

Modern CSS techniques

by Fatih Demir

AVISI

Most of my issues with CSS

- Constants
- Vendor prefixes
- CSS hacks

Preprocessors

Sass

{less}

stylus

and others..

Horizontal alignment

Let's position a div inside another div
horizontally..

Horizontal alignment

How to horizontally center a `<div>` in another `<div>`?



2938



794

How can I horizontally center a `<div>` within another `<div>` using `css` (if it's even possible)?

The outer `<div>` has `width:100%`:

```
<div id="outer" style="width:100%">
  <div id="inner">Foo foo</div>
</div>
```

asked 8 years, 6 months

viewed 2806264 times

active 13 days ago

Linked

46

[How to make a div](#)

Horizontal alignment



You can apply this CSS to the inner `<div>`:

3325



```
#inner {  
  width: 50%;  
  margin: 0 auto;  
}
```

Of course, you don't have to set the `width` to `50%`. Any width less than the containing `<div>` will work. The `margin: 0 auto` is what does the actual centering.

If you are targeting IE8+, it might be better to have this instead:

```
#inner {  
  display: table;  
  margin: 0 auto;  
}
```

Vertical alignment

Let's position a div inside another div
vertically.....

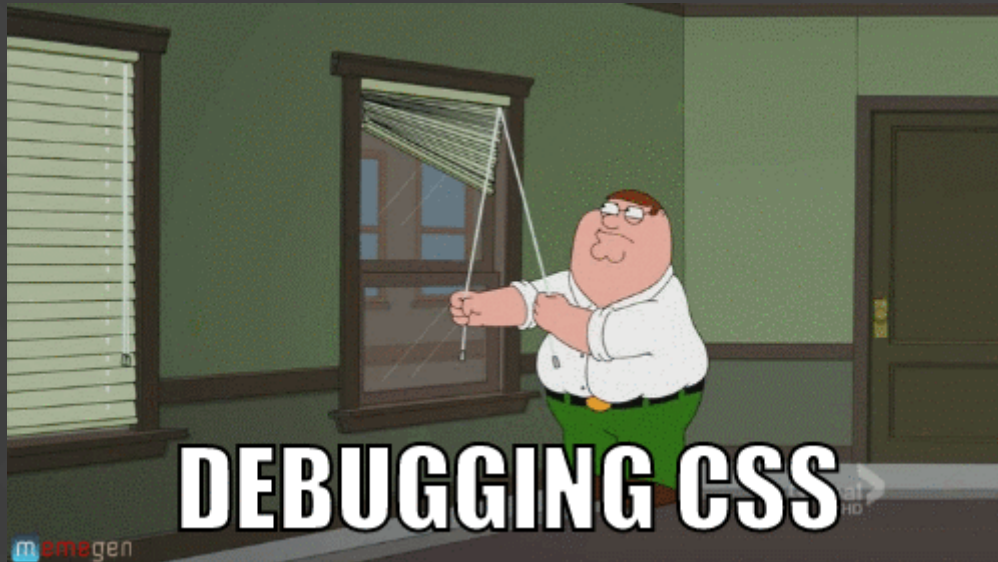


Vertical alignment

`tl;dr`

Vertical align middle works, but you will have to use `table-cell` on your parent element and `inline-block` on the child.

Why can't it be easy?



Flexbox

Positioning your elements in modern web apps
without pulling your hairs out.

Flexbox

Designed for laying out more complex applications and webpages.

Gives flex containers the ability to manipulate it's contents based on the available space.

Flexbox

Demo horizontal/vertical alignment.

Flex container

```
/* Flex container as a block element */  
display: flex;
```

```
/* Flex container as an inline element */  
display: inline-flex;
```

Flex container

One dimensional. Either horizontal or vertical.

Ability to manipulate positioning in both axis.

```
flex-direction: row; /* default, aligns horizontally.*/
```

Flex direction

```
flex-direction: row; /* default */
```

```
flex-direction: row-reverse; /* also works with column */
```



Flex container

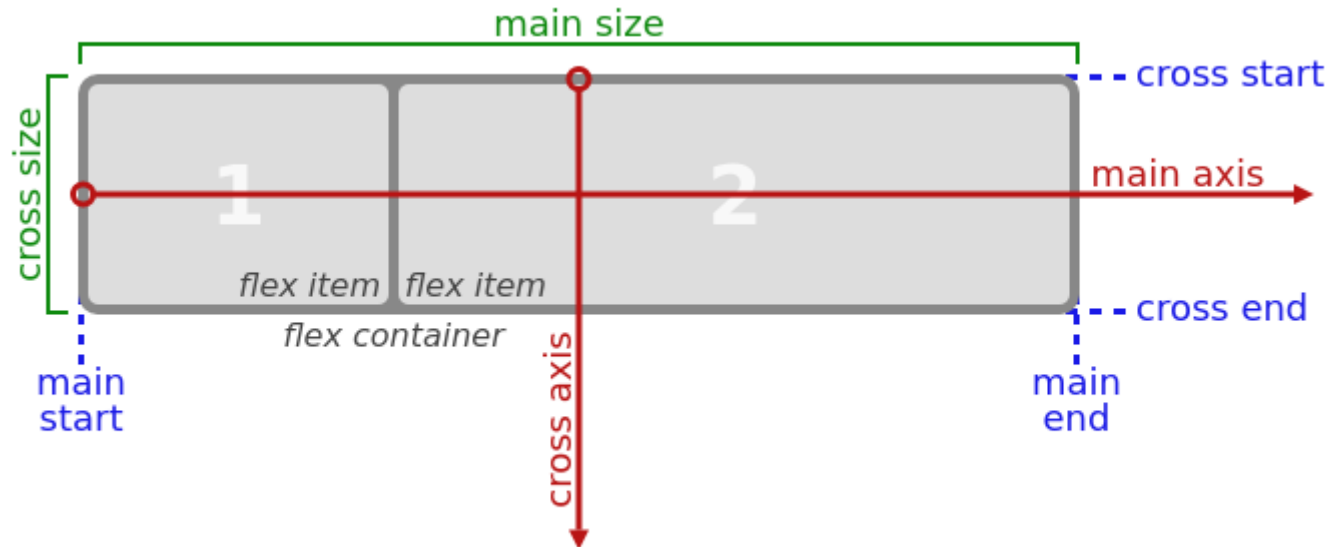


Figure 4 An illustration of the various directions and sizing terms as applied to a 'row' flex container.

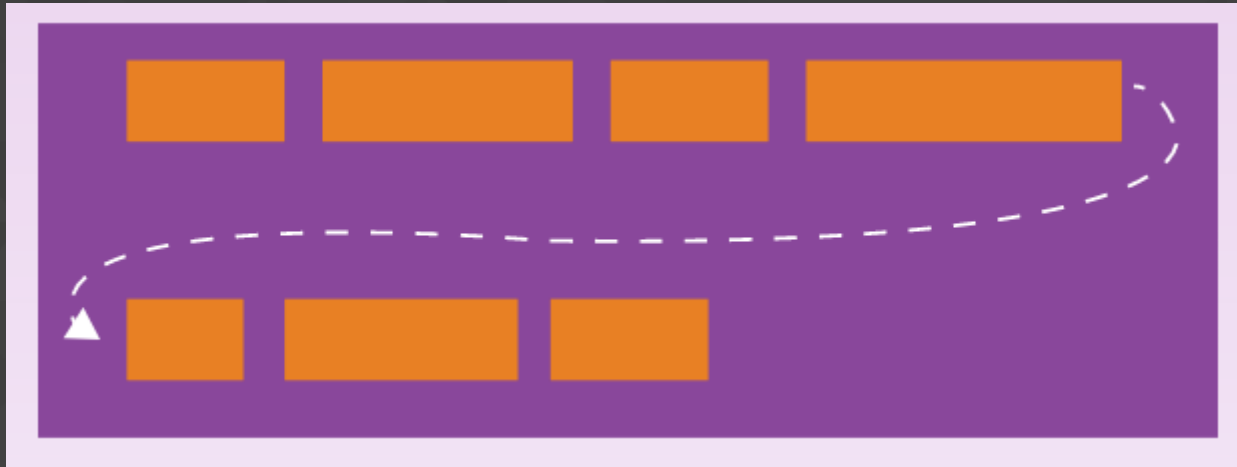
Flex container

```
justify-content: flex-start; /* main axis */
```

```
align-items: flex-start; /* cross axis */
```

Flex wrap

```
flex-wrap: wrap;
```



```
align-content: stretch;
```

Flex items

Direct children only

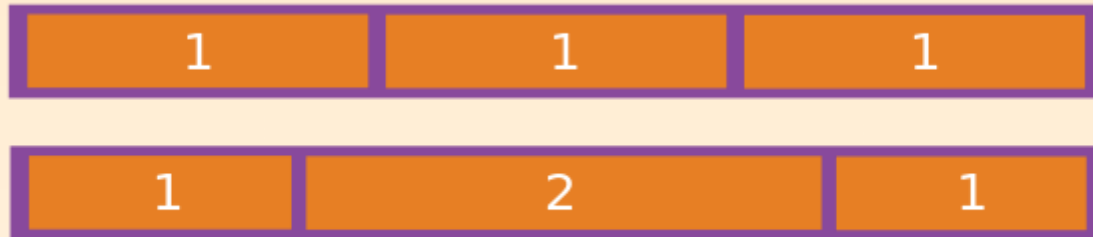
Able to nest flex-containers however.

Order

```
order: 0; /* default */
```

Flex-grow

```
flex-grow: 0; /* default */
```



Flex-shrink

```
flex-shrink: 1; /* default */
```

Flex-basis

```
flex-basis: auto | length; /* default auto */
```

Shorthands

```
flex: 0 1 auto; /* flex-grow flex-shrink flex-basis */
```

```
flex-flow: row nowrap; /* flex-direction flex-wrap */
```


Demo



Browser support

<http://caniuse.com/#search=flexbox>

#

Flexible Box Layout Module - CR

Global 83.35% + 14.36% = 97.71%

unprefixed: 82.83% + 3.52% = 86.35%

Method of positioning elements in horizontal or vertical stacks. Support includes the support for the all properties prefixed with **flex** as well as **display: flex**, **display: inline-flex**, **align-content**, **align-items**, **align-self**, **justify-content** and **order**.

| Current aligned | Usage relative | Date relative | Show all | | | | | | |
|-----------------|----------------|---------------|----------|--------|-------|--------------|--------------|-------------------|--------------------|
| IE | Edge * | Firefox | Chrome | Safari | Opera | iOS Safari * | Opera Mini * | Android Browser * | Chrome for Android |
| | | | 49 | | | | | | |
| | | | 55 | | | 9.3 | | 4.4 | |
| | 14 | 51 | 56 | 10 | 43 | 10.2 | | 4.4.4 | |
| ⁴ 11 | 15 | 52 | 57 | 10.1 | 44 | 10.3 | all | 56 | 57 |
| | | 53 | 58 | TP | 45 | | | | |
| | | 54 | 59 | | 46 | | | | |
| | | 55 | 60 | | | | | | |

Notes

Known issues (8)

Resources (13)

Feedback

Most partial support refers to supporting an **older version** of the specification or an **older syntax**.

⁴ Partial support is due to large amount of bugs present (see known issues)

Flexbox issues

Excellent resource for flexbox issues and cross-browser workarounds.

<https://github.com/philipwalton/flexbugs>

Learn more

<https://www.w3.org/TR/css-flexbox-1/#propdef-flex-wrap>

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

Questions so far about
flexbox?

CSS Grid

CSS Grid

Grid based two-dimensional layout.

Works great in combination with flexbox

There is no Flexbox vs Grid. They are friends!!

Browser support

<http://caniuse.com/#search=grid>

#

CSS Grid Layout 📄 - CR

Global 31.32% + 5.69% = 37.01%
unprefixed: 31.32%

Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for lay out into columns and rows using a set of predictable sizing behaviors

Current aligned Usage relative Date relative Show all

| IE | Edge * | Firefox | Chrome | Safari | Opera | iOS Safari * | Opera Mini * | Android Browser * | Chrome for Android |
|------|--------|---------|--------|--------|-------|--------------|--------------|-------------------|--------------------|
| | | | 1 49 | | | | | | |
| | | | 1 55 | | | 9.3 | | 4.4 | |
| | 2 14 | 3 51 | 1 56 | 10 | 1 43 | 10.2 | | 4.4.4 | |
| 2 11 | 2 15 | 52 | 4 57 | 10.1 | 44 | 10.3 | all | 56 | 4 57 |
| | | 53 | 58 | TP | 45 | | | | |
| | | 54 | 59 | | 46 | | | | |
| | | 55 | 60 | | | | | | |

Notes Known issues (0) Resources (11) Feedback

Supported in WebKit Nightly with `-webkit-` prefix.

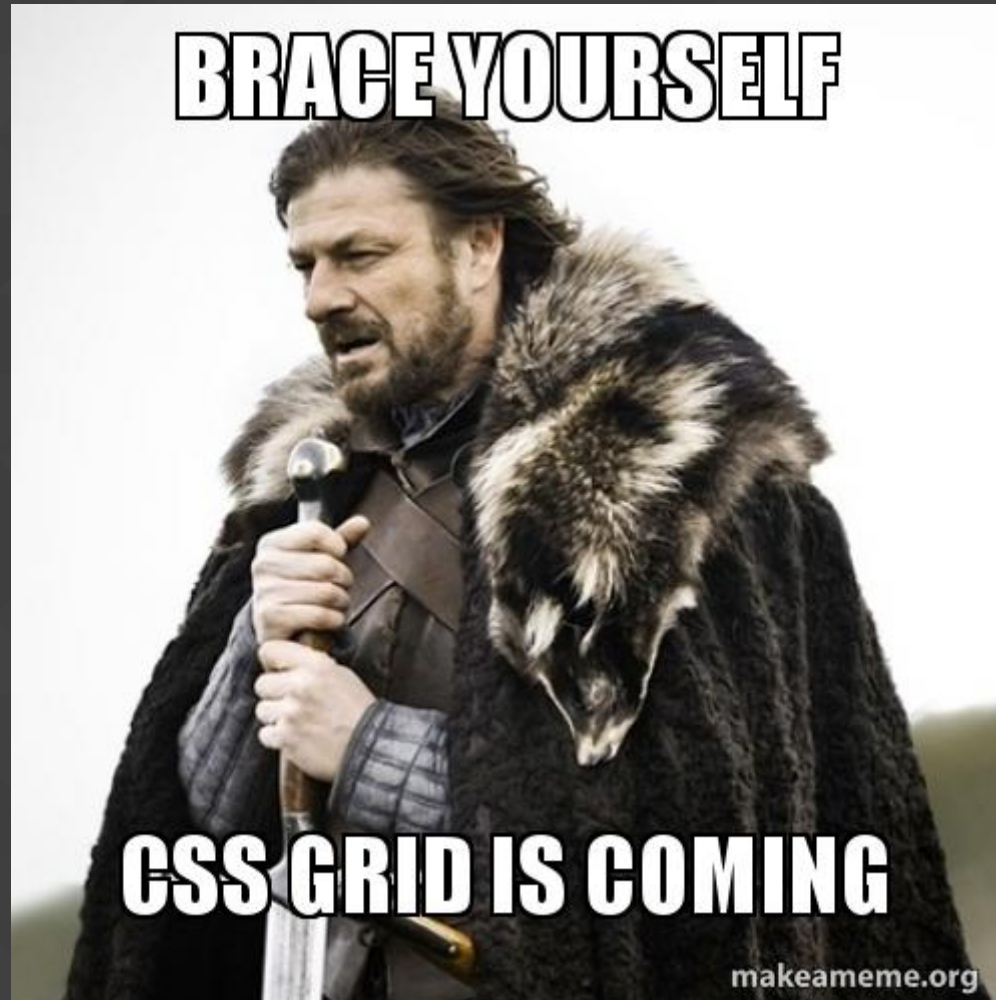
¹ Enabled in Chrome through the "experimental Web Platform features" flag in `chrome://flags`

² Partial support in IE refers to supporting an **older version** of the specification.

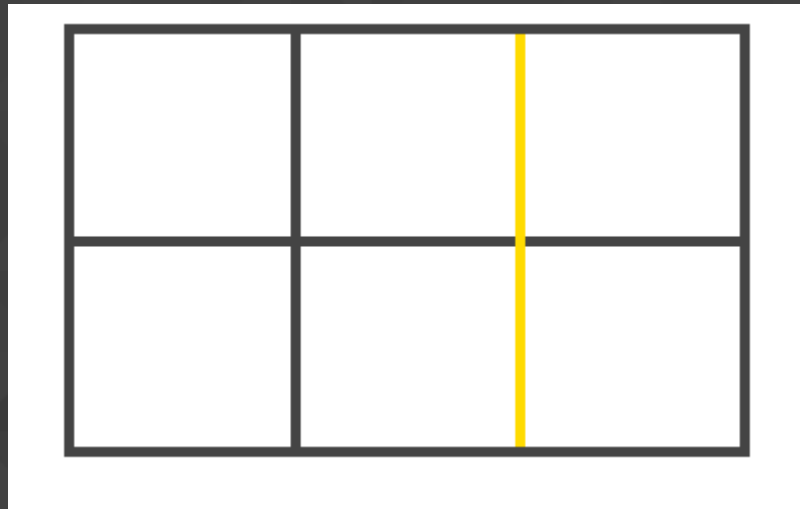
³ Enabled in Firefox through the `layout.css.grid.enabled` flag

⁴ Grid does not initially display with certain JavaScript ([bug report](#))

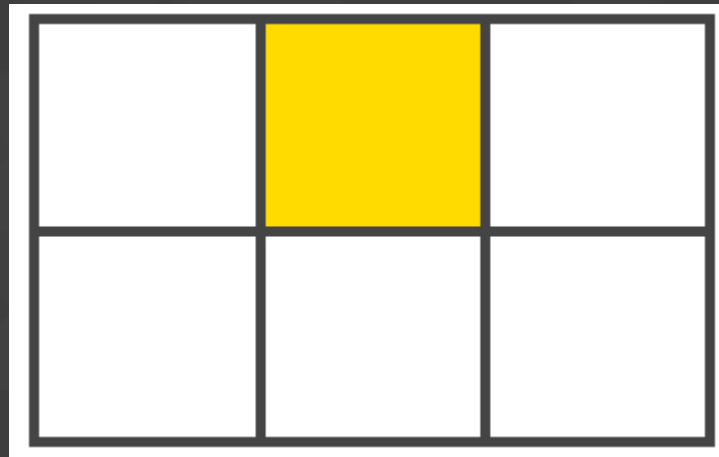
Let's get ready for Grid



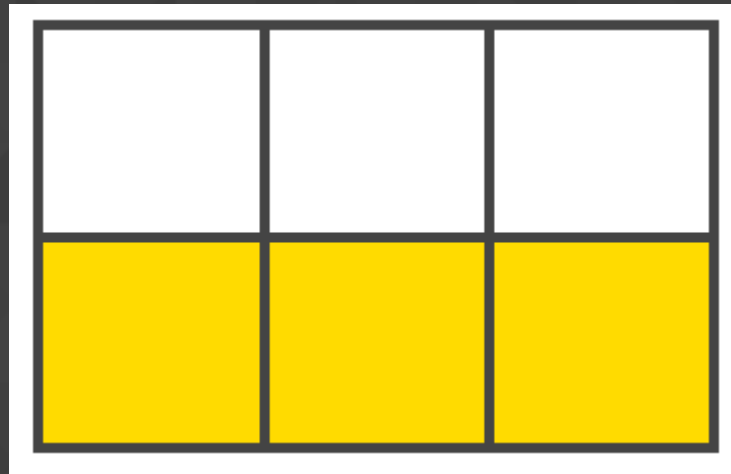
Grid line



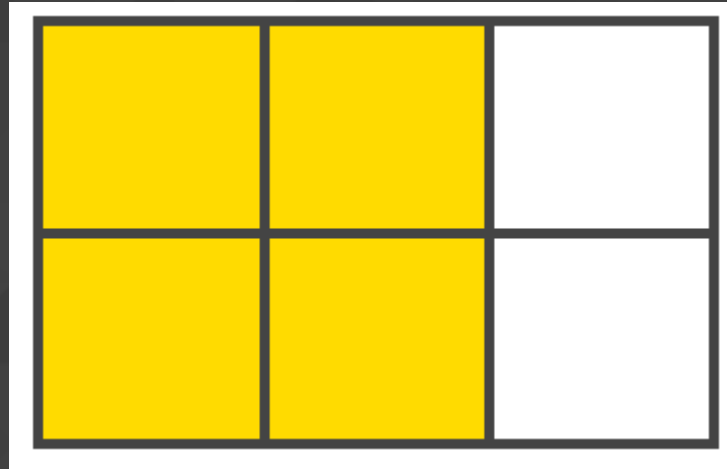
Grid cell



Grid track



Grid area



Demo

Let's build a simple application with CSS grid together

Grid container

```
display: grid | inline-grid | subgrid;
```


Grid template rows/columns

```
grid-template-rows: 10px 10% auto 5% 5px;
```

```
grid-template-columns: 1fr 1fr; /* repeat(2, 1fr) */
```

```
grid-template-columns: [col-1-custom-name] 1fr [end-col-1];
```

Grid template areas

```
.item-a { grid-area: header; }  
.item-b { grid-area: main; }  
.item-c { grid-area: sidebar; }  
.item-d { grid-area: footer; }  
  
.container {  
  grid-template-columns: 50px 50px 50px 50px;  
  grid-template-rows: auto;  
  grid-template-areas:  
    "header header header header"  
    "main main . sidebar"  
    "footer footer footer footer";  
}
```

Grid gaps

```
grid-column-gap: size;
```

```
grid-row-gap: size;
```

```
grid-gap: size size; /* 1 size for both or 2 sizes */
```

Aligning

```
justify-items: stretch; /* start | end | center */
```

```
align-items: stretch; /* Same as above */
```

```
justify-content: stretch; /* space-around -between & -evenly */
```

```
align-content: stretch; /* Same as above */
```

Grid is awesome however

Not production ready..

Learn more

<https://css-tricks.com/snippets/css/complete-guide-grid/>

<https://www.w3.org/TR/css3-grid-layout/>

<https://developer.mozilla.org/en-US/docs/Glossary/Grid>

<http://gridbyexample.com/>

Questions about CSS Grid?

Might need to enable flags
in your browser for CSS
Grid

[https://igalia.github.io/css-grid-
layout/enable.html](https://igalia.github.io/css-grid-layout/enable.html)

Slides on Github

<https://github.com/fatihict/modern-css-techday-presentation>

Handson

Pick a website (preferably one you've built yourself before) and create the same layout using CSS grid and flexbox.

Try making it responsive even if the original isn't.

Bonus: no floats, clears, tables & vertical-align.