

Graduating to Grid

An Event Apart Boston 2018
@rachelandrew

And there was great rejoicing.

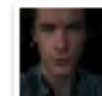
8,737

votes

Vote

Update CSS Grid

Update the IE implementation of Grid Layout to match the current spec. The current implementation from the IE 10 days is outdated.



mnsth shared this idea · October 02, 2014 · [Flag idea as inappropriate...](#)

CSS Grid Layout - CR

Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors. Includes support for all `grid-*` properties and the `fr` unit.

Usage

Global

unprefixed:

% of all users

84.14% + 3.42% = 87.56%

84.14%

Current alignedUsage relativeDate relativeShow all

IE	Edge*	Firefox	Chrome	Safari	Opera	iOS Safari*	Opera Mini*	Android Browser*	Opera Mobile*	Chrome for Android	Firefox for Android	Samsung Internet
			63									
		58	64									
211	16	59	65	11	50	11.2	all	62	37	64	57	6.2
	17	60	66	11.1	51	11.3						
		61	67	TP	52							
			68									

What's it like being in the middle of a launch of a big new CSS feature?

It involved a lot of email.

Grid is:

- **Exciting** - a real game changer
- **Confusing** - it doesn't seem to do what I thought it would
- **Scary** - there are so many new properties to learn!

“Q. How do you feel when you see a new CSS feature announced?”

–Me, in a survey question

“Excited but also worried about falling behind.”

–Survey response

“Excited, until I share it with colleagues and
they pick it apart”

–Survey response

“Oh no, a new way to have inconsistencies between web browsers.”

–Survey response

“Tired.”

–Survey response

I can't tell you what to do.

**I can help you develop the skills to
make those decisions yourself.**

I want **you** to be the amazing CSS layout person on your team.

You need to **understand** CSS.

Understanding CSS is not about learning every property and value by heart.

(my main skill is “can use a search engine”)

Core ideas that help CSS layout make **sense**.

Cascading Style Sheets

Inheritance - which properties will inherit values from their parent.

Specificity - which rule wins when two things could apply to the same element.



Cascade and inheritance

See also

Complete beginners start here!

▶ Getting started with the Web

HTML — Structuring the Web

▶ Introduction to HTML

▶ Multimedia and embedding

▶ HTML tables

CSS — Styling the Web

▼ Introduction to CSS

Introduction to CSS overview

How CSS works

CSS syntax

Selectors introduction

Simple selectors

Attribute selectors

Pseudo-classes and pseudo-elements

Combinators and multiple selectors

CSS values and units

Cascade and inheritance

The box model

Debugging CSS

▶ In This Article

← Previous

↑ Overview: Introduction to CSS

Next →

In a previous article, we got into the [various CSS selectors](#). At some point in your work, you'll find yourself in the situation where multiple CSS rules will have selectors matching the same element. In such cases, which CSS rule "wins", and ends up being the one that is finally applied to the element? This is controlled by a mechanism called the Cascade; this is also related to inheritance (elements will take some property values from their parents, but not others.) In this article we will define what the cascade is, what specificity is, what importance is, and how properties inherit from different rules.

Prerequisites:

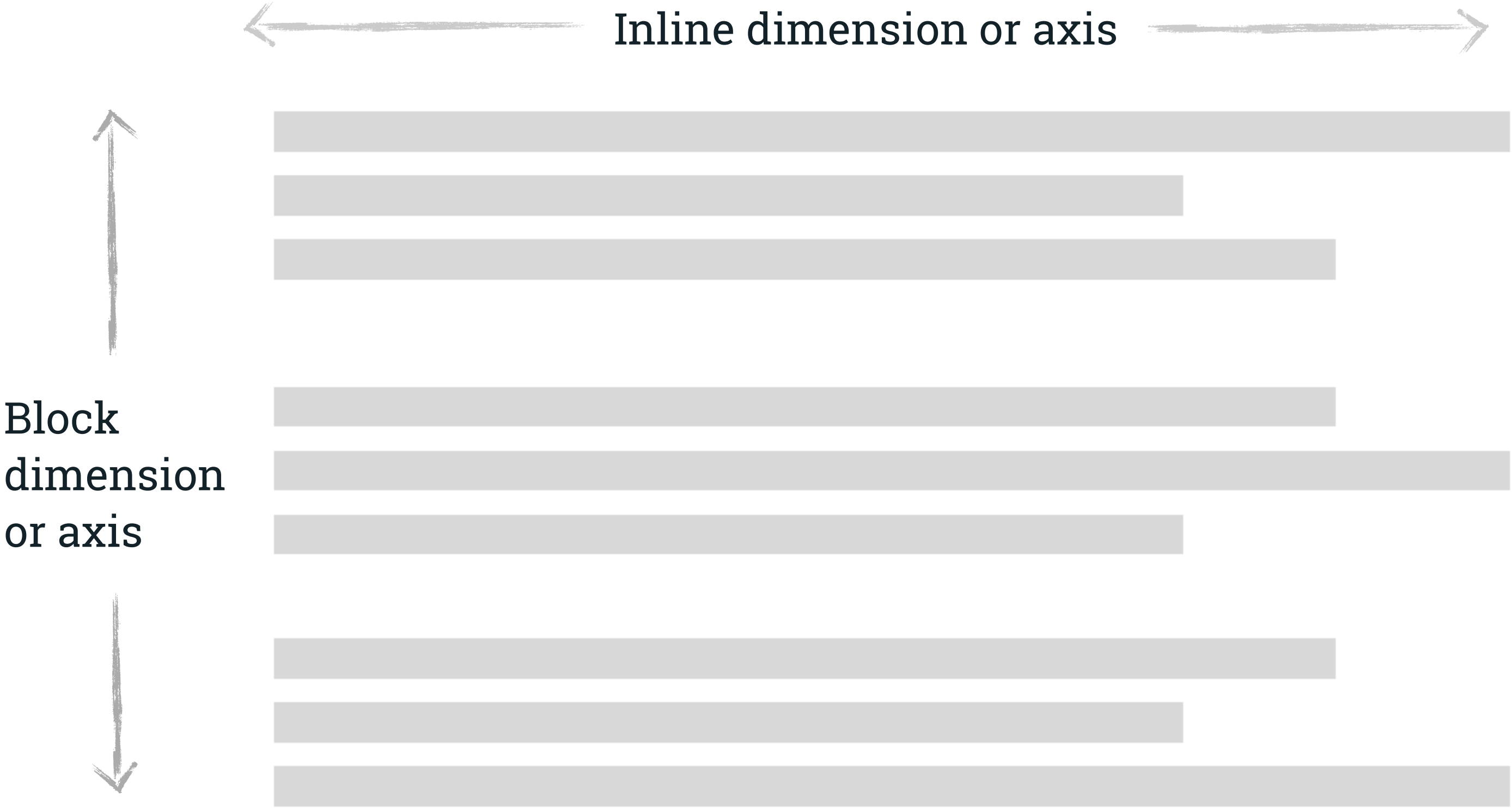
Basic computer literacy, [basic software installed](#), basic knowledge of [working with files](#), HTML basics (study [Introduction to HTML](#)), and an idea of How CSS works (study the previous articles in [this module](#).)

Objective:

To learn about the cascade and specificity, and how

Block and inline dimensions

Horizontal Writing Mode



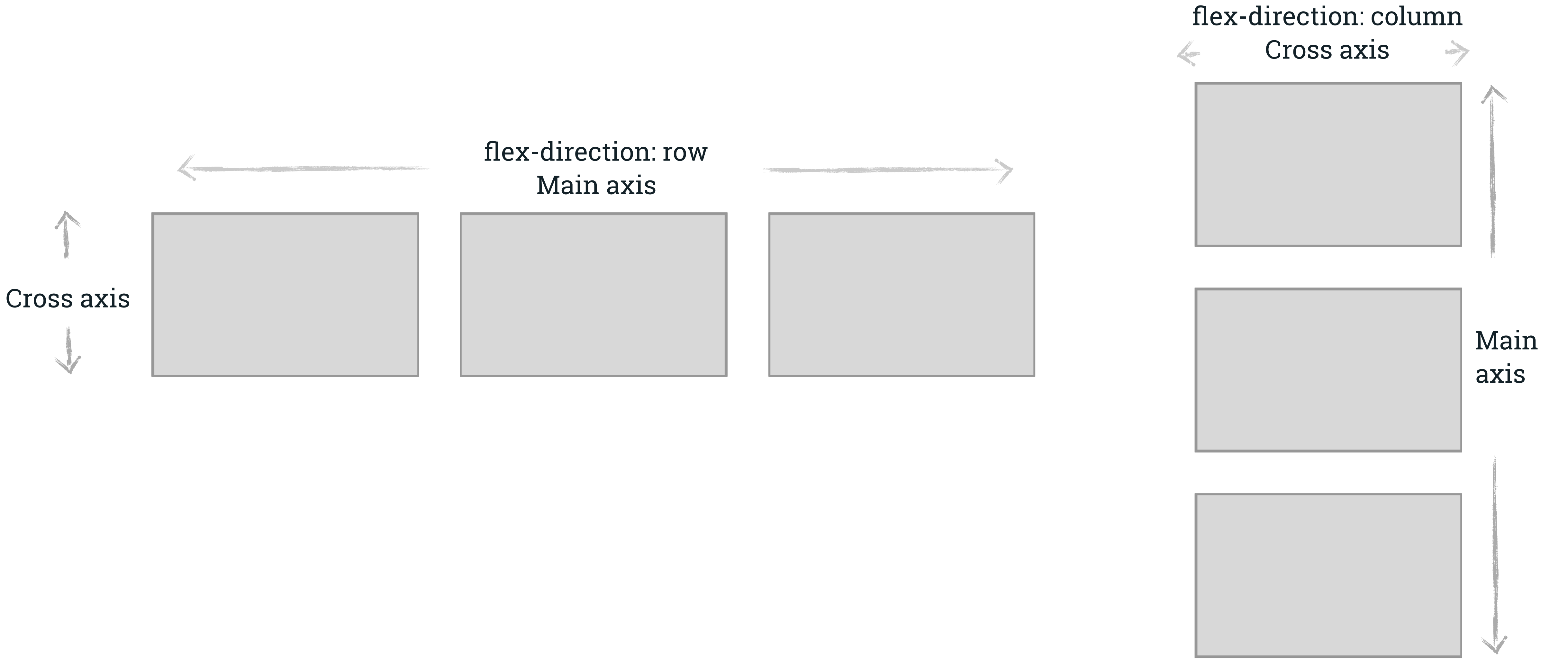
Vertical Writing Mode



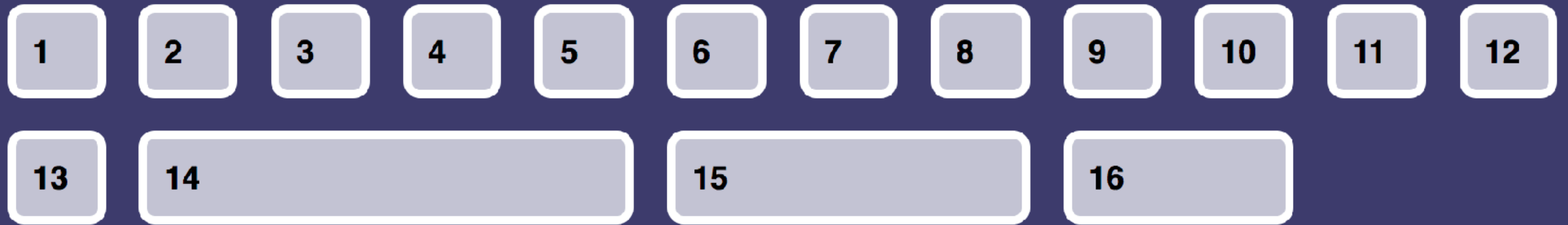
Grid Layout in Horizontal Writing Mode



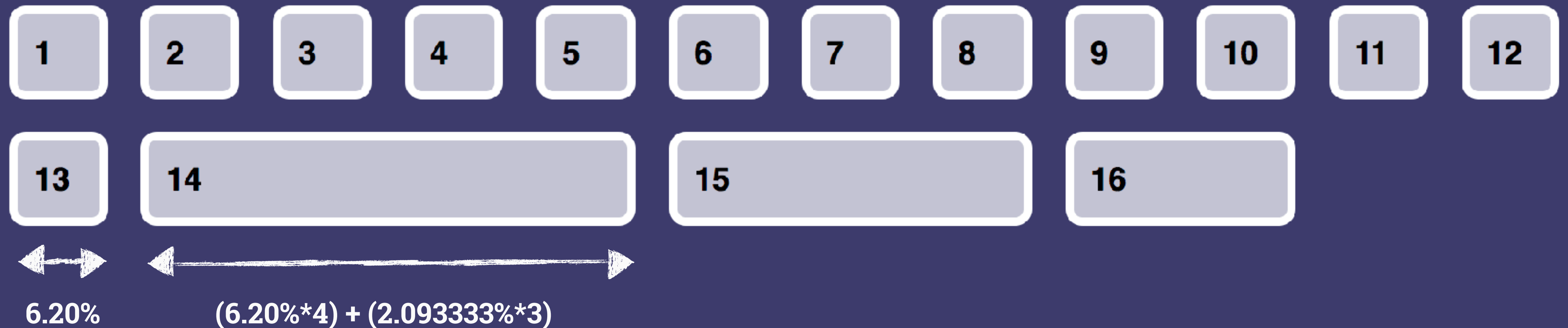
Flex Layout in Horizontal Writing Mode



Sizing Matters



row wrapper




```
.col {  
  margin-bottom: 1em;  
  margin-left: 2.093333%;  
  width: 6.20%;  
  float: left;  
}
```

Percentages!

```
.row::after {  
  content: "";  
  display: block;  
  clear: both;  
}
```

Clearing!

```
.col.span2 { width: calc((6.20%*2) + 2.093333%); }
```

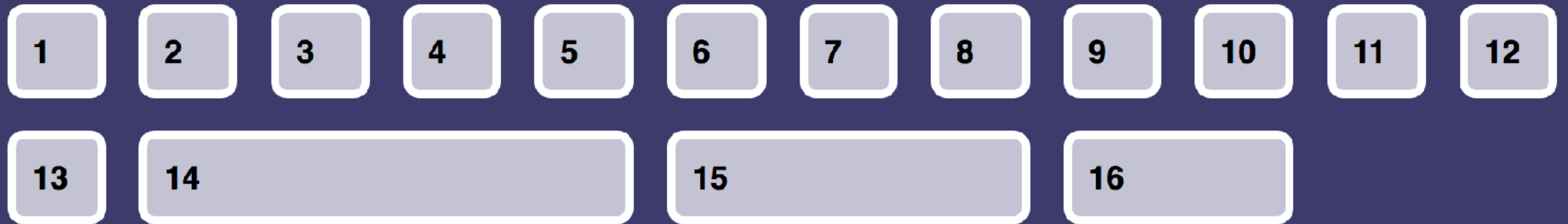
```
.col.span3 { width: calc((6.20%*3) + (2.093333%*2)); }
```

```
.col.span4 { width: calc((6.20%*4) + (2.093333%*3)); }
```

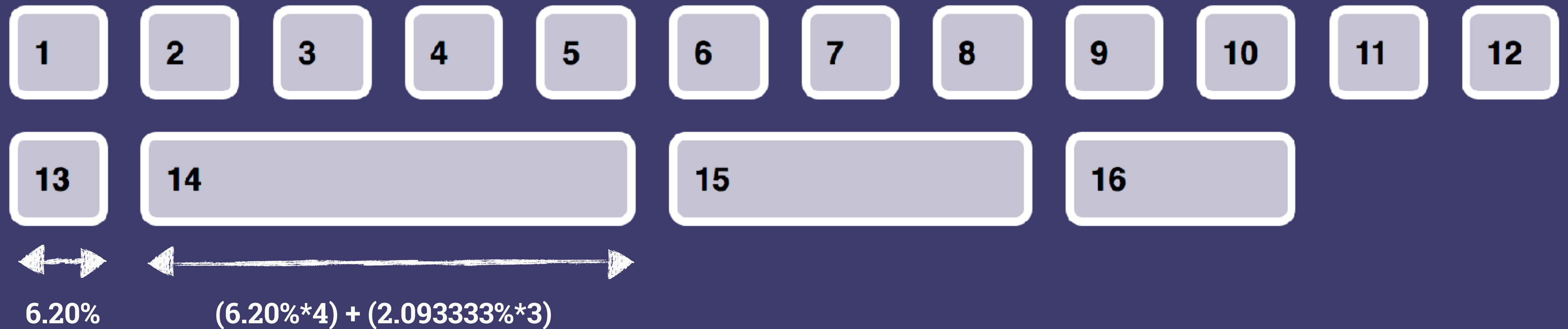
Fun math!

Percentages

- Ugly
- Easy to understand
- If they total more than 100% bad things happen.
- Can be converted from an ideal pixel size using a straightforward calculation.



Row wrapper as flex container



```
.wrapper .row {  
  display: flex;  
  flex-wrap: wrap;  
}
```

```
.col {  
  padding: 10px;  
  margin-bottom: 1em;  
  margin-left: 2.093333%;  
  width: 6.20%;  
  flex: 0 0 auto;  
}
```

Percentages!

Inflexible flex items!

Fun math!

```
.col.span2 { width: calc((6.20%*2) + 2.093333%); }
```

```
.col.span3 { width: calc((6.20%*3) + (2.093333%*2)); }
```

```
.col.span4 { width: calc((6.20%*4) + (2.093333%*3)); }
```

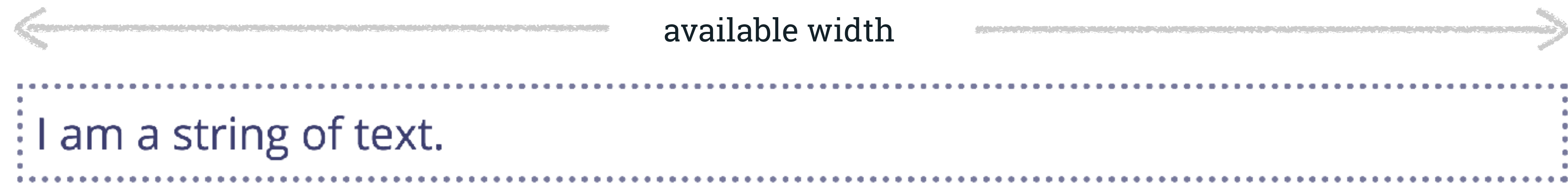

Past layout methods create the **appearance** of a grid, by lining things up.

CSS Intrinsic and Extrinsic Sizing

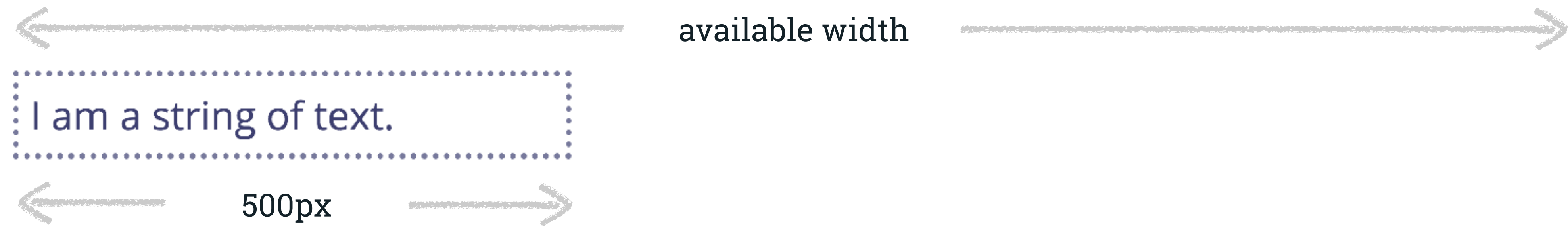
<https://drafts.csswg.org/css-sizing-3/>

```
/* html */
<div class="box">
  I am a string of text.
</div>

/* css */
.box {
  padding: 10px;
  border: 5px dotted rgba(255,255,255,.7);
  margin-bottom: 2em;
}
```

```
.box {  
  width: 500px;  
}
```



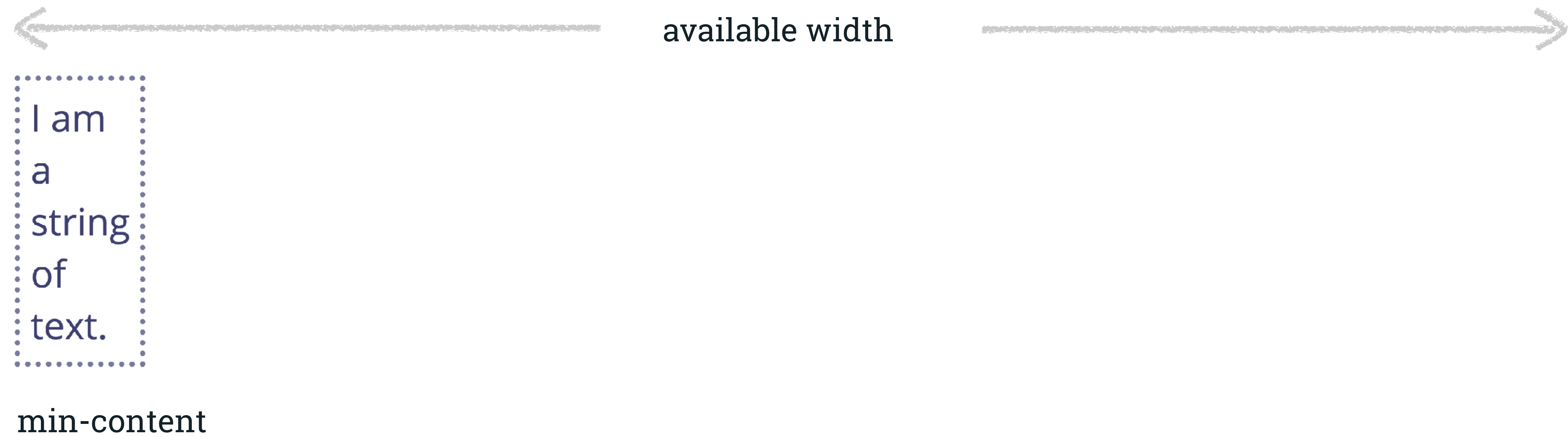
max content

I am a string of text.

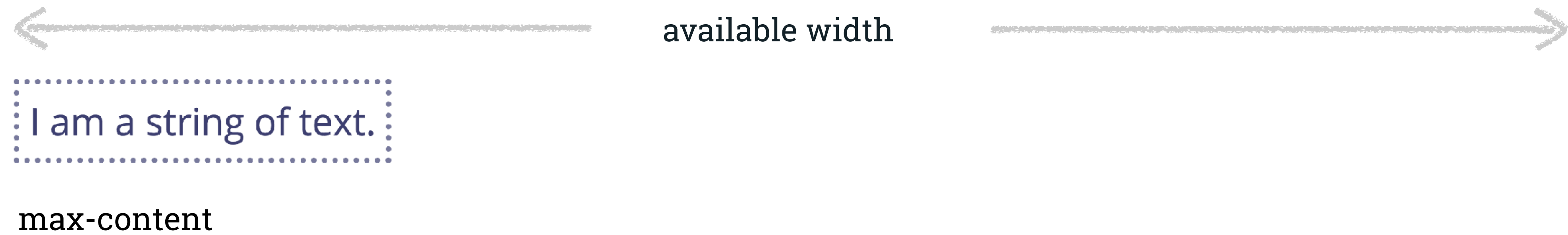
min content

I am
a
string
of
text.

```
.box {  
  width: min-content;  
}
```

```
.box {  
  width: max-content;  
}
```



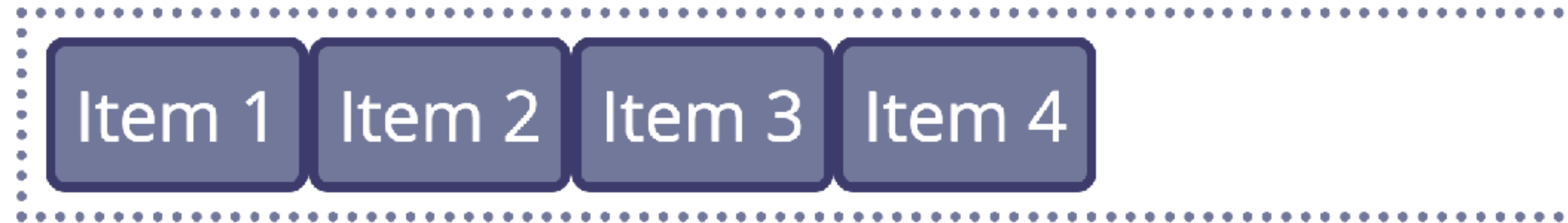

```
<div class="fixed-width">  
  <div class="box">  
    I am a longer string of text and max-content  
will cause me to overflow.  
  </div>  
</div>
```

```
.fixed-width {  
  width: 20em;  
  border: 5px solid rgb(255,255,255);  
  margin-bottom: 2em;  
}  
  
.box {  
  width: max-content;  
}
```

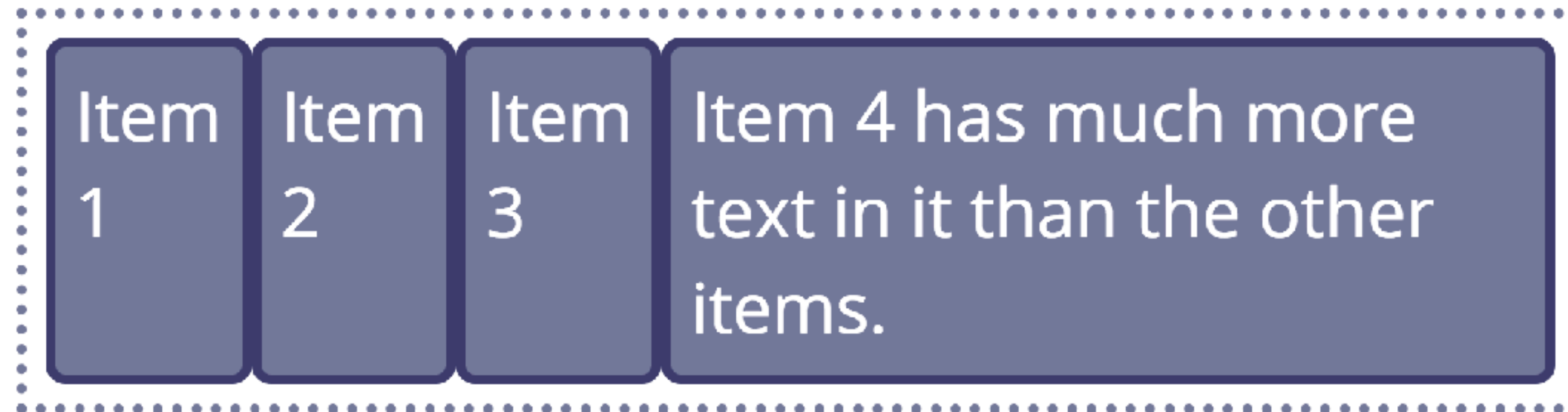



```
<div class="box">  
  <div>Item 1</div>  
  <div>Item 2</div>  
  <div>Item 3</div>  
  <div>Item 4</div>  
</div>
```

Items start by trying to display at max-content size.



Space is reduced according to the flex-basis. In this case the size of the content.

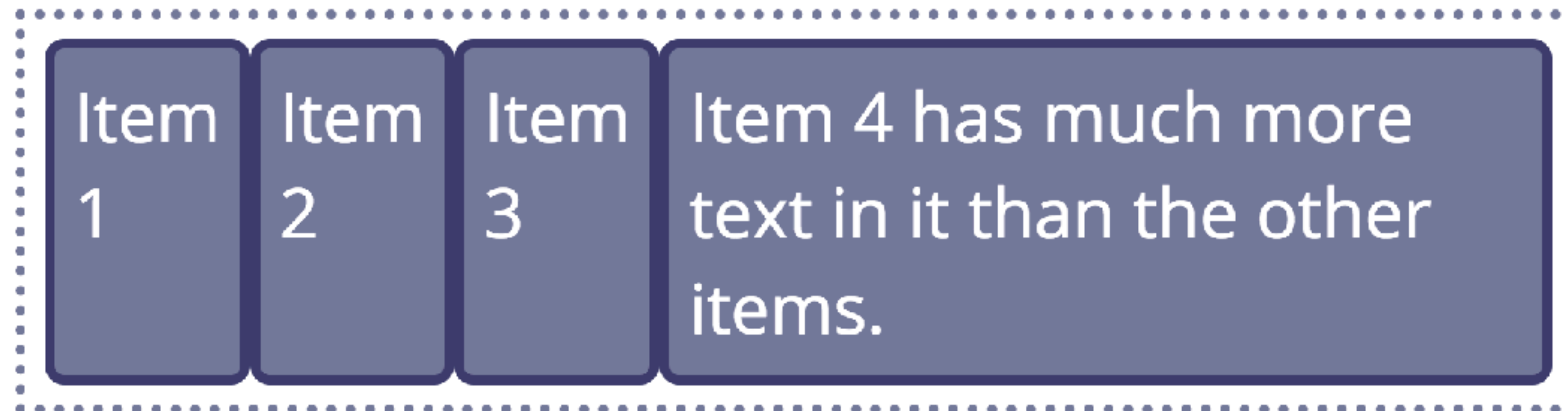



```
flex: 1 1 auto;
```

Items can grow and shrink so stretch to fill the container.



With no extra space, items shrink as before.

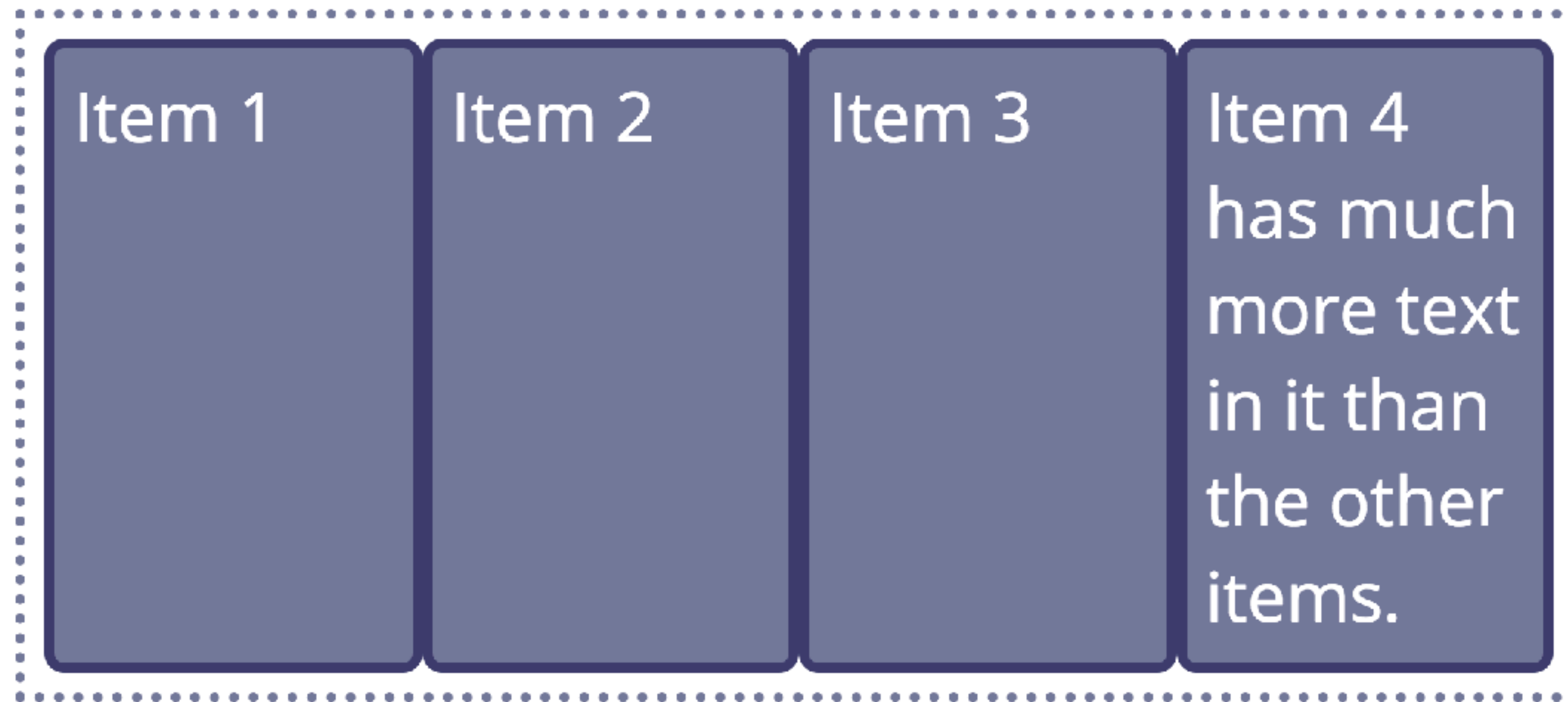


```
flex: 1 1 0;
```

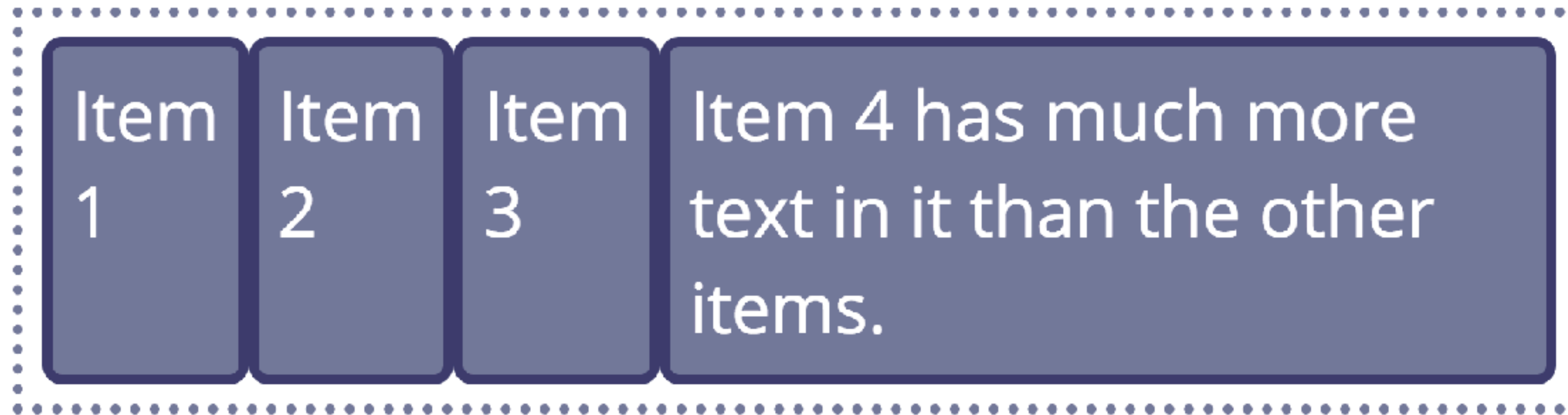
Items can grow and shrink so stretch to fill the container.



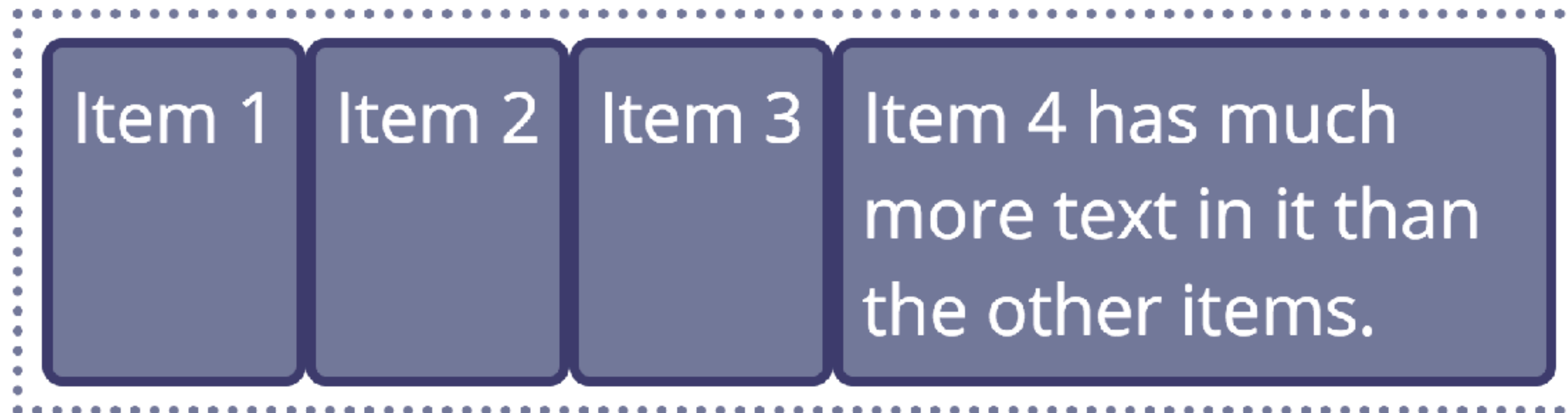
With a flex-basis of 0 space is distributed from 0, making equal columns.



Flex items at min-content size



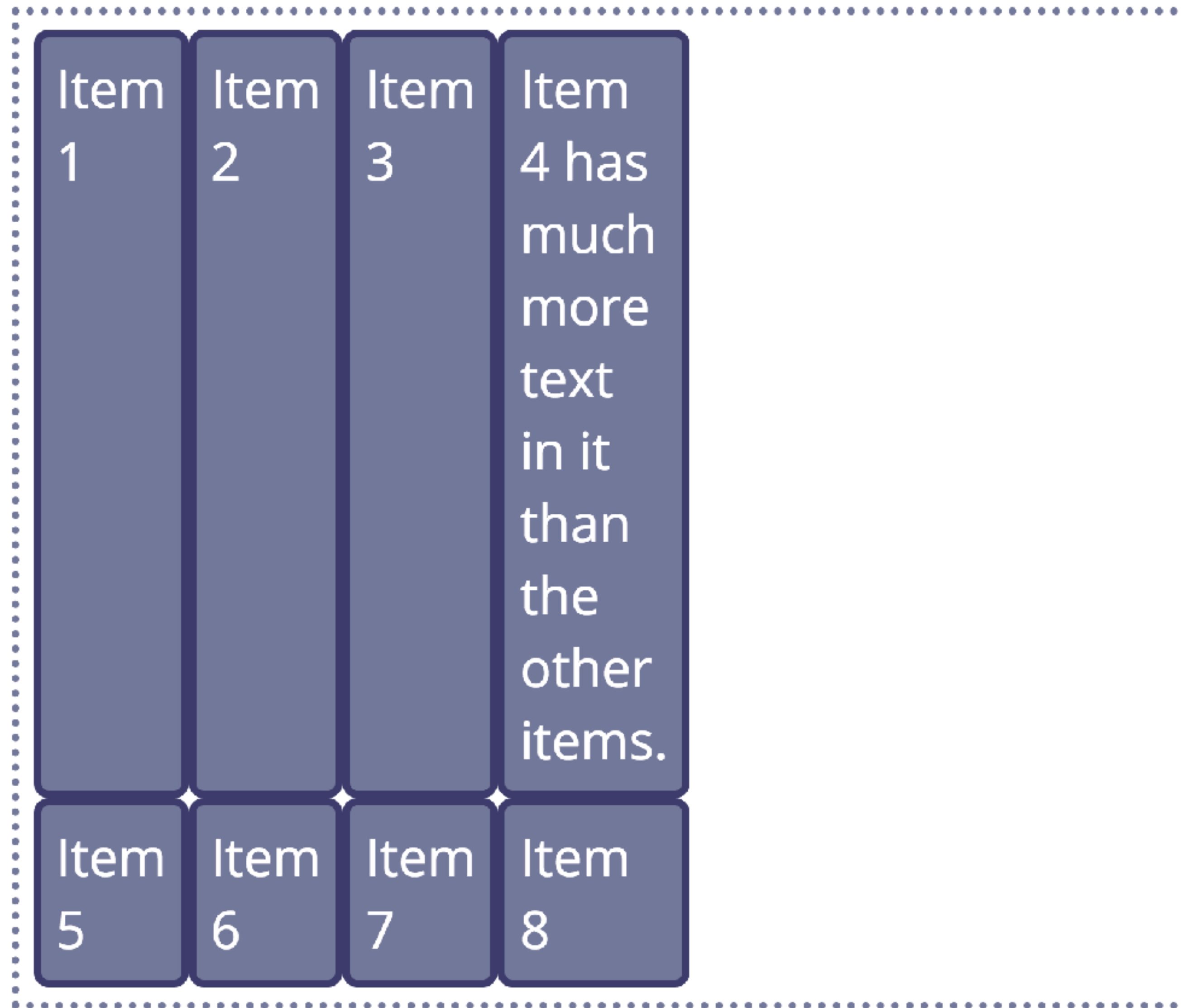
Grid items at max-content size



Flexbox is starting from **max-content** and taking space away. Grid starting at **min-content** and adding space.

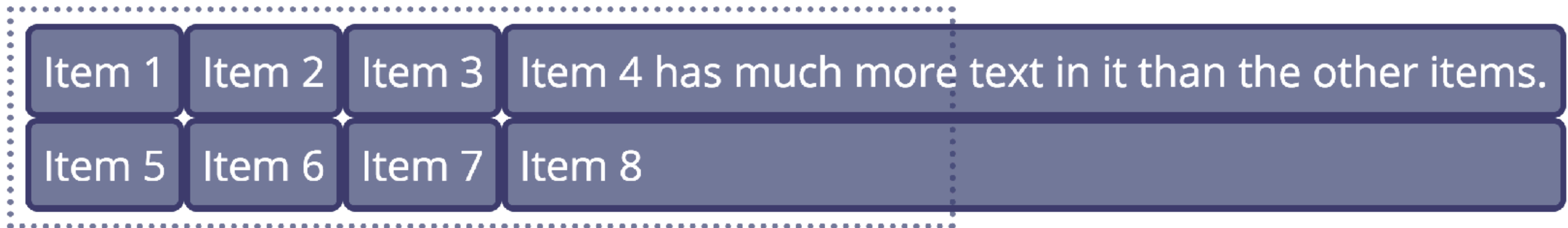
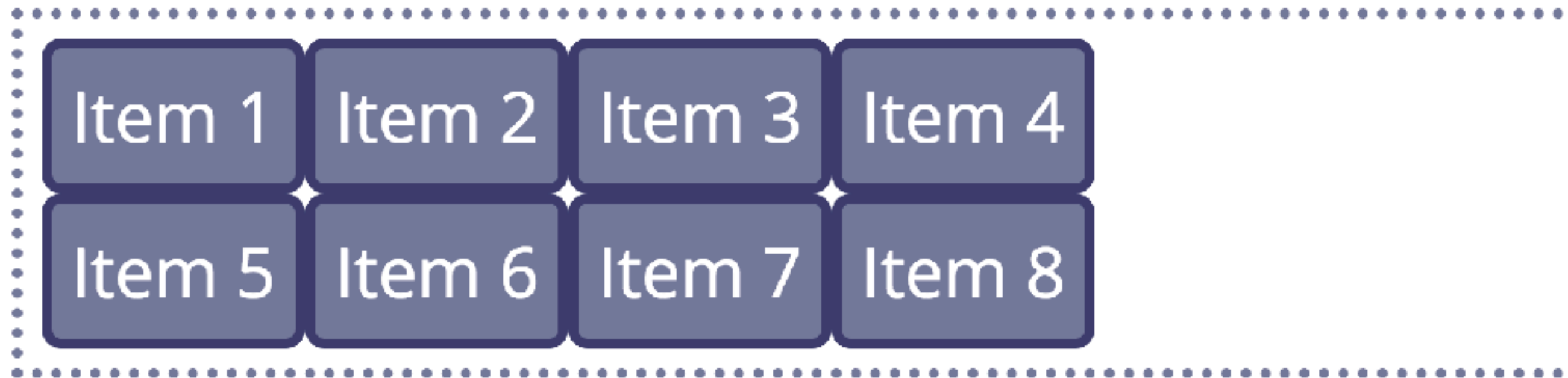
```
.grid {  
  display: grid;  
  grid-template-columns: repeat(4, min-content);  
}
```

min-content



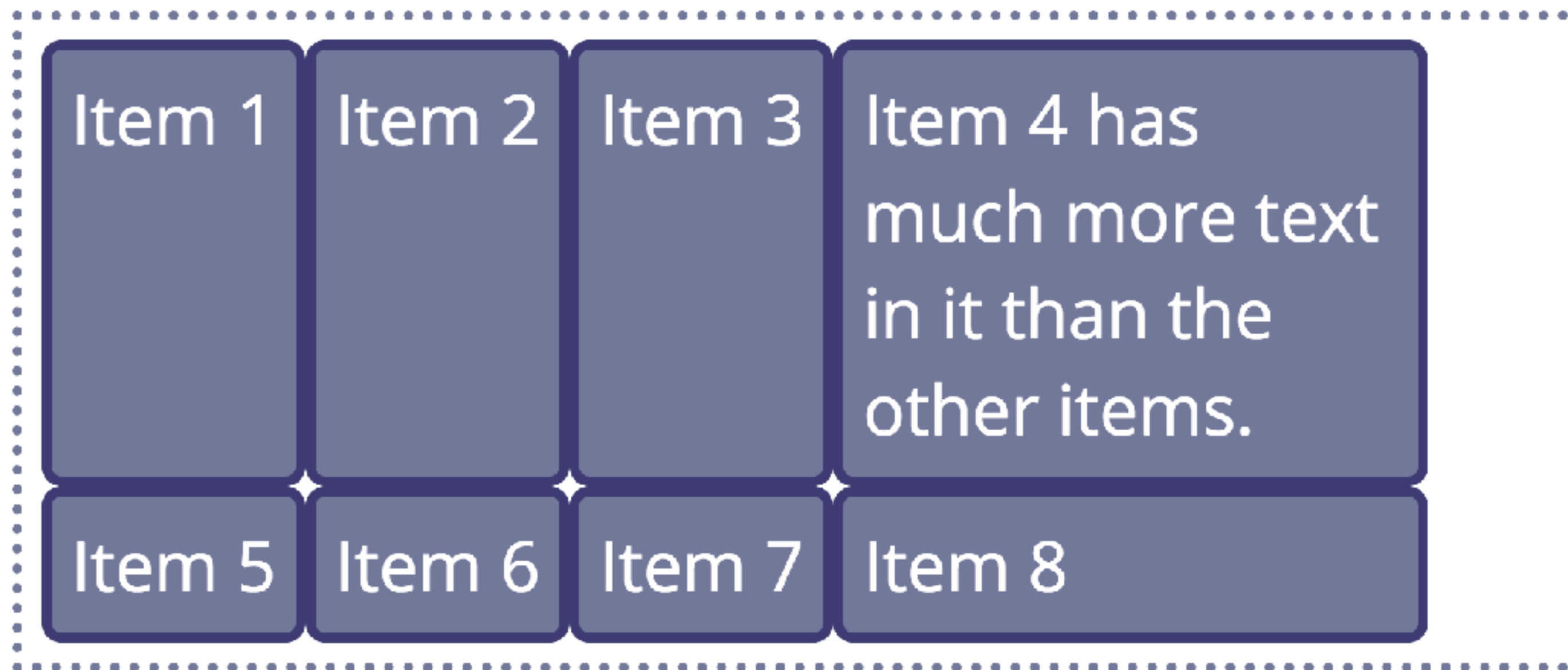

```
.grid {  
  display: grid;  
  grid-template-columns: repeat(4, max-content);  
}
```

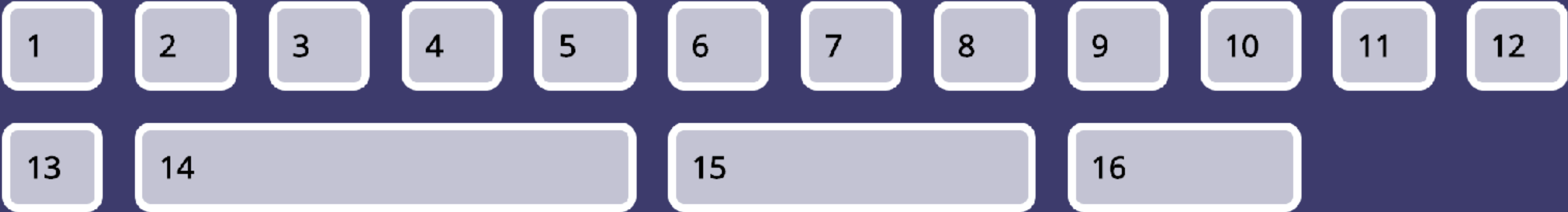
max-content

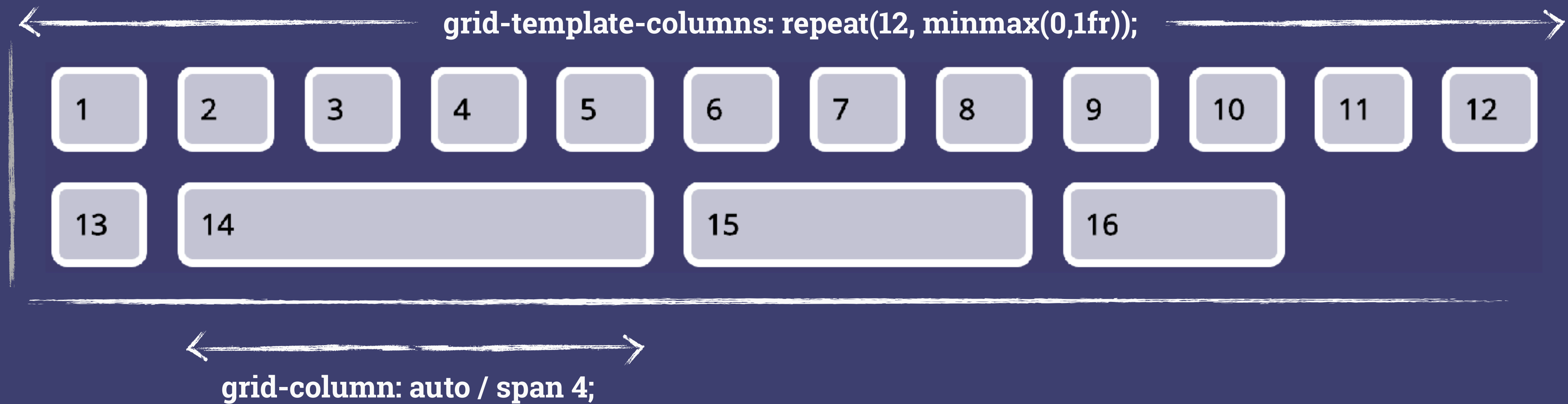



```
.grid {  
  display: grid;  
  grid-template-columns: repeat(4, fit-content(15ch));  
}
```

fit-content(15ch)







```
.wrapper {  
  display: grid;  
  grid-template-columns: repeat(12, minmax(0,1fr));  
  grid-gap: 20px;  
}  
  
.col.span2 {  
  grid-column: auto / span 2;  
}  
  
.col.span3 {  
  grid-column: auto / span 3;  
}  
  
.col.span4 {  
  grid-column: auto / span 4;  
}
```

```
.wrapper {  
  display: grid;  
  grid-template-columns: repeat(12, minmax(0,1fr));  
  grid-column-gap: 2.093333%;  
  grid-row-gap: 20px;  
}  
  
.col.span2 {  
  grid-column: auto / span 2;  
}  
  
.col.span3 {  
  grid-column: auto / span 3;  
}  
  
.col.span4 {  
  grid-column: auto / span 4;  
}
```


“In the end, we discovered that a column-based grid wasn’t actually needed. Since Grid allows you to create a custom grid to match whatever layout you have, we didn’t need to force it into 12 columns. Instead, we created CSS Grid objects for some of the common layout patterns in the designs.”

–Rebuilding Slack.com



500 x 320

We now near the confines of the clouds, and dip swiftly into the thickest of them; we experience a decided chill, and hear the rustling of the collapsing balloon, which is now but one-third full, but cannot see it, so dense is the mass of vapour. One, two, three, or more minutes pass, and we are still in the cloud. How thick it must be, considering the rapidity of the descent!



120 x 120

We now near the confines of the clouds, and dip swiftly into the thickest of them; we experience a decided chill, and hear the rustling of the collapsing balloon, which is now but one-third full, but cannot see it, so dense is the mass of vapour. One, two, three, or more minutes pass, and we are still in the cloud. How thick it must be, considering the rapidity of the descent!

← 300px →



We now near the confines of the clouds, and dip swiftly into the thickest of them; we experience a decided chill, and hear the rustling of the collapsing balloon, which is now but one-third full, but cannot see it, so dense is the mass of vapour. One, two, three, or more minutes pass, and we are still in the cloud. How thick it must be, considering the rapidity of the descent!

← 120px →



We now near the confines of the clouds, and dip swiftly into the thickest of them; we experience a decided chill, and hear the rustling of the collapsing balloon, which is now but one-third full, but cannot see it, so dense is the mass of vapour. One, two, three, or more minutes pass, and we are still in the cloud. How thick it must be, considering the rapidity of the descent!

```
.media {  
  display: grid;  
  grid-template-columns: fit-content(300px) 1fr;  
  grid-gap: 20px;  
}
```




Chapter Six:
Ascents by Messrs
Glaisher and
Coxwell



We now come to that point in our subject where it is appropriate to give more detailed and graphic accounts of the recent doings of aeronauts. An extremely interesting description of a scientific balloon ascent is given by the celebrated aeronaut, Mr Glaisher, in a pamphlet, from which we shall make a few extracts. (See Note 1.) His description is illustrative of the subject of ballooning, and contains the salient points of several ascents.

He asks us to imagine the balloon somewhat more than half inflated, eager for flight, with only one link connecting it with earth, namely, a rope attached to an instrument, called a liberating iron catch. When all the ballast, instruments, etcetera, were placed in the car, Mr Coxwell brought the balloon to a nice and even balance, so that the addition of twenty pounds would have prevented it from rising.

As the moment for departure drew near, friends became impatient, and every one anxiously watched the final arrangements, which were made by Mr Coxwell, on whom was laid the important duty of letting go. His hand was on the catch, his countenance was fixed, and his expression stern, as he gazed up into the heavens. He was waiting for the right moment, for the sky was partially cloudy, and it was necessary to wait until the balloon was midway between the cloud that had just passed and the next that was approaching, so that the aeronauts might have a clear sky, and be able

to see the earth they were about to quit for a time. Nor was this all; he knew that in every wind, however strong it might be, there are periods of calm. If he could start in one of these he would avoid much rotatory motion. The deciding, therefore, of the exact moment for making a fair start was not so easy a matter as one might suppose.

Some one at this critical time, with the characteristic eagerness of poor human nature to "put its finger in the pie," cried out "Now!" and another shouted "Pull!" but Mr Coxwell, regardless of every one, decided for himself; and, just when the wind lulled and the sun shone bright, and the balloon stood proudly erect, he pulled the trigger and they were free.

But they were more than free. They were suddenly in profound repose, for—however high the wind may be, however agitated the balloon, swaying to and fro with sudden and violent action, despite the efforts of many hands that endeavour to restrain it,—no sooner do aeronauts quit their hold of earth, than, in an instant, all agitation ceases and they are in perfect stillness, without any sense of motion whatever; and this freedom continues throughout the entire flight—except, indeed, when they sink so low as to come into contact with mother earth, when the serenity of their flight is terribly and violently interrupted, as shall be seen in the case of another balloon voyage hereafter to be described.



Chapter Six:
Ascents by Messrs
Glaisher and
Coxwell



We now come to that point in our subject where it is appropriate to give more detailed and graphic accounts of the recent doings of aeronauts. An extremely interesting description of a scientific balloon ascent is given by the celebrated aeronaut, Mr Glaisher, in a pamphlet, from which we shall make a few extracts. (See Note 1.) His description is illustrative of the subject of ballooning, and contains the salient points of several ascents.

He asks us to imagine the balloon somewhat more than half inflated, eager for flight, with only one link connecting it with earth, namely, a rope attached to an instrument, called a liberating iron catch. When all the ballast, instruments, etcetera, were placed in the car, Mr Coxwell brought the balloon to a nice and even balance, so that the addition of twenty pounds would have prevented it from rising.

As the moment for departure drew near, friends became impatient, and every one anxiously watched the final arrangements, which were made by Mr Coxwell, on whom was laid the important duty of letting go. His hand was on the catch, his countenance was fixed, and his expression stern, as he gazed up into the heavens. He was waiting for the right moment, for the sky was partially cloudy, and it was necessary to wait until the balloon was midway between the cloud that had just passed and the next that was approaching, so that the aeronauts might have a clear sky, and be able to see the earth they were about to quit for a time. Nor was this all; he knew that in every wind, however strong it might be, there are periods of calm. If he could start in one of these he would avoid much rotatory motion. The deciding, therefore, of the exact moment for making a fair start was not so easy a matter as one might suppose.

```
.panel {  
  display: grid;  
  grid-gap: 1px;  
  grid-template-columns: 1fr 1fr 3fr;  
  grid-template-rows:  
    minmax(100px, auto)  
    minmax(50px, auto)  
    minmax(250px, auto)  
    minmax(50px, auto)  
    minmax(150px, auto);  
}
```




Chapter Six: Ascents by Messrs Glaisher and Coxwell

We now come to that point in our subject where it is appropriate to give more detailed and graphic accounts of the recent doings of aeronauts. An extremely interesting description of a scientific balloon ascent is given by the celebrated aeronaut, Mr Glaisher, in a pamphlet, from which we shall make a few extracts. (See Note 1.) His description is illustrative of the subject of ballooning, and contains the salient points of several ascents.

He asks us to imagine the balloon somewhat more than half inflated, eager for flight, with only one link connecting it with earth, namely, a rope attached to an instrument, called a liberating iron catch. When all the ballast, instruments, etcetera, were placed in the car, Mr Coxwell brought the balloon to a nice and even balance, so that the addition of twenty pounds would have prevented it from rising.

As the moment for departure drew near, friends became impatient, and every one anxiously watched the final arrangements, which were made by Mr Coxwell, on whom was laid the important duty of letting go. His hand was on the catch, his countenance was fixed, and his expression stern, as he gazed up into the heavens. He was waiting for the right moment, for the sky was partially cloudy, and it was necessary to wait until the balloon was midway between the cloud that had just passed and the next that was approaching, so that the aeronauts might have a clear sky, and be able to see the earth they were about to quit for a time. Nor was this all; he knew that in every wind, however strong it might be, there are periods of calm. If he could start in one of these he would avoid much rotatory motion. The deciding, therefore, of the exact moment for making a fair start was not so easy a matter as one might suppose.



↑
Min 50px
Max auto
↓



Chapter Six: Ascents by Messrs Glaisher and Coxwell



We now come to that point in our subject where it is appropriate to give more detailed and graphic accounts of the recent doings of aeronauts. An extremely interesting description of a scientific balloon ascent is given by the celebrated aeronaut, Mr Glaisher, in a pamphlet, from which we shall make a few extracts. (See Note 1.) His description is illustrative of the subject of ballooning, and contains the salient points of several ascents.

He asks us to imagine the balloon somewhat more than half inflated, eager for flight, with only one link connecting it with earth, namely, a rope attached to an instrument, called a liberating iron catch. When all the ballast, instruments, etcetera, were placed in the car, Mr Coxwell brought the balloon to a nice and even balance, so that the addition of twenty pounds would have prevented it from rising.

As the moment for departure drew near, friends became impatient, and every one anxiously watched the final arrangements, which were made by Mr Coxwell, on whom was laid the important duty of letting go. His hand was on the catch, his countenance was fixed, and his expression stern, as he gazed up into the heavens. He was waiting for the right moment, for the sky was partially cloudy, and it was necessary to wait until the balloon was midway between the cloud that had just passed and the next that was approaching, so that the aeronauts might have a clear sky, and be able

to see the earth they were about to quit for a time. Nor was this all; he knew that in every wind, however strong it might be, there are periods of calm. If he could start in one of these he would avoid much rotatory motion. The deciding, therefore, of the exact moment for making a fair start was not so easy a matter as one might suppose.

Some one at this critical time, with the characteristic eagerness of poor human nature to "put its finger in the pie," cried out "Now!" and another shouted "Pull!" but Mr Coxwell, regardless of every one, decided for himself; and, just when the wind lulled and the sun shone bright, and the balloon stood proudly erect, he pulled the trigger and they were free.

But they were more than free. They were suddenly in profound repose, for—however high the wind may be, however agitated the balloon, swaying to and fro with sudden and violent action, despite the efforts of many hands that endeavour to restrain it,—no sooner do aeronauts quit their hold of earth, than, in an instant, all agitation ceases and they are in perfect stillness, without any sense of motion whatever; and this freedom continues throughout the entire flight—except, indeed, when they sink so low as to come into contact with mother earth, when the serenity of their flight is terribly and violently interrupted, as shall be seen in the case of another balloon voyage hereafter to be described.

↑
Min 50px
Max auto
↓

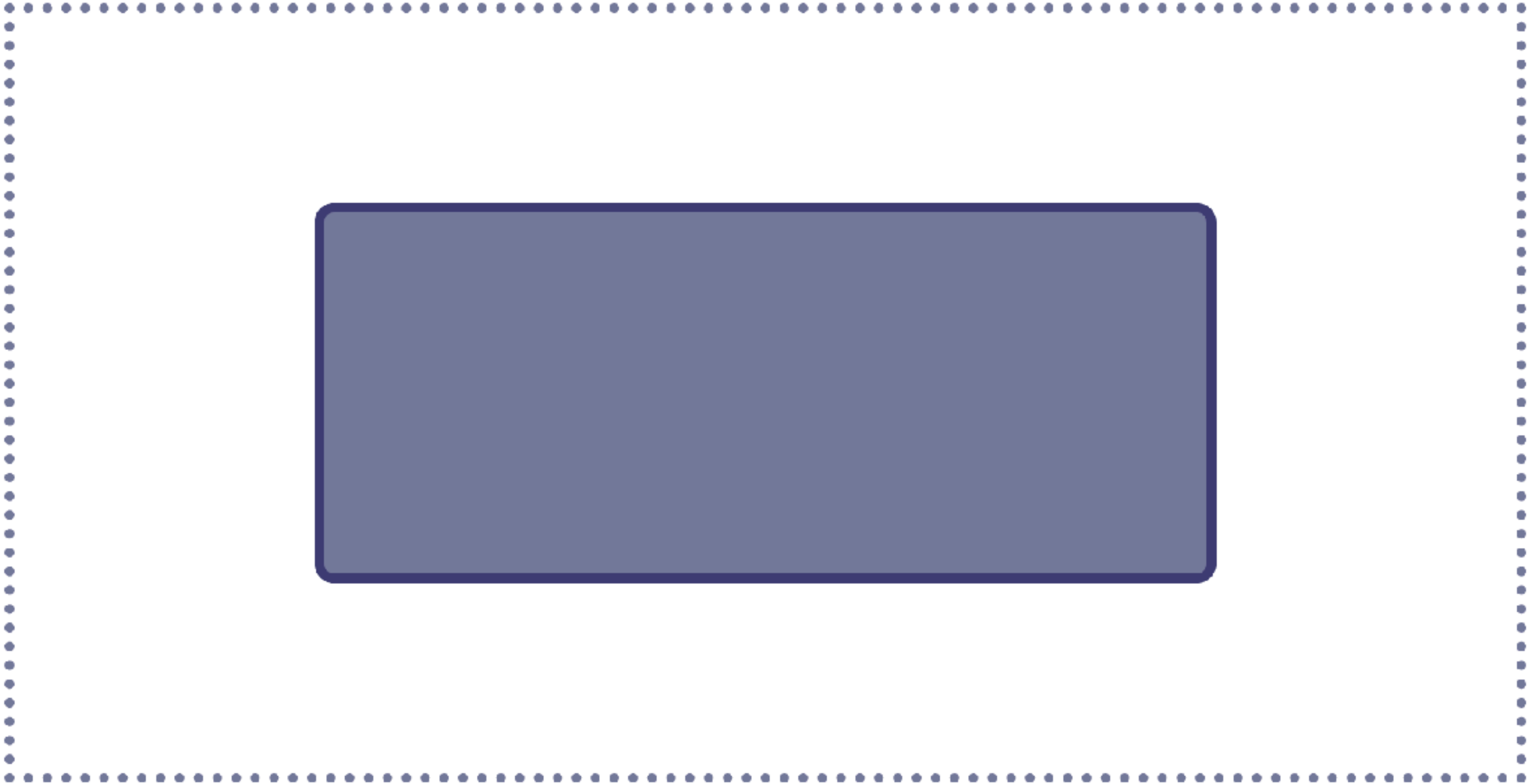
↑
Min 50px
Max auto
↓

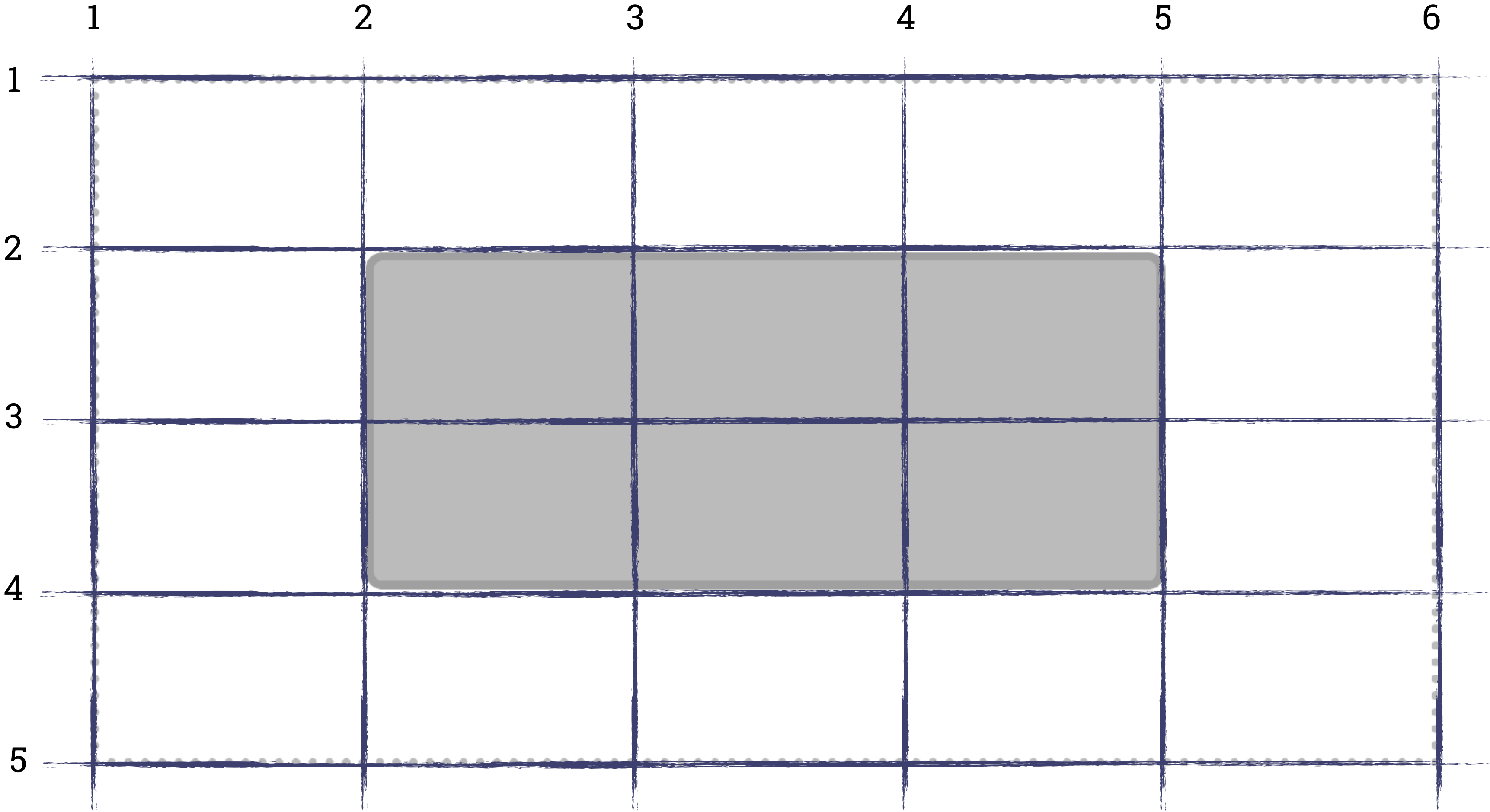
This is not exciting. But it will let you do exciting things.

Why so complicated?

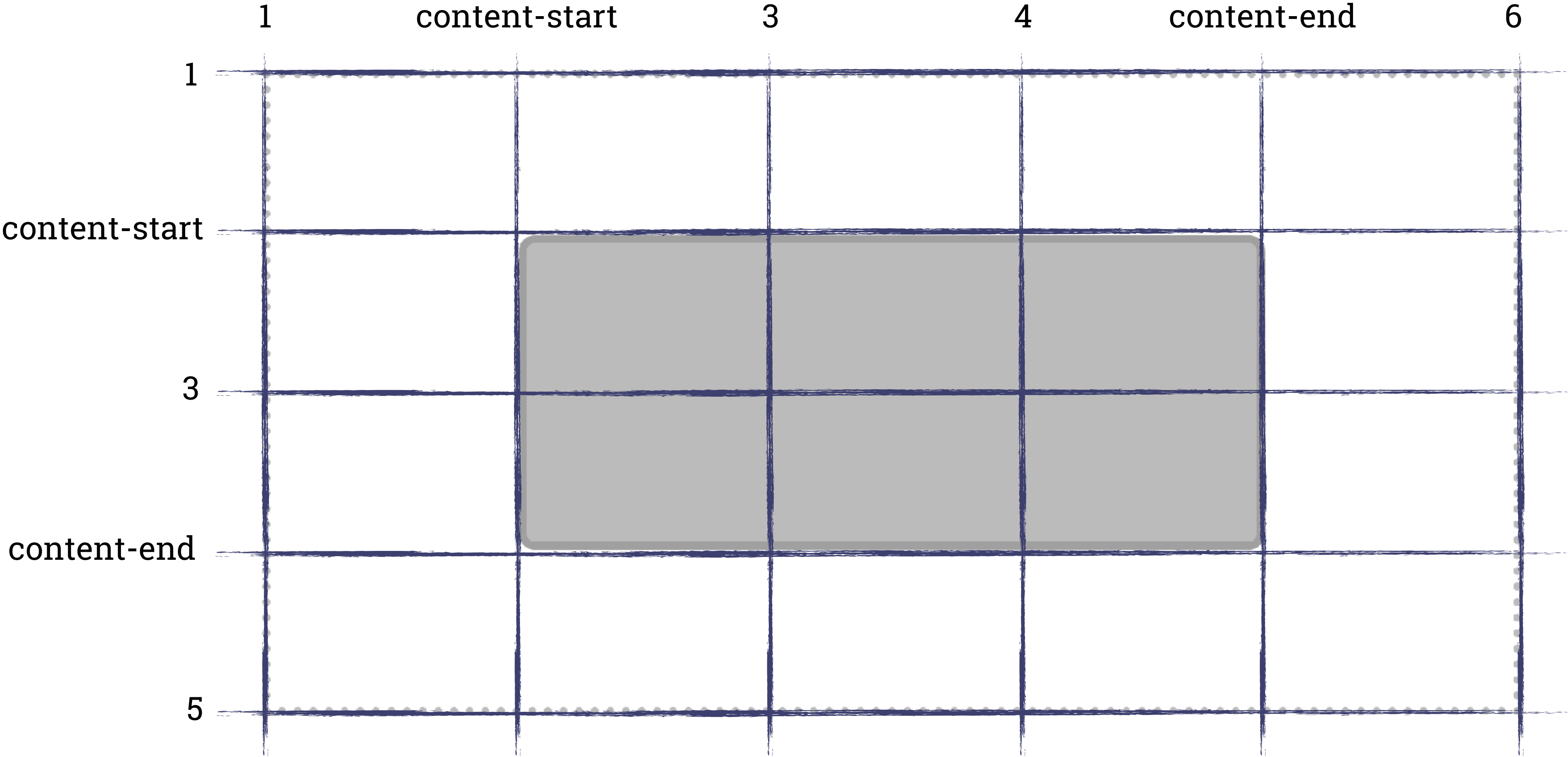
More capability & flexibility means
more to learn

It is all just **lines**.

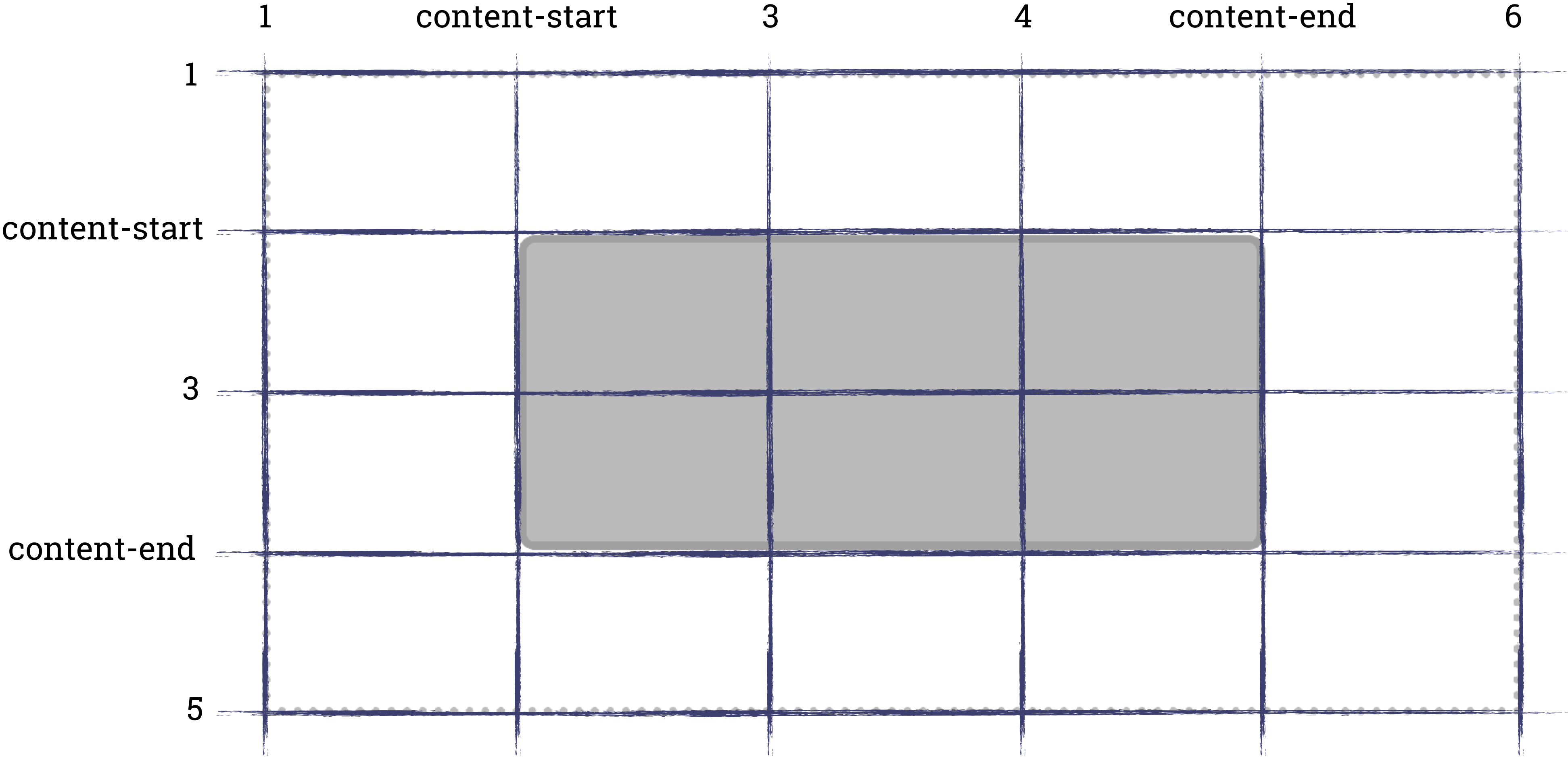


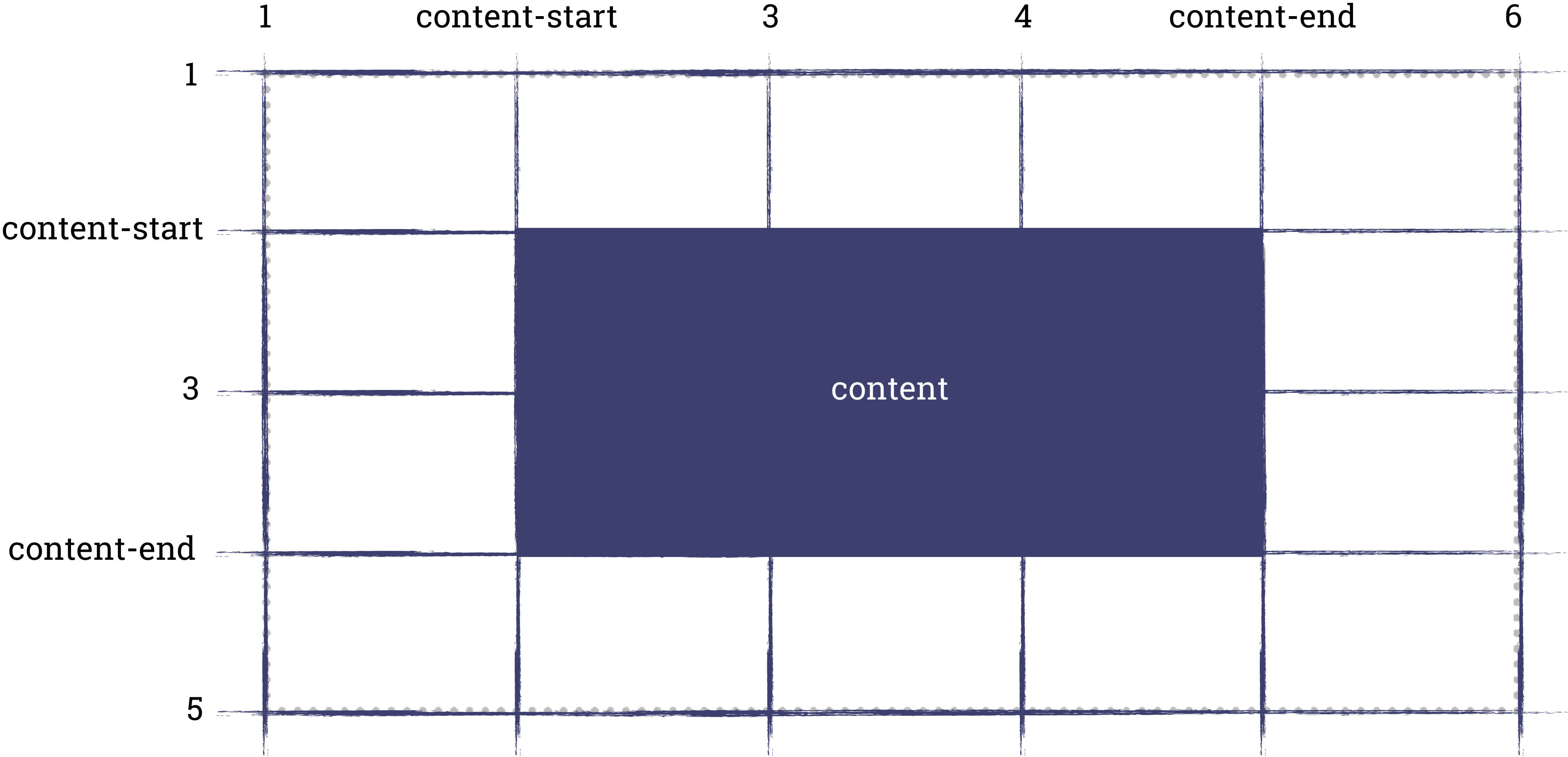


```
.grid {  
  display: grid;  
  grid-template-columns: 1fr 1fr 1fr 1fr 1fr;  
  grid-template-rows: 100px 100px 100px 100px;  
}  
  
.item {  
  grid-column: 2 / 5;  
  grid-row: 2 / 4;  
}
```




```
.grid {  
  display: grid;  
  grid-template-columns: 1fr [content-start] 1fr  
1fr 1fr [content-end] 1fr;  
  grid-template-rows: 100px [content-start] 100px  
100px [content-end] 100px;  
}  
  
.item {  
  grid-column: content-start / content-end;  
  grid-row: content-start / content-end;  
}
```



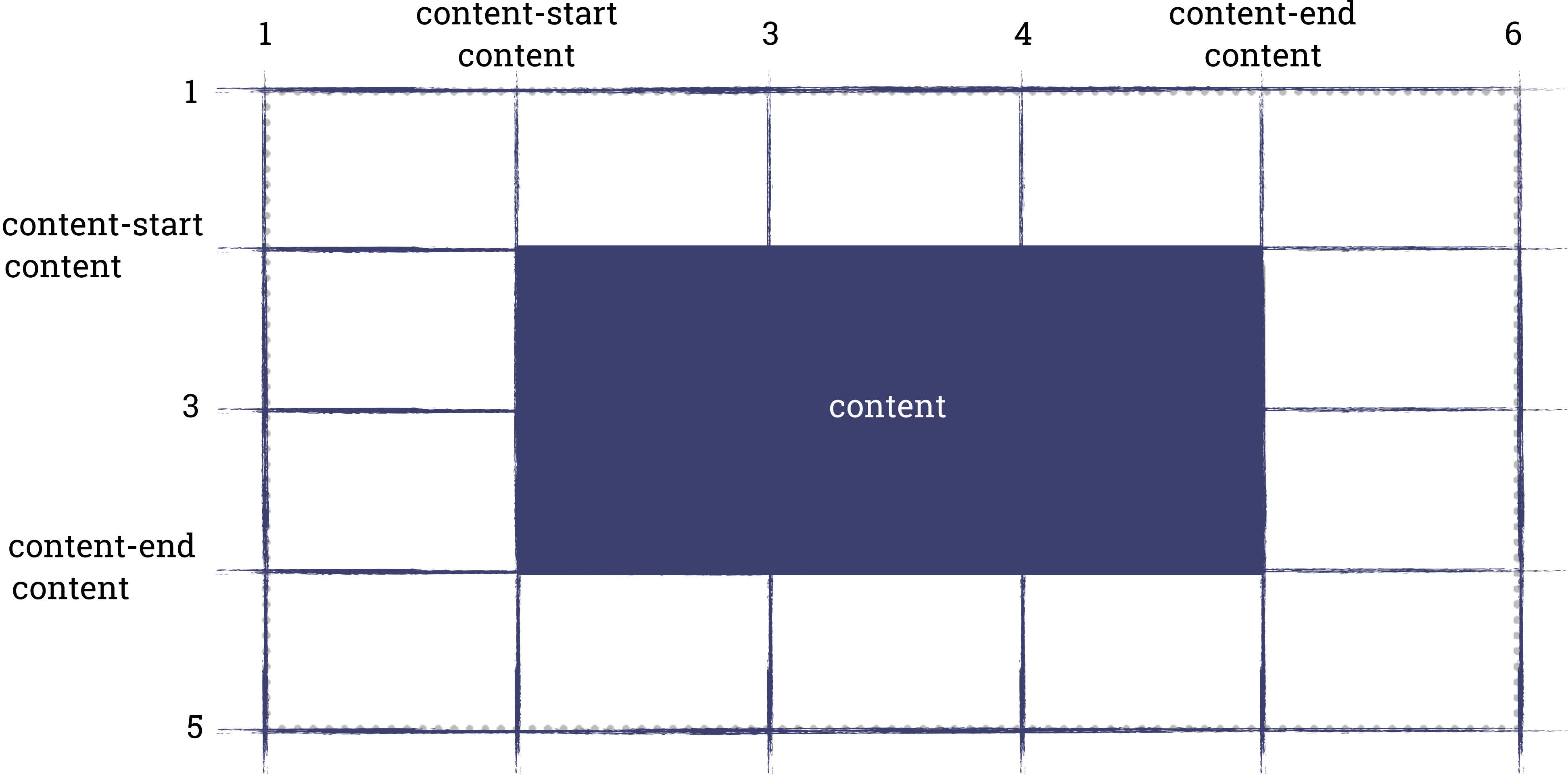


```
.grid {  
  display: grid;  
  grid-template-columns: 1fr [content-start] 1fr  
1fr 1fr [content-end] 1fr;  
  grid-template-rows: 100px [content-start] 100px  
100px [content-end] 100px;  
}  
  
.item {  
  grid-area: content;  
}
```

```
.grid {  
  display: grid;  
  grid-template-columns: 1fr [content-start] 1fr  
1fr 1fr [content-end] 1fr;  
  grid-template-rows: 100px [content-start] 100px  
100px [content-end] 100px;  
}  
  
.item {  
  grid-area: content / content / content / content;  
}
```



```
grid-area: content / content / content / content;  
           grid-row-start  grid-column-start  grid-row-end  grid-column-end
```



```
grid-area: content / content / content / content;  
           grid-row-start  grid-column-start  grid-row-end  grid-column-end
```



```
grid-area: content / content / content ;  
           grid-row-start  grid-column-start  grid-row-end
```

- grid-column-end is set to the value used for grid-column-start, which is 'content'.

```
grid-area: content / content ;  
           grid-row-start  grid-column-start
```

- grid-row-end is set to the value used for grid-column-start, which is 'content'.
- grid-column-end is set to the value used for grid-column-start, which is 'content'.


```
grid-area: content ;  
grid-row-start
```

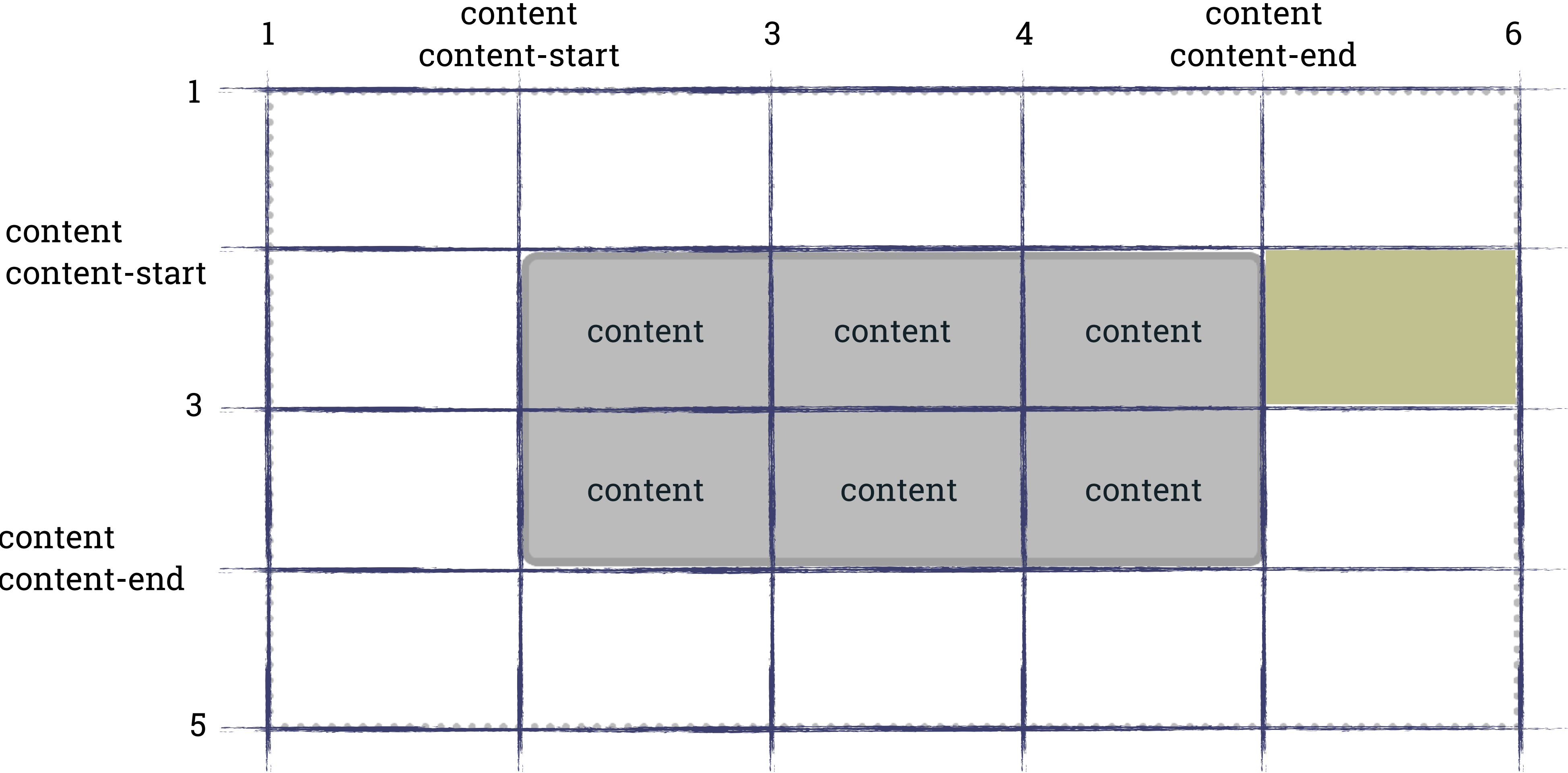
- The other three values are set to the same as grid-row-start, so all are set to 'content'

```
.grid {
  display: grid;
  grid-template-columns: 1fr 1fr 1fr 1fr 1fr;
  grid-template-rows: 100px 100px 100px 100px;
  grid-template-areas:
    ". . . . ."
    ". content content content ."
    ". content content content ."
    ". . . . ."
}

.item {
  grid-area: content ;
}
```

	1	content	3	4	content	6
1						
content						
		content	content	content		
3						
		content	content	content		
content						
5						

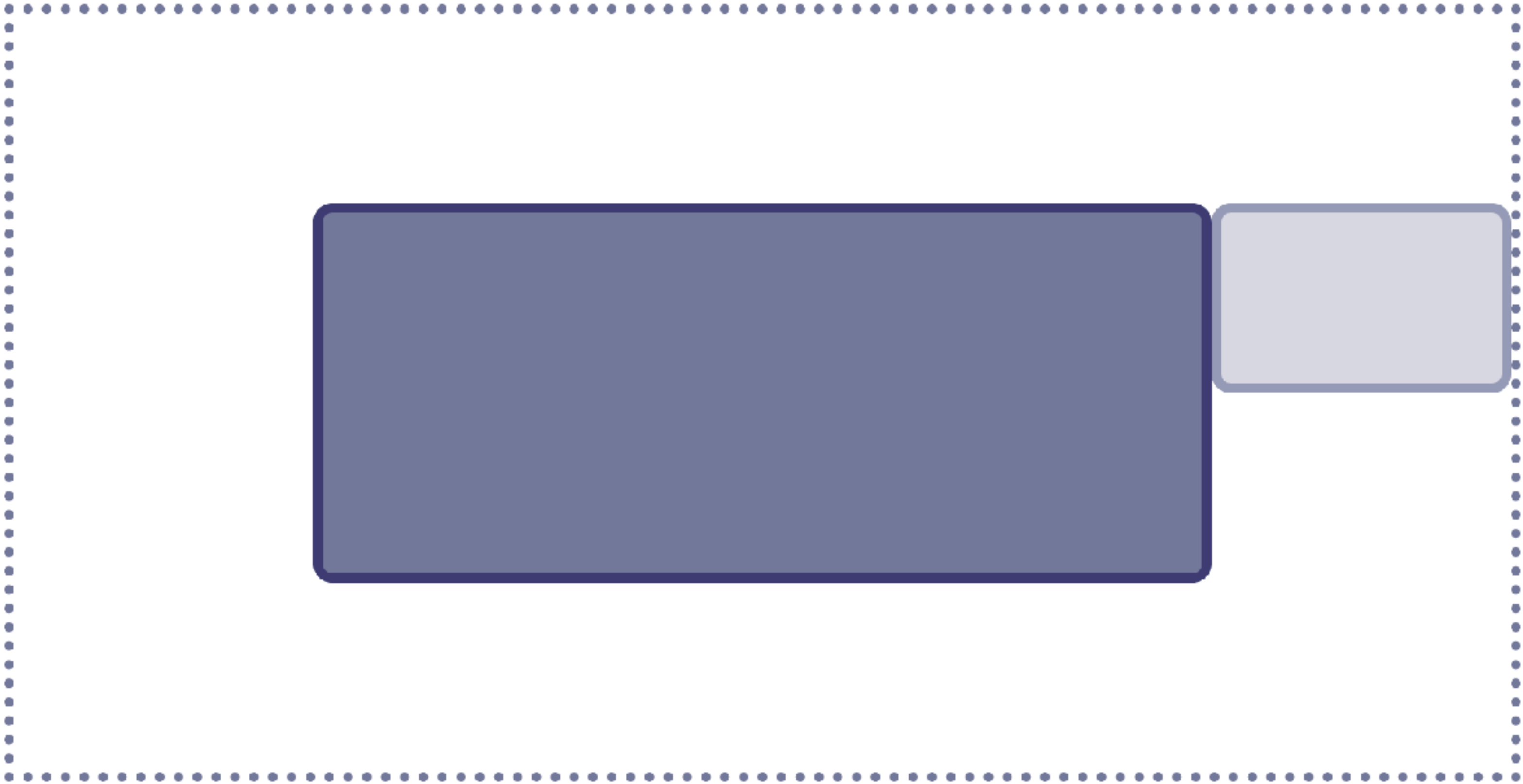
	1	content	3	4	content	6
1						
content		content	content	content		
3		content	content	content		
content						
5						



```
.grid {
  display: grid;
  grid-template-columns: 1fr 1fr 1fr 1fr 1fr;
  grid-template-rows: 100px 100px 100px 100px;
  grid-template-areas:
    ". . . . ."
    ". content content content ."
    ". content content content ."
    ". . . . ."
}

.item {
  grid-area: content ;
}

.item2 {
  grid-row-start: content-start;
  grid-column-start: content-end;
}
```

You have **real choice** for the first time.

What would be the **best method** to achieve this design pattern?

Could we solve this problem with a new design pattern?

Instead of “which patterns does our framework give us to use?”

How old is the oldest CSS in your project?

368

people working on existing projects

29

had CSS in their codebase written 10 years or more ago.

Old CSS in your project doesn't mean you
can't use new CSS.

This is where **understanding** CSS comes in really useful.

A two column Bootstrap 3 Layout

Using markup from the Bootstrap website

About

The main content area and sidebar are positioned using Bootstrap 3. The grid in the main content area is a component using CSS Grid Layout.

A two column Bootstrap 3 Layout

Using markup from the Bootstrap website

Caption



Caption



Caption



Caption



Caption



Caption



About

The main content area and sidebar are positioned using Bootstrap 3. The grid in the main content area is a component using CSS Grid Layout.

```
img {
  max-width: 100%;
}

.gallery {
  display: grid;
  grid-template-columns: repeat(auto-fill,minmax(200px,1fr));
  grid-gap: 10px 20px;
  grid-auto-flow: dense;
}

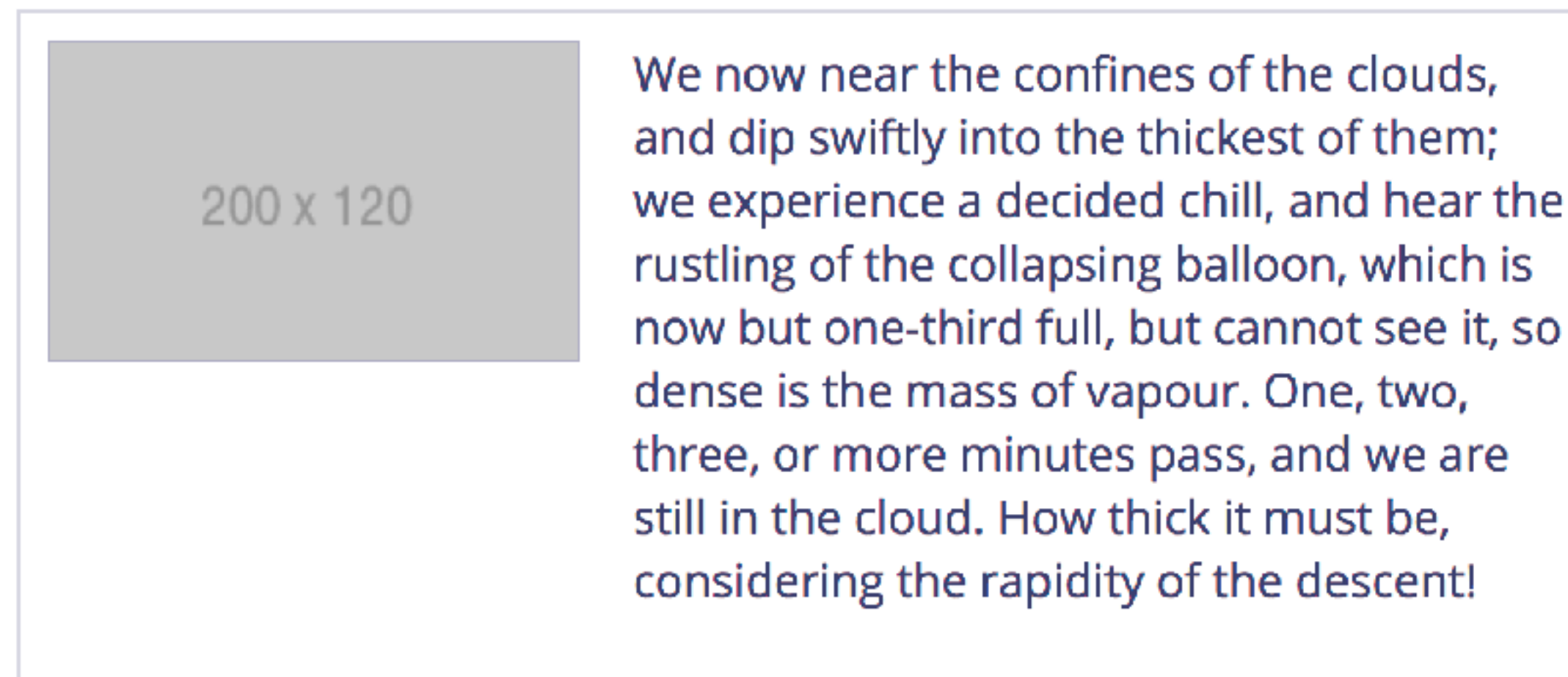
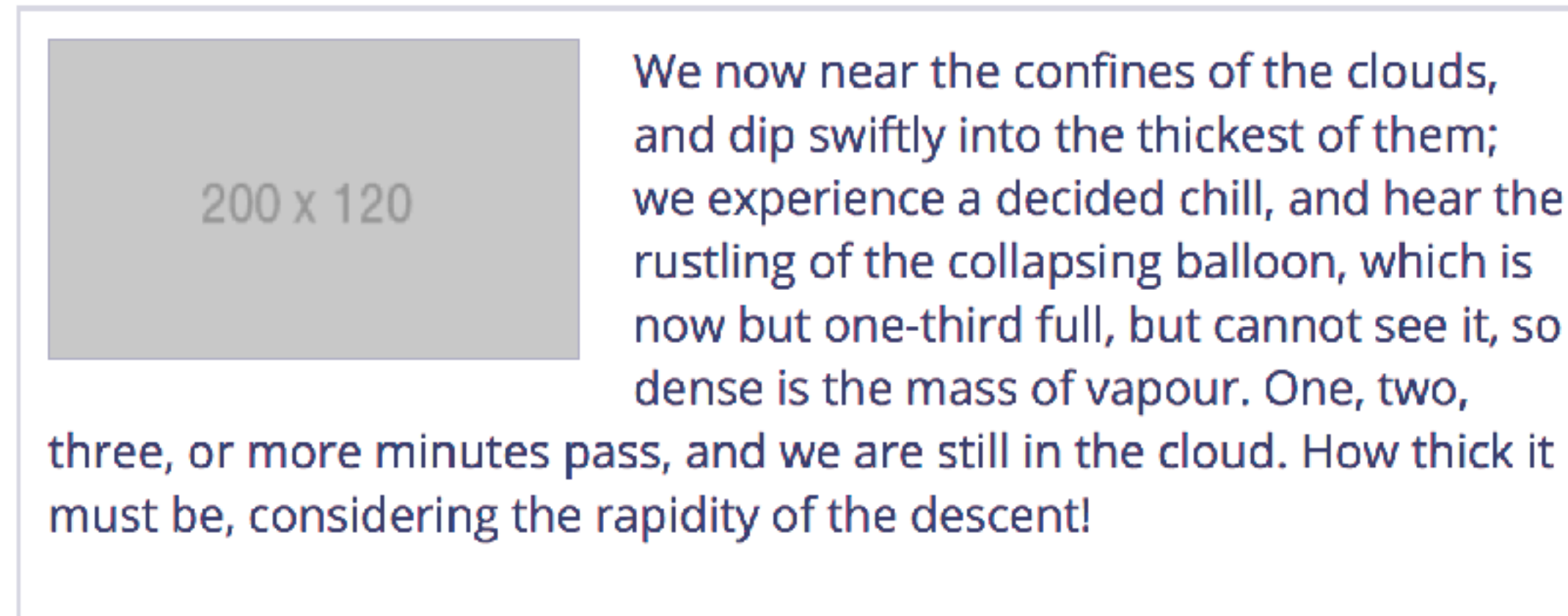
.portrait {
  grid-row-end: span 2;
}

figcaption {
  text-align: center;
  font-size: 1.5em;
}
```

Creating **systems** with new layout.

Other layout methods **still exist.**

Floating the image means the text wraps round.



Defining a grid on the container means we don't get the wrapping behaviour.

Multi-column layout
splits content into equal
width columns.

Choose tags

☐ alfa
☐ bravo
☐ charlie
☐ delta

☐ echo
☐ foxtrot
☐ golf
☐ hotel

☐ indigo
☐ juliet
☐ kilo
☐ lima

☐ mike
☐ november

Using column-width of
200px means we get
more columns if there
is room, fewer with less
available width.

Flex items with the
value of justify-content
set to space-between.



I also use flexbox to
centre the word in the
circle.

You don't need a **grid-shaped hammer** for every layout task.

Off-the-shelf frameworks are designed to solve **generic** problems.

Do you want your project to inherit the CSS
issues of the **rest of the world**?

Build your own framework*

* **framework** doesn't mean "all-encompassing behemoth"

Solving **your specific problems only** will
result in lighter, easier to understand code

```
@mixin gridded($type: grid, $col: 20em, $gap: 20px) {
  @if ($type == 'flex') {
    display: flex;
    flex-wrap: wrap;
    margin-left: -#{$gap} ;
    > * {
      flex: 1 1 $col;
      margin: 0 0 $gap $gap;
    }
  }
  @if ($type == 'grid') {
    display: grid;
    grid-template-columns: repeat(auto-fill, minmax($col,1fr));
    grid-gap: $gap;
  }

  @if ($type == 'multicol') {
    column-gap: $gap;
    column-width: $col;
  }
}
```

```
.multicol {  
  @include gridded('multicol',200px,20px);  
}
```

```
.grid{  
  @include gridded('grid',200px,20px);  
}
```

```
.flex {  
  @include gridded('flex',200px,20px);  
}
```


I am flex item 1	I am flex item 2	I am flex item 3	I am flex item 4
I am grid item 1	I am grid item 2	I am grid item 3	I am grid item 4

As the moment for departure drew near, friends became impatient, and every one anxiously watched the final arrangements, which were made by Mr Coxwell, on whom was laid the important duty of letting go. His hand was on the catch, his countenance was fixed, and his expression stern, as he gazed up into the heavens. He was waiting for the right moment, for the sky was partially cloudy, and it was necessary to wait until the balloon was midway between the cloud that had just passed and the

next that was approaching, so that the aeronauts might have a clear sky, and be able to see the earth they were about to quit for a time. Nor was this all; he knew that in every wind, however strong it might be, there are periods of calm. If he could start in one of these he would avoid much rotatory motion. The deciding, therefore, of the exact moment for making a fair start was not so easy a matter as one might suppose.

Some one at this critical time, with the characteristic eagerness of

poor human nature to "put its finger in the pie," cried out "Now!" and another shouted "Pull!" but Mr Coxwell, regardless of every one, decided for himself; and, just when the wind lulled and the sun shone bright, and the balloon stood proudly erect, he pulled the trigger and they were free.

But they were more than free. They were suddenly in profound repose, for—however high the wind may be, however agitated the balloon, swaying to and fro with sudden and violent action, despite the efforts of many hands that

endeavour to restrain it,—no sooner do aeronauts quit their hold of earth, than, in an instant, all agitation ceases and they are in perfect stillness, without any sense of motion whatever; and this freedom continues throughout the entire flight—except, indeed, when they sink so low as to come into contact with mother earth, when the serenity of their flight is terribly and violently interrupted, as shall be seen in the case of another balloon voyage hereafter to be described.

Working with **less capable** browsers.

These may not always be **old browsers**.

A lack of **understanding** on one side.
A lack of **confidence** on the other.

Building confidence in your CSS skills will help you to make your case to use newer methods.

(Although quite often asking permission is optional)

Old browser versions

39%

of survey respondents cited IE11 as oldest IE supported.

Old browser versions

63%

of survey respondents support IE10+

IE10 & 11 have the `-ms` prefixed older version of grid layout.

For other desktop browsers supporting
the last 2 versions is common.

TABLE OF CONTENTS

1 Introduction

2 Subgrids

2.1 Establishing a Subgrid

2.2 Characteristics of a Subgrid Item

2.3 Subgrid Sizing Algorithm

3 Aspect-ratio-controlled Gutters

4 Changes

Changes

Changes since the April 2018 CSS Grid Layout Level 2 Working Draft

5 Acknowledgements

Conformance

Document conventions

Conformance classes

Requirements for Responsible Implementation of CSS

Partial Implementations

Implementations of Unstable and Proprietary Features

Implementations of CR-level Features

Index

Terms defined by this specification

CSS Grid Layout Module Level 2

Editor's Draft, 11 June 2018



► Specification Metadata

Copyright © 2018 W3C® (MIT, ERCIM, Keio, Beihang). W3C [liability](#), [trademark](#) and [permissive document license](#) rules apply.

Abstract

This CSS module defines a two-dimensional grid-based layout system, optimized for user interface design. In the grid layout model, the children of a grid container can be positioned into arbitrary slots in a predefined flexible or fixed-size layout grid. Level 2 expands Grid by adding “subgrid” capabilities for nested grids to participate in the sizing of their parent grids; and aspect-ratio-controlled gutters.

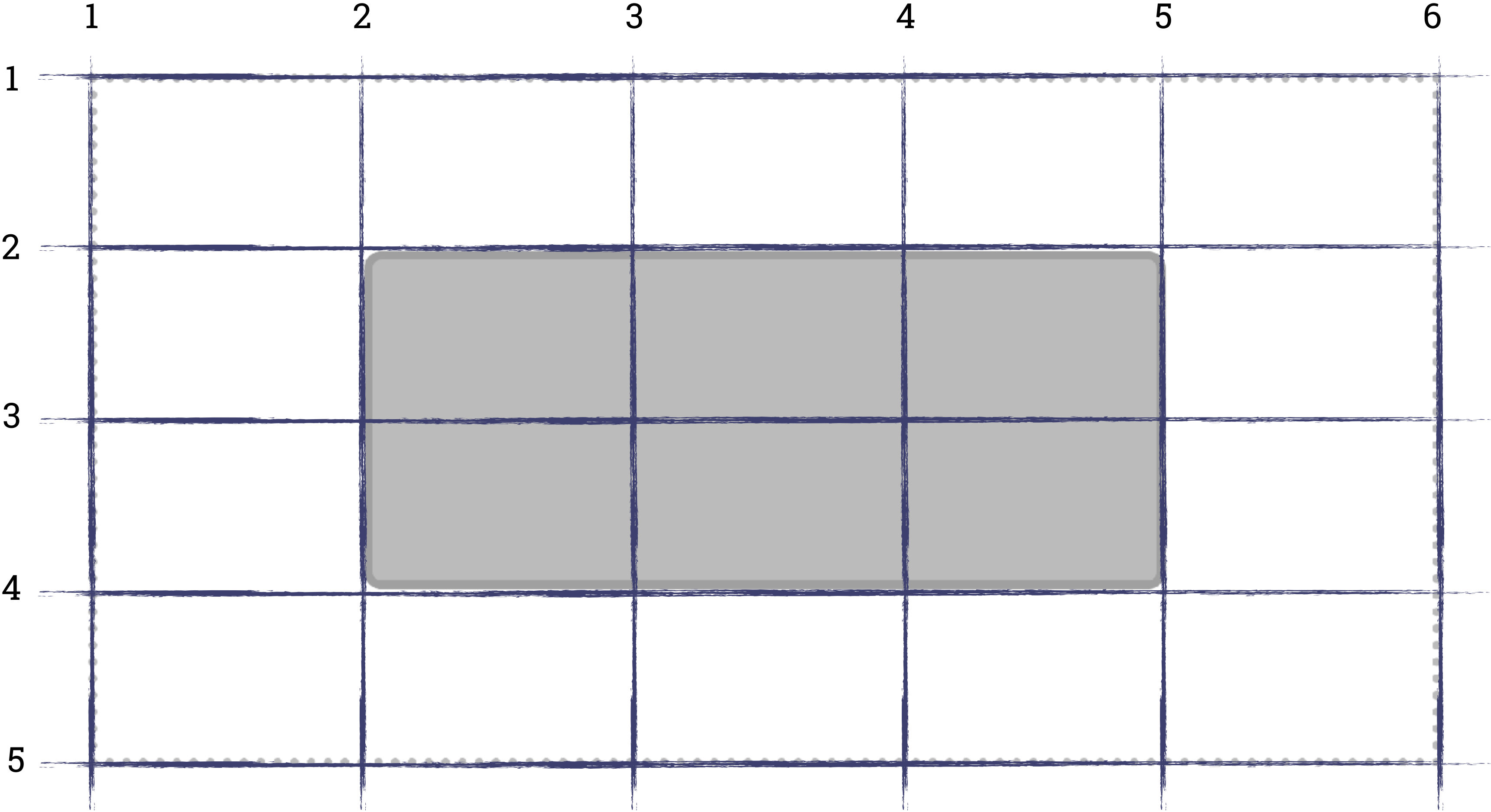
[CSS](#) is a language for describing the rendering of structured documents (such as HTML and XML) on screen, on paper, in speech, etc.

Status of this document

This is a public copy of the editors' draft. It is provided for discussion only and may change at any moment. Its publication here does not imply endorsement of its contents by W3C. Don't cite this document other than as work in progress.

[GitHub Issues](#) are preferred for discussion of this specification. When filing an issue, please put the text “css-grid” in the title, preferably like this: “[css-grid] ...summary of comment...”. All issues and comments are [archived](#), and there is also a [historical archive](#).

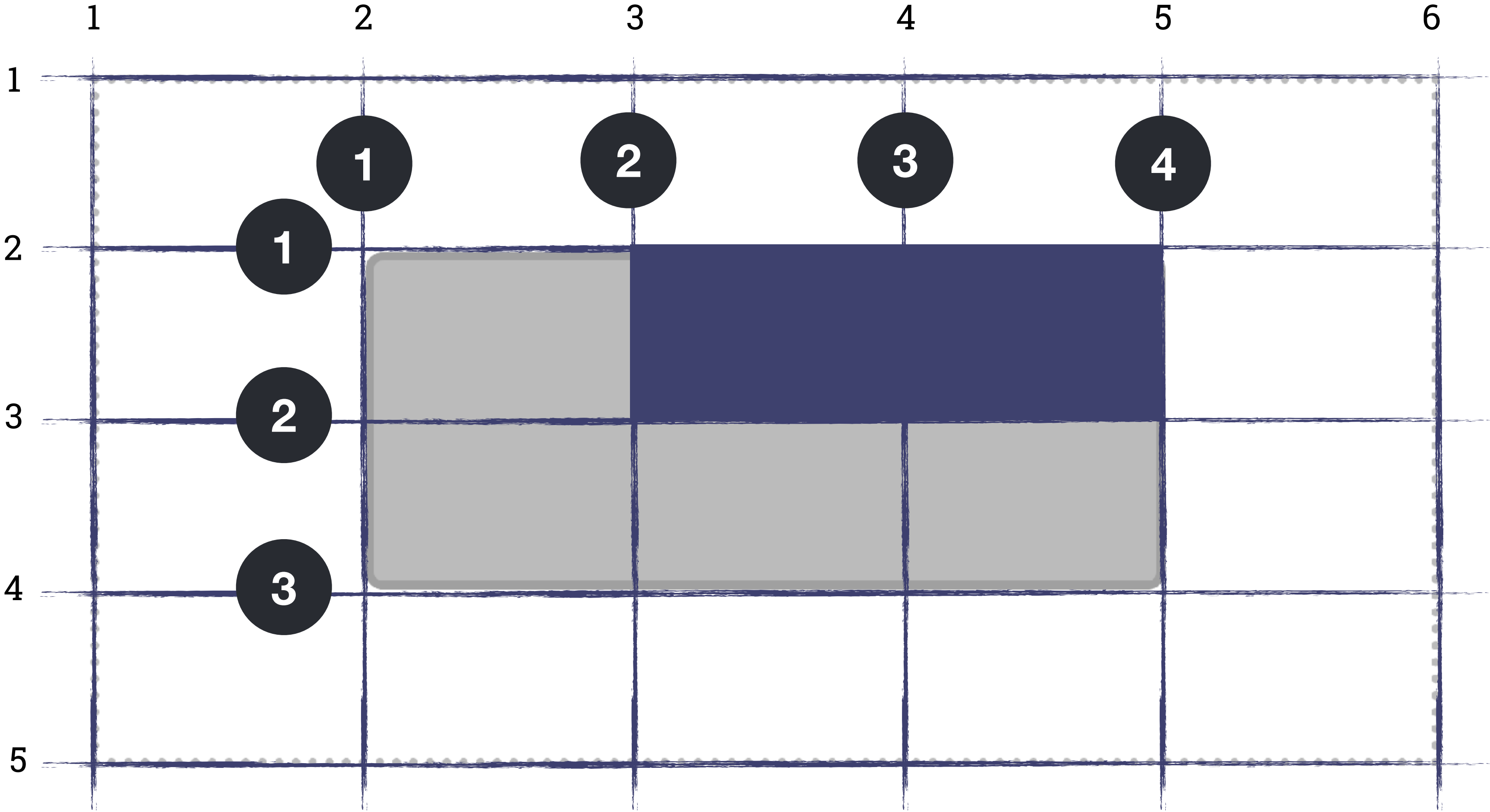
This document was produced by the [CSS Working Group](#) (part of the [Style Activity](#)).



```
.grid {  
  display: grid;  
  grid-template-columns: 1fr 1fr 1fr 1fr 1fr;  
  grid-template-rows: 100px 100px 100px 100px;  
}  
  
.item {  
  grid-column: 2 / 5;  
  grid-row: 2 / 4;  
  
}
```

```
.grid {  
  display: grid;  
  grid-template-columns: 1fr 1fr 1fr 1fr 1fr;  
  grid-template-rows: 100px 100px 100px 100px;  
}
```

```
.item {  
  grid-column: 2 / 5;  
  grid-row: 2 / 4;  
  display: grid;  
  grid-template-columns: subgrid;  
  grid-template-rows: subgrid;  
}
```

```
.item {  
  grid-column: 2 / 5;  
  grid-row: 2 / 4;  
  display: grid;  
  grid-template-columns: subgrid;  
  grid-template-rows: subgrid;  
}
```

```
.item > div {  
  grid-column: 2 / 4;  
  grid-row: 1;  
}
```

CSS Grid Layout - CR

Global71.42% + 5.5% = 76.92%

unprefixed:71.42%

Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors. Includes support for all `grid-*` properties and the `fr` unit.

Current alignedUsage relativeDate relativeShow all

IE	Edge*	Firefox	Chrome	Safari	Opera	iOS Safari*	Opera Mini*	Android Browser*	Opera Mobile*	Chrome for Android	UC Browser for Android	Samsung Internet
			1 49									
			61			10.2						4
	2 15	56	62	10.1		10.3		4.4				5
2 11	16	57	63	11	48	11.2	all	62	37	62	11.4	6.2
	17	58	64	TP	49							
		59	65		50							
		60	66									

NotesKnown issues (2)Resources (13)Feedback

1

Enabled in Chrome through the "experimental Web Platform features" flag in chrome://flags

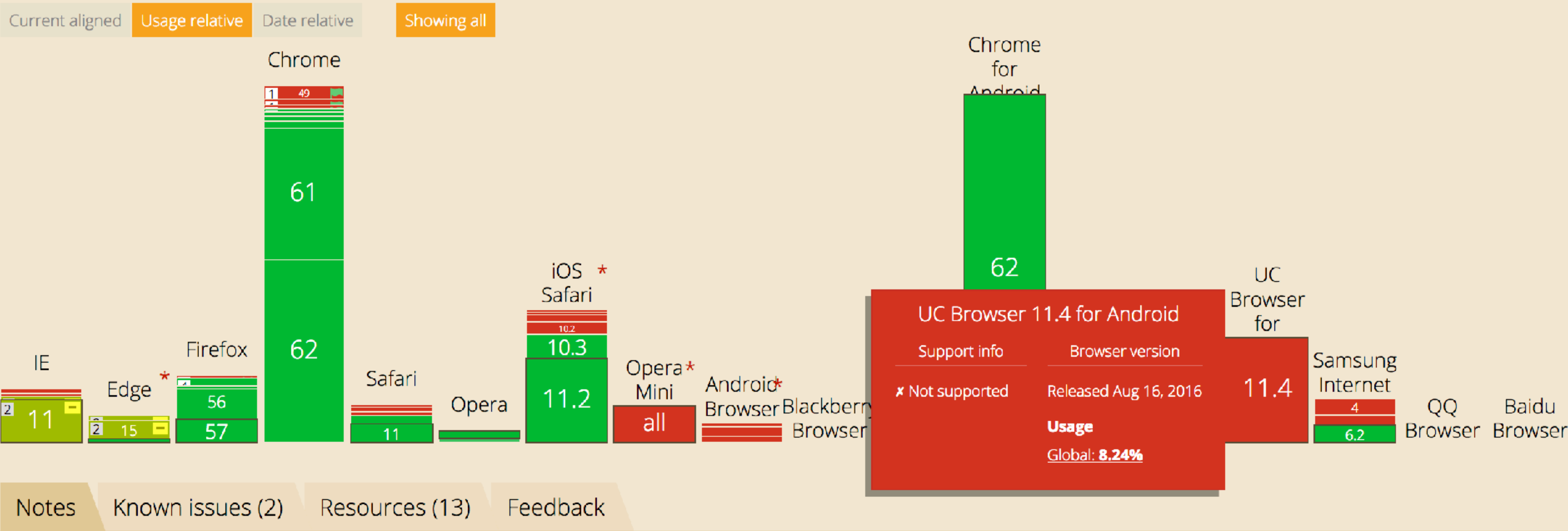
2

Partial support in IE refers to supporting an older version of the specification.

CSS Grid Layout - CR

Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors. Includes support for all `grid-*` properties and the `fr` unit.

Global 71.42% + 5.5% = 76.92%
unprefixed: 71.42%



1. For details on Chrome's support for the `grid-*` properties, see the [Chrome DevTools Grid Layout Inspector](#).



[Press Releases](#) [FAQ](#) [About](#) [Feedback](#)

Chrome

45.42%

UC Browser

35.63%

Opera

7.92%

Firefox

3.92%

Safari

2.52%

Samsung Internet

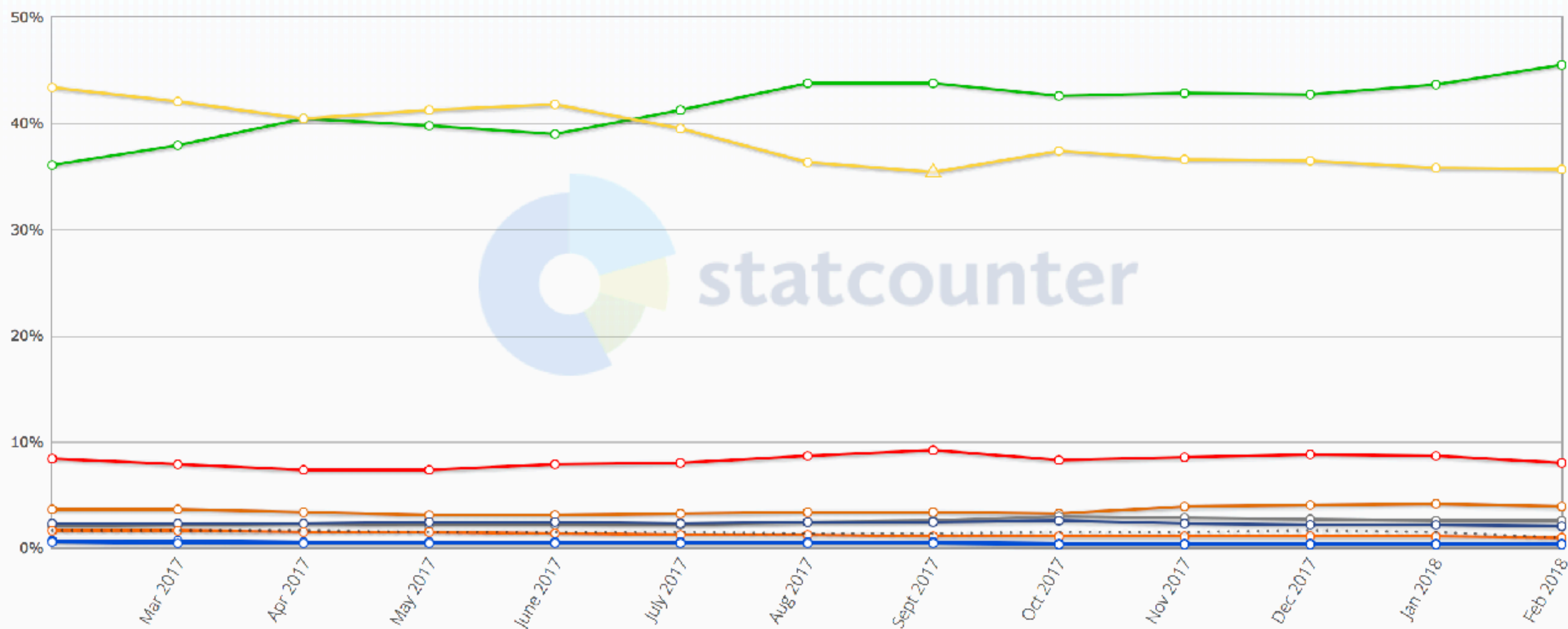
2.05%

Browser Market Share in India - February 2018

Browser Market Share India

Feb 2017 - Feb 2018

[Edit Chart Data](#)



Many browsers without support for Grid and other new CSS at this point are **mobile browsers**.

Many of the browsers without support are most popular in areas where data is **expensive** or devices **less powerful**.

“Grid too young and would need a ton of polyfills.”

“Lack of a good css grid polyfill that works with postcss
and supports not so old browsers”

–Survey responses

Stop looking for polyfills and shims. They will make the experience **worse** for less capable browsers and devices.

“The time it takes your customer to get
the information she came for.”

–Jeffrey Zeldman

Using Grid rather than loading a big framework can help create a better experience even for browsers that **don't support Grid**.

Feature Queries - use CSS to ask if the browser supports a feature before using it.

**Alexandre Dieulot**

@Dieulot

Follow



CSS grid's final hurdle has just been lifted!
🎉 @UCBrowser 11.8 (in beta) is based on
Chrome 57 and support them!!! /cc
@jensimmons @rachelandrew

All lights green for CSS grid going
mainstream in 2018.

full page

HTML

CSS

Result

Edit

A

B

C

D



[Press Releases](#) [FAQ](#) [About](#) [Feedback](#)

Chrome for Android
30.82%

UC Browser 12.0
11.76%

Chrome 63.0
6.68%

UC Browser 11.5
6.3%

Chrome 64.0
6.23%

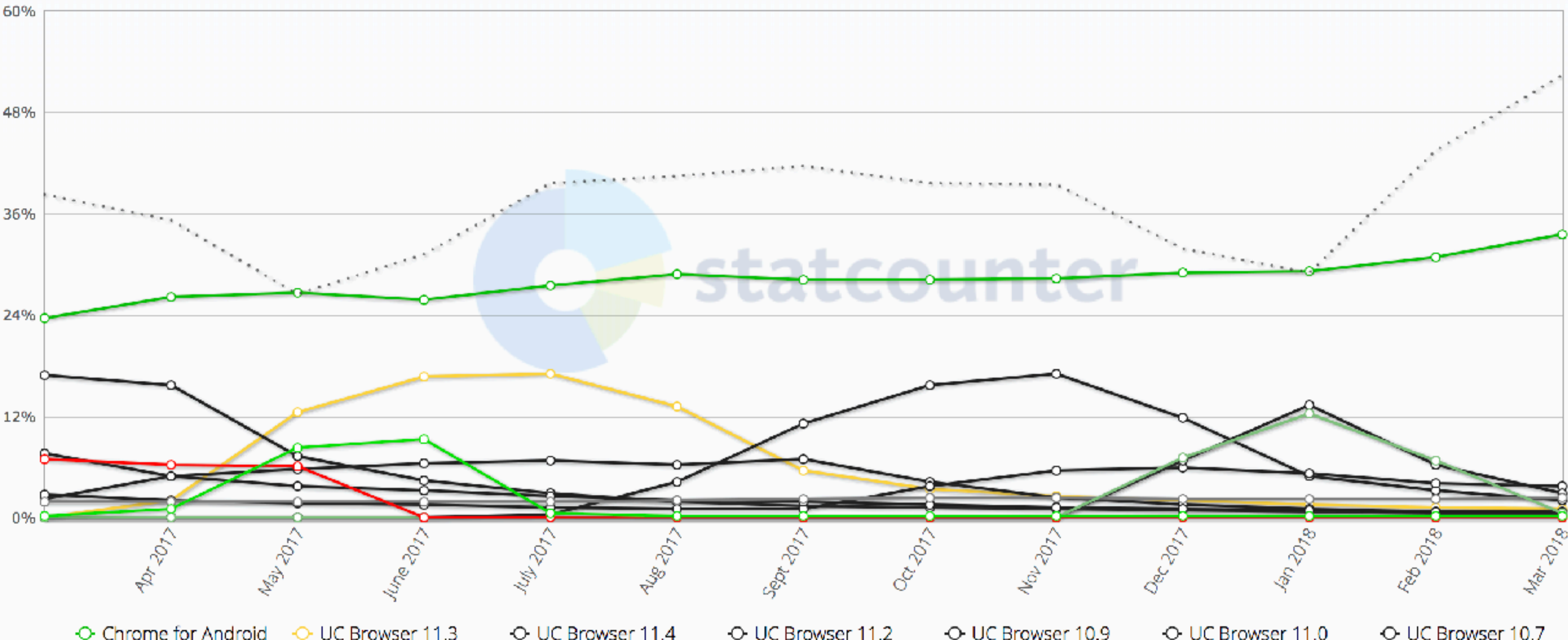
UC Browser 11.0
4.05%

Browser Version Market Share in India - February 2018

Browser Version Market Share India

Mar 2017 - Mar 2018

Edit Chart Data



Create complex layouts for browsers that support them **with a few lines of CSS.**

Making the web available to **everyone**.
That's exciting.

“Q. How do you feel when you see a new CSS feature announced?”

–Me, in a survey question

“Excited!”

Thank you!

@rachelandrew

