Internet and Web Principal

Week 7

CSS 2

Content

1. Box Element
2. Floating and Positioning
3. Flexbox

Box Element

Every element in a page, both inline and block-level, creates a box-shaped element.

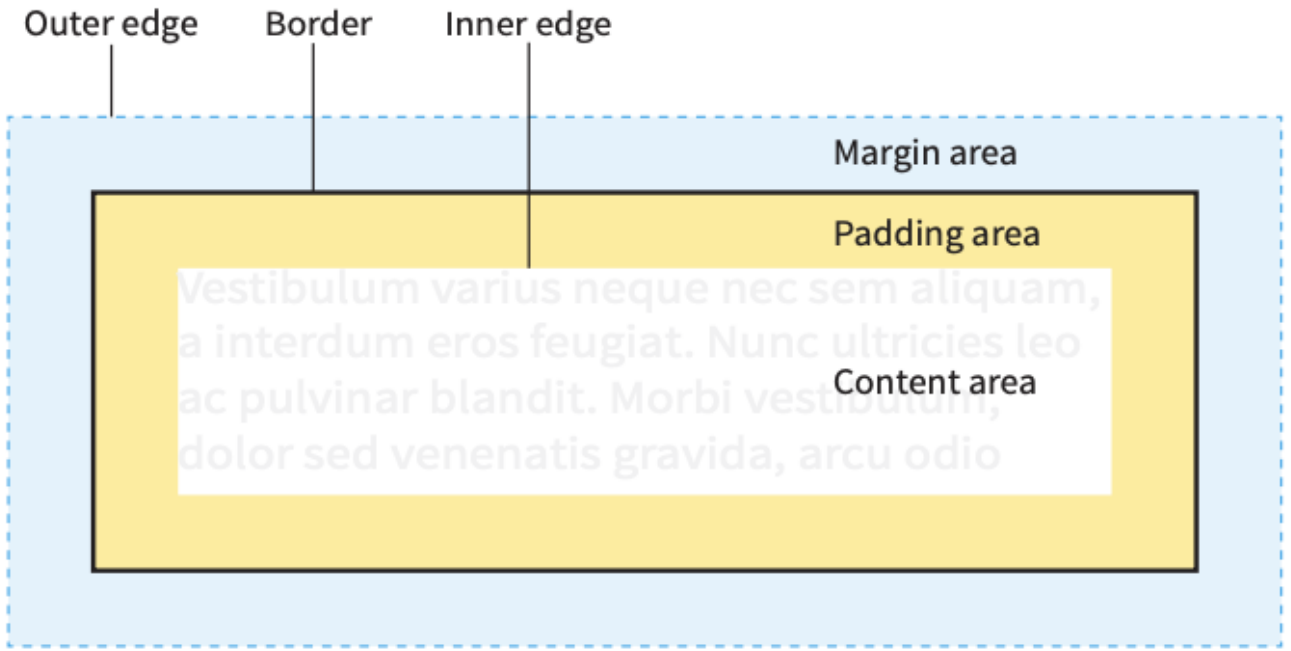


FIGURE 14-1. The parts of an element box according to the CSS box model.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 355

Box Element

Content Area

The actual content is what makes up the element box. The content region is denoted by a white box in FIGURE 14-1.

Inner Edges

The inner margins of the element box are the boundaries of the content area. In real pages, the content area's edge is invisible, even if FIGURE 14-1's inner boundaries are highlighted by a color change.

Box Element

Padding

The space between the content area and an optional border is known as the padding. The padding area is denoted in the diagram with a yellow-orange tint. Padding is not required.

Border

The element and its padding are surrounded by a line (or stylised line) known as the border. Borders are also not required.

Box Element

Margin

The margin is extra space that can be inserted outside of the border. Although the margin is shown in the diagram as having light-blue coloring, in actuality, margins are always transparent, allowing the parent element's background to be visible.

Outside edge

The outer margins of the element box are the outer edges of the margin area. This is the entire space occupied by the element on the page, which includes the width of the content area as well as all padding, borders, and margins. On the diagram, the outside margin edge is represented by a dotted line; however, in actual web pages, the margin border is not visible.

Box Element

SPECIFYING BOX DIMENSIONS

width

Values: *length* | *percentage* | auto

Default: *Auto*

Applies to: block-level elements and replaced inline elements (such as images)

Inherits: no

height

Values: *length* | *percentage* | auto

Default: Auto

Applies to: block-level elements and replaced inline elements (such as images)

Inherits: no

box-sizing

Values: content-box | border-box

Default: content-box

Applies to: all elements

Inherits: no

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 357

Box Element

By default, a block element's width and height are determined automatically by the browser (thus the default auto value). The box will be as tall as necessary to fit the content and as wide as the browser window or any containing block element. However, you may specify the width and height of the content area of an element using the width and height properties.

Box Element

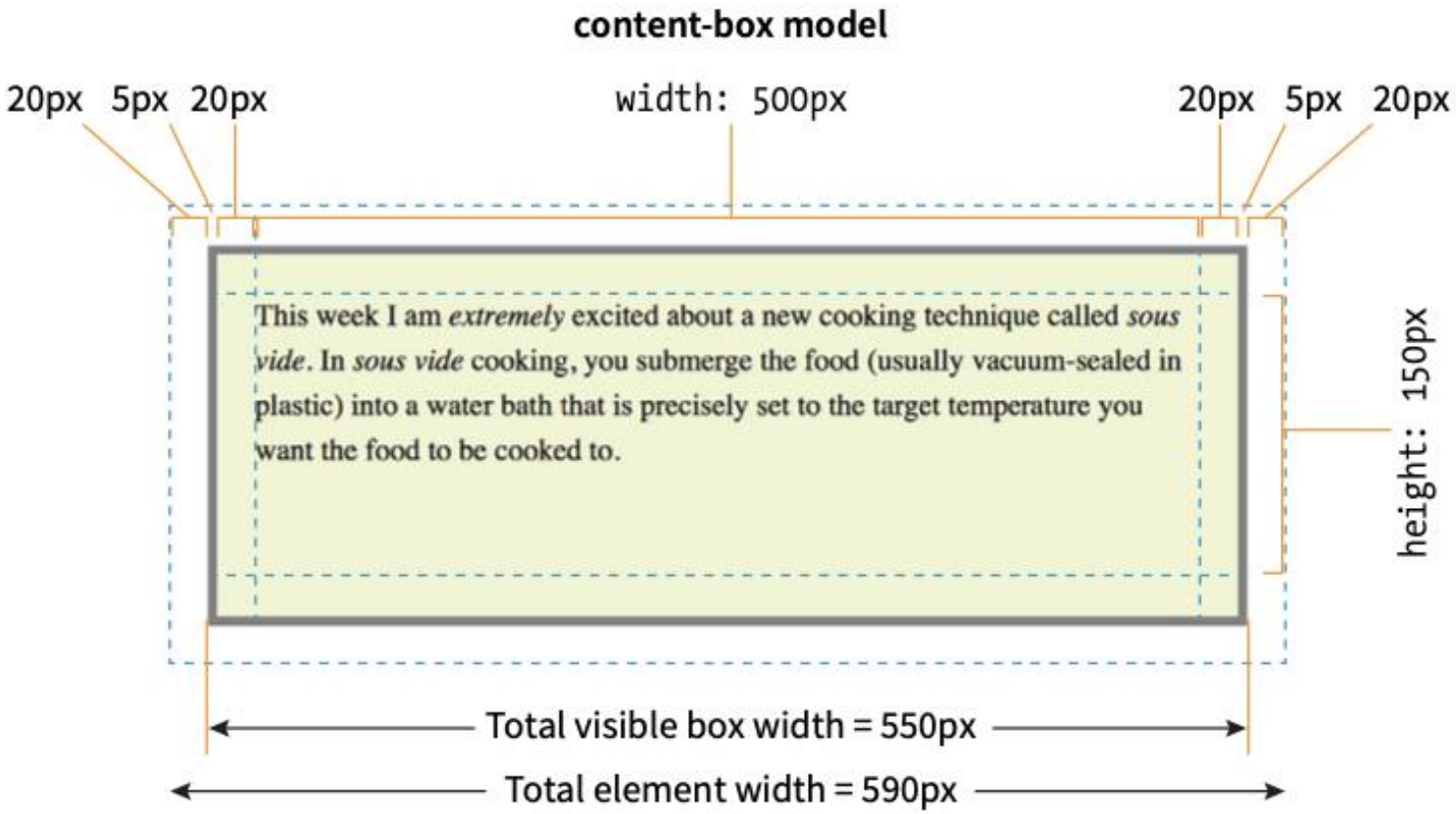
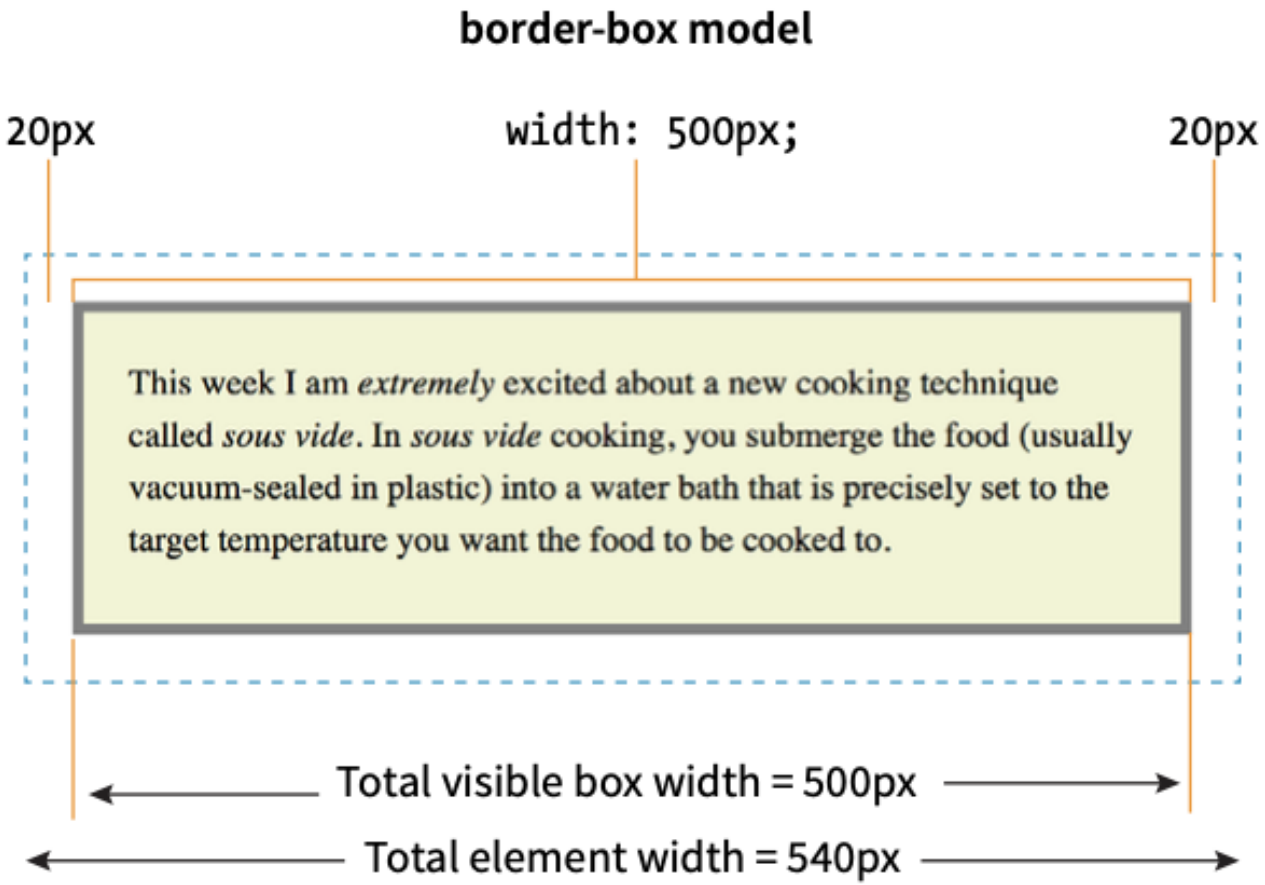


FIGURE 14-2. Specifying the **width** and **height** with the **content-box** model.

```
p{
background: #f2f5d5; width: 500px;
height: 150px;
padding: 20px;
border: 5px solid gray; margin: 20px;
}
```

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 358

Box Element




```
p{
 ...
box-sizing: border-box;
width: 500px;
 height: 150px;
}
```

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 359

Box Element

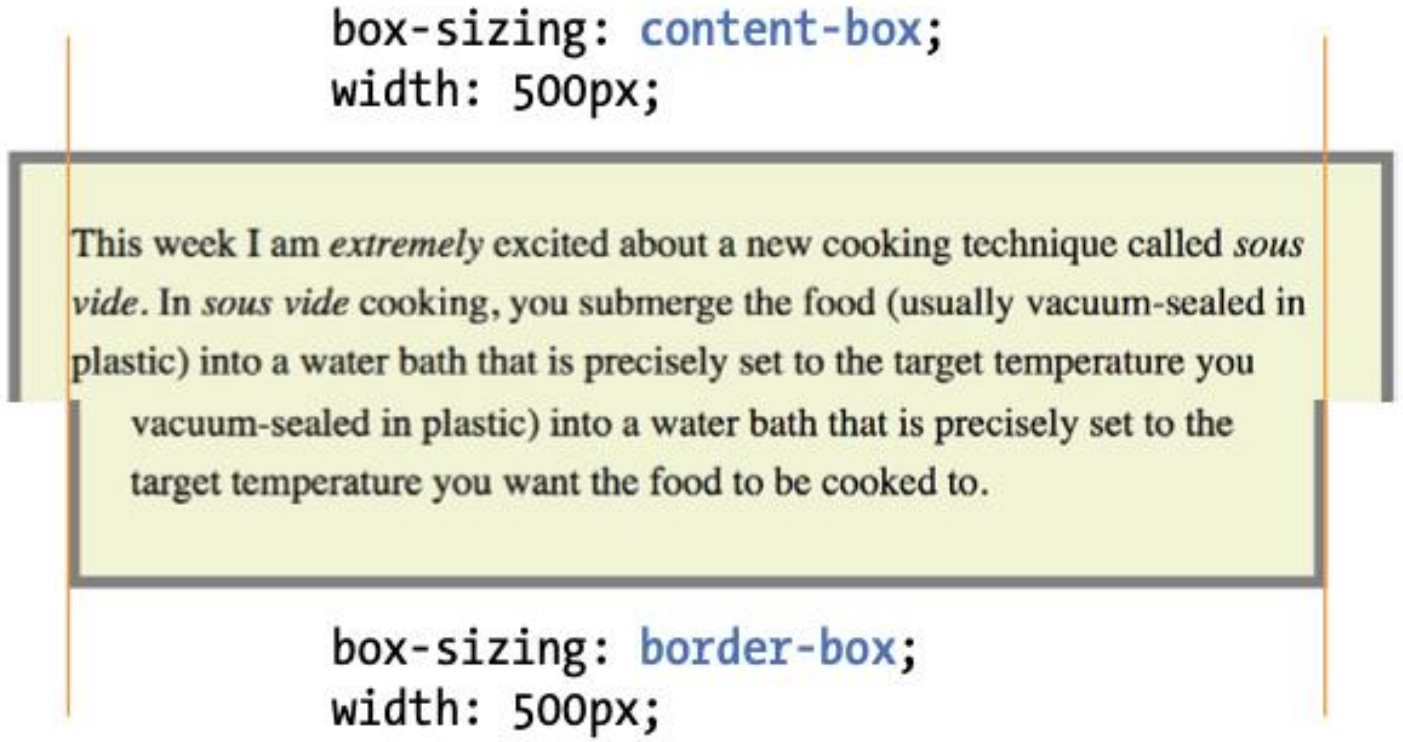


FIGURE 14-3. Sizing an element with the **border-box** method. The bottom diagram compares the resulting boxes from each sizing method.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 359

Box Element

Handling Overflow

When an element is sized too small for its contents, you can specify what to do with the content that doesn't fit by using the **overflow** property.

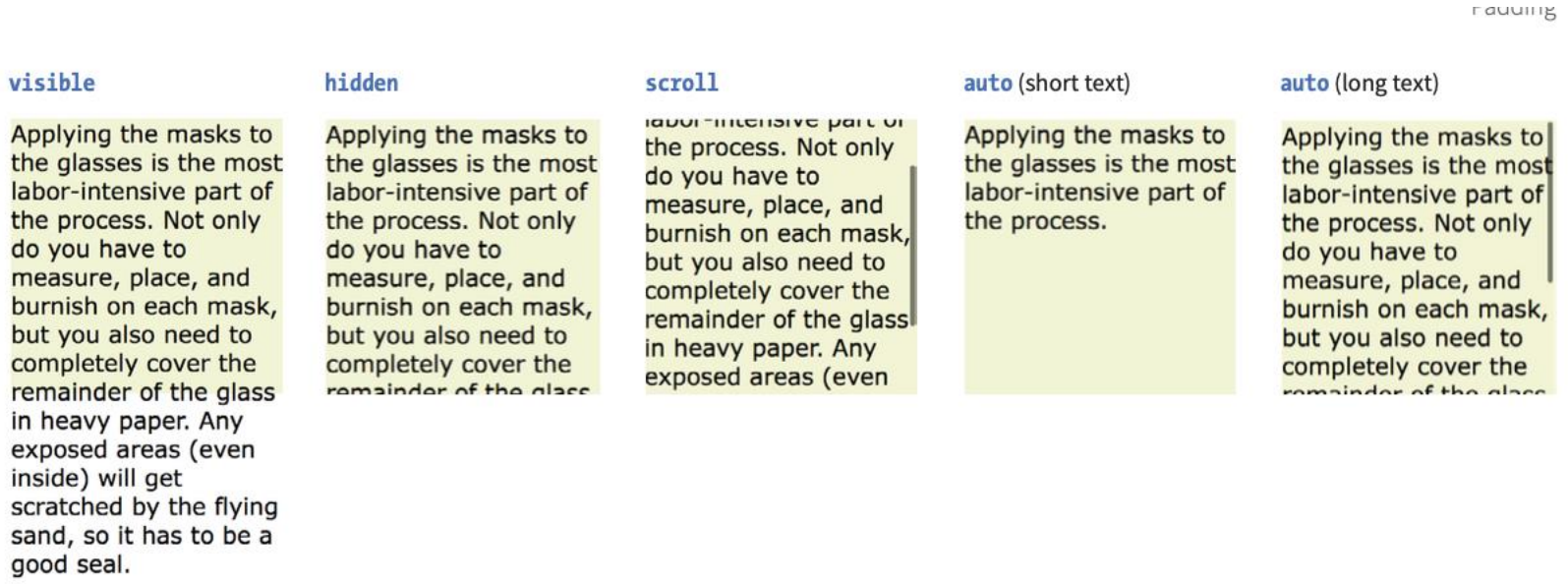


FIGURE 14-4. Options for handling content overflow. The scroll and auto options have narrow gray scrollbars to the right of the text (as rendered on macOS).

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 361

Box Element

overflow

Values: visible | hidden | scroll | auto

Default: visible

Applies to: block-level elements and replaced inline elements (such as images)

Inherits: no

- **visible**
  The default value is **visible**, which allows the content to hang out over
- the element box so that it all can be seen.
- **hidden**
  When **overflow** is set to **hidden**, the content that does not fit gets clipped
- off and does not appear beyond the edges of the element's content area.
- **scroll**
- When **scroll** is specified, scrollbars are added to the element box to let users scroll through the content. Be aware that they may become visible only when you click the element to scroll it.
- **auto**
- The **auto** value allows the browser to decide how to handle overflow. In most cases, scrollbars are added only when the content doesn't fit and they are needed.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 361

Box Element

Padding is the area that exists between the content area and the border (or, in the absence of a border, the location where the boundary would be). When utilizing a border or a background color, I find it useful to give padding to the elements. It offers the text some breathing room and keeps the background's edge or border from touching the text directly.

Any element's individual sides can have padding added (block-level or inline). Additionally, a shorthand padding property exists that enables you to add padding to both sides simultaneously.

Box Element

Padding, padding-top, padding-right, padding-bottom, padding-left

- Values: *length* | *percentage*
- Default: 0
- Applies to: all elements
- Inherits: no

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 361

Box Element

```
blockquote {
  padding-top: 2em;
  padding-right: 4em;
  padding-bottom: 2em;
  padding-left: 4em;
  background-color: #D098D4; /* light green */
}
```

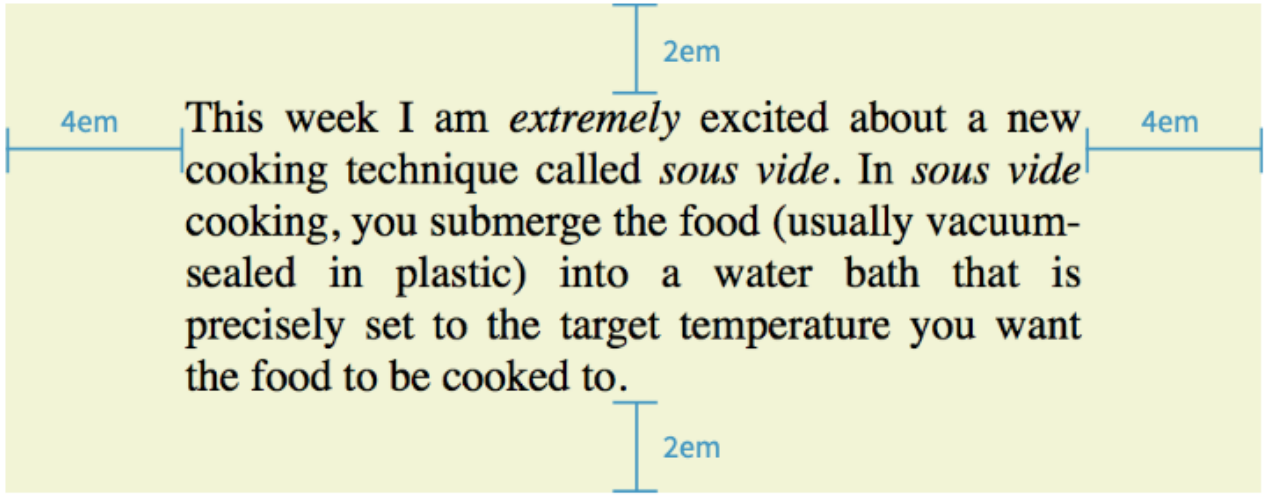


FIGURE 14-5. Adding padding around the content of an element.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 362

Box Element

Border

The style is the most crucial border property since, in accordance with the CSS definition, a border does not exist if a border style is not supplied (the default is none). To put it another way, if you don't define the border style, the other border attributes won't be taken into account.

Box Element

Border, border-top, border-right, border-bottom, border-left

Values: none | solid | hidden | dotted | dashed | double | groove | ridge | inset | outset

Default: none

Applies to: all elements

Inherits: no

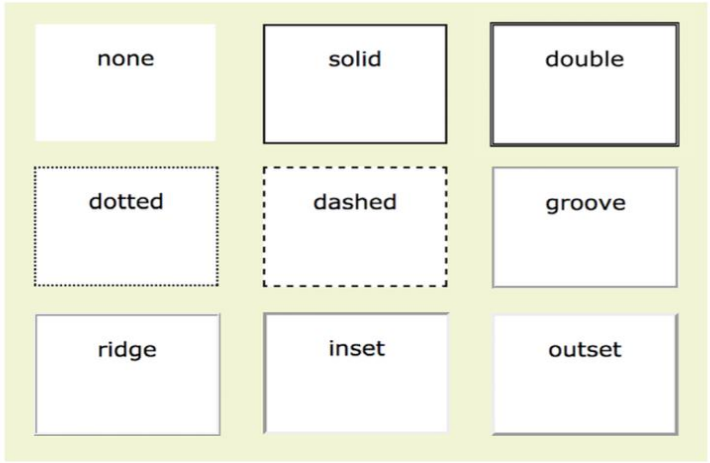


FIGURE 14-8. The available border styles (shown at the default medium width).

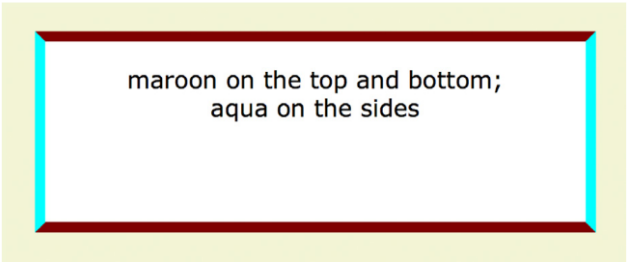


FIGURE 14-11. Specifying the color of borders.

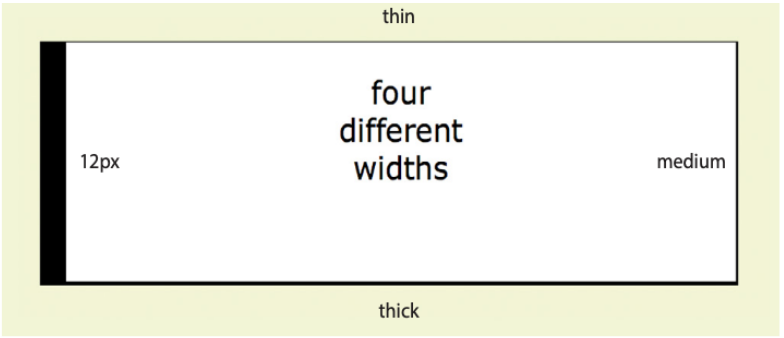


FIGURE 14-10. Specifying the width of borders.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 367

Box Element

Rounded Corners with border-radius

Perhaps you'd like your element boxes to look a little softer and rounder. Well, then, the **border-radius** property is for you! There are individual corner properties as well as a **border-radius** shorthand.

Border-radius, border-top-left-radius, border-top-right-radius, border-bottom-right-radius, border-bottom-left-radius

Values: *length* | *percentage*

Default: 0

Applies to: all elements

Inherits: no

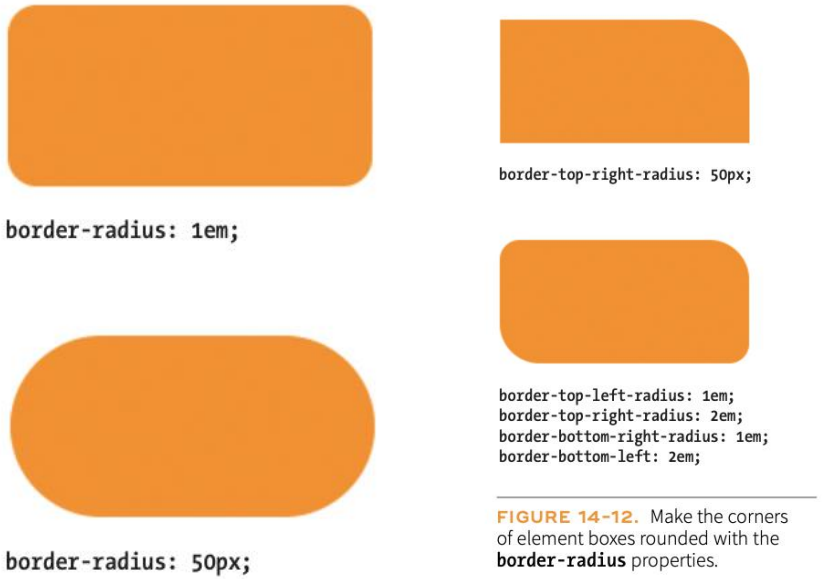


FIGURE 14-12. Make the corners of element boxes rounded with the **border-radius** properties.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 367

Box Element

Property	Description
`border`	A shorthand property that combines border properties

`border-top`
`border-right`
`border-bottom`
`border-left`
	Combines border properties for each side of the element

`border-color`	Shorthand property for specifying the color of borders

`border-top-color`
`border-right-color`
`border-bottom-color`
`border-left-color`
	Specifies the border color for each side of the element

`border-image`	Adds an image inside the border area
`border-image-outset`	How far the border image should be positioned away from the border area.
`border-image-repeat`	The manner in which the image fills the sides of the border
`border-image-slice`	The points at which the border image should be divided into corners and sides
`border-image-source`	The location of the image file to be used for the border image

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 384

Box Element

Property	Description
border-image-width	The width of the space the border image should occupy
border-radius	Shorthand property for rounding the corners of the visible element box
border-top-left-radius border-top-right-radius border-bottom-right-radius border-bottom-left-radius	Specifies the radius curve for each individual corner
border-style	Shorthand property for specifying the style of borders
border-top-style border-right-style border-bottom-style border-left-style	Specifies the border style for each side of the element
border-width	Shorthand property for specifying the width of borders
border-top-width border-right-width border-bottom-width border-left-width	Specifies the border width for each side of the element
box-sizing	Specifies whether width and height dimensions apply to the content box or the border box
box-shadow	Adds a drop shadow around the visible element box
display	Defines the type of element box an element generates
height	Specifies the height of the element's content box or border box
margin	Shorthand property for specifying margin space around an element
margin-top margin-right margin-bottom margin-left	Specifies the margin amount for each side of the element
max-height	Specifies the maximum height of an element
max-width	Specifies the maximum width of an element
min-height	Specifies the minimum height of an element
min-width	Specifies the minimum width of an element
outline	Shorthand property for adding an outline around an element
outline-color	Sets the color of the outline
outline-offset	Sets space between an outline and the outer edge of the border
outline-style	Sets the style of the outline
outline-width	Sets the width of the outline
overflow	Specifies how to handle content that doesn't fit in the content area
padding	Shorthand property for specifying space between the content area and the border
padding-top padding-right padding-bottom padding-left	Specifies the padding amount for each side of the element
width	Specifies the width of an element's content box or border box

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 384

Floating and Positioning

Normal Flow

Objects in the normal flow affect the layout of the objects around them. This is the behavior you've come to expect in web pages—elements don't overlap or bunch up. They make room for one another.

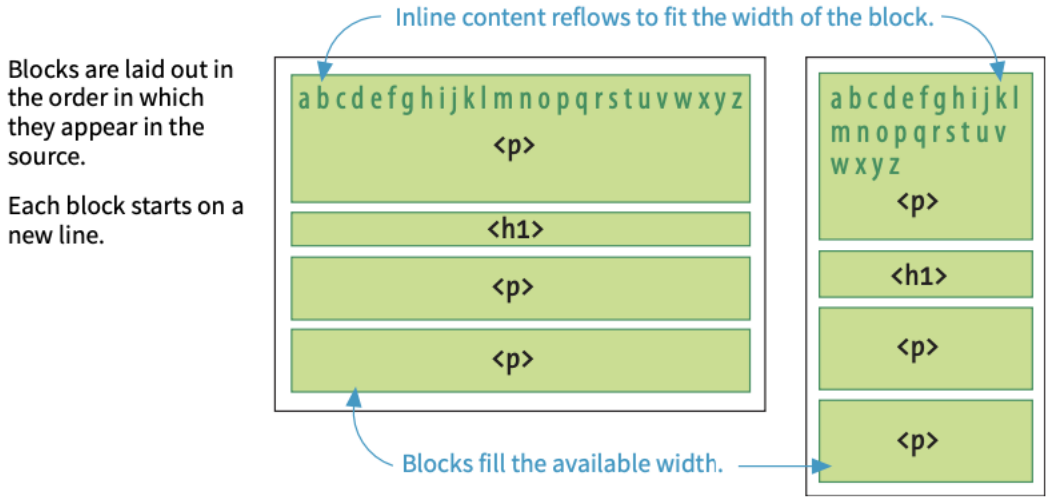


FIGURE 15-1. One more example of the normal flow behavior.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 387

Floating and Positioning

Floating

Simply said, the float attribute shifts an element as far to the left or right as feasible so that the information below it may wrap around it. It is a distinct feature with some intriguing behaviour that is integrated into CSS.

- **float**
- **Values:** left | right | none
- **Default:** none
- **Applies to:** all elements
- **Inherits:** no

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 387

Floating and Positioning

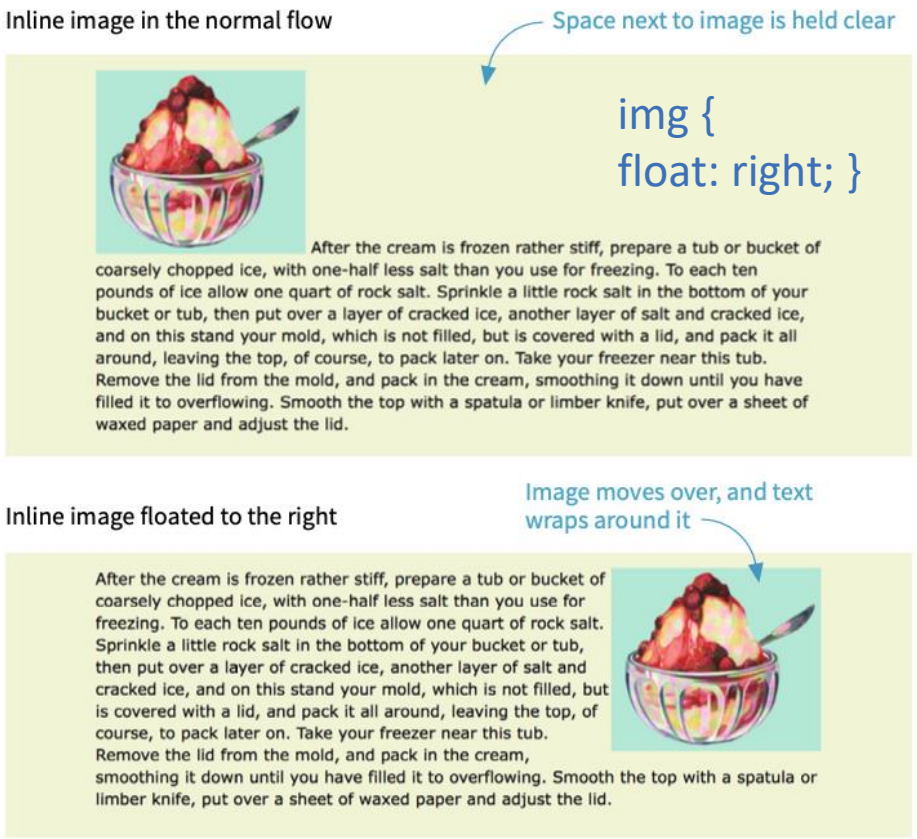


FIGURE 15-2. The layout of an image in the normal flow (top), and with the **float** property applied (bottom).

```
img {
float: right;
margin: 1em;
}
```

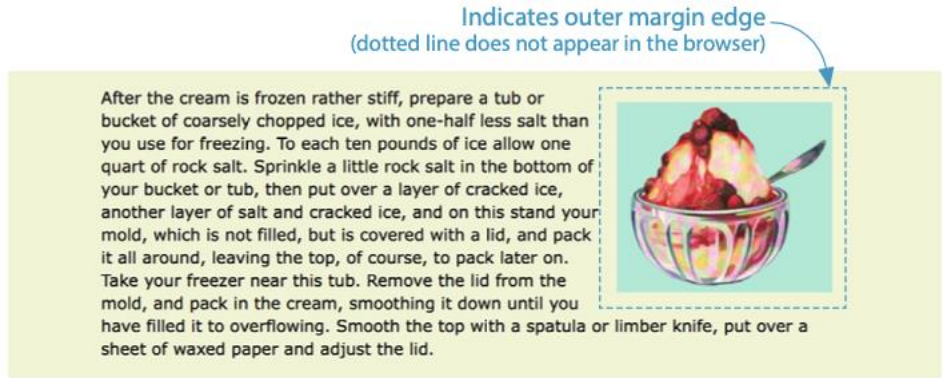


FIGURE 15-3. Adding a 1em margin around the floated image.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 389

Floating and Positioning

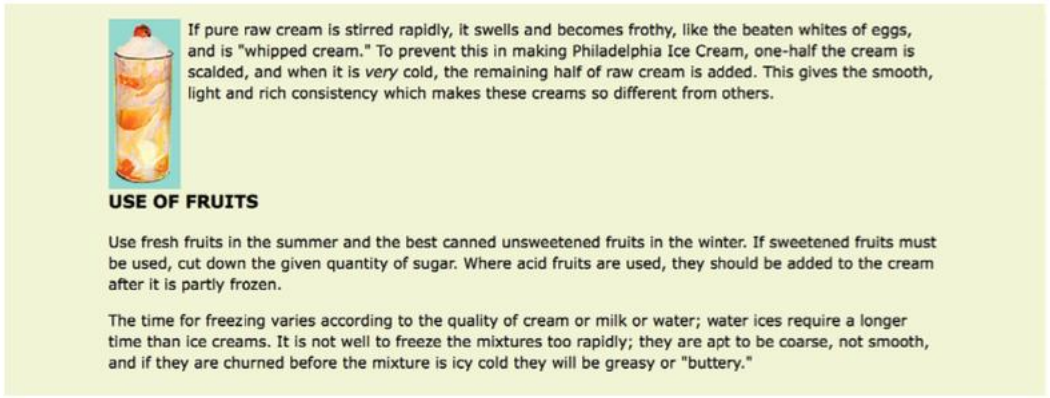
Clearing Float

Knowing how to disable text wrapping and resume the regular flow is crucial if you're going to be floating items about. You accomplish this by removing the element from underneath the float that you wish to start with. When an element has the clear attribute set, it can no longer appear next to a floating element and must instead start against the following "clear" space below the float.

- **clear**
- **Values:** left | right | both | none
- **Default:** none
- **Applies to:** bloack-level elements only
- **Inherits:** no

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 393

Floating and Positioning



If pure raw cream is stirred rapidly, it swells and becomes frothy, like the beaten whites of eggs, and is "whipped cream." To prevent this in making Philadelphia Ice Cream, one-half the cream is scalded, and when it is *very* cold, the remaining half of raw cream is added. This gives the smooth, light and rich consistency which makes these creams so different from others.

USE OF FRUITS

Use fresh fruits in the summer and the best canned unsweetened fruits in the winter. If sweetened fruits must be used, cut down the given quantity of sugar. Where acid fruits are used, they should be added to the cream after it is partly frozen.

The time for freezing varies according to the quality of cream or milk or water; water ices require a longer time than ice creams. It is not well to freeze the mixtures too rapidly; they are apt to be coarse, not smooth, and if they are churned before the mixture is icy cold they will be greasy or "buttery."

FIGURE 15-6. Clearing a left-floated element.

```
img {
float: left;
margin-right: .5em; }
h2 {
clear: left;
margin-top: 2em; }
```

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 394

Floating and Positioning

Text Warp Alternative

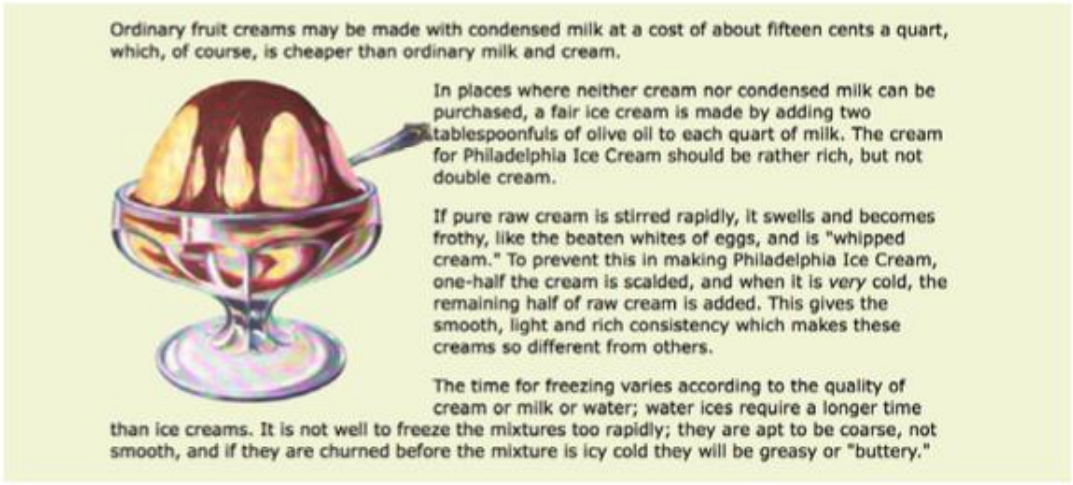
You can see that the text always forms a rectangular border around a floated image or element box if you look at the previous float examples. However, you may modify the form of the wrapped text to a circle, ellipse, polygon, or any image shape by using the shape-outside property.

shape-outside

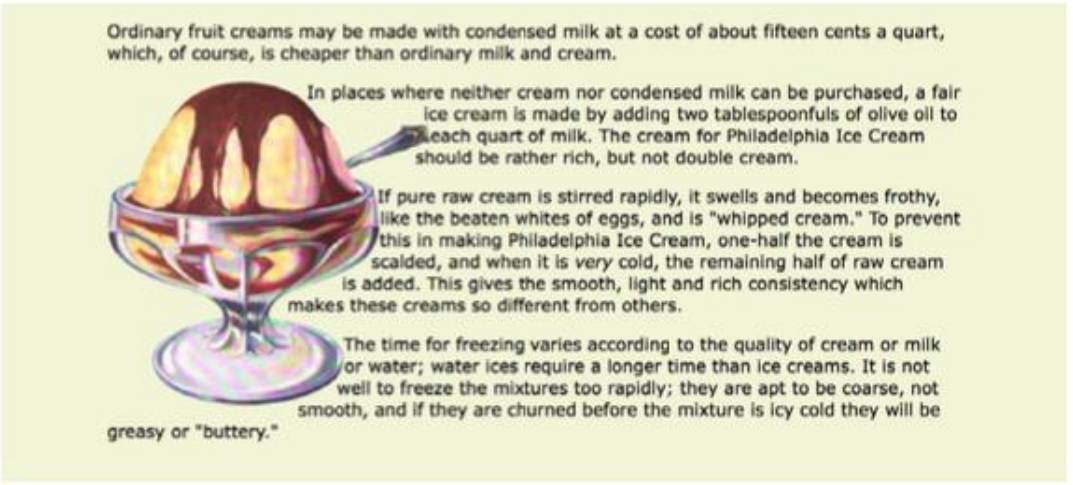
- **Values:** none | circle() | ellipse() | polygon() | url() | [margin-box | padding-box | content-box]
- **Default:** none
- **Applies to:** floats
- **Inherits:** no

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 389

Floating and Positioning



Default text wrap



Text wrap with **shape-outside** using the transparent areas of the image as a guide

FIGURE 15-12. Example of text wrapping around an image with **shape-outside**.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 389

Floating and Positioning

Text Warp Alternative

shape-margin

- **Values:** length | percentage
- **Default: 0**
- **Applies to:** floats
- **Inherits:** no

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 389

Floating and Positioning

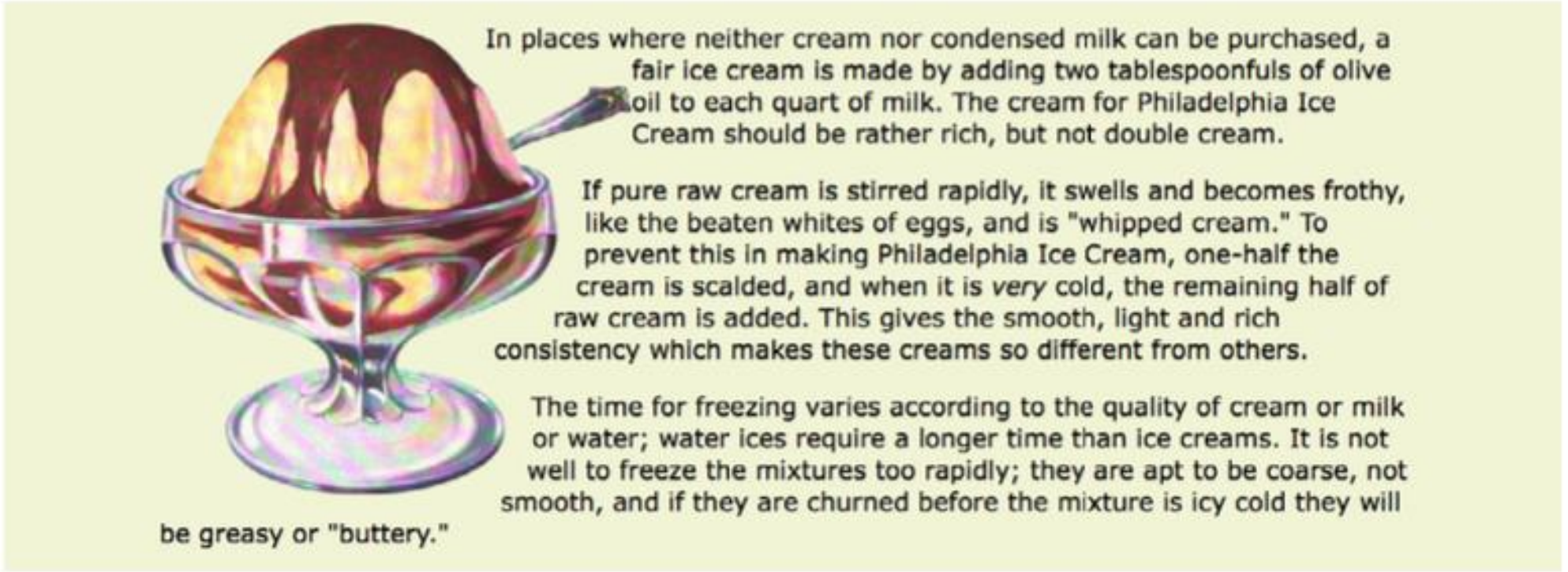


FIGURE 15-13. Adding a margin between the shape and the wrapped text.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 389

Floating and Positioning

Types of Positioning

- **position**
- **Values:** static | relative | absolute | fixed
- **Default:** static
- **Applies to:** all elements
- **Inherits:** no

The **position** property indicates that an element is to be positioned and specifies which positioning method to use.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 389

Floating and Positioning

static

This is the normal positioning scheme in which elements are positioned as they occur in the normal document flow.

relative

Relative positioning moves the element box relative to its original position in the flow. The distinctive behavior of relative positioning is that the space the element would have occupied in the normal flow is preserved as empty space.

absolute

Absolutely positioned elements are removed from the document flow entirely and positioned with respect to the viewport or a containing ele- ment (we'll talk more about this later). Unlike relatively positioned ele- ments, the space they would have occupied is closed up. In fact, they have no influence at all on the layout of surrounding elements.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 389

Floating and Positioning

fixed

The distinguishing characteristic of fixed positioning is that the element stays in one position in the viewport even when the document scrolls. Fixed elements are removed from the document flow and positioned rela- tive to the viewport rather than another element in the document.

sticky

Sticky positioning is a combination of relative and fixed in that it behaves as though it is relatively positioned, until it is scrolled into a specified position relative to the viewport, at which point it remains fixed

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 389

Flexbox

Flexbox is another layout mode with its own behaviors. To turn on flexbox mode for an element, set its display property to flex or inline-flex. It is now a flex container, and all of its direct child elements (whether they are divs, list items, paragraphs, etc.) automatically become flex NOTE items in that container. The flex items (the beads) are laid out and aligned along flex lines (the string).

Flexbox

THE MARKUP

```
<div id="container">
<div class="box box1">1</div>
 <div class="box box2">2</div>
<div class="box box3">3</div>
<div class="box box4">4</div>
<div class="box box5">5</div>
</div>
```

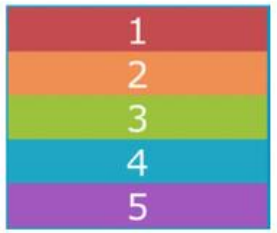
THE STYLES

```
#container {
display: flex;
}
```

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 421

Flexbox

By default, the **divs** display as block elements, stacking up vertically. Turning on flexbox mode makes them line up in a row.



block layout mode

```
display: flex;
```

flexbox layout mode

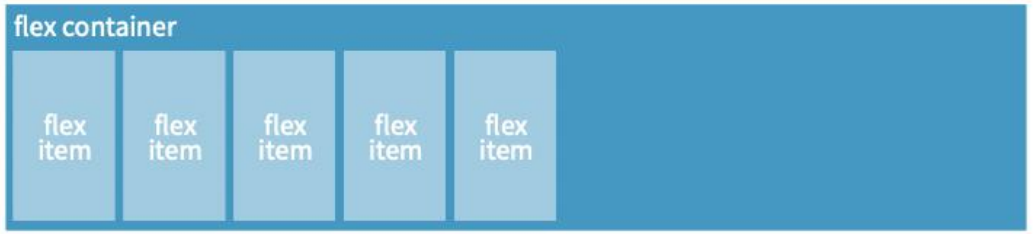


FIGURE 16-2. Applying the flex display mode turns the child elements into flex items that line up along one axis. You don't need to do anything to the child elements themselves.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 422

Flexbox

Specifying flow direction

- **flex-direction**
- **Values:** row | column | row-reverse | column-reverse
- **Default:** row
- **Applies to:** flex containers
- **Inherits:** no

The default value is row. You can also specify that items get aligned vertically in a column. The other options, row-reverse and column-reverse, arrange items in the direction you would expect, but they start at the end and get filled in the opposite direction.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 421

Flexbox

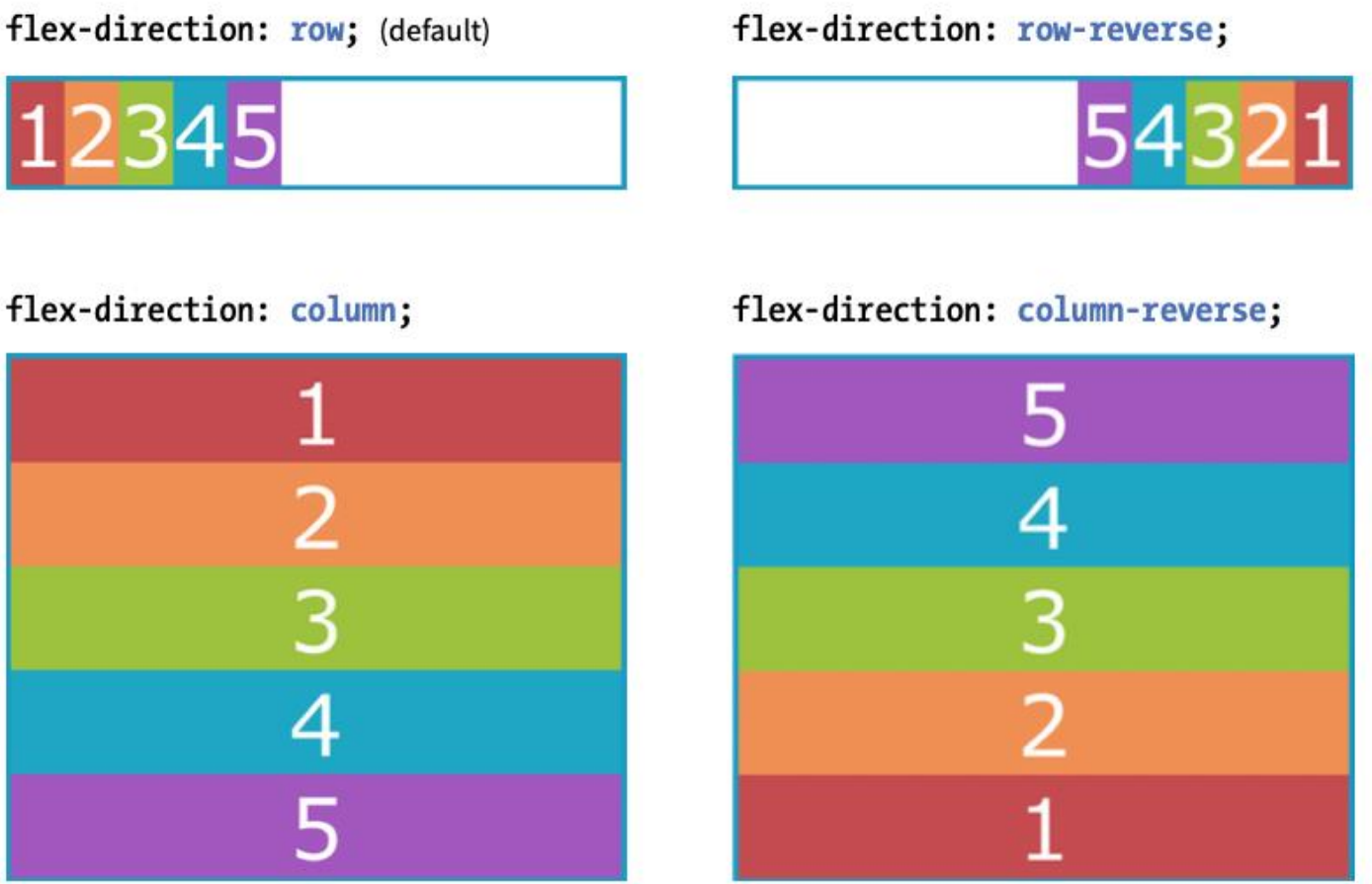


FIGURE 16-3. Examples of **flex-direction** values **row**, **row-reverse**, **column**, and **column-reverse**.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 423

Flexbox

Wrapping onto multiple lines

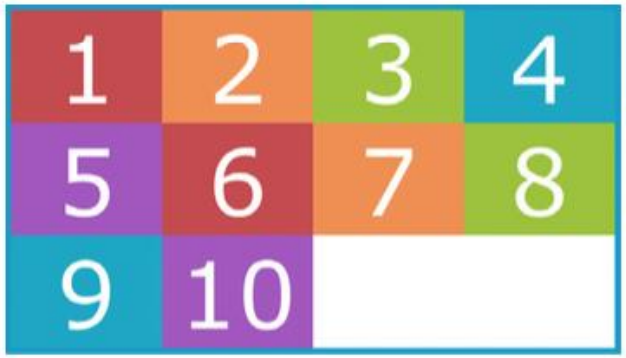
If you have a large or unknown number of flex items in a container and don’t want them to get all squished into the available space, you can allow them to break onto additional lines with the flex-wrap property.

- **flex-wrap**
- **Values:** nowrap | wrap | wrap-reverse
- **Default:** nowrap
- **Applies to:** flex containers
- **Inherits: no**

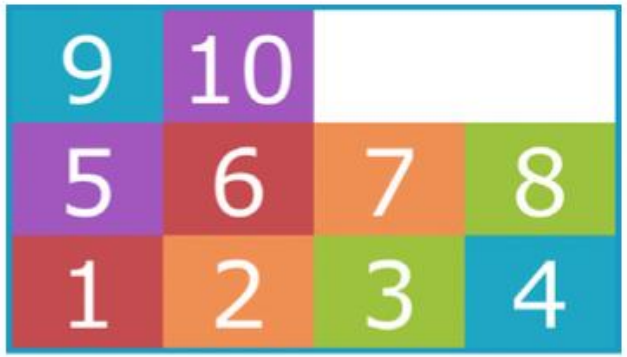
Learning Web Design, Jennifer Niederst Robbins, O’Reilly Media, Inc., 2018, Page 423

Flexbox

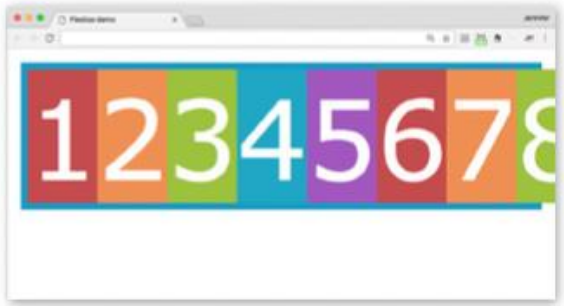
`flex-wrap: wrap;`



`flex-wrap: wrap-reverse;`



`flex-wrap: nowrap;` (default)



When wrapping is disabled, flex items squish if there is not enough room, and if they can't squish any further, may get cut off if there is not enough room in the viewport.

FIGURE 16-5. Comparing the effects of **nowrap**, **wrap**, and **wrap-reverse** keywords for **flex-wrap**.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 425

Flexbox

Controlling the Alignment of Flex Items in the Container

So far we've seen how to turn flexbox mode on, turning an element into a flex container and its children into flex items. We've also learned how to change the direction in which items flow, and allow them to wrap onto multiple lines. The remaining set of container properties affects the alignment of items along the main axis (justify-content) and cross axis (align-items and align-content).

Flexbox

- **justify-content**
- **Values:** flex-start |flex-end |center |space-between |space-around
- **Default:** flex-start
- **Applies to:** flex containers
- **Inherits:** no

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 427

Flexbox

justify-content: flex-start; (default)

justify-content: flex-end;

justify-content: center;

justify-content: space-between;

justify-content: space-around;

FIGURE 16-8. Options for aligning items along the main axis with **justify-content**.

Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, Page 428

Thank You

Alfred Tenggono, S.Kom., M.Kom.

alfred.tenggono@jiu.ac

Reference

- Learning Web Design, Jennifer Niederst Robbins, O'Reilly Media, Inc., 2018, ISBN: 978-1-491-96020-2