Now You See It.

Rachel Andrew @ An Event Apart San Francisco 2019

CSS1 Recommendation

December 1996

Cascading Style Sheets, level 1

W3C Recommendation 17 Dec 1996

This is:

http://www.w3.org/pub/WWW/TR/REC-CSS1

Authors:

[Håkon Wium Lie](#) (howcome@w3.org)
[Bert Bos](#) (bert@w3.org)

Status of this document

This document is a W3C Recommendation. It has been reviewed by [W3C](#) (http://www.w3.org/) Members and general consensus that the specification is appropriate for use has been reached. It is a stable document and may be used as reference material or cited as a normative reference from another document. W3C promotes widespread deployment of this Recommendation.
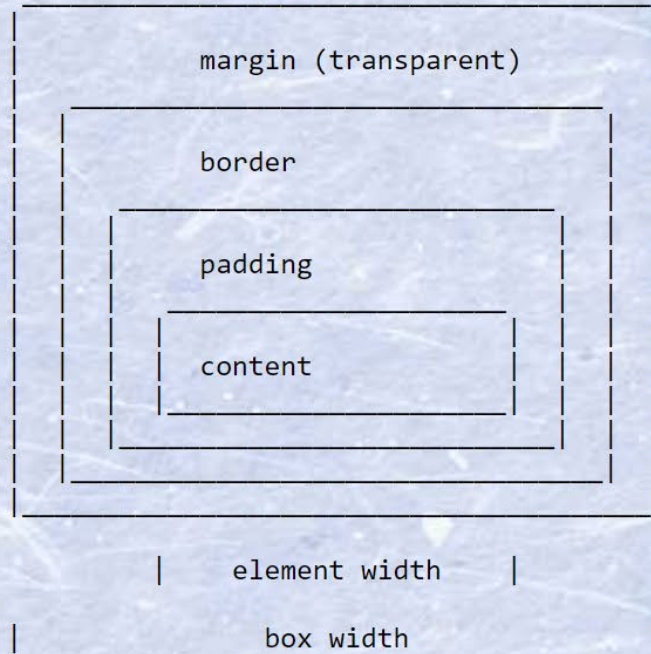
A list of current W3C Recommendations and other technical documents can be found at http://www.w3.org/pub/WWW/TR/.

Abstract

This document specifies level 1 of the Cascading Style Sheet mechanism (CSS1). CSS1 is a simple style sheet mechanism that allows authors and readers to attach style (e.g. fonts, colors and spacing) to HTML documents. The CSS1 language is human readable and writable, and expresses style in common desktop publishing terminology.

4   Formatting model

CSS1 assumes a simple box-oriented formatting model where each element results in one or more rectangular boxes. (Elements that have a 'display' value of 'none' are not formatted and will therefore not result in a box.) All boxes have a core content area with optional surrounding padding, border and margin areas.

```
 _____
|                                            |
|            margin (transparent)            |
|     _____      |
|    |                                 |     |
|    |  border                         |     |
|    |     _____     |     |
|    |    |                       |    |     |
|    |    |  padding              |    |     |
|    |    |     _____     |    |     |
|    |    |    |             |    |     |    |
|    |    |    |  content    |    |     |    |
|    |    |    |_____|    |     |    |
|    |    |_____|    |     |
|    |_____|     |
|_____|

        |      element width      |

   |              box width              |
```

The size of the margin, border and padding are set with the margin (5.5.1-5.5.5), padding (5.5.6-5.5.10), and border (5.5.11-5.5.22) properties respectively. The padding area uses the same background as the element itself (set with the background properties (5.3.2-5.3.7). The color and style for the border is set with the border properties. The margins are always transparent, so the parent element will shine through.

The size of the box is the sum of the element width (i.e. formatted text or image) and the padding, the border and the margin areas.

From the formatter's point of view there are two main types of elements: block-level and inline.

4.1   Block level elements

Some things are **block** things other things are **inline** things.

This is a heading

Veggies es bonus vobis, proinde vos postulo essum magis kohlrabi welsh onion daikon amaranth tatsoi tomatillo melon azuki bean garlic.

Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini.

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

A level 2 heading

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

Celery quandong swiss chard chicory earthnut pea potato. Salsify taro catsear garlic gram celery bitterleaf wattle seed collard greens nori. Grape wattle seed kombu beetroot horseradish carrot squash brussels sprout chard.

This is a heading

Veggies es bonus vobis, proinde vos postulo essum magis kohlrabi welsh onion daikon amaranth tatsoi tomatillo melon azuki bean garlic. Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini. Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

A level 2 heading

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify. Celery quandong swiss chard chicory earthnut pea potato. Salsify taro catsear garlic gram celery bitterleaf wattle seed collard greens nori. Grape wattle seed kombu beetroot horseradish carrot squash brussels sprout chard.

A heading is a block thing

Paragraphs are block by default.

Block-level boxes can have a size in both dimensions, this will push other boxes away.

- Lists are block
- So are list items

Veggies es bonus vobis, proinde vos postulo essum magis kohlrabi welsh onion daikon amaranth tatsoi tomatillo melon azuki bean garlic. Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini. Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea.

Veggies es bonus vobis, proinde vos postulo essum magis kohlrabi welsh onion daikon amaranth tatsoi tomatillo melon azuki bean garlic. Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini. Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea.

Being able to change the value of display is important.

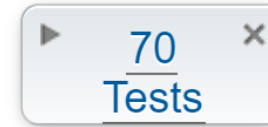We can choose the correct semantic HTML element for the job, then use CSS to change how it looks.

Teaching CSS over the past 20 years

- Here is a block thing
- Here is an inline thing
- This is the Box Model. It is **Very Important** and also kind of weird …

display

CSS Display Module Level 3

W3C Candidate Recommendation, 11 July 2019

70
Tests

This version:

https://www.w3.org/TR/2019/CR-css-display-3-20190711/

Latest published version:

https://www.w3.org/TR/css-display-3/

Editor's Draft:

https://drafts.csswg.org/css-display/

Previous Versions:

https://www.w3.org/TR/2018/CR-css-display-3-20180828/

https://www.w3.org/TR/2018/WD-css-display-3-20180809/

https://www.w3.org/TR/2018/WD-css-display-3-20180420/

https://www.w3.org/TR/2017/WD-css-display-3-20170720/

https://www.w3.org/TR/2017/WD-css-display-3-20170126/

https://www.w3.org/TR/2015/WD-css-display-3-20151015/

https://www.w3.org/TR/2015/WD-css-display-3-20150721/

https://www.w3.org/TR/2014/WD-css-display-3-20140911/

Test Suite:

http://test.csswg.org/suites/css-display-3_dev/nightly-unstable/

Editors:

Tab Atkins Jr. (Google)

Elika J. Etemad / fantasai (Invited Expert)

Suggest an Edit for this Spec:

block and inline

Paragraphs are block by default.

They display one after the other in the block direction.

inline

block

Paragraphs are block by default.

They display one after the other in the block direction.

Block and inline

- The block dimension is the direction that paragraphs lay out in your writing mode.
- The inline direction is the direction in which sentences run in your writing mode.

block

inline

writing-mode: vertical-rl

They display one after the other in the block direction.
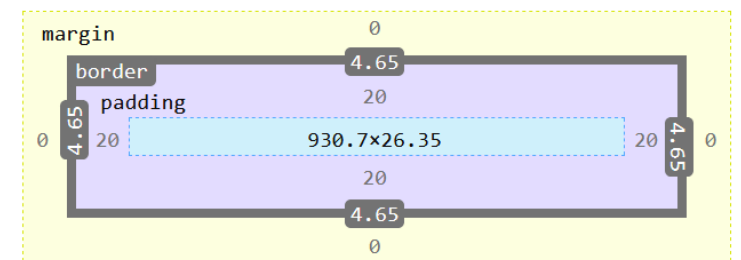
Paragraphs are block by default.
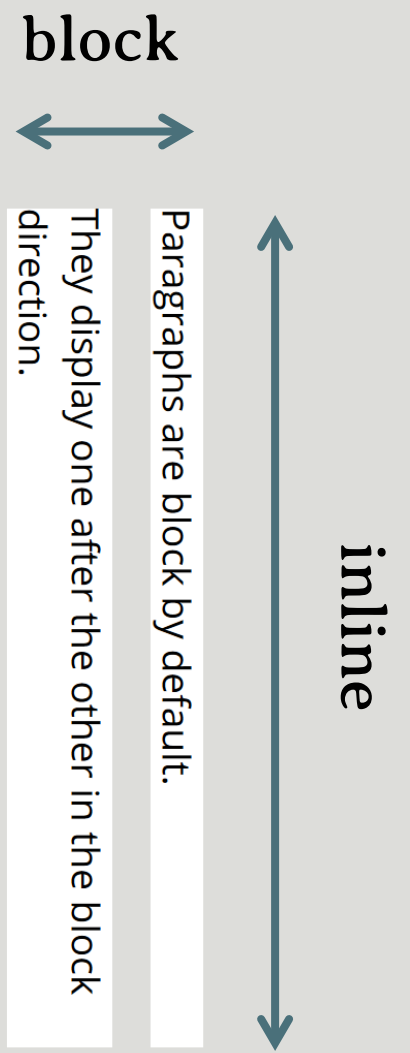
Normal Flow

block and inline layout

Layout always returns to Normal Flow

A well-structured HTML document means you are working with the browser rather than against it.

Content.

Box Model


```
            margin
                        0
        border
            padding     4.65
                        20
  0    20          930.7×26.35          20    0
4.65                                              4.65
                        20
                       4.65
                        0
```

980×75.65   static

Box Model Properties

box-sizing	border-box	line-height	normal
display	block	position	static
float	none	z-index	auto

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

For each element, CSS generates one or more boxes as specified by that element's display property.

display: block

Creates a block-level box.

display: inline

Creates an inline-level box.

display: flex

Creates a block-level box with flex children.

Creating a Flex Formatting Context

Formatting context

Describes the behavior of the child elements of a box.

display: flex

Flex Item One | Item Two | Three

display: flex

<table>
<tr><td>Flex Item One</td><td>Item Two</td><td>Three</td></tr>
</table>

Some text that follows the flex container.

justify-content: space-between

Flex Item One

Item Two

Three

display: inline-flex

<table>
<tr>
<td>Flex Item One</td>
<td>Item Two</td>
<td>Three</td>
</tr>
</table>
Some text

that follows the flex container.

display: grid

Creates a block-level box with children participating in a grid formatting context.

```
display: grid;
grid-template-columns: 1fr 2fr 2fr;
```

Grid Item One

Item Two

Three

4

The value of display

- Changes how the box behaves among other boxes – is it block or inline?
- Changes the behavior of the direct children – for example they become flex items.

Two values for display

Refactoring the display specification.

display: block flex

Flex Item One	Item Two	Three

Some text that follows the flex container.

display: inline flex

| Flex Item One | Item Two | Three | Some text

that follows the flex container.

display: block grid

Grid Item One

Item Two

Three

4

display: block flow

Paragraphs are block by default.

They display one after the other in the block direction.

Old Value	New Value(s)
block	block flow
flow-root	block flow-root
inline	inline flow
inline-block	inline flow-root
flex	block flex
inline-flex	inline flex
grid	block grid
inline-grid	inline grid

display: block flow-root

Starting over with a new formatting context for normal flow.

Cat ipsum dolor sit amet, purr as loud as possible, be the most annoying cat that you can, and, knock everything off the table meow for food, then when human fills food dish, take a few bites of food and continue meowing so jumps off balcony gives owner dead mouse at present then poops in litter box snatches yarn and fights with dog cat chases laser then plays in grass finds tiny spot in cupboard and sleeps all day jumps in bathtub and meows when owner fills food dish the cat knocks over the food dish cat slides down the water slide and into pool and swims even though it does not like water.

overflow: auto

Cat ipsum dolor sit amet, purr as loud as possible, be the most annoying cat that you can, and, knock everything off the table meow for food, then when human fills food dish, take a few bites of food and continue meowing so jumps off balcony gives owner dead mouse at present then poops in litter box snatches yarn and fights with dog cat chases laser then plays in grass finds tiny spot in cupboard and sleeps all day jumps in bathtub and meows when owner fills food dish the cat knocks over the food dish cat slides down the water slide and into pool and swims even though it does not like water.

display: block flow-root

Cat ipsum dolor sit amet, purr as loud as possible, be the most annoying cat that you can, and, knock everything off the table meow for food, then when human fills food dish, take a few bites of food and continue meowing so jumps off balcony gives owner dead mouse at present then poops in litter box snatches yarn and fights with dog cat chases laser then plays in grass finds tiny spot in cupboard and sleeps all day jumps in bathtub and meows when owner fills food dish the cat knocks over the food dish cat slides down the water slide and into pool and swims even though it does not like water.

display: inline flow-root

An inline box that creates a new formatting context.

Cat ipsum dolor sit amet, purr as loud as possible.

display: inline-block

Cat ipsum dolor sit amet, purr as loud as possible.

display: inline flow-root

Cat ipsum dolor sit amet, purr as loud as possible.

Margins

Do not collapse through a new formatting context.

One

Two

Three

div | 399.8 × 46

One

Two

Three

display: flow-root

div | 399.8 × 46

One

Two

Three

display: flex

div | 57.5 × 46 | Flex Item

One   Two   Three

Anonymous boxes

```
<p>One <span>two</span> three</p>
```

Onetwo**three**

```
p {
  display: flex;
  justify-content: space-between;
}
```

One   two   three

Anonymous Boxes

Ensure that everything is in a box.

```
p > * {
  color: rgb(74,112,122);
}
```

One two three

Anonymous Boxes

Fixing up the box tree.

```
li {
  display: table-cell;
}
```

Card One

Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini.

Card Two

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi.

Card Three

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

When Layout Methods Collide

Changing the formatting context away from block and inline layout, means some things no longer do what we are used to.

Floating and positioning

Taking items out of flow.

Cat ipsum dolor sit amet, purr as loud as possible, be the most annoying cat that you can, and, knock everything off the table meow for food, then when human fills food dish, take a few bites of food and continue meowing so jumps off balcony gives owner dead mouse at present then poops in litter box snatches yarn and fights with dog cat chases laser then plays in grass finds tiny spot in cupboard and sleeps all day jumps in bathtub and meows when owner fills food dish the cat knocks over the food dish cat slides down the water slide and into pool and swims even though it does not like water.

ud as possible, be the most annoying cat that you can, and, knock everything off the table meow for sh, take a few bites of food and continue meowing so jumps off balcony gives owner dead mouse at tches yarn and fights with dog cat chases laser then plays in grass finds tiny spot in cupboard and meows when owner fills food dish the cat knocks over the food dish cat slides down the water slide gh it does not like water.

Cat ipsum dolor sit amet, purr as loud as possible, be the most annoying cat that you can, and, knock everything off the table meow for food, then when human fills food dish, take a few bites of food and continue meowing so jumps off balcony gives owner dead mouse at present then poops in litter box snatches yarn and fights with dog cat chases laser then plays in grass finds tiny spot in cupboard and sleeps all day jumps in bathtub and meows when owner fills food dish the cat knocks over the food dish cat slides down the water slide and into pool and swims even though it does not like water.

Out of flow

The floating and positioning behavior we understand is specified for normal flow, for block and inline layout. They behave differently, or don't work at all in other formatting contexts.

```
.box blockquote {
    float: left;
}
```

```
.box {
    display: grid;
    grid-template-columns: 1fr 2fr
}
```

Dandelion cucumber earthnut pea peanut soko zucchini. Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea.

Dandelion cucumber earthnut pea peanut soko zucchini. Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea.

```
li {
    float: left;
}
```

Card One

Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini.

Card Two

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi.

Card Three

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

```
ul {
    display: grid;
}
```

Card One

Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini.

Card Two

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi.

Card Three

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

Editorial Layouts, Floats, and CSS Grid

CSS Grid is in many ways a dream come true, but there's at least one basic thing it can't do.

8 NOV 2018

I've been getting my feet wet with CSS Grid this year. The new possibilities it introduces for page layout on the web are exciting, but before venturing too far into those waters, I wanted to see how it handles the kind of layouts I'm already designing. Most of the work I do on the web—at ProPublica and here on my site—is editorial design, so I was keen to learn how Grid accommodates the conventions of the form.

One of those conventions is the interspersion of photographs, illustrations, figures, and asides throughout the text of an article. A typical template is based on a columnar grid, which establishes the text area and various size and position options for supporting material. Take this simple example of a five-column article layout, with the main text flowing around a two-column image:

TL;DR

Since floats aren't compatible with CSS Grid, aligning floated elements to an established grid can be a real headache. This is a nontrivial impediment to traditional editorial layouts' ability to take advantage of Grid's benefits.
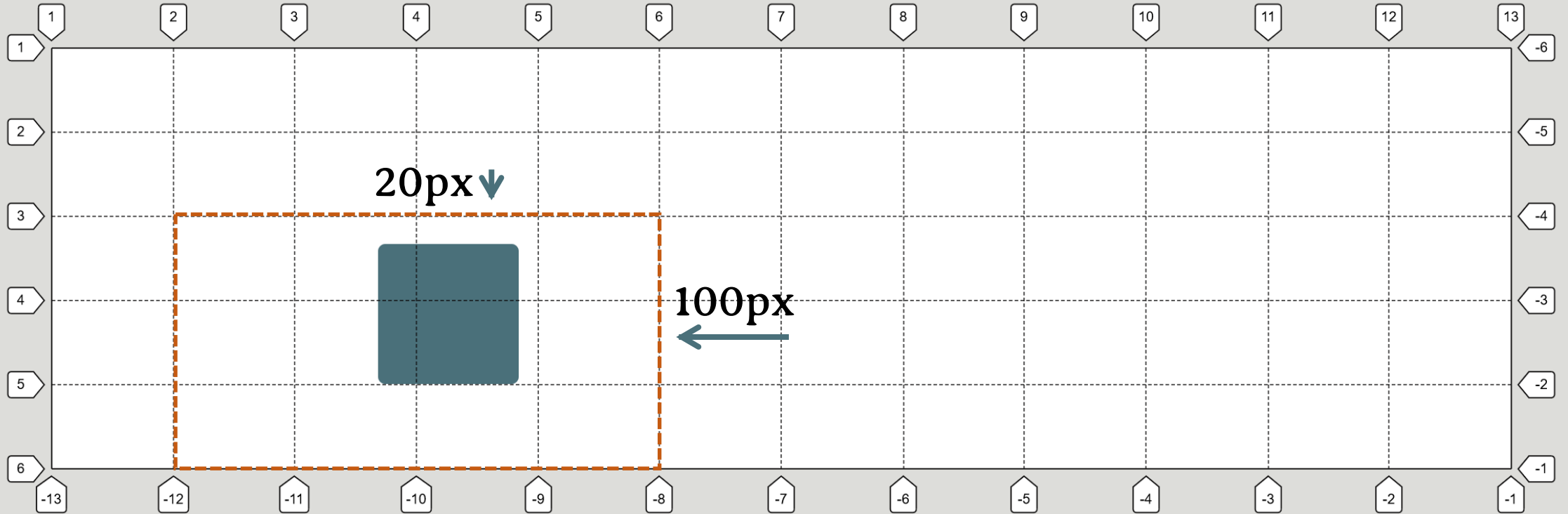
Grid and position

You can absolutely position items in a grid layout.

grid-column: 2 / 6;
grid-row: 3 / -1;
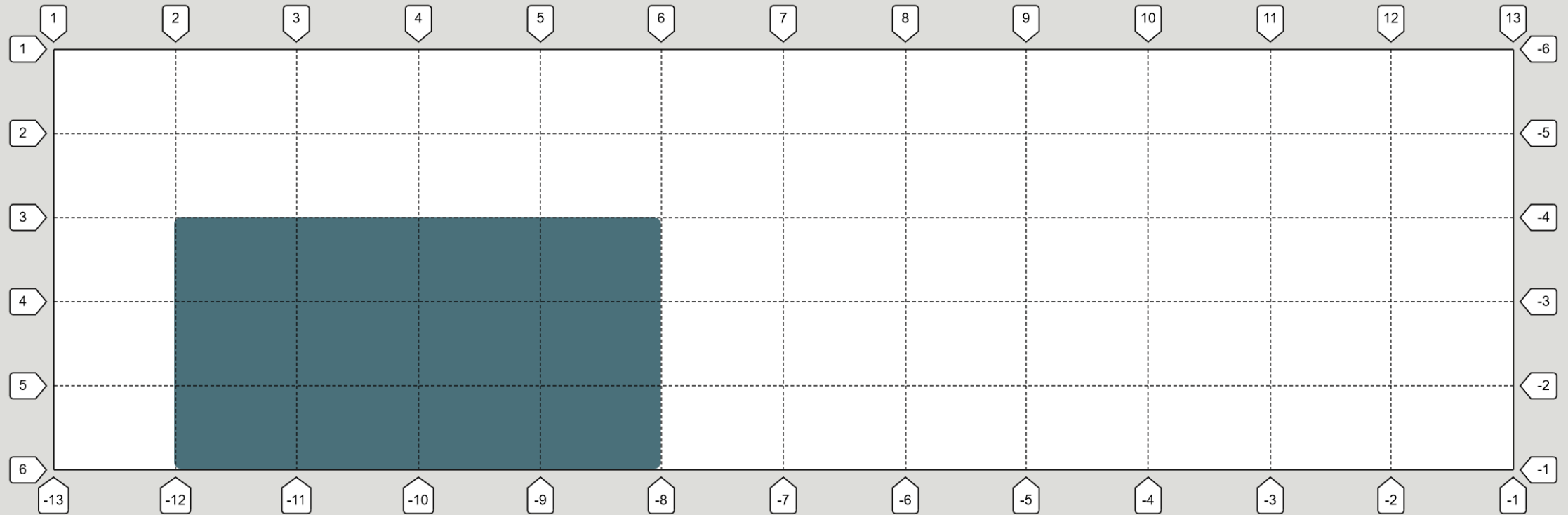
```
position: absolute;
top: 20px;
right: 100px;
```

display: table-cell

Anonymous boxes created to fix up the box tree do not get generated once the item participates in a grid formatting context.

This is all as specified

Precise details ensure that each browser does the same thing, makes for happier web developers!

This is all testable

The Web Platform Tests project has tests against web platform specifications, so user agents can check they are conforming.

animations/	13 / 13	13 / 13	13 / 13	13 / 13
display-change-iframe.html	1 / 1	1 / 1	1 / 1	1 / 1
display-change-object-iframe.html	1 / 1	1 / 1	1 / 1	0 / 1
display-contents-alignment-001.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-alignment-002.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-before-after-001.html	0 / 1	0 / 1	1 / 1	1 / 1
display-contents-before-after-002.html	0 / 1	0 / 1	0 / 1	1 / 1
display-contents-before-after-003.html	0 / 1	0 / 1	0 / 1	1 / 1
display-contents-block-001.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-block-002.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-blockify-dynamic.html	2 / 2	2 / 2	2 / 2	2 / 2
display-contents-button.html	1 / 1	1 / 1	1 / 1	0 / 1
display-contents-computed-style.html	6 / 6	6 / 6	6 / 6	6 / 6
display-contents-details-001.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-details.html	0 / 1	0 / 1	1 / 1	0 / 1
display-contents-dynamic-before-after-001.html	1 / 1	1 / 1	0 / 1	1 / 1
display-contents-dynamic-before-after-first-letter-001.html	1 / 1	1 / 1	0 / 1	1 / 1
display-contents-dynamic-flex-001-inline.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-flex-001-none.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-flex-002-inline.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-flex-002-none.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-flex-003-inline.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-flex-003-none.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-generated-content-fieldset-001.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-inline-flex-001-inline.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-inline-flex-001-none.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-list-001-inline.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-list-001-none.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-multicol-001-inline.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-multicol-001-none.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-pseudo-insertion-001.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-table-001-inline.html	0 / 1	0 / 1	1 / 1	0 / 1
display-contents-dynamic-table-001-none.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-table-002-inline.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-dynamic-table-002-none.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-fieldset-nested-legend.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-fieldset.html	1 / 1	1 / 1	0 / 1	1 / 1
display-contents-first-letter-001.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-first-letter-002.html	1 / 1	1 / 1	0 / 1	0 / 1
display-contents-first-line-001.html	1 / 1	1 / 1	0 / 1	1 / 1
display-contents-first-line-002.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-flex-001.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-flex-002.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-flex-003.html	1 / 1	1 / 1	1 / 1	1 / 1
display-contents-float-001.html	1 / 1	1 / 1	1 / 1	1 / 1

Not Generating Boxes

display: none and display: contents

display: none

Do not generate a box for the element, or for the children of the element.

Aside from the none value, which also affects the aural/speech output and interactivity of an element and its descendants, the display property only affects visual layout
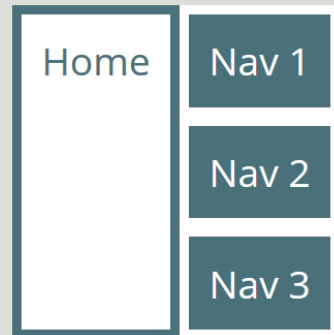
https://www.w3.org/TR/css-display-3/#the-display-properties

display: contents

Like display: none but only the box it is applied to is removed. The children remain.

```
<nav>
    <a class="home" href="/">Home</a>
    <ul>
      <li><a href="">Nav 1</a></li>
      <li><a href="">Nav 2</a></li>
      <li><a href="">Nav 3</a></li>
    </ul>
</nav>
```

Home

Nav 1

Nav 2

Nav 3

```
nav {
  display: flex;
}
```

```css
nav {
  display: flex;
  justify-content: space-between;
}

nav ul {
  display: contents;
}
```

Home
Nav 1
Nav 2
Nav 3

⊕ Browser and Spec Bugs

Ten days before my tests, Hidde de Vries had already filed bugs with browsers based on use of `display: contents` in grid layouts:

- [Firefox bug 1455357](): Setting grid item to display:contents resets its accessible role

  - **Update, 15 May 2018:** [Verified fixed]() in Firefox 62.0a1 (20180515220059).

  - **Update, 22 October 2018:** fixed only for lists, as narrowly defined in the original bug.

- [Chromium Issue 835455](): Element not exposed to accessibility tree when it is set to display: contents

- [Safari bug 3963O24O]() (which I cannot see because my AppleID may not have the needed permissions to see it)

After a Twitter conversation this morning with Ilya Streltsyn, he took the initiative to file an issue against the CSS spec:

- [CSSWG #2632](): [css-display][css-aam][selectors-4] How elements with `display:contents` get focused?

Exciting boxes!

Boxes generating boxes.

Principal Box

- List item one
- List item two
- List item three

- List item one
- List item two
- List item three

```
li:first-child {
  color: white;
}
```

- **List item one**
- **List item two**
- **List item three**

::marker

```css
li:first-child::marker {
  color: black;
}
```

- **List item one**
- **List item two**
- **List item three**

li::marker 32.8333 × 17.9

- List item one
- List item two
- List item three

Assets   Shortcuts

ole

□ Inspector   ⊡ Console   ◗ Debugger   ⇅ Network   { } Style Editor   ◎ Performance   ◉ Memory   ▤ Storage   ⚙ Accessibility   ▦ What's New

ch HTML

```
▼<div class="output-sizer"> flex
  ▼<div id="result_div" class="result">
    ▼<iframe id="iFrameKey-3ab7aee7-aed7-a42f-6c69-a2fa0164b5f1" class="result-iframe" src="https://cdpn.io/boomboom/v2/index.html?template=61f1bbe4593b…e45f4ccfd&key=iFrameKey-3ab7aee7-aed7-a42f-6c69-a2fa0164b5f1" name="CodePen"
      allowfullscreen="true" sandbox="allow-forms allow-modals allow-pointer-lock allow-popups allow-presentation allow-same-origin allow-scripts" allow="geolocation; microphone; camera; midi; vr; accelerometer; gyroscope; payment
      ambient-light-sensor; encrypted-media" allowtransparency="true" allowpaymentrequest="true"> event
      ▼#document
          <!DOCTYPE html>
        ▼<html class="" lang="en"> event scroll
          ▶<head> ⋯ </head>
          ▼<body>
            ▼<div class="container">
              ▼<ul>
                ▼<li>
                    ::marker
                    List item one
                  </li>
                ▶<li> ⋯ </li>
                ▶<li> ⋯ </li>
              </ul>
```

```css
li::marker {
  content: "Step: " counter(list-item) ": ";
}
```

Step 1: **List item one**

Step 2: **List item two**

Step 3: **List item three**

```
h1 {
  display: list-item;
}
```

- # Veggies!

Veggies es bonus vobis, proinde vos postulo essum magis kohlrabi welsh onion daikon amaranth tatsoi tomatillo melon azuki bean garlic. Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini. Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea.

```
h1::marker {
  content: "🥦";
}
```

🥦 Veggies!

Veggies es bonus vobis, proinde vos postulo essum magis kohlrabi welsh onion daikon amaranth tatsoi tomatillo melon azuki bean garlic. Gumbo beet greens corn soko endive gumbo gourd. Parsley shallot courgette tatsoi pea sprouts fava bean collard greens dandelion okra wakame tomato. Dandelion cucumber earthnut pea peanut soko zucchini. Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea.

It's all a value of display

Values of display do not inherit.

They act on the principal box and its direct children; the grandchildren go back to normal flow.

subgrid

subgrid

Allowing track definitions to be inherited by a grid on a child.

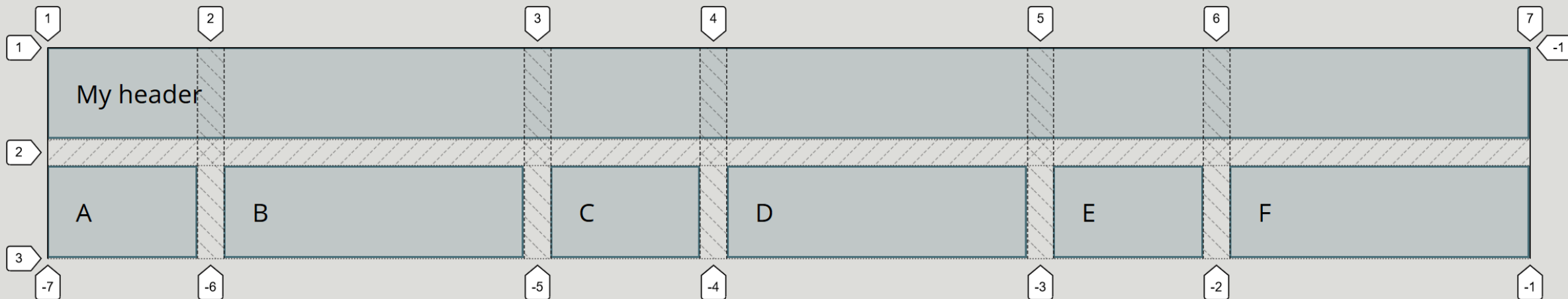# grid-template-columns: repeat(3,1fr 2fr)

My header

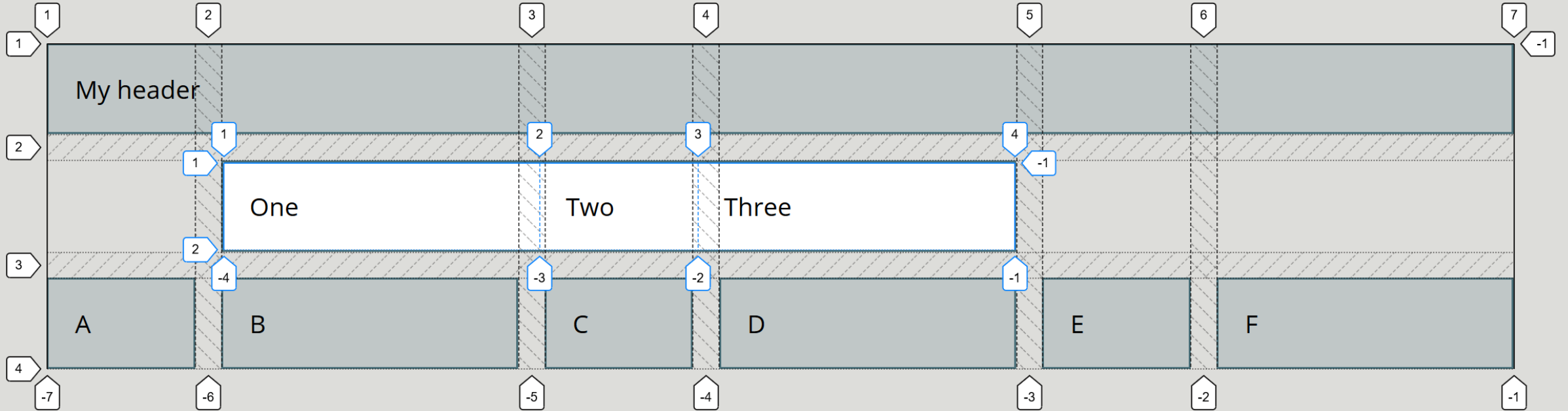| A | B | C | D | E | F |

grid-column: 2 / 5
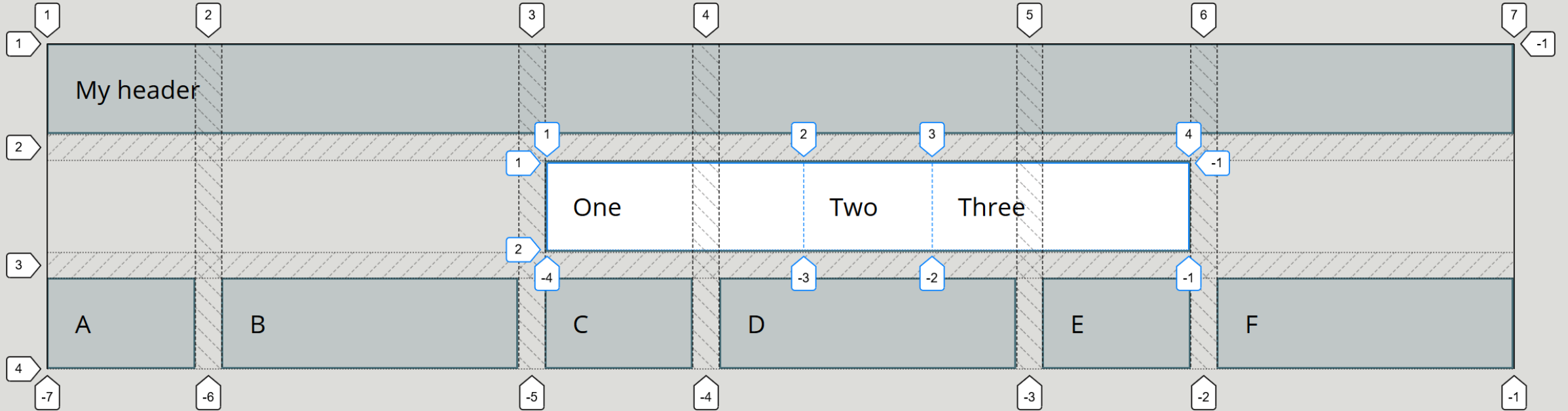
My header

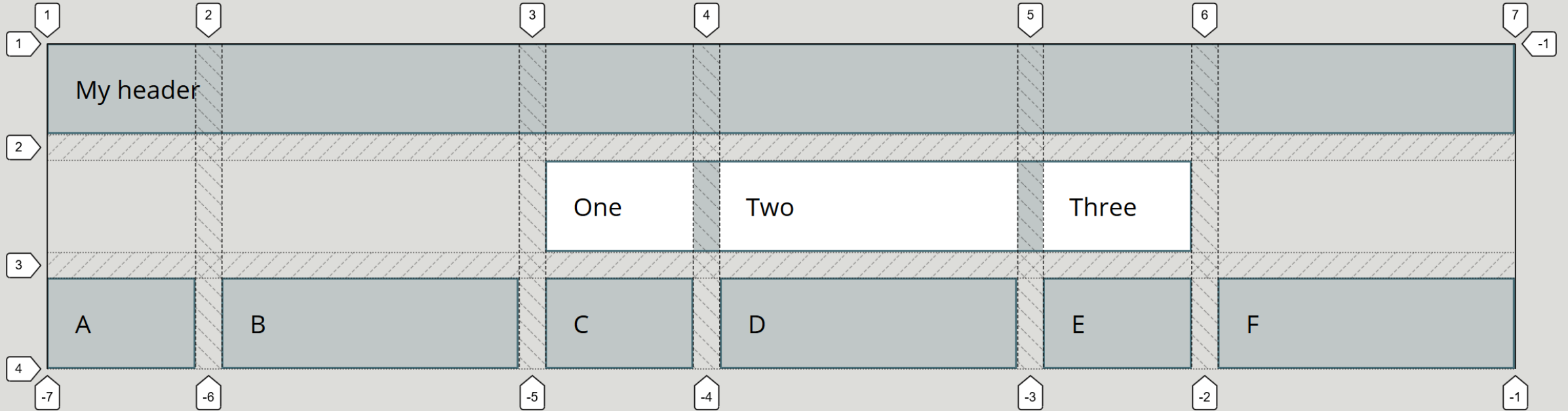One   Two   Three

A   B   C   D   E   F

grid-column: 2 / 5

My header

| One | Two | Three |

| A | B | C | D | E | F |

grid-column: 3 / 6

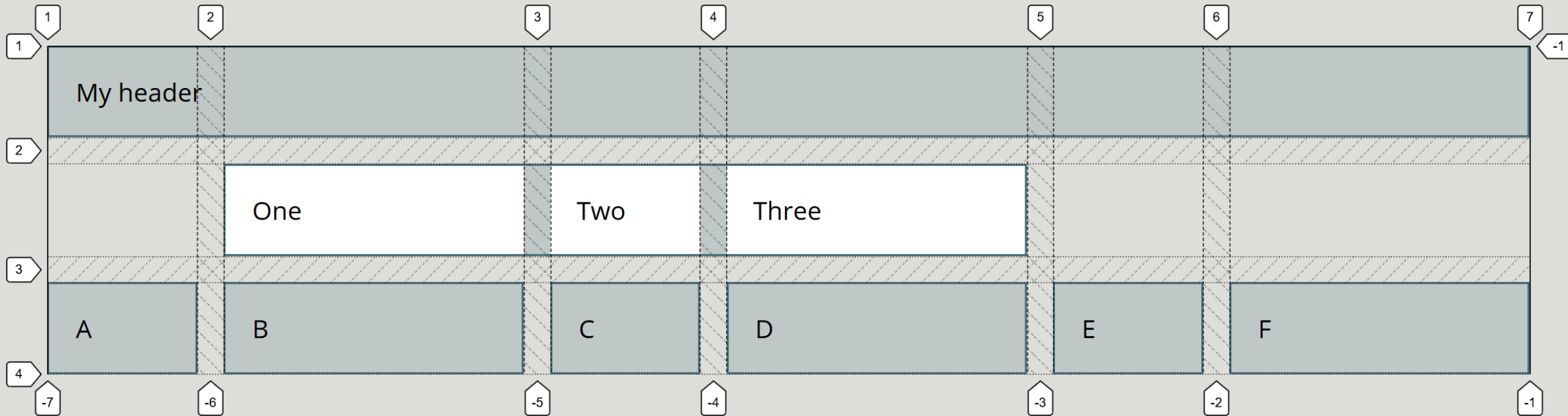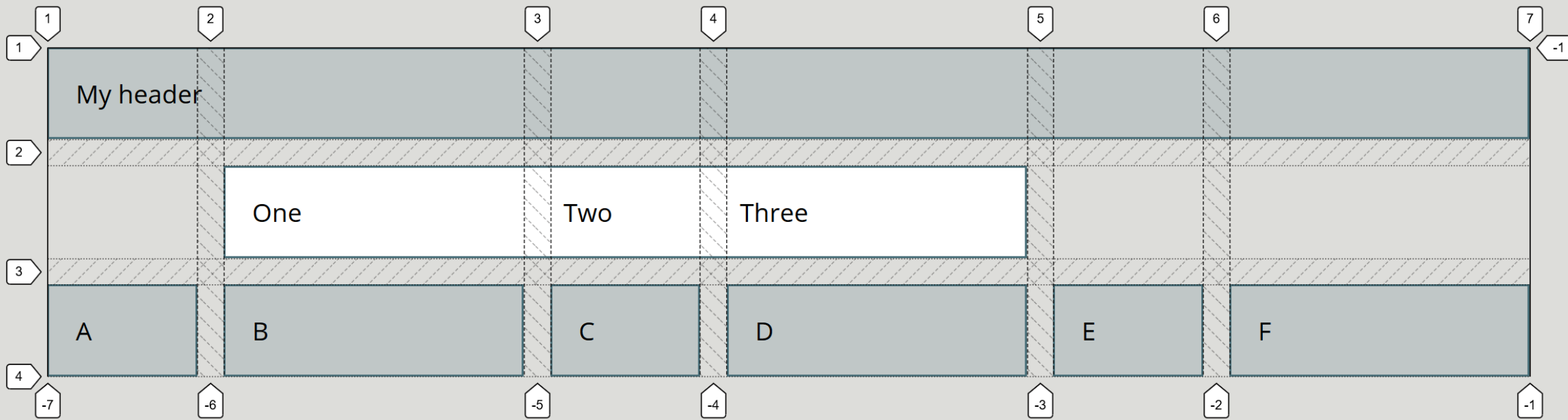My header

One   Two   Three

A   B   C   D   E   F

grid-template-columns: subgrid

To use subgrid

First create a grid formatting context with display: grid. Then opt in columns or rows with the subgrid value.

My header

One Two Three

A B C D E F

My header

One   Two   Three

A   B   C   D   E   F

subgrid

The subgrid must be participating in grid layout and a grid container itself.

Putting it all together ...

A demo.

Thank you!

@rachelandrew