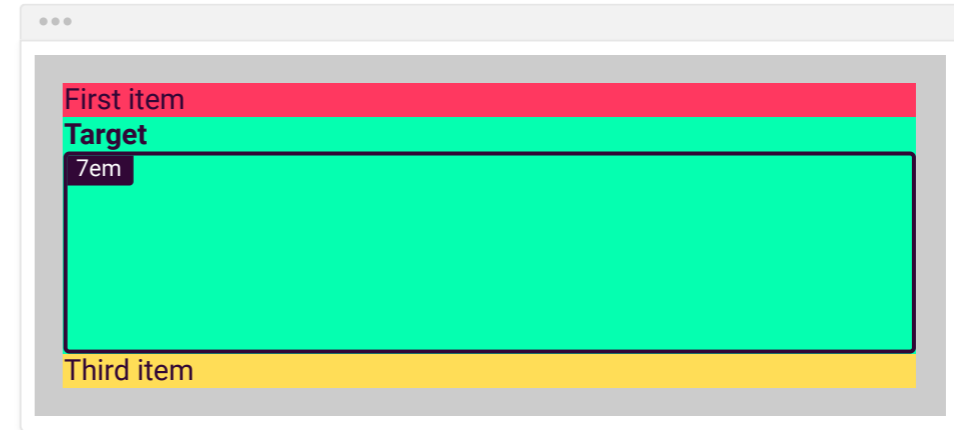
The value is **relative** to the font size:

- **em**: relative to the *element's* current font size
- **rem**: relative to `<html>` the **root element's** font size

```
padding-bottom: 30%;
```

You can use **percentage** values.

The percentage is based on the *width* of the *element*.



## # padding-left

Defines the space **inside** the element, on the **left** side.

```
padding-left: 0;
```

Removes any padding on the left.

default

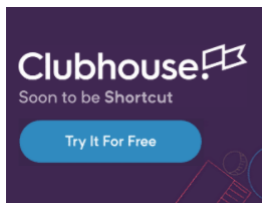


In collection: **Box-Model** Permalink Share MDN



Search for a property

- border-bottom-width
- border-left-width
- border-right-width
- border-top-width
- border-width
- box-sizing
- height
- line-height
- margin-bottom
- margin-left
- margin-right
- margin-top
- margin
- max-height
- max-width
- min-height
- min-width

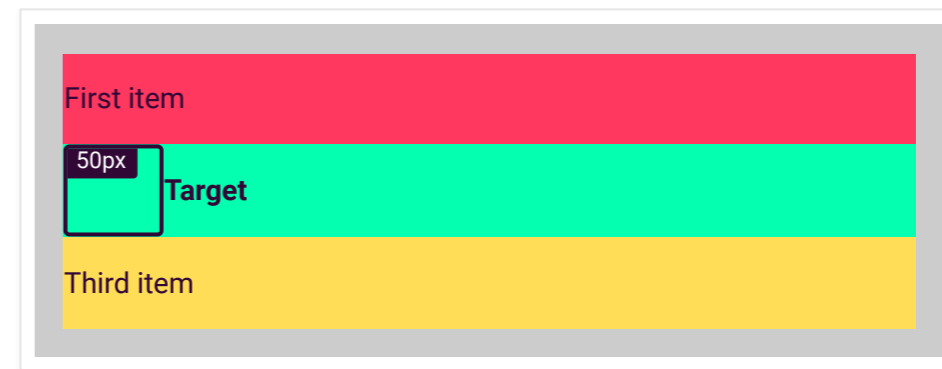


Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

```
padding-left: 50px;
```

You can use **pixel** values.

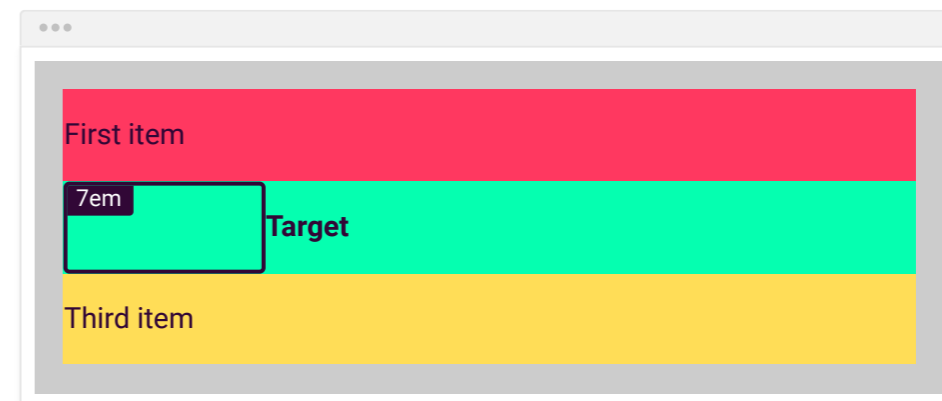


```
padding-left: 7em;
```

You can use **(r)em** values.

The value is **relative** to the font size:

- **em**: relative to the *element's* current font size
- **rem**: relative to `<html>` the **root element's** font size



```
padding-left: 30%;
```

You can use **percentage** values.

The percentage is based on the *width* of the *element*.



## # padding-right

Defines the space **inside** the element, on the **right** side.

```
padding-right: 0;
```

Removes any padding on the right.

default



In collection: [Box-Model](#) [Permalink](#) [Share](#) [MDN](#)

Search for a property

- border-bottom-width
- border-left-width
- border-right-width
- border-top-width
- border-width
- box-sizing
- height
- line-height
- margin-bottom
- margin-left
- margin-right
- margin-top
- margin
- max-height
- max-width
- min-height
- min-width

```
padding-right: 50px;
```

You can use **pixel** values.

```
padding-right: 7em;
```

You can use **(r)em** values.

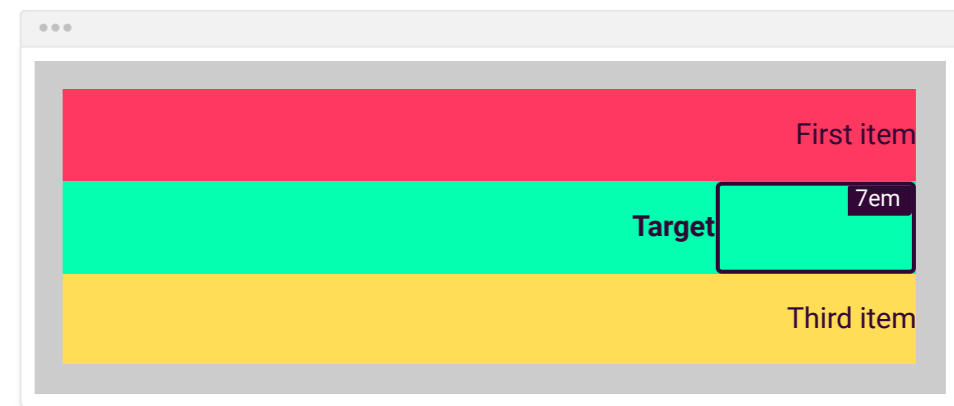
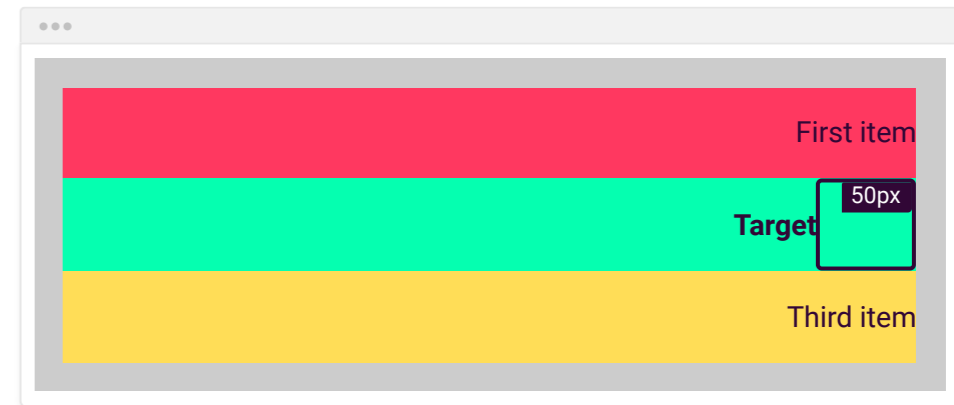
The value is **relative** to the font size:

- **em**: relative to the *element's* current font size
- **rem**: relative to `<html>` the **root element's** font size

```
padding-right: 30%;
```

You can use **percentage** values.

The percentage is based on the *width* of the *element*.

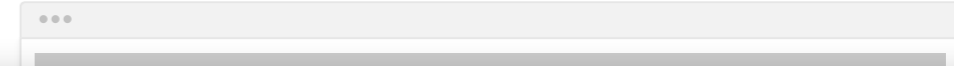


## # padding-top

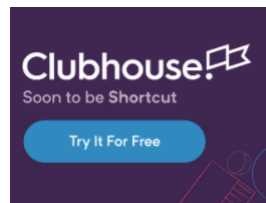
Defines the space **inside** the element, on the **top** side.

```
padding-top: 0;
```

default



In collection: [Box-Model](#) [Permalink](#) [Share](#) [MDN](#)



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

A free visual guide to CSS Created by [@jgthms](#)

Share

Star 4,261

Search for a property

- [border-bottom-width](#)
- [border-left-width](#)
- [border-right-width](#)
- [border-top-width](#)
- [border-width](#)
- [box-sizing](#)
- [height](#)
- [line-height](#)
- [margin-bottom](#)
- [margin-left](#)
- [margin-right](#)
- [margin-top](#)
- [margin](#)
- [max-height](#)
- [max-width](#)
- [min-height](#)
- [min-width](#)



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

```
padding-top: 50px;
```

You can use **pixel** values.

```
padding-top: 7em;
```

You can use **(r)em** values.

The value is **relative** to the font size:

- **em**: relative to the *element's* current font size
- **rem**: relative to `<html>` the **root element's** font size

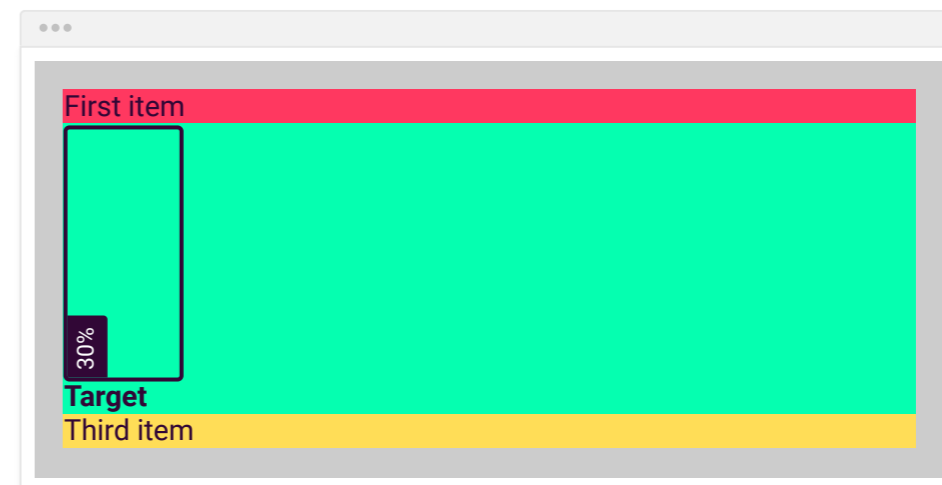
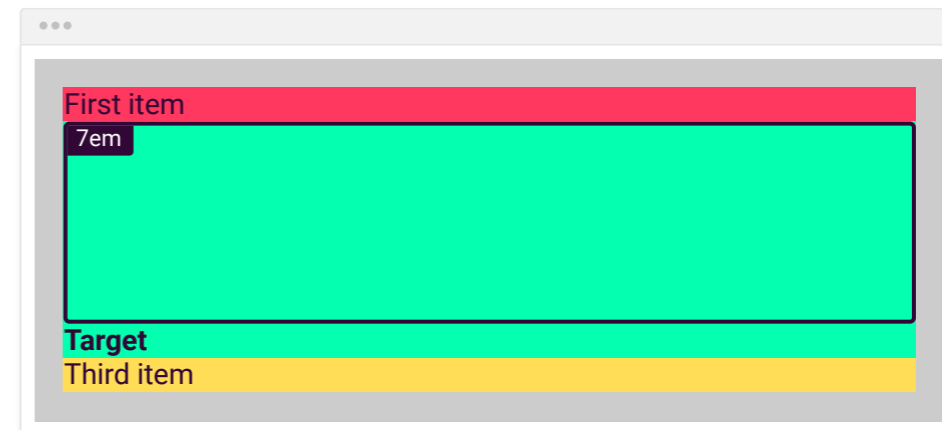
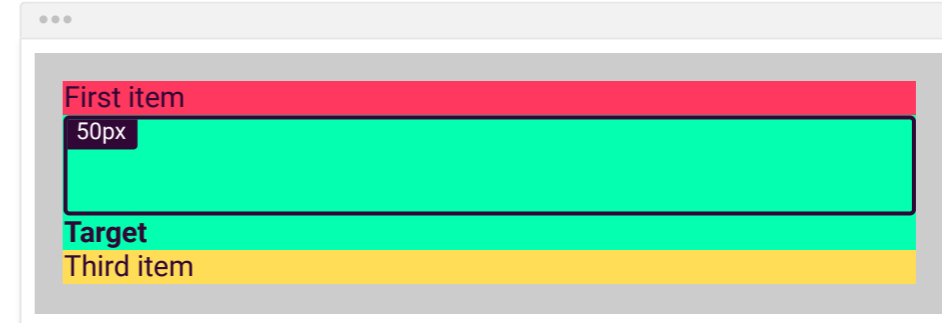
```
padding-top: 30%;
```

You can use **percentage** values.

The percentage is based on the *width* of the *element*.

```
padding: 0;
```

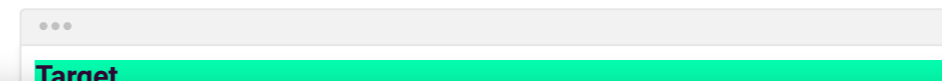
default



## # padding

Shorthand property for `padding-top` `padding-right` `padding-bottom` and `padding-left`.

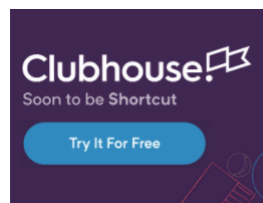
In collection: [Box-Model](#) [Permalink](#) [Share](#) [MDN](#)



A free visual guide to CSS Created by @jgthms

Share

Star 4,261

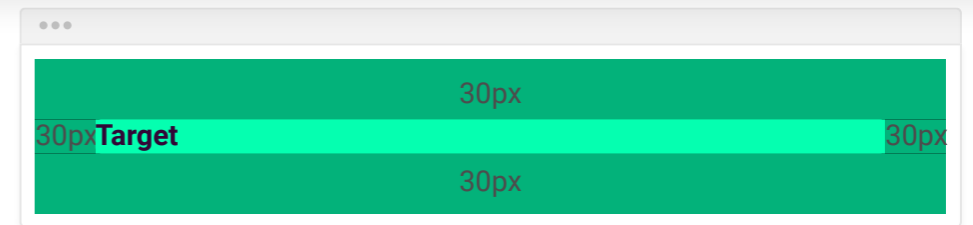


Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

`padding: 30px;`

When using **1** value, the padding is set for **all 4 sides**.



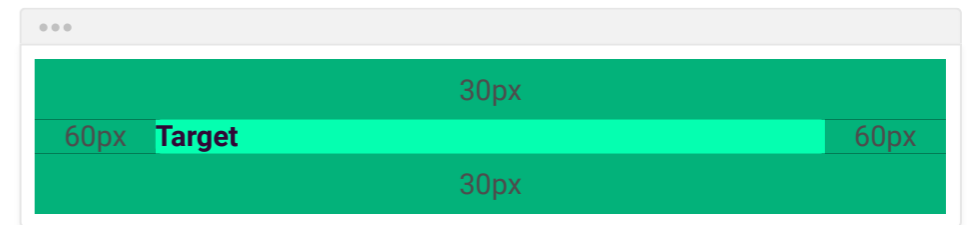
`padding: 30px 60px;`

When using **2** values:

- the **first** value is for **top/bottom**
- the **second** value is for **right/left**

To remember the **order** think about the values you **haven't defined**.

If you enter **2** values (top/right), you *omit* setting bottom and left. Because bottom is the vertical counterpart of top, it will use top's value. And because left is the horizontal counterpart of right, it will use right's value.



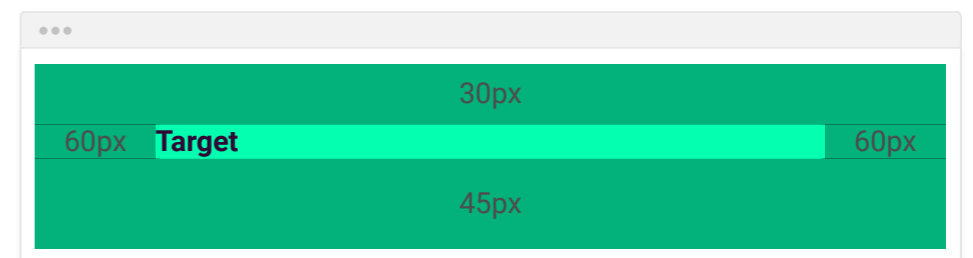
`padding: 30px 60px 45px;`

When using **3** values:

- the **first** value is for **top**
- the **second** value is for **right/left**
- the **third** value is for **bottom**

To remember the **order** think about the values you **haven't defined**.

If you enter **3** values (top/right/bottom), you omit setting left. As right's counterpart, it will use its value.

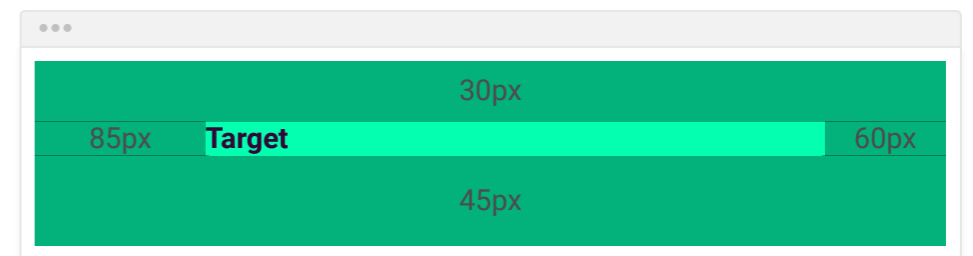


`padding: 30px 60px 45px 85px;`

When using **4** values:

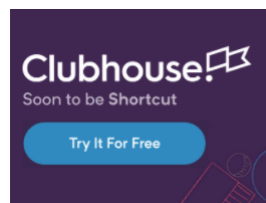
- the **first** value is for **top**
- the **second** value is for **right**
- the **third** value is for **bottom**
- the **fourth** value is for **left**

To remember the **order**, start at the **top** and go **clockwise**.



Search for a property

border-bottom-width  
border-left-width  
border-right-width  
border-top-width  
border-width  
box-sizing  
height  
line-height  
margin-bottom  
margin-left  
margin-right  
margin-top  
margin  
max-height  
max-width  
min-height  
min-width



Clubhouse provides speedy task management, reporting, and collaboration for developers. Try it free today.

ads via Carbon

Defines the width of the element.

```
width: auto;
```

default

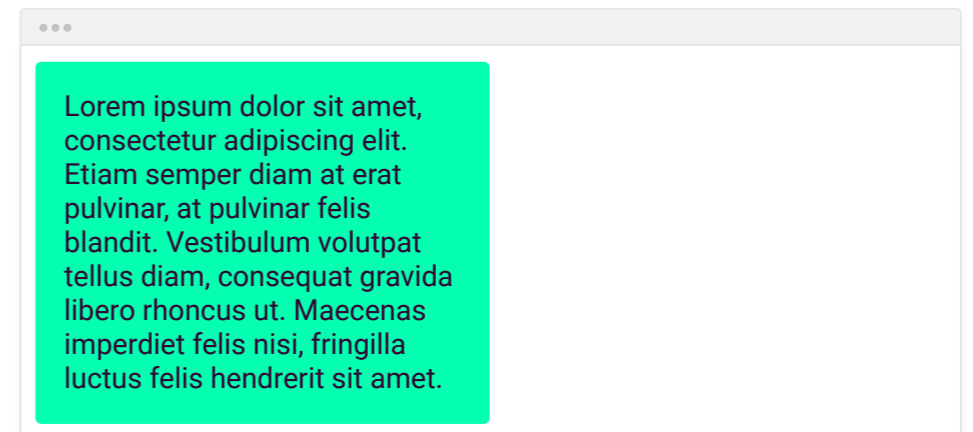
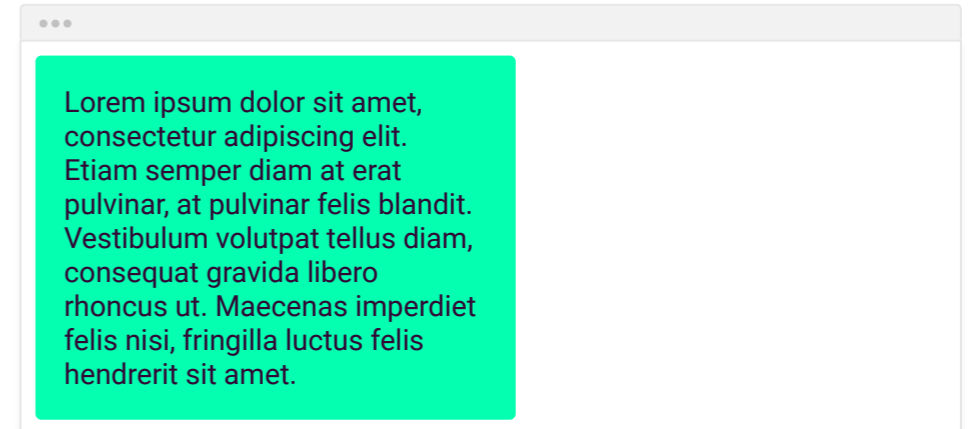
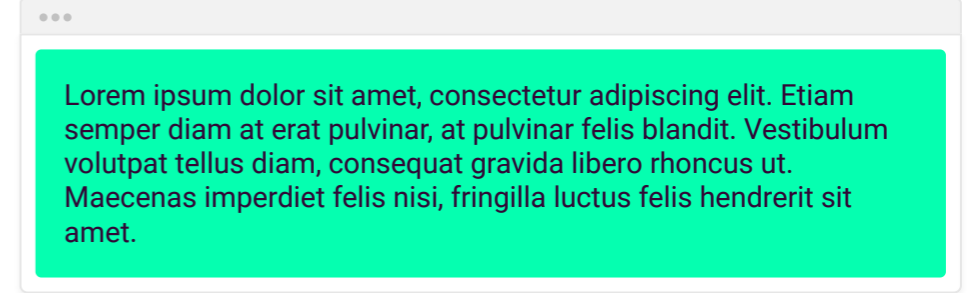
The element will **automatically** adjust its width to allow its content to be displayed correctly.

```
width: 240px;
```

You can use **numeric values** like **pixels, (r)em, percentages...**

```
width: 50%;
```

If you use **percentages**, the value is relative to the container's width.



# 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 ensures that the page structure and organization remain unchanged.

```
.invisible-elements {  
  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;  
}
```