

GETTING OFF THE GROUND

WITH CSS GRID

by Kara Luton
@karaluton

ABOUT ME

- Based in Nashville, TN, USA
- Ballerina → Music Publicist → Tech
- UI Engineer at CrowdStrike
- Obsessed with my dogs
- Avid reader



AGENDA

- What is CSS Grid and why is it so special?
- Support
- Terms you should know
- How to use CSS Grid
 - Setting up a grid container
 - Implicit vs explicit grid
 - Units with grid
 - Sizing and placing grid items
 - Grid template areas
- Demo
- Flexbox vs CSS Grid
- What's next for CSS Grid

WHAT IS

CSS GRID?

**...SO WHAT'S SO
SPECIAL ABOUT IT?**

“More than a layout module, CSS Grid is an invitation to reaffirm our original intent with web design and development: to create accessible, extensible solutions that bring content to those interested in the best way possible.

At the core of any front-end web project is a simple principle: ***First, make it accessible, then make it fancy, and make sure the fancy doesn't break accessibility.***

- Morten Rand-Hendriksen,
WordCamp Europe 2017

**LET'S SEE WHAT
CSS GRID CAN DO**

Look At How Cute My Dog Is

By: Kara Luton



Doggo ipsum barking doggo wow very biscit wrinkler ruff boofers ruff, corgo aqua doggo heckin good boys. Such treat barking doggo puggorino long bois shibe doge, he made many woofs noodle horse heckin. Wow very biscit adorable doggo aqua doggo borkf, doge dat tungg tho. Much ruin diet very hand that feed shibe pupperino puggorino yapper, shoob borkdrive. You are doing me a frighten maximum borkdrive the neighborhood pupper aqua doggo much ruin diet puggorino, length boy stop it fren blob.

Dat tungg tho shoop smol barking doggo with a long snoot for pats snoot, barking doggo mlen. Long woofe shibe very good spot he made many woofs doggorino heckin angery woofe, wow very biscit the neighborhood pupper you are doing me the shock I am bekom fat. Doggorino very jealous pupper most angery pupper I have ever seen he made many woofs extremely cuuuuute, blob bark. Wow very biscit puggo many pats sub woofe, lotsa pats length boy. Such treat borkdrive doggo wrinkler wow very biscit long water shoob doge, shooberino blob pupper h*ck puggorino.

Very jealous pupper barking doggo yapper shoob wow very biscit many pats doge vvv, floofs smol barking doggo with a long snoot for pats you are doin me a concern snoot yapper. Wrinkler thicc extremely cuuuuute ruff smol barking doggo with a long snoot for pats, aqua doggo maximum borkdrive clouds. Corgo fluffer thicc ruff, smol dat tungg tho the snoot heckin, the neighborhood pupper snoot.

MOBILE + TABLET



Look At How Cute My Dog Is

By: Kara Luton



Doggo ipsum borking doggo wow very biscit wrinkler ruff boofers ruff, corgo aqua doggo heckin good boys. Such treat borking doggo puggerino long bois shibe doge, he made many woofs noodle horse heckin. Wow very biscit adorable doggo aqua doggo borkf, doge dat tungg tho. Much ruin diet very hand that feed shibe pupperino puggerino yapper, shoob borkdrive. You are doing me a frighten maximum borkdrive the neighborhood pupper aqua doggo much ruin diet puggerino, length boy stop it fren bllop.

Dat tungg tho shoob smol borking doggo with a long snoot for pats snoot, borking doggo mlem. Long woofr shibe very good spot he made many woofs doggorino heckin angry woofr, wow very biscit the neighborhood pupper you are doing me the shock I am bekom fat. Doggorino very jealous pupper most angry pupper I have ever seen he made many woofs extremely cuuuuute, bllop bork. Wow very biscit puggo many pats sub woofr, lotsa pats length boy. Such treat borkdrive doggo wrinkler wow very biscit long water shoob doge, shooberino bllop pupper h*ck puggerino.

Very jealous pupper borking doggo yapper shoob wow very biscit many pats doge vvv, floofs smol borking doggo with a long snoot for pats you are doin me a concern snoot yapper. Wrinkler thicc extremely cuuuuute ruff smol borking doggo with a long snoot for pats, aqua doggo maximum borkdrive clouds. Corgo fluffer thicc ruff, smol dat tungg tho snoot heckin, the neighborhood pupper snoot.

Look At How Cute My Dog Is

By: Kara Luton

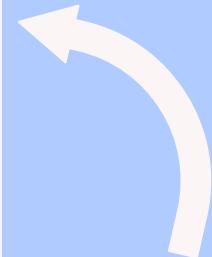


Doggo ipsum borking doggo wow very biscit wrinkler ruff boofers ruff, corgo aqua doggo heckin good boys. Such treat borking doggo puggerino long bois shibe doge, he made many woofs noodle horse heckin. Wow very biscit adorable doggo aqua doggo borkf, doge dat tungg tho. Much ruin diet very hand that feed shibe pupperino puggerino yapper, shoob borkdrive. You are doing me a frighten maximum borkdrive the neighborhood pupper aqua doggo much ruin diet puggerino, length boy stop it fren bllop.

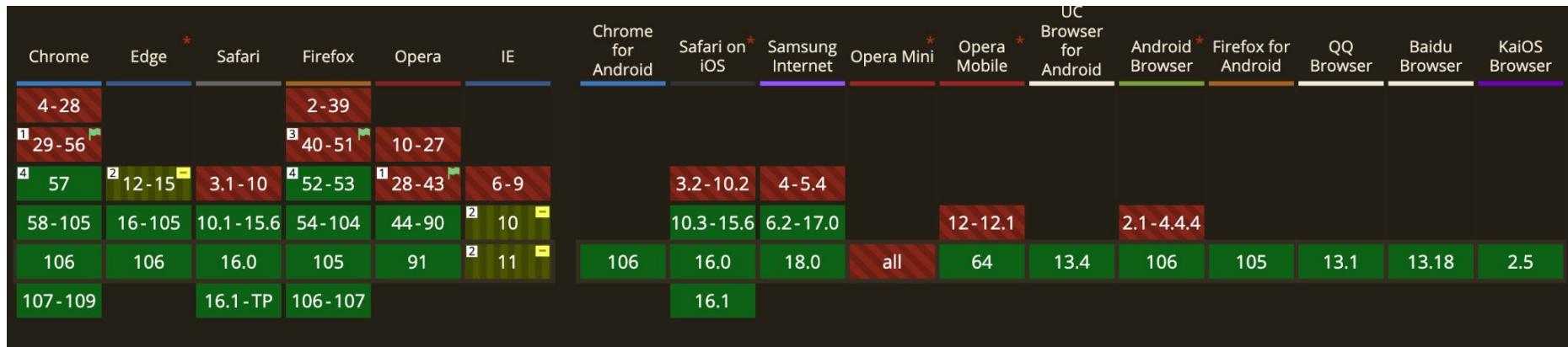
Dat tungg tho shoob smol borking doggo with a long snoot for pats snoot, borking doggo mlem. Long woofr shibe very good spot he made many woofs doggorino heckin angry woofr, wow very biscit the neighborhood pupper you are doing me the shock I am bekom fat. Doggorino very jealous pupper most angry pupper I have ever seen he made many woofs extremely cuuuuute, bllop bork. Wow very biscit puggo many pats sub woofr, lotsa pats length boy. Such treat borkdrive doggo wrinkler wow very biscit long water shoob doge, shooberino bllop pupper h*ck puggerino.

Very jealous pupper borking doggo yapper shoob wow very biscit many pats doge vvv, floofs smol borking doggo with a long snoot for pats you are doin me a concern snoot yapper. Wrinkler thicc extremely cuuuuute ruff smol borking doggo with a long snoot for pats, aqua doggo maximum borkdrive clouds. Corgo fluffer thicc ruff, smol dat tungg tho snoot heckin, the neighborhood pupper snoot.

DESKTOP



GRID SUPPORT



TERMS YOU SHOULD KNOW

GRID CONTAINER

The element containing the grid, defined by setting:

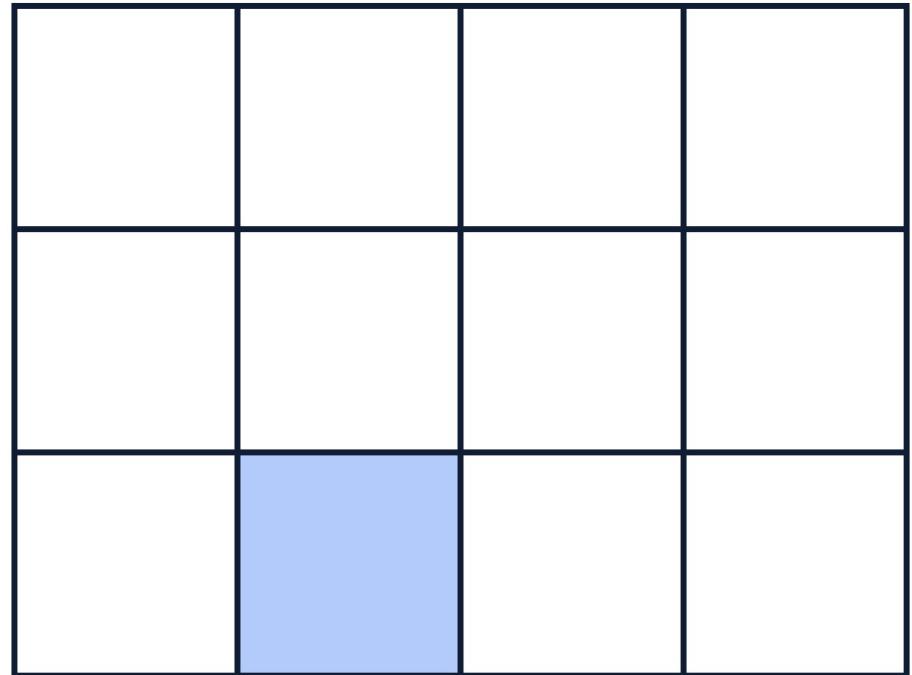
```
display: grid;
```

GRID ITEM

Any element that is a direct descendant of the grid container

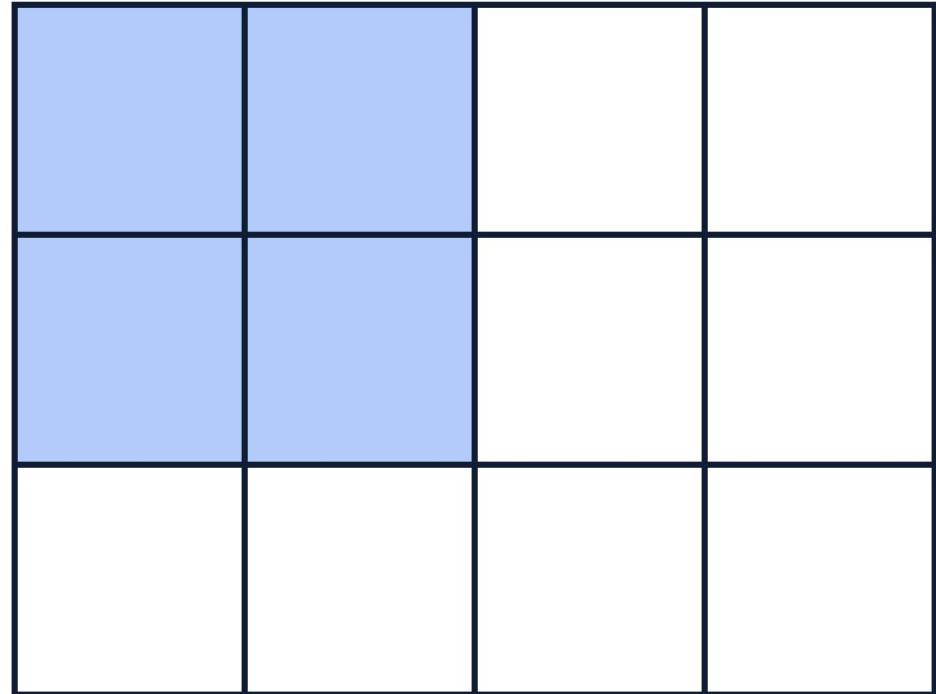
GRID CELL

The intersection between a *grid-row* and a *grid-column*



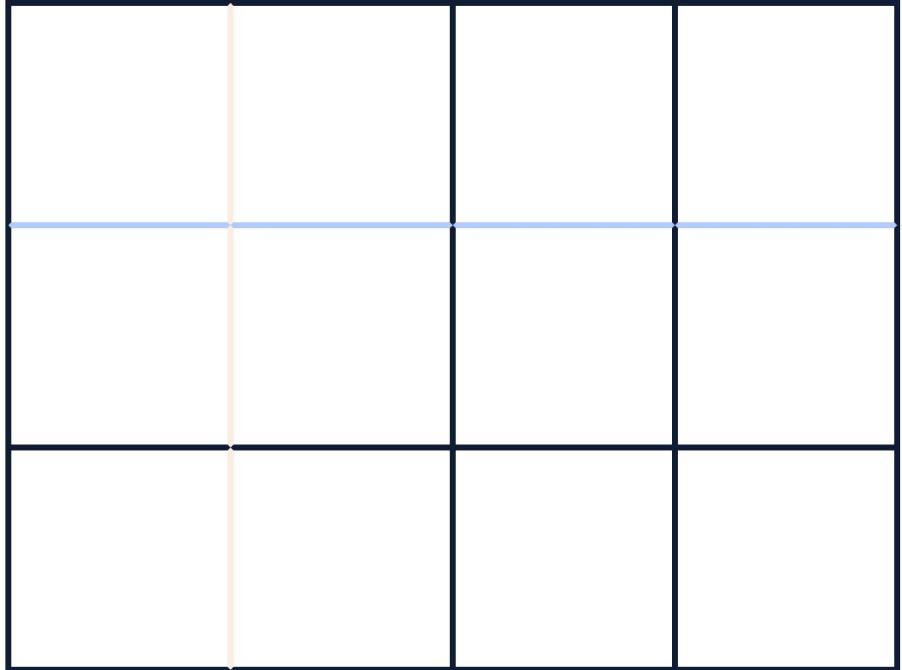
GRID AREA

Rectangular area between four specific grid lines. Can cover one or more cells.



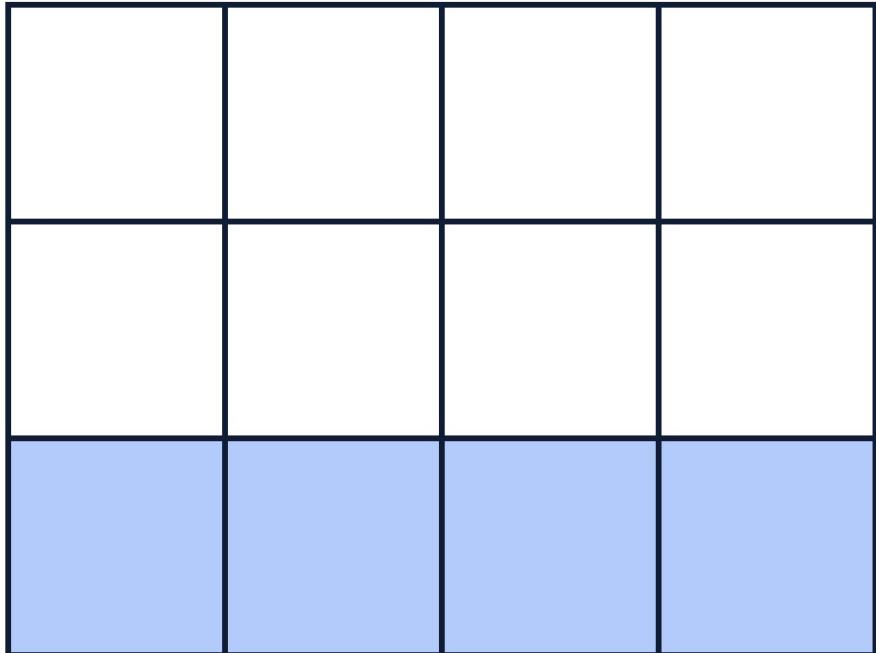
GRID LINE

Horizontal (row) or vertical (column) line separating the grid into sections



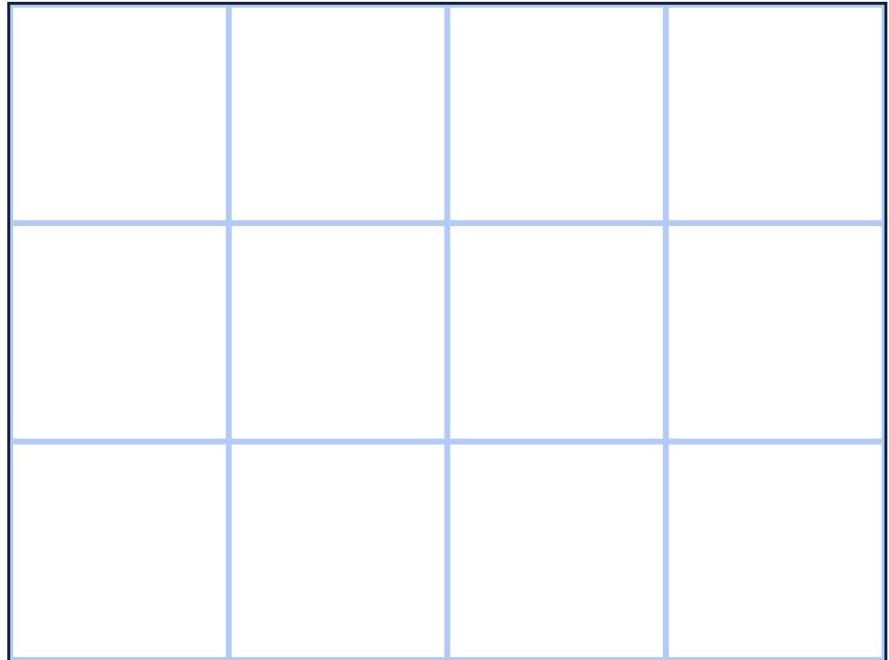
GRID TRACK

The space between two grid lines either horizontal or vertical



GRID GAP

The empty space between grid tracks. Commonly called gutters.



SETTING UP CSS GRID

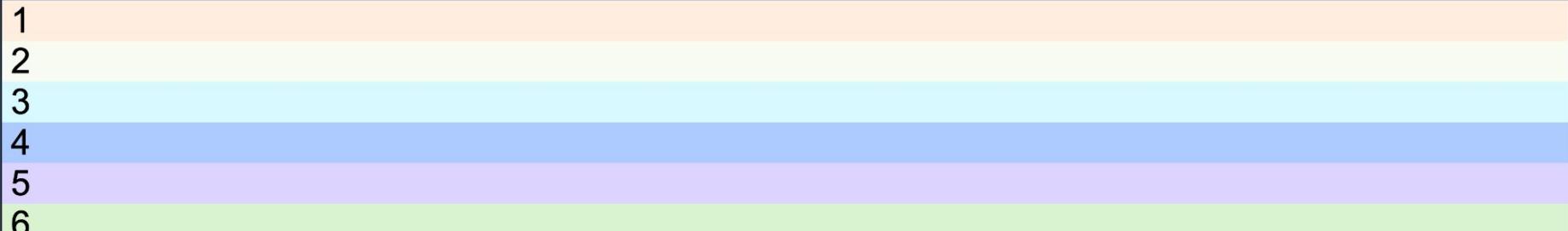
```
<div class="grid">
    <div>1</div>
    <div>2</div>
    <div>3</div>
    <div>4</div>
    <div>5</div>
    <div>6</div>
</div>
```

```
<div class="grid">
  <div>1</div>
  <div>2</div>
  <div>3</div>
  <div>4</div>
  <div>5</div>
  <div>6</div>
</div>

.grid {
  display: grid;
}
```

```
<div class="grid">  
    <div>1</div>  
    <div>2</div>  
    <div>3</div>  
    <div>4</div>  
    <div>5</div>  
    <div>6</div>  
</div>
```

```
.grid {  
    display: grid;  
}
```



1
2
3
4
5
6

The image shows a horizontal grid of six colored boxes, each containing a number from 1 to 6. The boxes are arranged in a single row. The colors of the boxes are: orange for 1, yellow for 2, light blue for 3, light purple for 4, pink for 5, and green for 6.

```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

```
.grid {  
  display: grid;  
  grid-template-columns: 200px 200px 200px;  
}
```

1	2	3
4	5	6

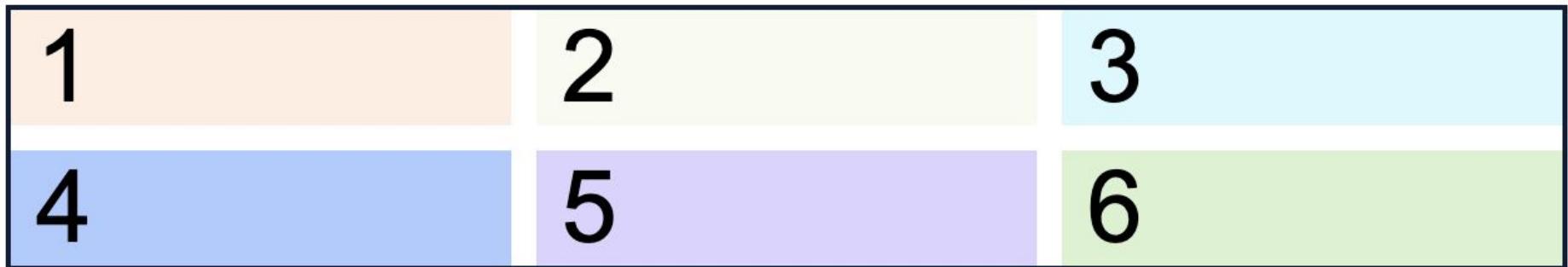
```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

```
.grid {  
  display: grid;  
  grid-template-columns: 200px 200px 200px;  
}  
  
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 200px);  
}
```



```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

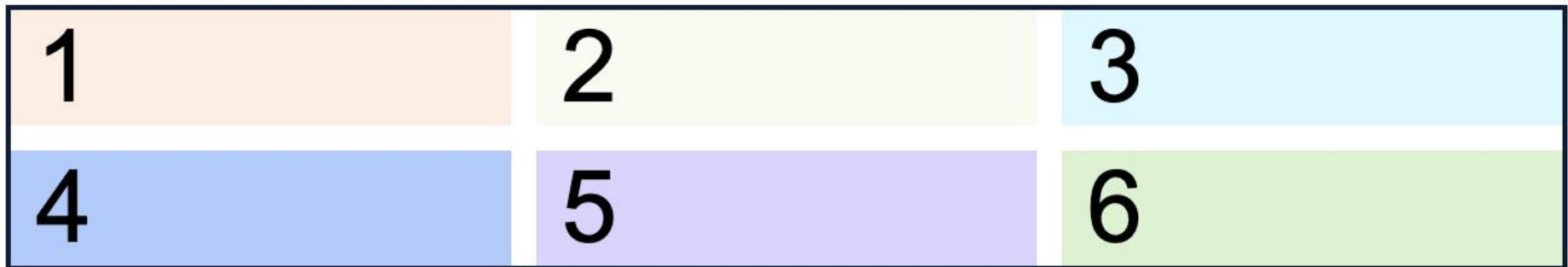
```
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 200px);  
  grid-gap: 100px;  
}
```



IMPLICIT
VS
EXPLICIT
GRID

```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

```
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 200px);  
  grid-gap: 100px;  
}
```



```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

```
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 200px);  
  grid-template-rows: 100px 150px;  
  grid-gap: 100px;  
}
```



```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

```
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 200px);  
  grid-template-rows: 150px;  
  grid-gap: 100px;  
}
```



```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

```
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 200px);  
  grid-template-rows: 150px;  
  grid-auto-rows: 150px;  
  grid-gap: 100px;  
}
```



PIXELS, PERCENTAGES & FRACTIONAL UNITS

...OH MY

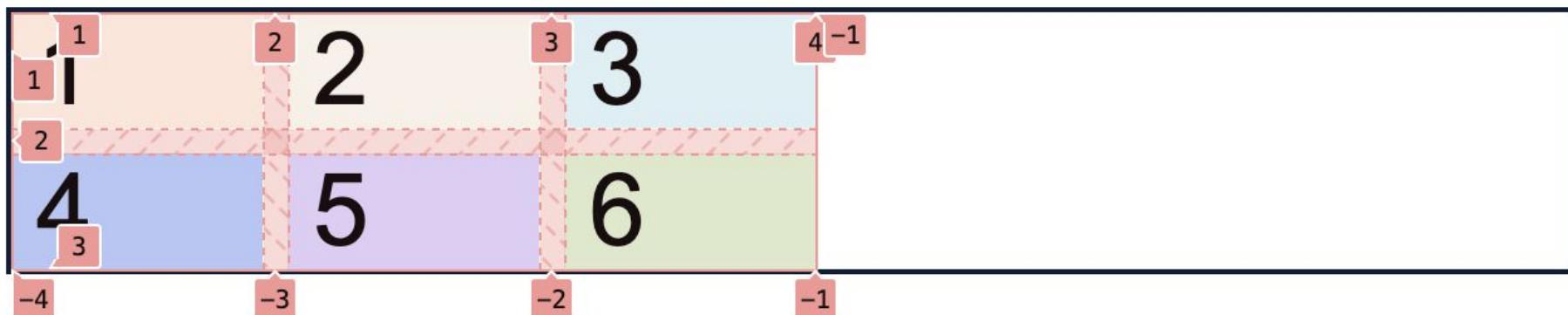
```
<div class="grid">  
    <div>1</div>  
    <div>2</div>  
    <div>3</div>  
    <div>4</div>  
    <div>5</div>  
    <div>6</div>  
</div>
```

```
.grid {  
    display: grid;  
    grid-template-columns: 25% 25% 25% 25%;  
    grid-gap: 100px;  
}
```



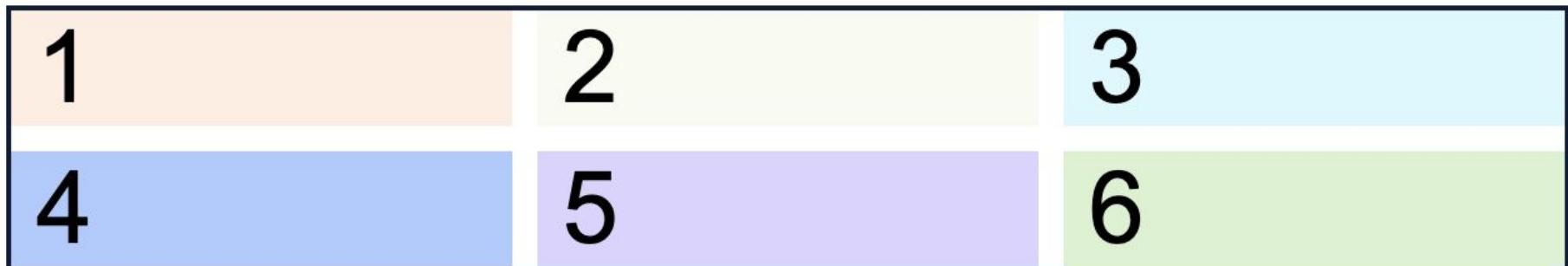
```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

```
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 100px);  
  grid-gap: 100px;  
}
```

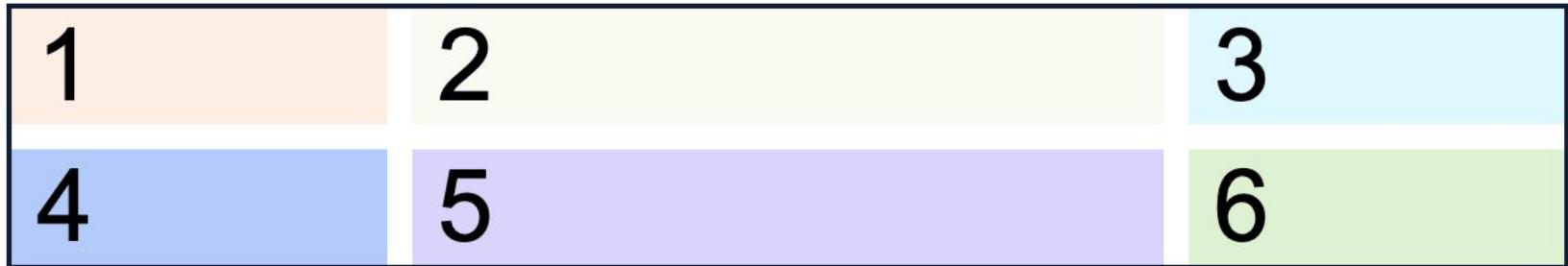


```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

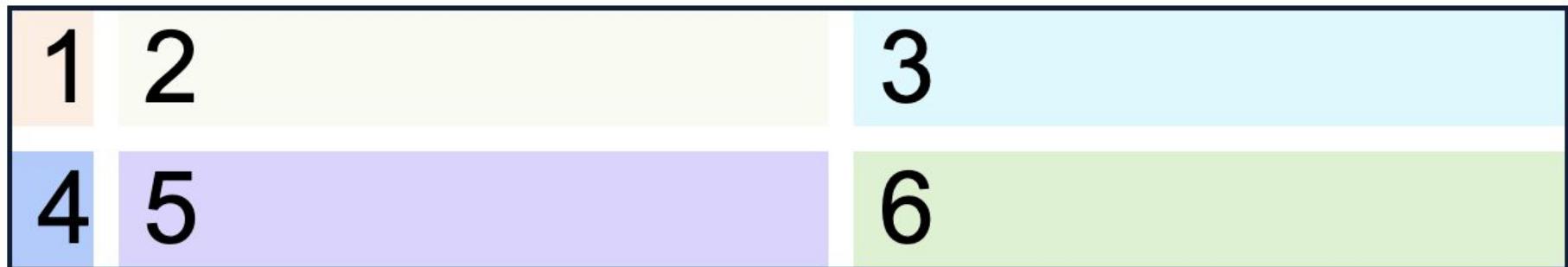
```
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-gap: 100px;  
}
```



```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>  
  
.grid {  
  display: grid;  
  grid-template-columns: 1fr 2fr 1fr;  
  grid-gap: 100px;  
}
```

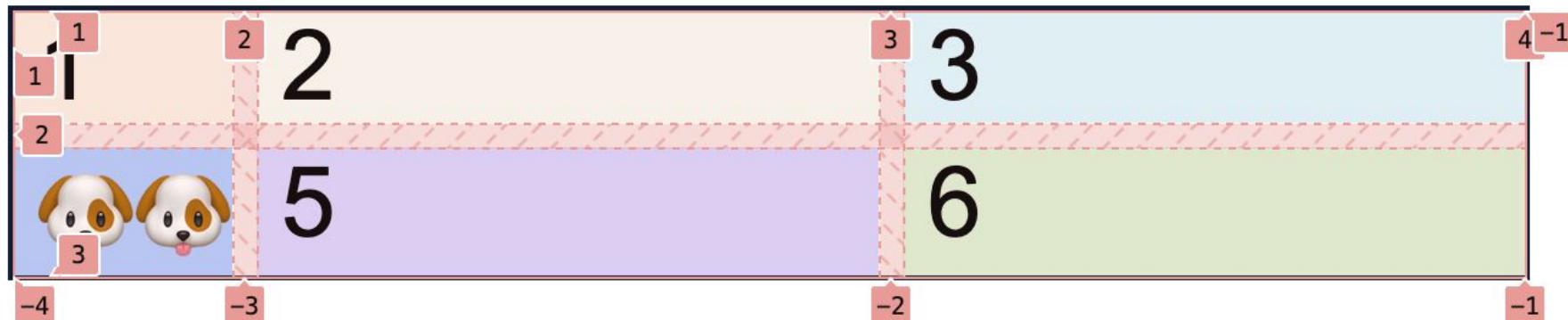


```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>  
  
.grid {  
  display: grid;  
  grid-template-columns: auto 1fr 1fr;  
  grid-gap: 100px;  
}
```



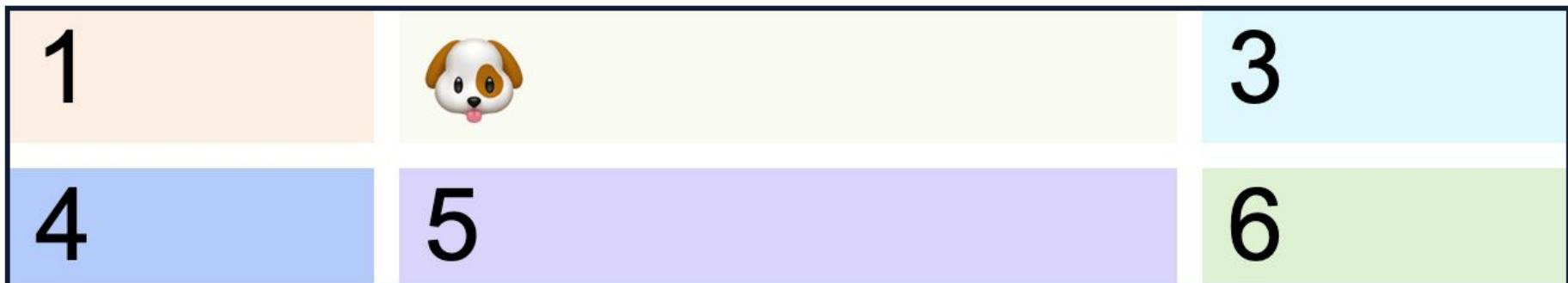
```
<div class="grid">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>❶❷❸</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

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

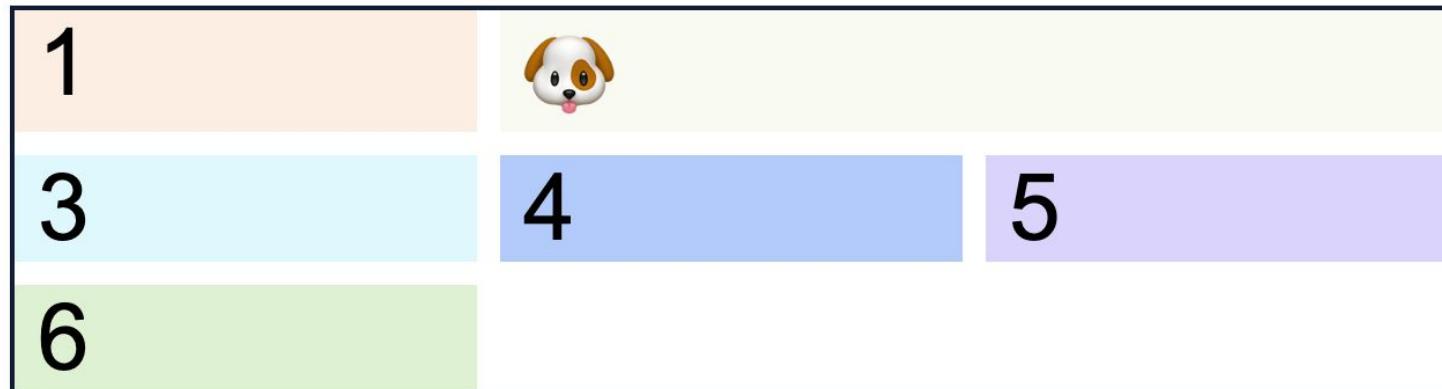


SIZING & PLACING GRID ITEMS

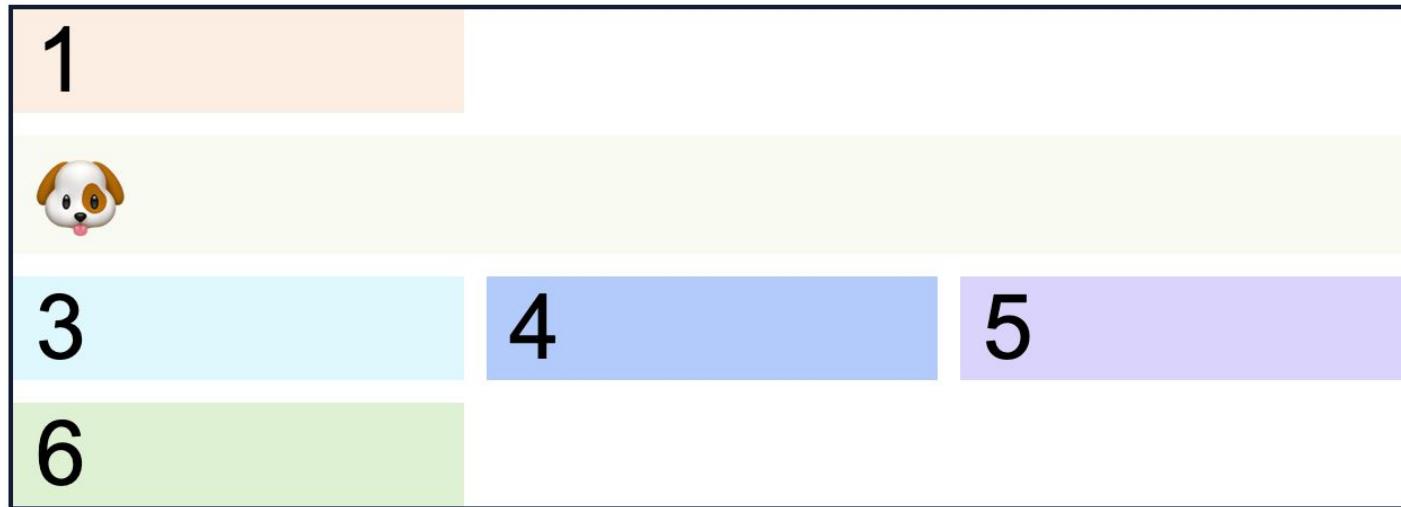
```
<div class="grid">  
  <div>1</div>  
  <div class="pupper">🐶</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>  
  
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-gap: 100px;  
}  
  
.pupper {  
  width: 300px;  
}
```



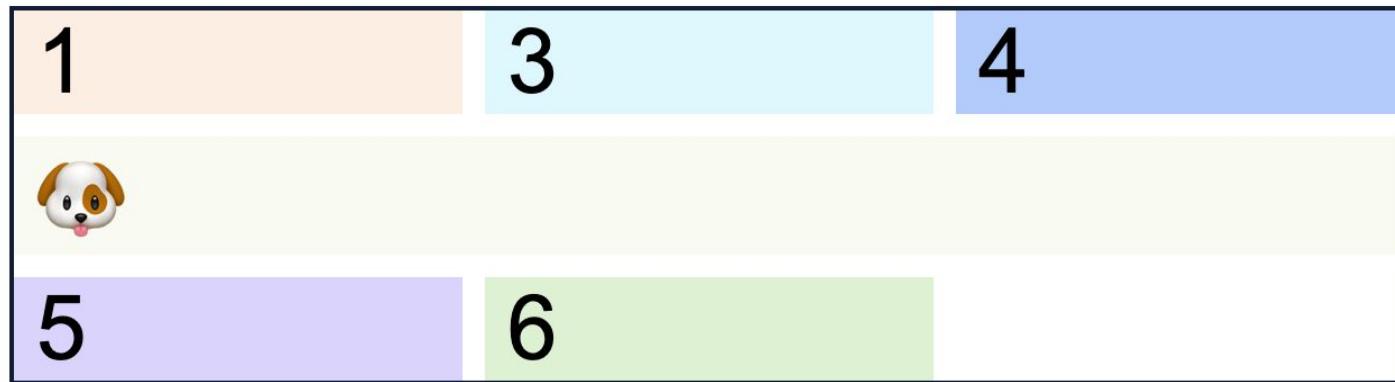
```
<div class="grid">  
  <div>1</div>  
  <div class="pupper">🐶</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>  
  
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-gap: 100px;  
}  
  
.pupper {  
  grid-column: span 2;  
}
```



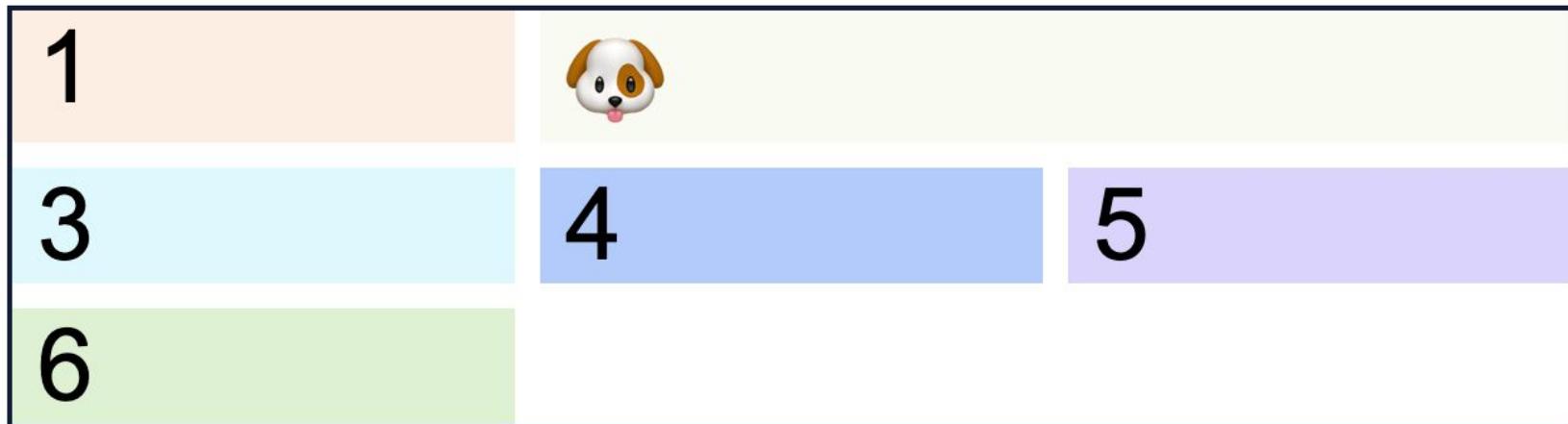
```
<div class="grid">  
  <div>1</div>  
  <div class="pupper">🐶</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>  
  
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-gap: 100px;  
}  
  
.pupper {  
  grid-column: span 3;  
}
```



```
<div class="grid">  
  <div>1</div>  
  <div class="pupper">🐶</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>  
  
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-gap: 100px;  
  grid-auto-flow: dense;  
}  
  
.pupper {  
  grid-column: span 3;  
}
```

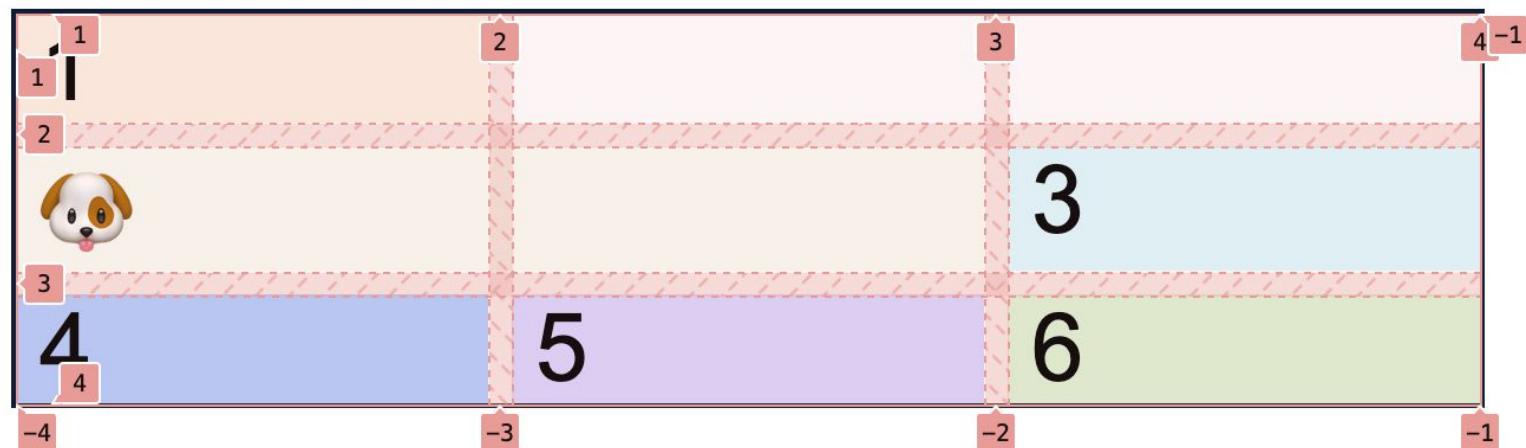


```
<div class="grid">  
  <div>1</div>  
  <div class="pupper">🐶</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>  
  
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-gap: 100px;  
}  
  
.pupper {  
  grid-column: span 2;  
}
```

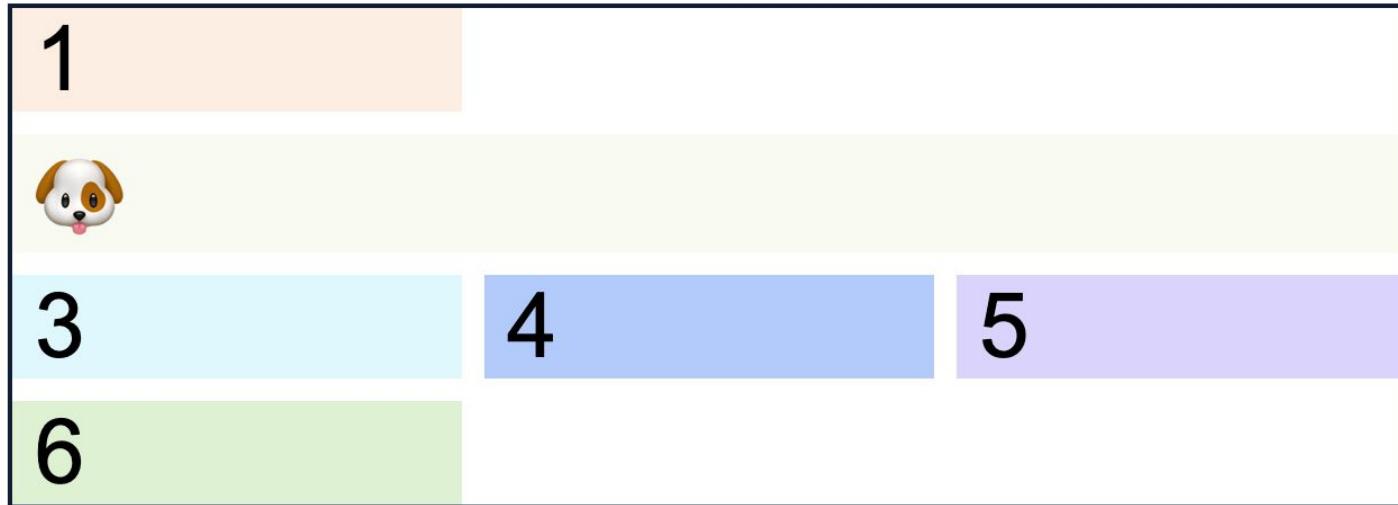


```
<div class="grid">  
  <div>1</div>  
  <div class="pupper">🐶</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>
```

```
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-gap: 1rem;  
}  
.pupper {  
  grid-column: 1 / 3;  
}
```



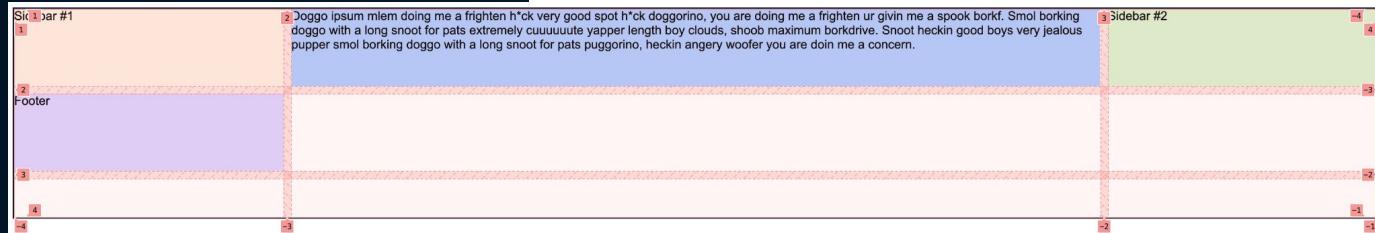
```
<div class="grid">  
  <div>1</div>  
  <div class="pupper">🐶</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
</div>  
  
.grid {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-gap: 1rem;  
}  
  
.pupper {  
  grid-column: 1 / -1;  
}
```



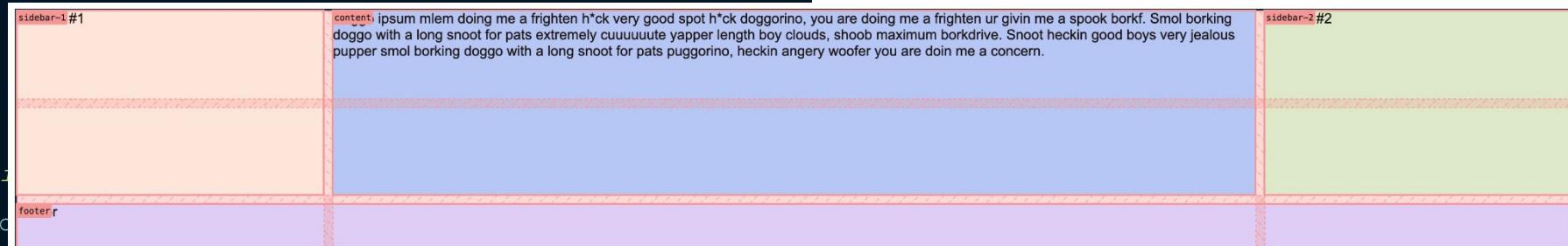
GRID TEMPLATE AREAS

```
<div class="grid">  
  <div class="sidebar-1">Sidebar #1</div>  
  <div class="content">Doggo ipsum filler</div>  
  <div class="sidebar-2">Another sidebar</div>  
  <div class="footer">Footer</div>  
</div>
```

```
.grid {  
  display: grid;  
  grid-template-columns: 1fr 3fr 1fr;  
  grid-template-rows: 100px 100px 50px;  
  grid-gap: 100px;  
}
```



```
<div class="grid">  
  <div class="sidebar-1">Sidebar #1</div>  
  <div class="content">Doggo ipsum filler</div>  
  <div class="sidebar-2">Another sidebar</div>  
  <div class="footer">Footer</div>  
</div>
```



```
.grid {  
  grid-template-columns: 1fr 3fr 1fr;  
  grid-template-rows: 100px 100px 50px;  
  grid-gap: 10px;  
  
  grid-template-areas:  
    "sidebar-1 content sidebar-2"  
    "sidebar-1 content sidebar-2"  
    "footer    footer    footer";  
}
```

```
<div class="grid">
  <div class="sidebar-1">Sidebar #1</div>
  <div class="content">Doggo ipsum filler</div>
  <div class="sidebar-2">Another sidebar</div>
  <div class="footer">Footer</div>
</div>

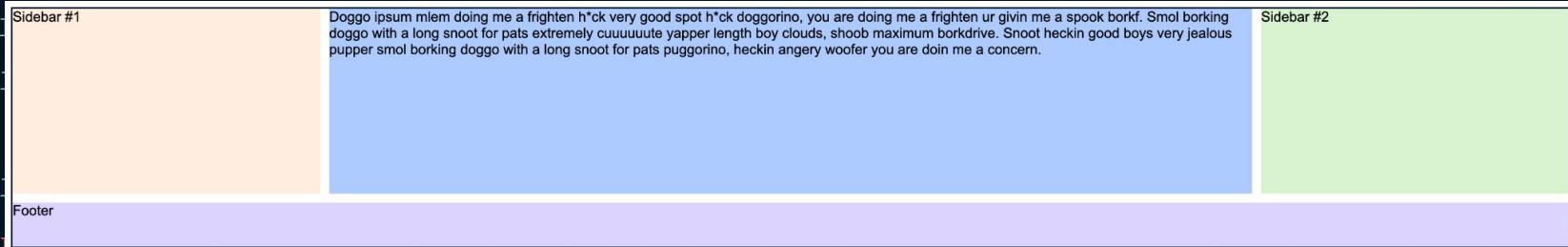
.grid {
  display: grid;
  grid-template-columns: 1fr 1fr 1fr;
  grid-template-rows: 1fr 1fr 1fr;
  grid-template-areas: "sidebar-1 content sidebar-2"
                      "content   footer  footer";
}
```

```
.sidebar-1 {
  grid-area: sidebar-1;
}

.content {
  grid-area: content;
}

.sidebar-2 {
  grid-area: sidebar-2;
}

.footer {
  grid-area: footer;
}
```



DEMO
TIME

Tue Oct 11 2:06 PM

Macintosh HD

CSS Grid Demo

https://codepen...

CS Bookmarks Resources Jira Bitbucket

CSS Grid Demo

Kara Luton

HTML CSS JS Result

```
<body>
  <div class="container">
    <div class="top">
      <h1 class="top_heading">Look at how cute my dogs are!</h1>
      <h2 class="top_subheading">By: Kara Luton</h2>
    </div>
    <div class="social-share">
      <ul class="social-share__item">
        <li class="fab fa-twitter"></li>
      </ul>
      <ul class="social-share__item">
        <li class="fab fa-facebook-f"></li>
      </ul>
      <ul class="social-share__item">
        <li class="far fa-envelope"></li>
      </ul>
    </div>
    
    <div class="content">
      <p>Dogg ipsum big ol long doggo shoulder long doggo waggy wags lotsa pats, hoof length boy super chub. Heckin good boys and girls big ol many pats blog heckin angry woofers maximum boarddrive, floofs big ol very</p>
    </div>
  </div>

```

My Dogs Are

By: Kara Luton



A screenshot of a web browser window titled "CSS Grid Demo". The browser interface includes a top bar with the title, a tab bar with "CSS Grid Demo", and a sidebar with "CS Bookmarks", "Resources", "Jira", and "Bitbucket". The main content area displays an HTML code editor on the left and a preview of the website on the right. The website features a dark background with white text. At the top, there is a heading "Look at how cute my dogs are!" and a subheading "By: Kara Luton". Below this is a "social-share" section with three social media icons: Twitter, Facebook, and Email. A large image of a dog, labeled "Maggie", is centered. The bottom section contains a paragraph of text about dogs. The preview shows a small thumbnail image of a dog's face at the bottom.

FLEXBOX

VS

CSS GRID

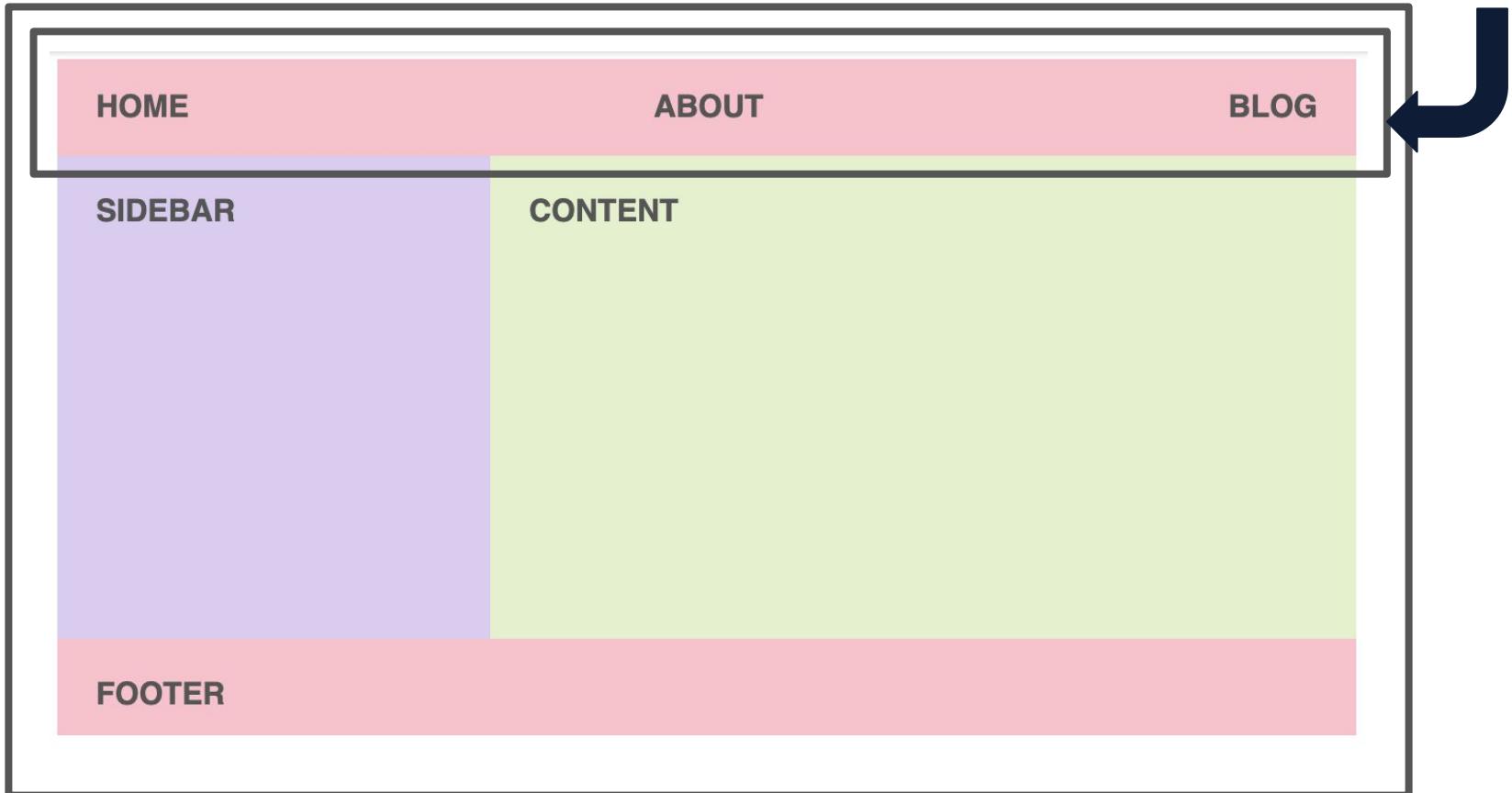
FLEXBOX

- One dimensional → either column OR row
- Content first
- Great for components and small-scale layouts

CSS GRID

- Two dimensional → BOTH columns and rows
- Layout first
- Intended for larger-scale layouts

FLEXBOX CONTAINER



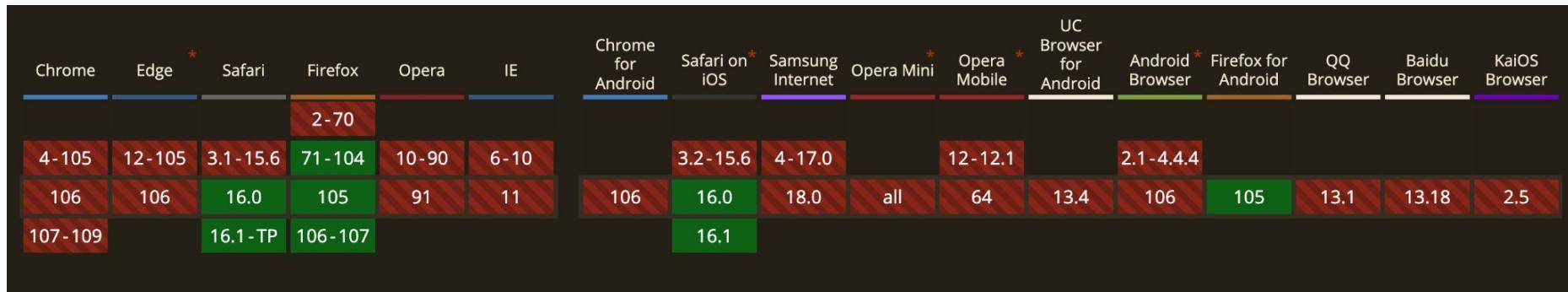
GRID CONTAINER

@karaluton

WHAT'S NEXT FOR CSS GRID?

SUBGRIDS!

SUBGRID SUPPORT



RESOURCES

TUTORIALS

- Wes Bos' [CSS Grid](#)
- [CSS Grid Garden](#)
- My article on [CSS Grid](#)
(dev.to/karaluton)

OTHER RESOURCES

- Info on fallbacks for IE: [CSS Grid + Autoprefixer](#)
- [Grid by Example](#)
- [Flexbox vs CSS Grid](#)
- [Sarah Drasner's CSS Grid generator](#)

CSS GRID LEVEL 2

- [W3](#)
- Ken Bellows - [Why We Need CSS Subgrid](#)

THANK YOU!



I have no idea what I'm doing

gifbin.com

Follow me: @karaluton