

CSS Grid

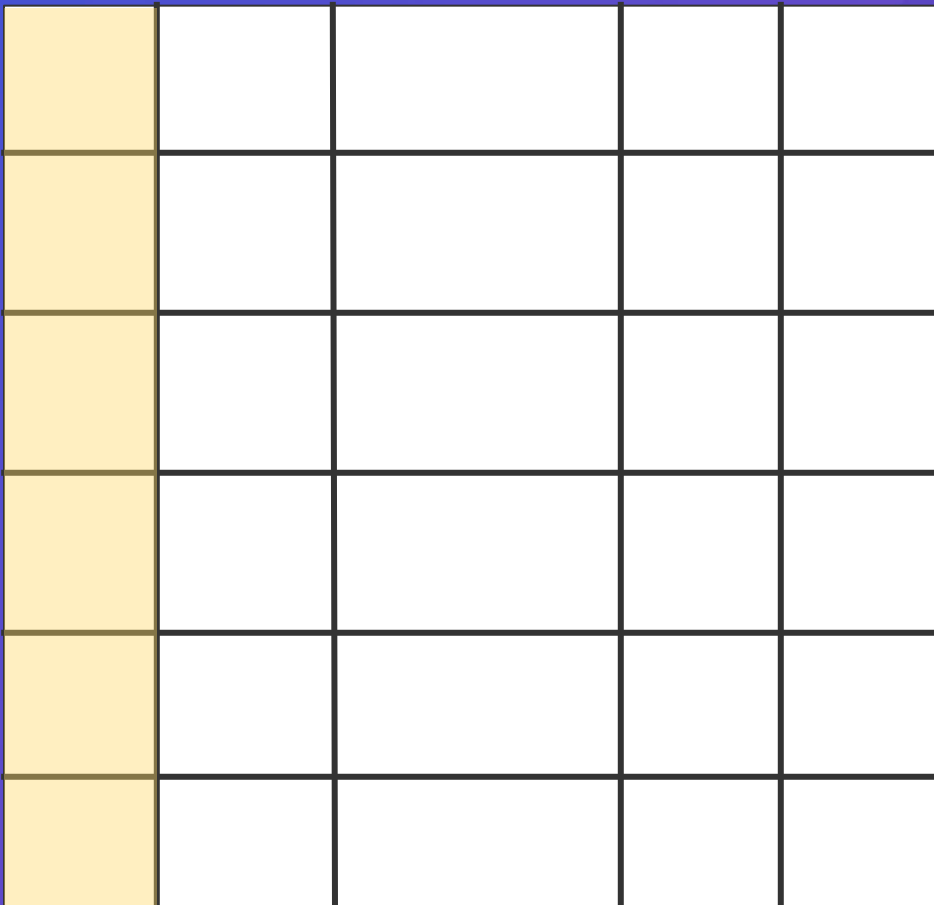
Front End Workshop

CSS Grid



Lets us take any div or element on a page and opens up the ability to create an actual grid

CSS Grid



```
display: grid;
```

We can define how many rows and columns are in our grid

i.e. **6 rows**, 5 columns

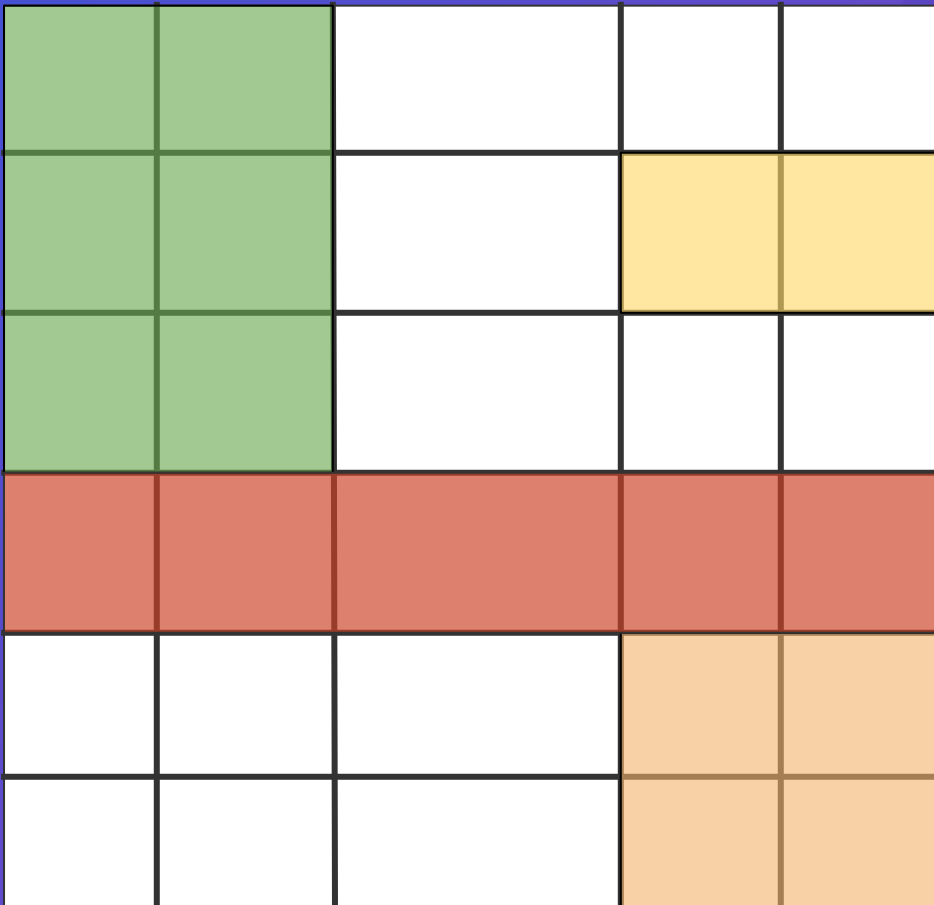
CSS Grid


```
display: grid;
```

We can define how many rows and columns are in our grid

i.e. 6 rows, **5 columns**

CSS Grid



Once we have our grid defined, we can now place elements wherever we want on the grid space

Defining Grid Container

grid-template-columns
grid-template-rows
grid-gap

defines the number of columns in the grid layout
defines the number of rows in the grid layout
defines the spacing between grid items (like margin)

```
.container {  
  display: grid;  
  grid-gap: 10px;  
  grid-template-columns: 100px auto 100px;  
  grid-template-rows: 100px 70px;  
}
```

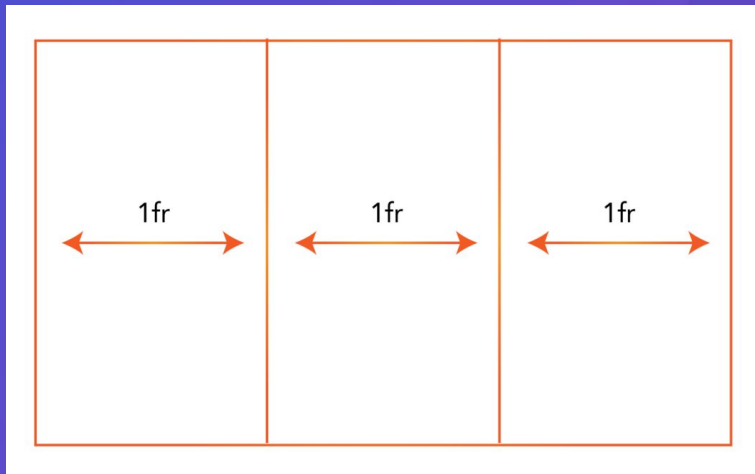


Understanding fr (fraction) Unit

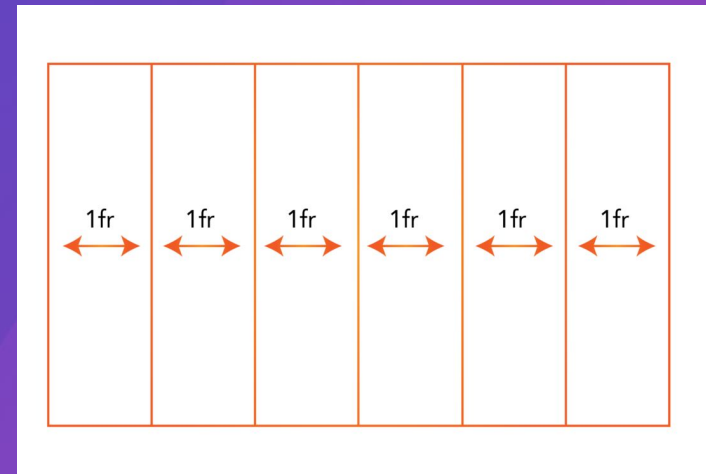
Measures space like **em**, **rem**, **px**, **%**

1fr = 1 part of the available space

If you specify a width of **1fr**, you can go on to add as many child elements as possible and it will be taken care of. The widths will be equally shared across all child elements.



3 child elements



6 child elements

Understanding fr (fraction) Unit

```
.container {  
  display: grid;  
  grid-gap: 10px;  
  grid-template-columns: 2fr 1fr 1fr;  
}
```

Two parts are given to the first track and one part each to the next two tracks.

1**2****3****4****5****6****7****8****9****10**

Repeat Function

Instead of typing 1fr a bunch of times, we can use repeat()

```
.container {  
  display: grid;  
  grid-gap: 10px;  
  grid-template-columns: 1fr 1fr 1fr;  
}
```

```
.container {  
  display: grid;  
  grid-gap: 10px;  
  grid-template-columns: repeat(3, 1fr);  
}
```



Sizing Grid Items

<https://stackblitz.com/edit/grid-sizing>

What if we want to size individual grid items? Or groups of items? Use **span**!

```
.item {  
  grid-column: span 2;  
}
```

Placing Grid Items

<https://stackblitz.com/edit/grid-sizing>

What if we want to control where grid items are placed in the layout? Use **start** / **stop**

```
.item {  
  grid-column: 2 / 5;  
}
```

Tip:

- Adding the “stop” value as -1 will extend to the end of the grid

Responsive Grids

No more media queries!

```
.container {  
  display: grid;  
  grid-gap: 10px;  
  grid-template-columns: repeat(auto-fit, minmax(min, max));  
}
```

auto-fit

minmax(min, max)

of columns/rows that can fit inside the container
calculates minimum size and maximum size an element can be

Responsive Grids

We use **auto-fit** and **minmax** to create responsiveness.

```
.container {  
  display: grid;  
  grid-gap: 10px;  
  grid-template-columns: repeat(auto-fit, minmax(150px, 1fr));  
}
```

We are repeating each column until it reaches the end of the grid

&

Giving each column a minimum width of 150px, and max width of 1fr (width of the grid)

BONUS

<https://twitter.com/wesbos/status/928731575684091904?lang=en>



Wes Bos 🔥 
@wesbos

Follow



Bootstrap 12 column grid in 5 lines of CSS.

One of many ways to lay out of a page in CSS grid. You won't be needing frameworks for layout much longer 😊

```
.wrapper {  
  display: grid;  
  grid-template-columns: repeat(12, 1fr);  
  grid-gap: 10px;  
}  
  
.sidebar {  
  grid-column: span 3;  
}  
  
.content {  
  grid-column: span 9;  
}
```

Grid Practice

Link to practice exercise: <https://stackblitz.com/edit/grid-cardio>

Answer: <https://stackblitz.com/edit/grid-cardio-answer>