

# Flexbox

---



**Brian Treece**

CHIEF OF USER EXPERIENCE AT SOCREATE.IT

@brianmtreece

# What is Flexbox?

A Set of CSS  
Properties

Creating  
Flexible  
Layouts

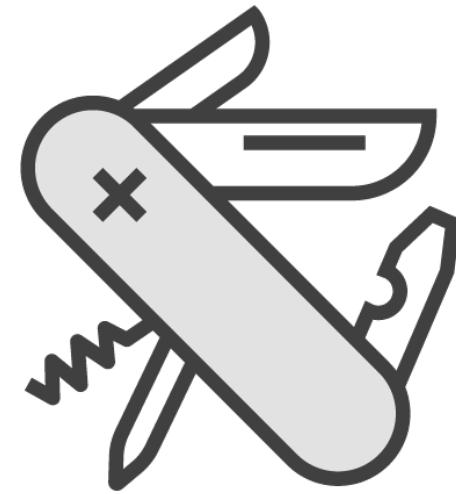
Distributing  
Extra Space

Aligning  
Content

**Not Flexible Enough!**

Block  
Inline  
Table  
Positioning

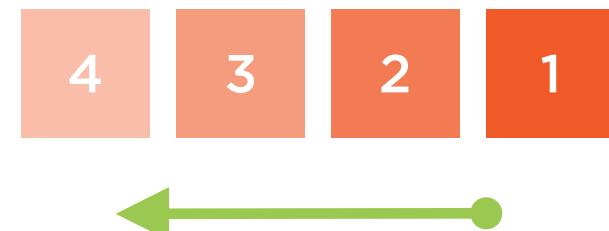
# Flexbox!



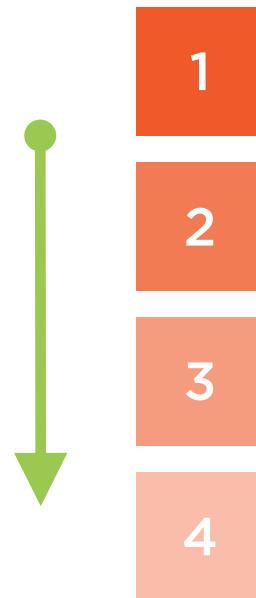
# Flexbox!



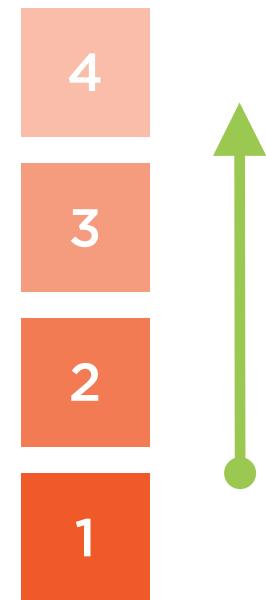
# Flexbox!



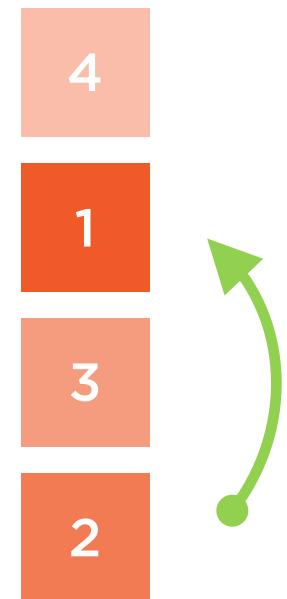
# Flexbox!



# Flexbox!



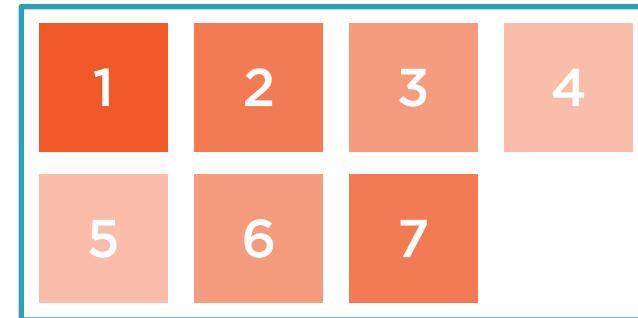
# Flexbox!



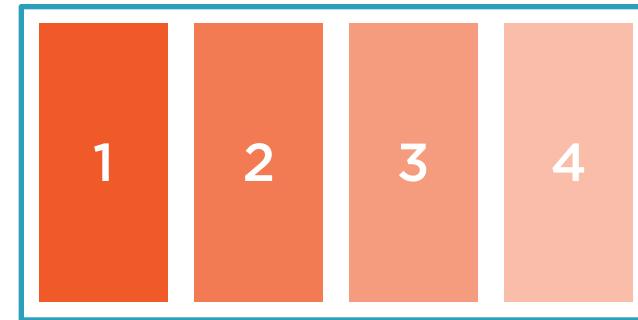
# Flexbox!



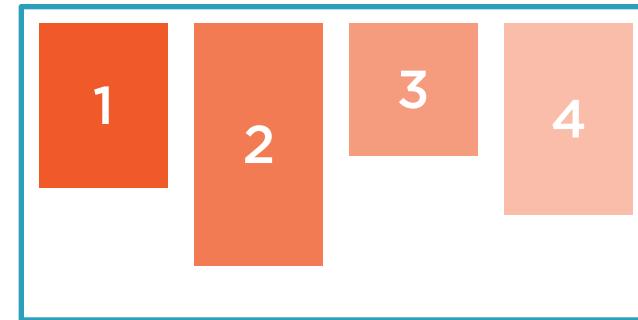
# Flexbox!



# Flexbox!



# Flexbox!



# Flexbox The Basics

---

1

2

3

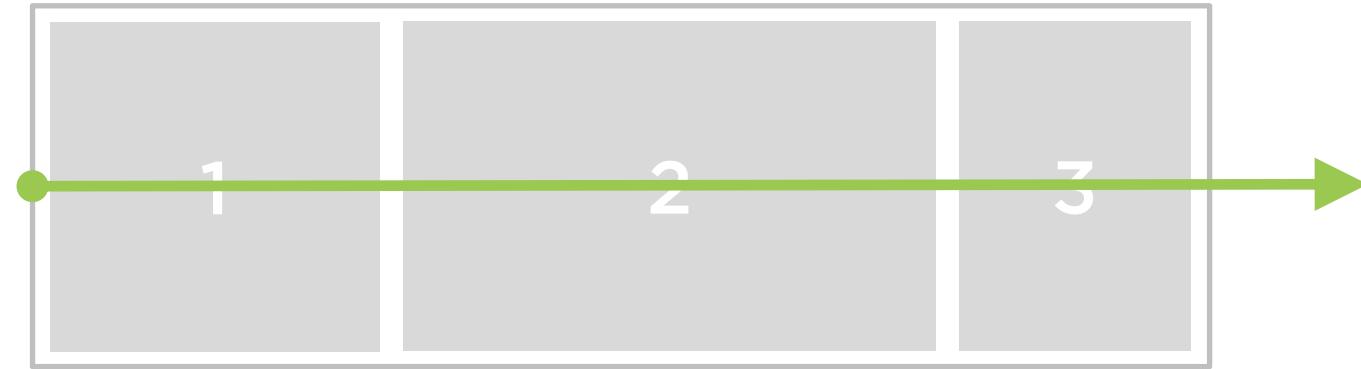
# Flex Container



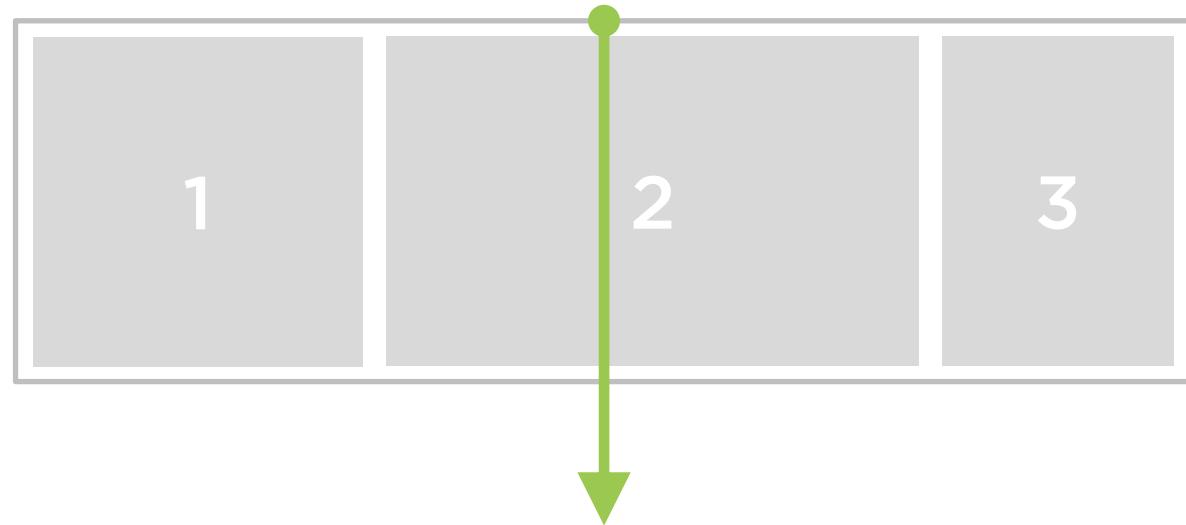
# Flex Items



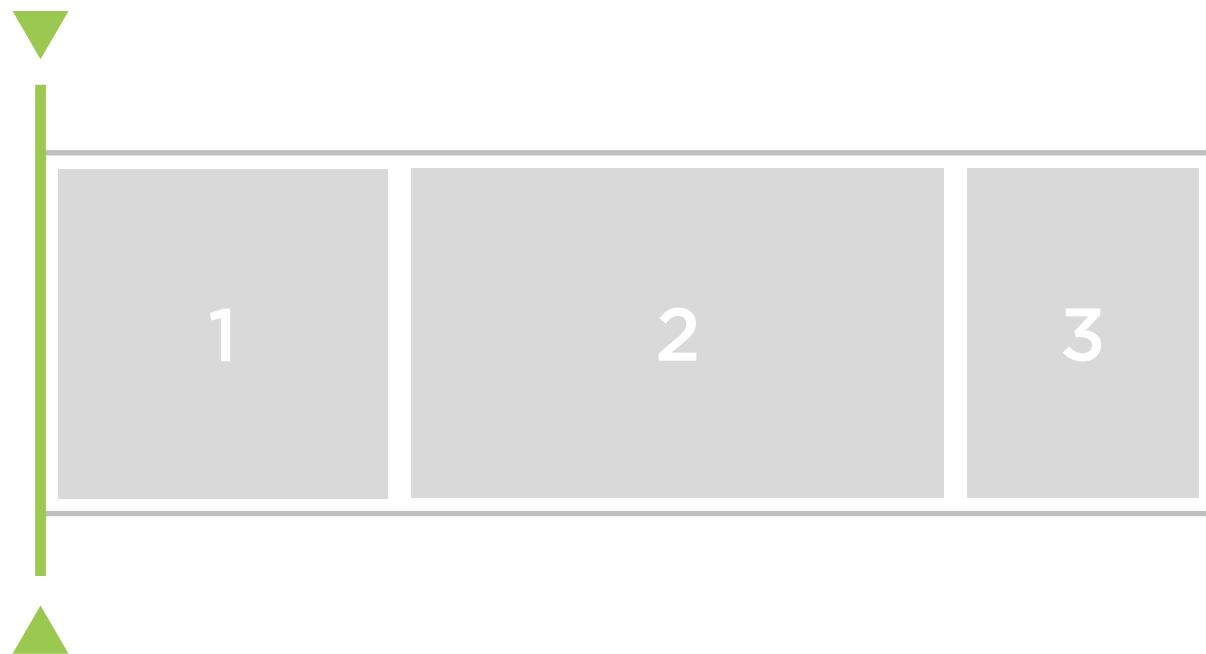
# Main Axis



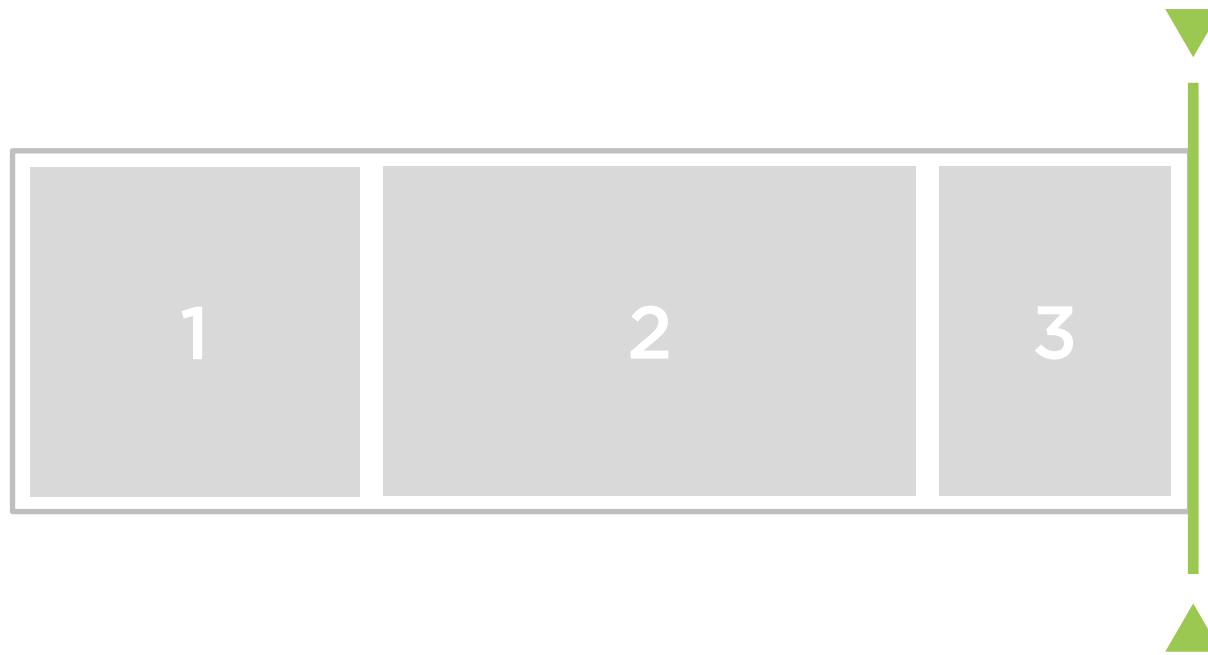
# Cross Axis



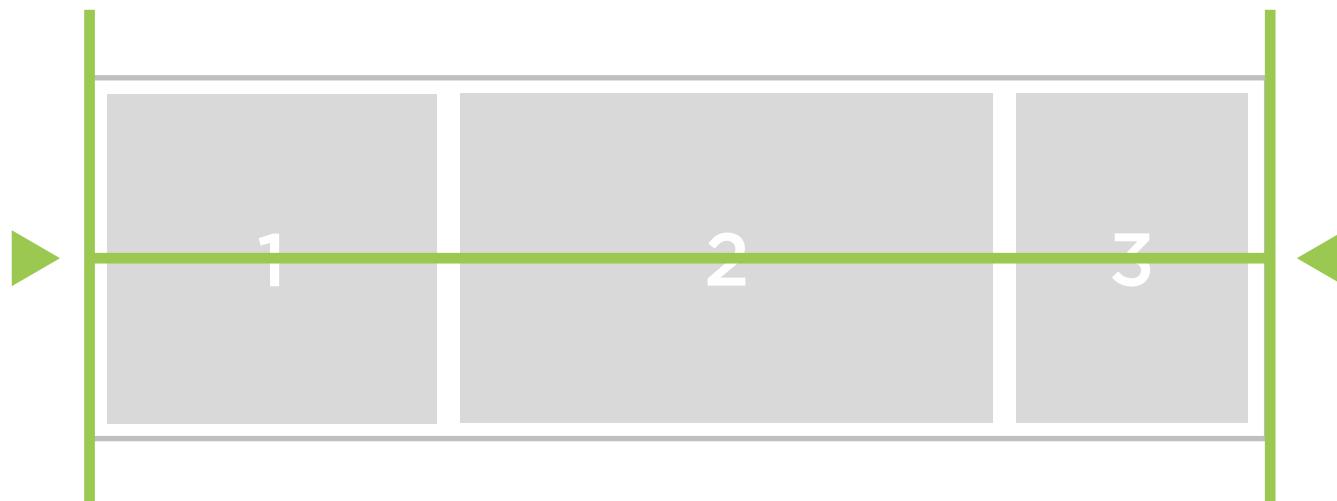
# Main Start



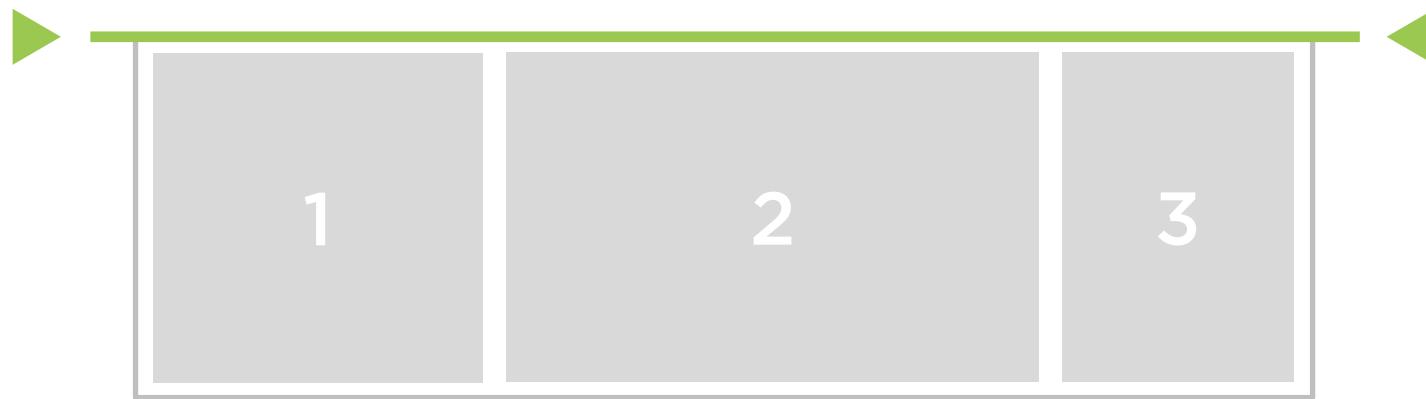
# Main End



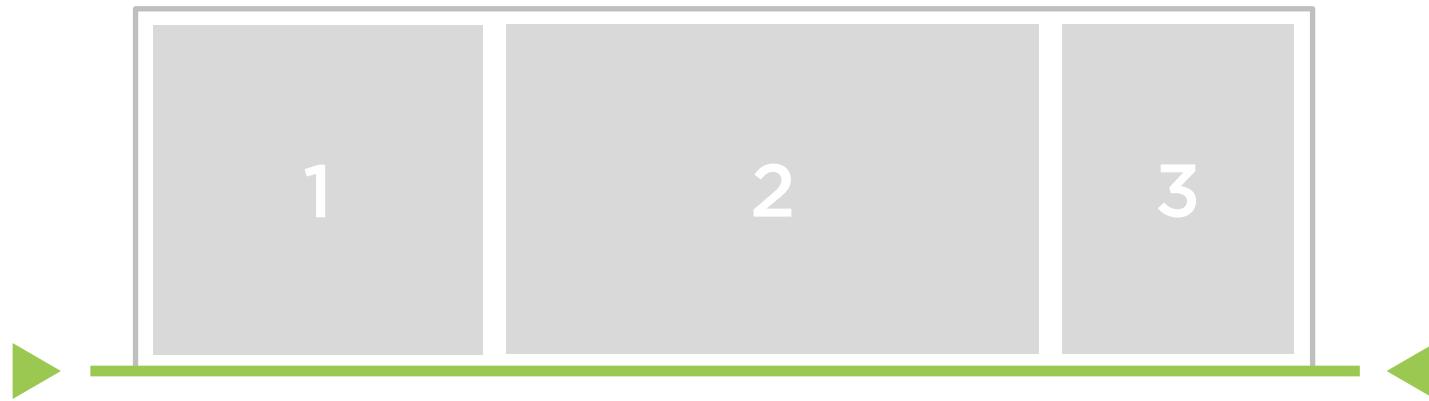
# Main Size



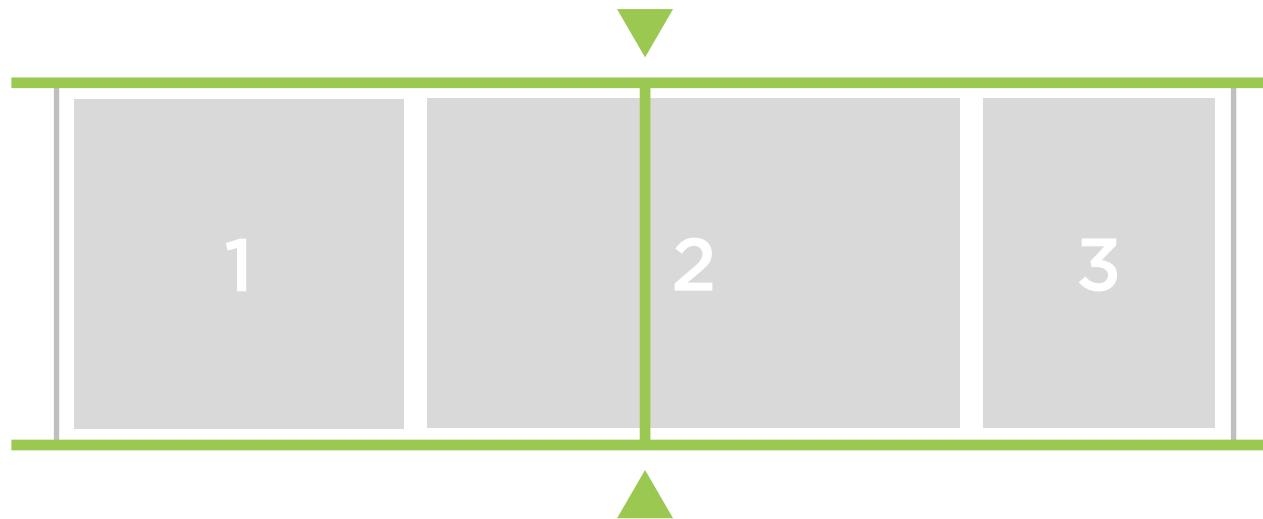
# Cross Start



# Cross End



# Cross Size



# The Flex Container

---

# Flex Container

display

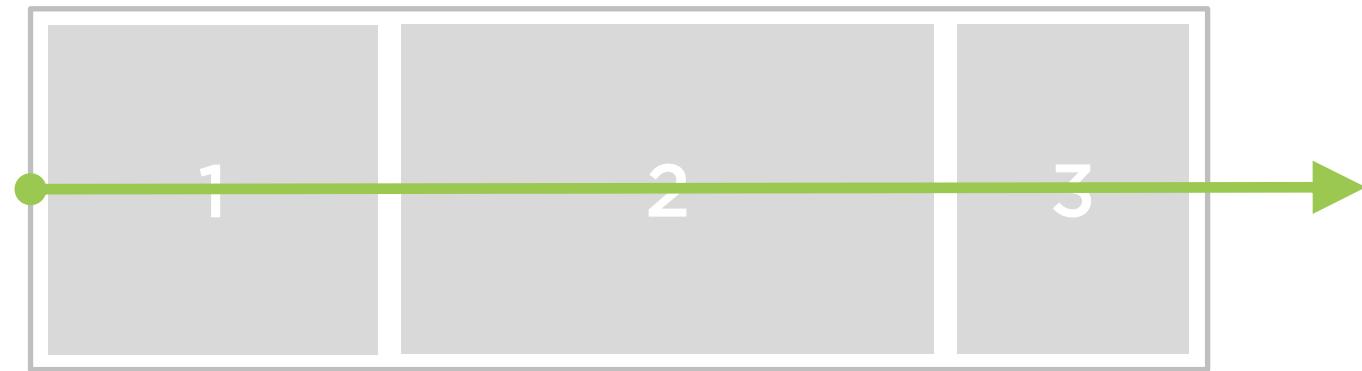
# Flex Container

display: flex;

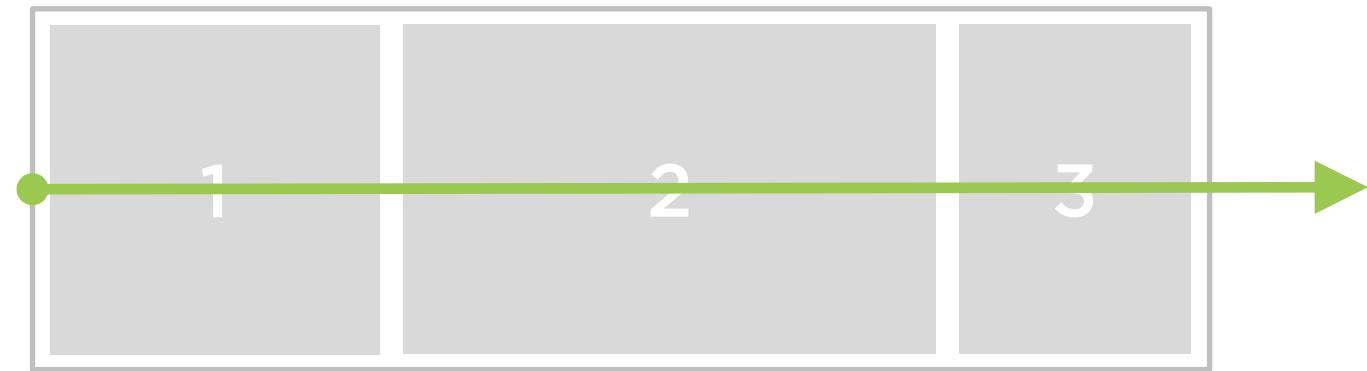
# Flex Flow Direction

---

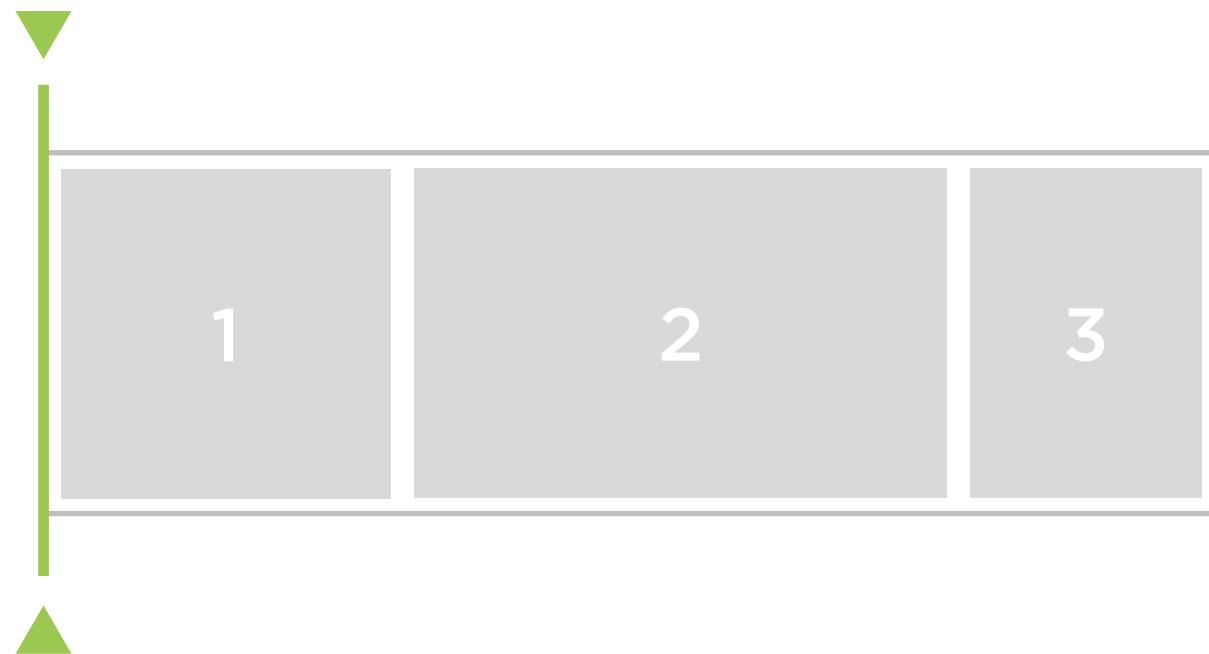
# Flex Items Flow Left to Right



This is the Main Axis



# Flex Start



# Flex End







# Flex Line Wrapping

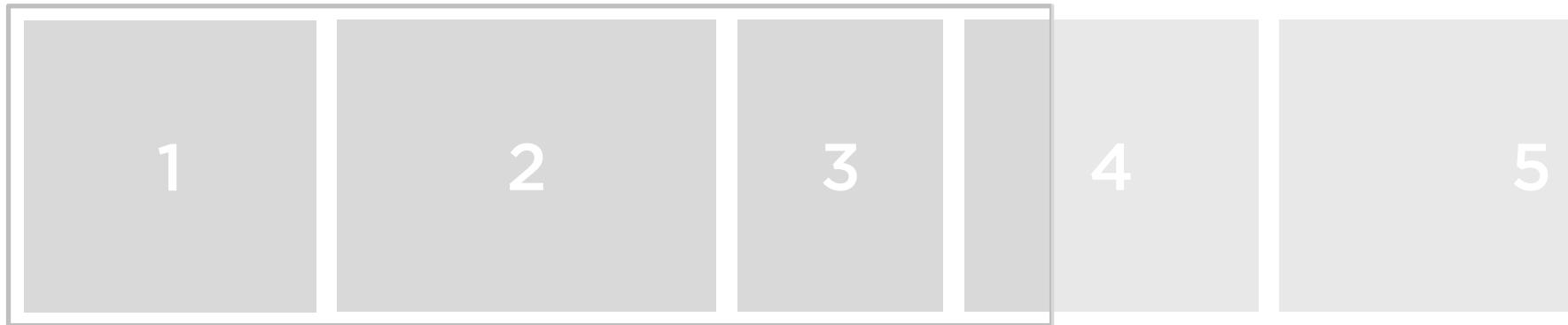
---



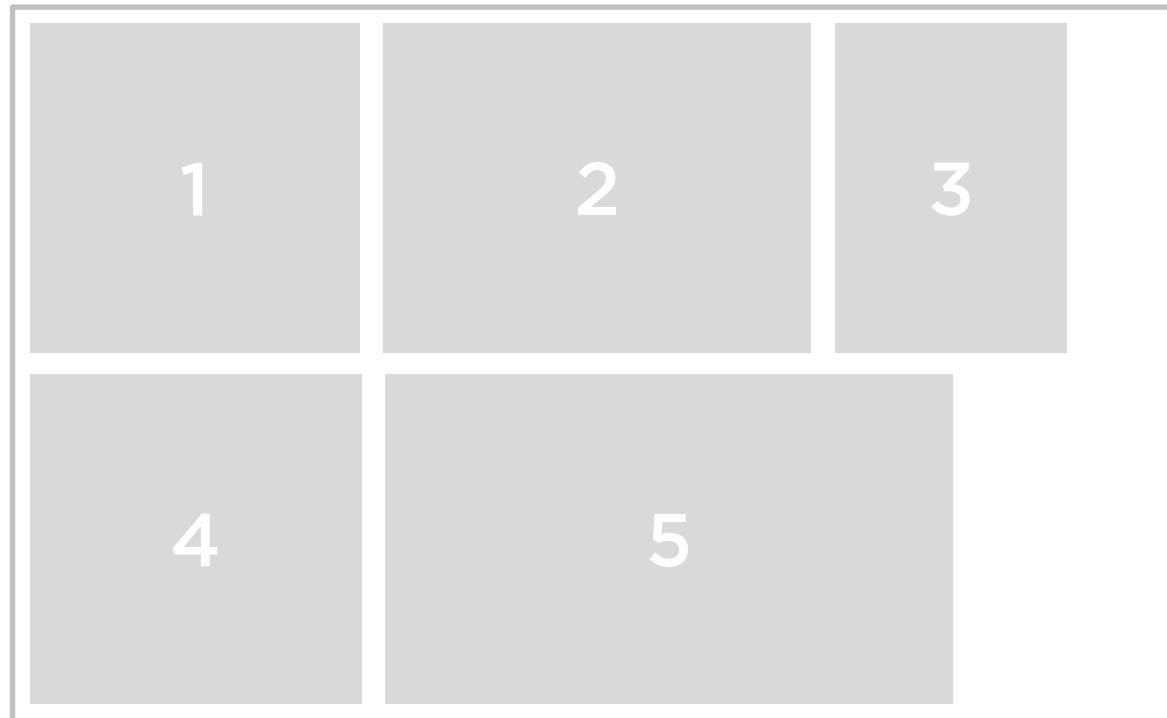
`text-wrap`

Flex Line Wrapping is Familiar...

# Single Line Formatting

- 
- 1
  - 2
  - 3
  - 4
  - 5

# Multi-Line Formatting



```
text-wrap: normal;
```

I'm text and I flow left to right and will wrap if I don't have enough space, but I won't if I'm set to nowrap

```
text-wrap: nowrap;
```

I'm text and I flow left to right and will wrap



Shorthand



# Display Order

---

A photograph of a person from behind, wearing a brown horse mask with a mane. They are leaning against a chain-link fence with their hands. In the background, a horse is grazing in a field. A purple rectangular overlay in the upper right corner contains the text "Say What?!"

Say What?!

# Source Order

```
<div class="flex-container">
  <div class="flex-item">
    1
  </div>
  <div class="flex-item">
    2
  </div>
  <div class="flex-item">
    3
  </div>
</div>
```

# Source Order

```
<div class="flex-container">  
  <div class="flex-item">  
    1  
  </div>  
  <div class="flex-item">  
    2  
  </div>  
  <div class="flex-item">  
    3  
  </div>  
</div>
```

# Display Order



# Default Ordinal Group

O

ZERO



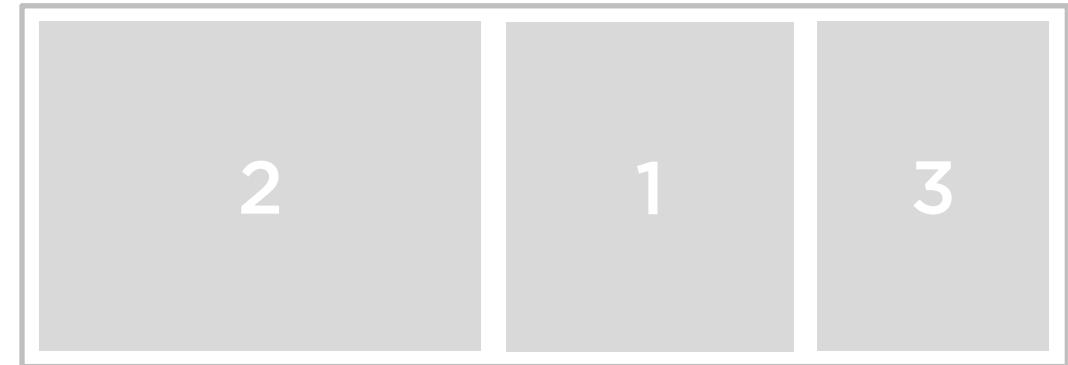
# Accessibility



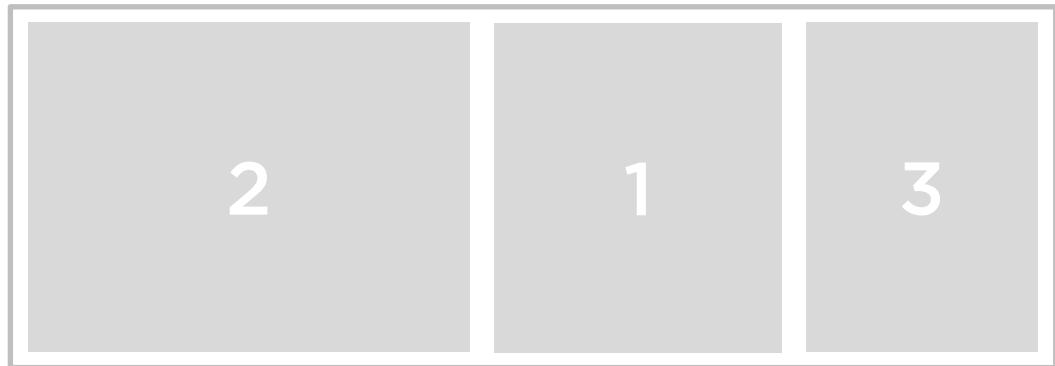
# Source Order

```
<div class="flex-container">  
  <div class="flex-item">  
    1  
  </div>  
  <div class="flex-item">  
    2  
  </div>  
  <div class="flex-item">  
    3  
  </div>  
</div>
```

# Display Order



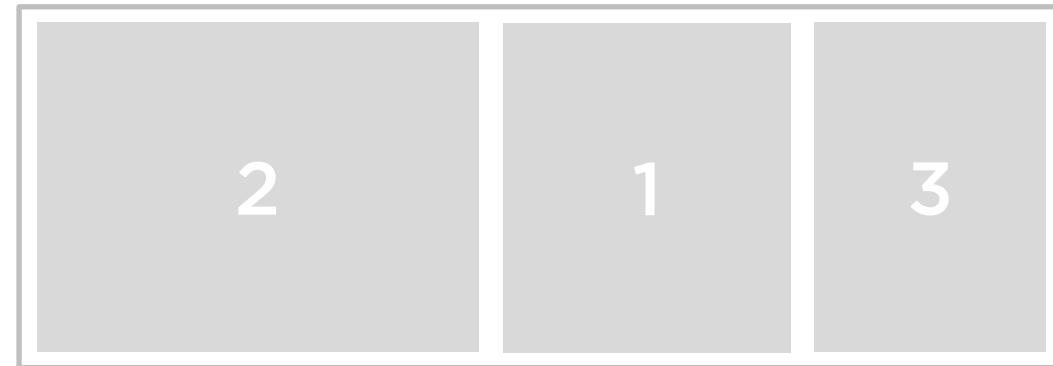
# Display Order Changes



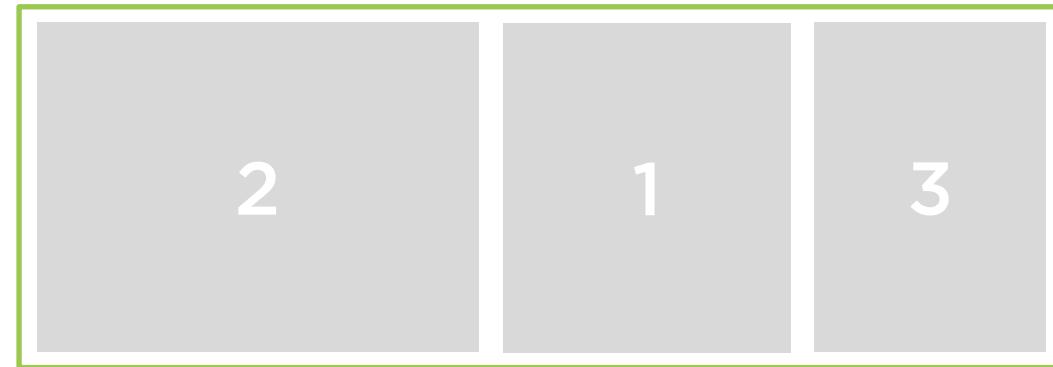
# Source Order Does Not

```
<div class="flex-container">  
  <div class="flex-item">  
    1  
  </div>  
  <div class="flex-item">  
    2  
  </div>  
  <div class="flex-item">  
    3  
  </div>  
</div>
```

# Tab Order

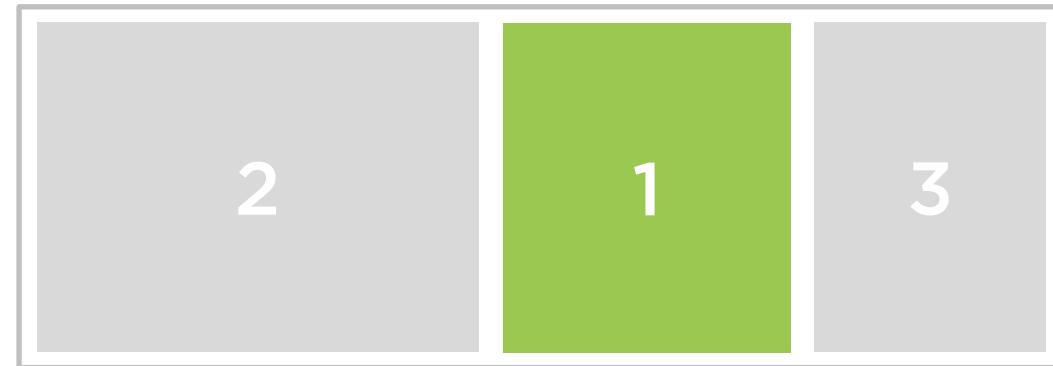


# Tab Order



Tab 01

# Tab Order



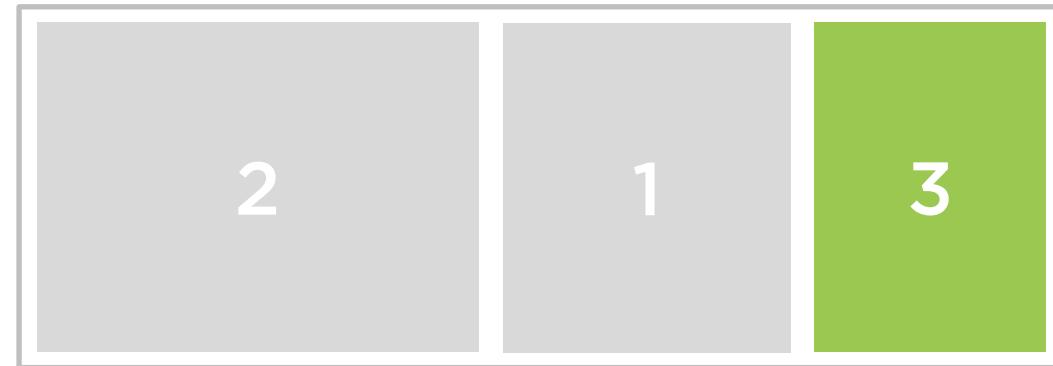
Tab 02

# Tab Order



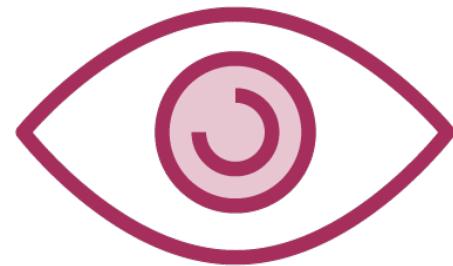
Tab 03

# Tab Order

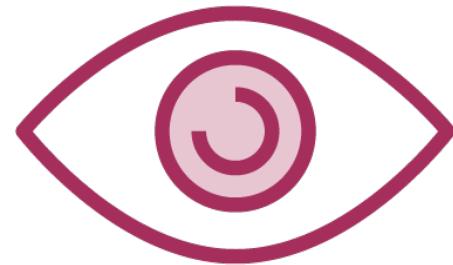


Tab 04

For Visual Purposes Only



For Visual Purposes Only

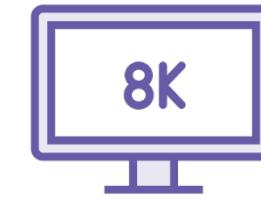
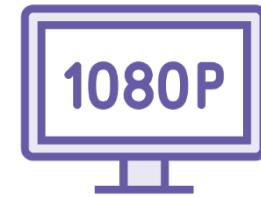
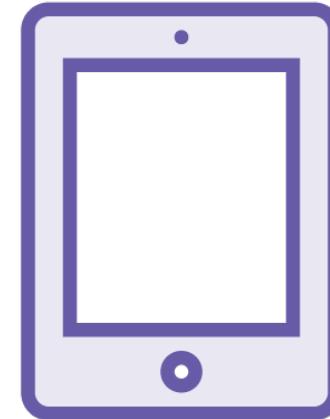
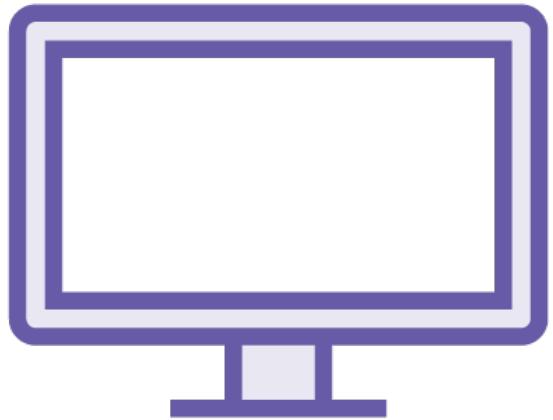


Not For Logical Reordering

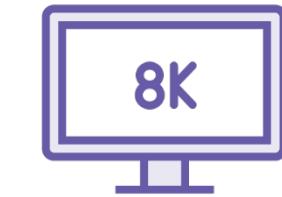
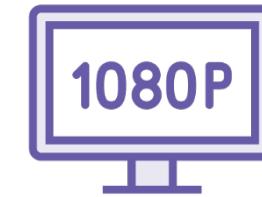
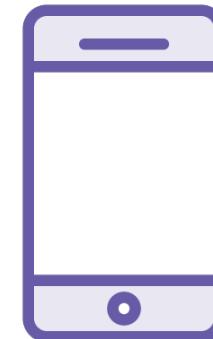
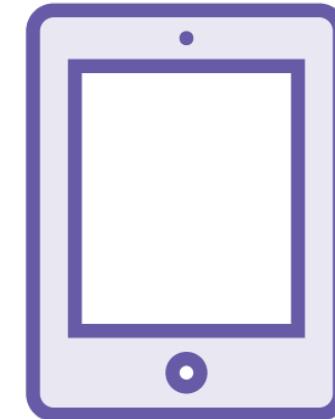
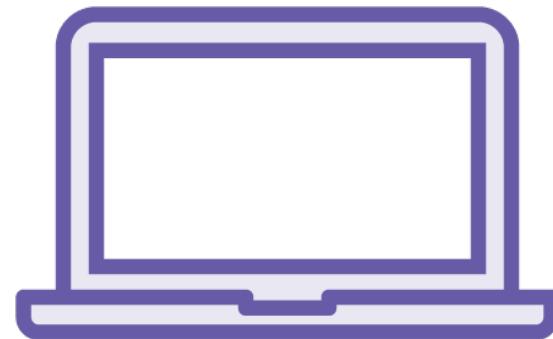
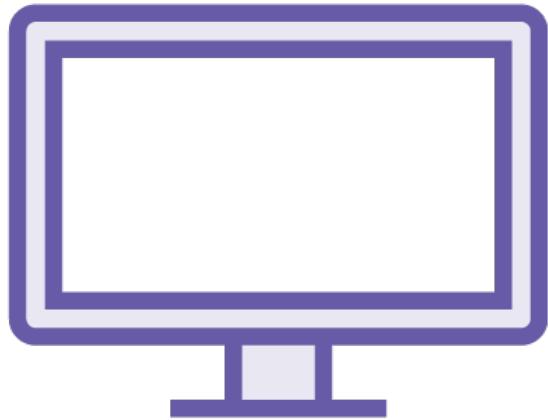
# Flexibility

---

# So Many Devices, Dimensions, & Densities



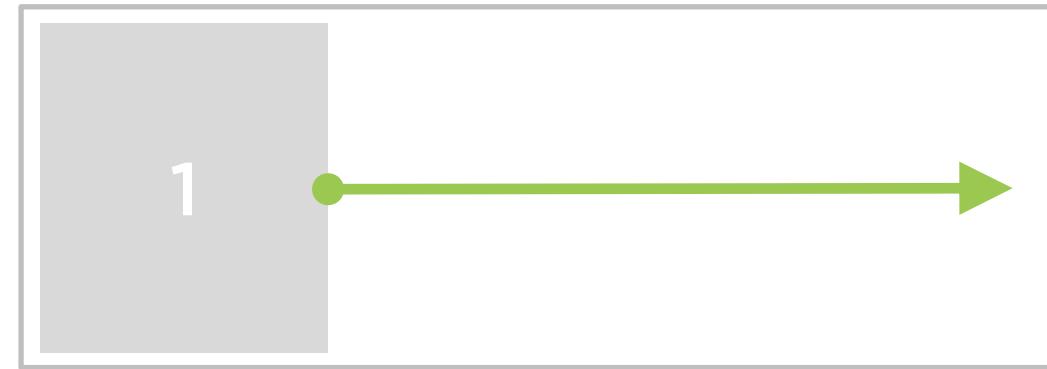
# We Need Help!



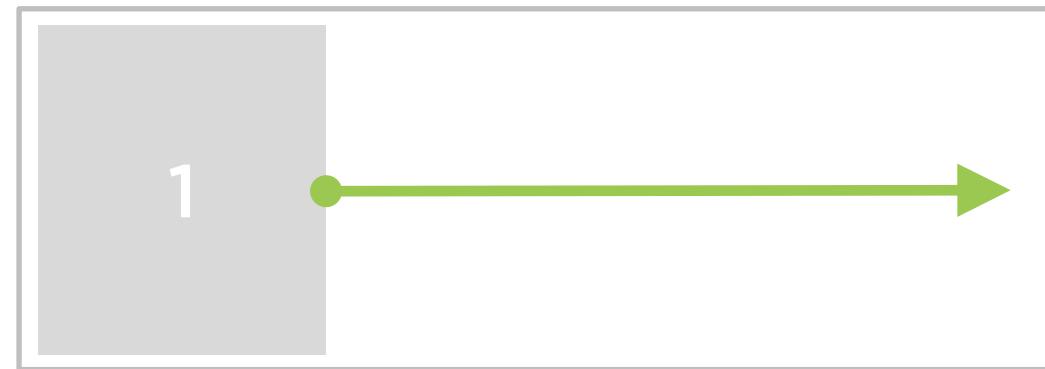
# Yay Flexbox!



# Flex Grow Factor

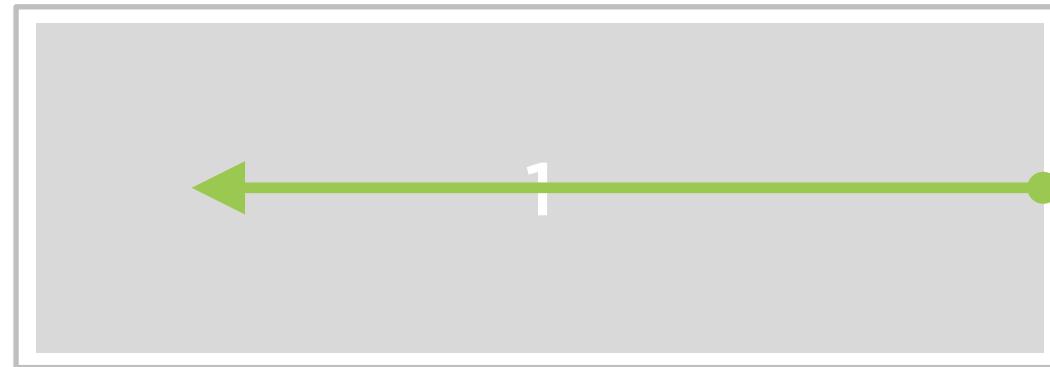


# Flex Grow Factor

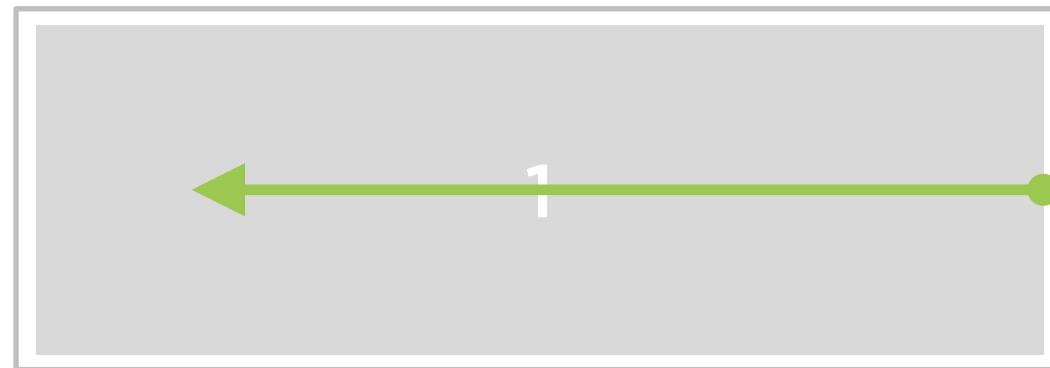


flex-grow

# Flex Shrink Factor

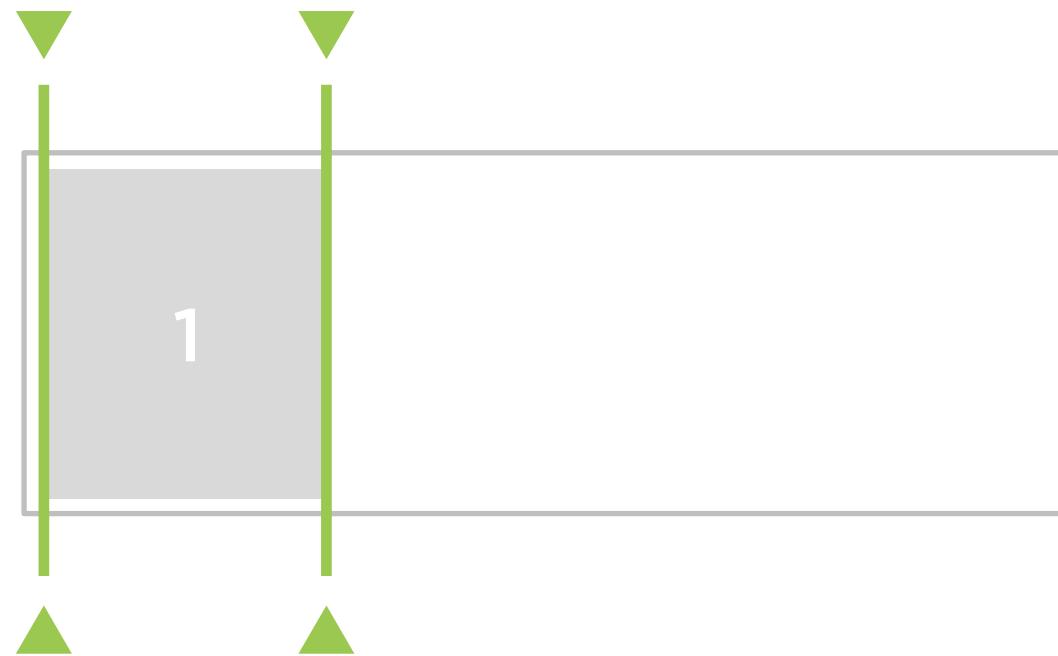


# Flex Shrink Factor

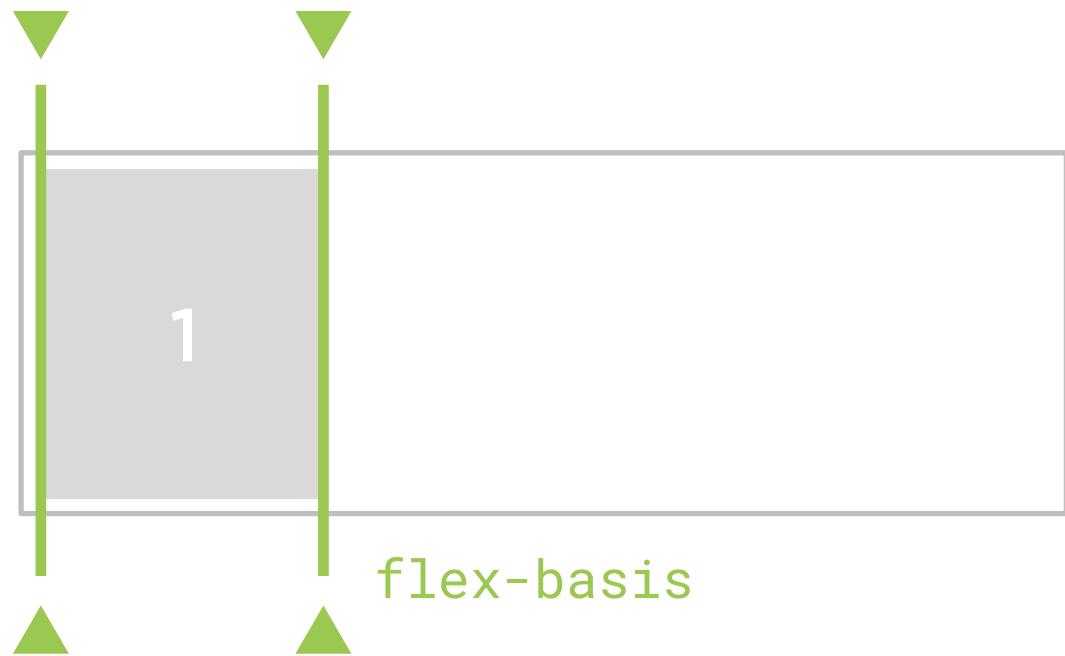


flex-shrink

# Initial Size



# Initial Size



# Shorthand

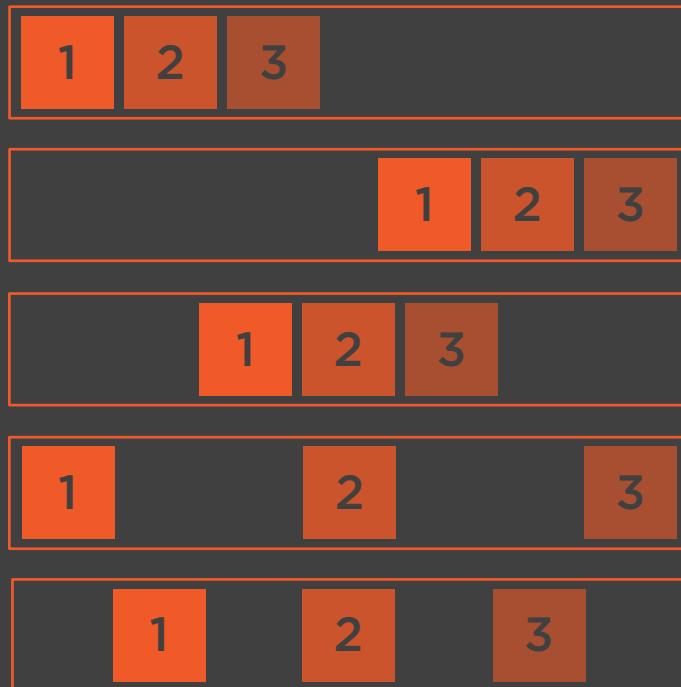
# Shorthand

`flex: flex-grow flex-shrink flex-basis;`

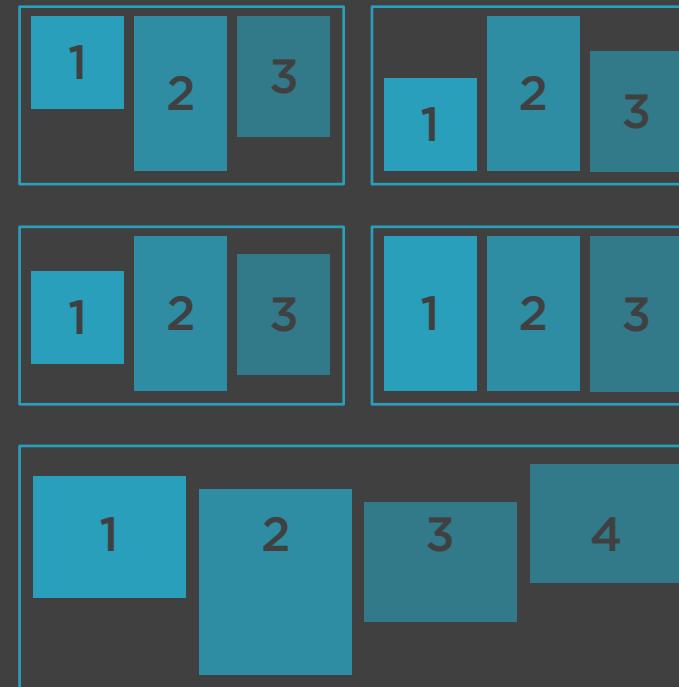
# Alignment

---

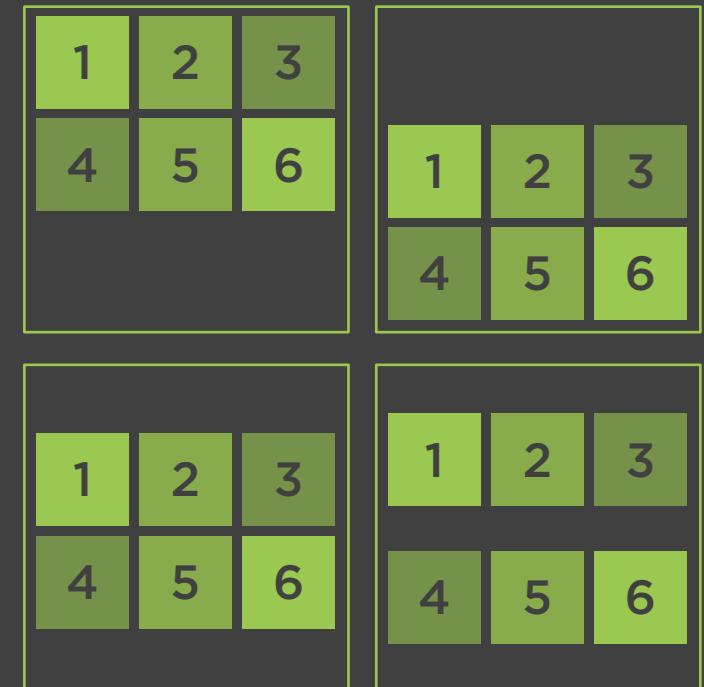
# On the Container



`justify-content`

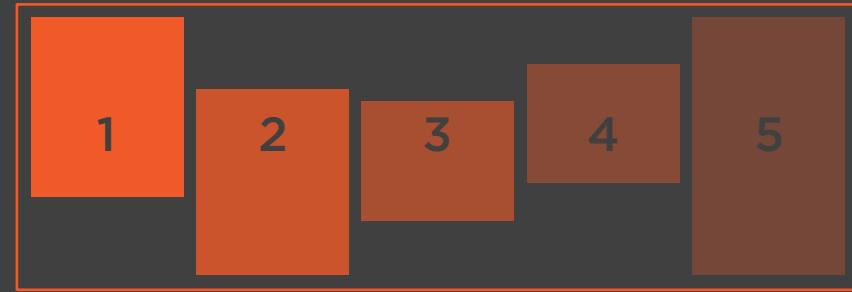


`align-items`



`align-content`

# On the Item



align-self







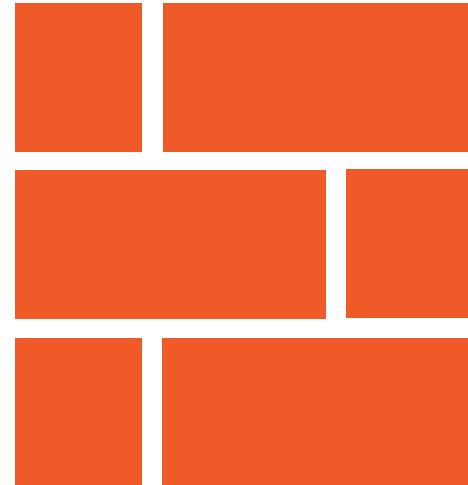




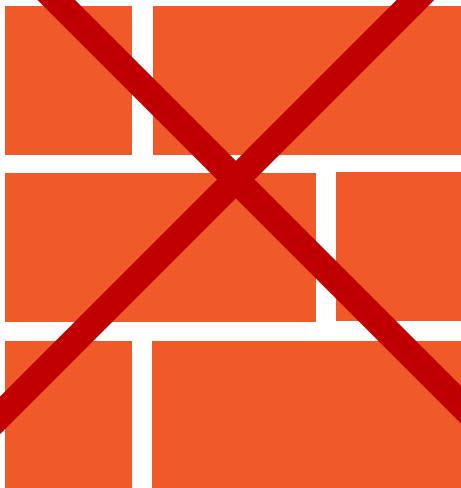
# Flexbox in the Real World

---

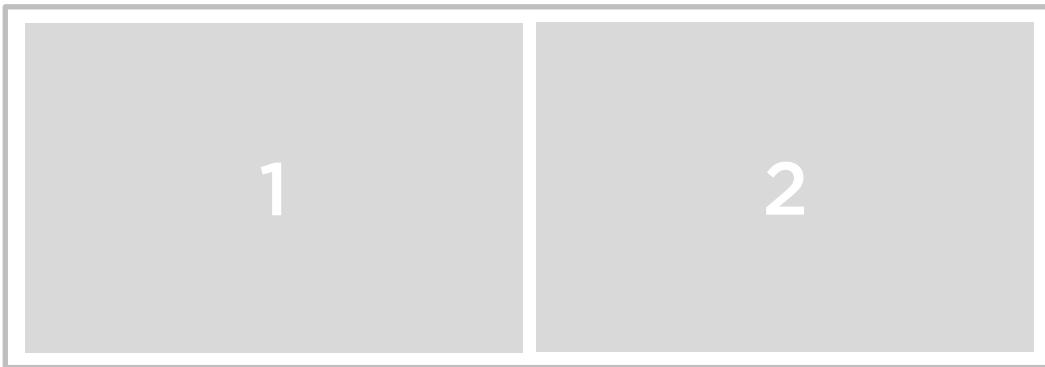
# Complex Grid Systems



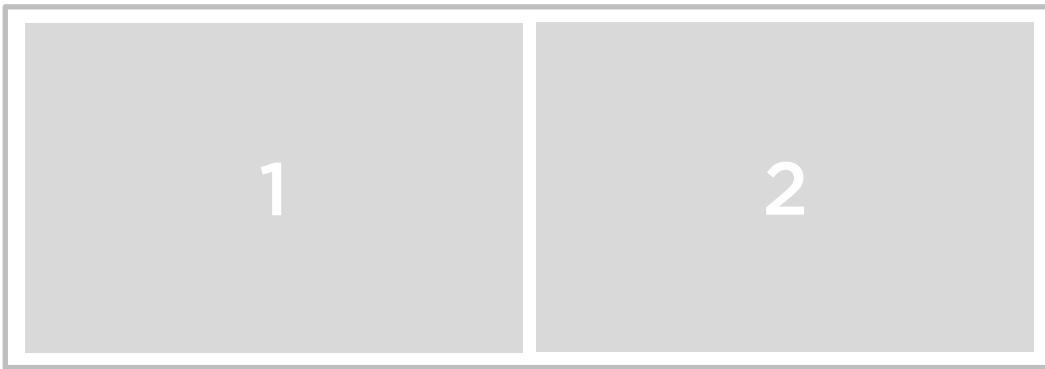
# Complex Grid Systems



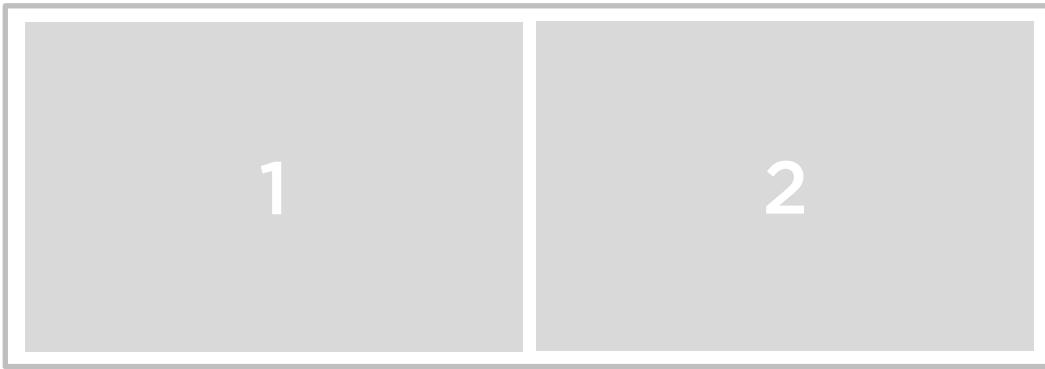




Float



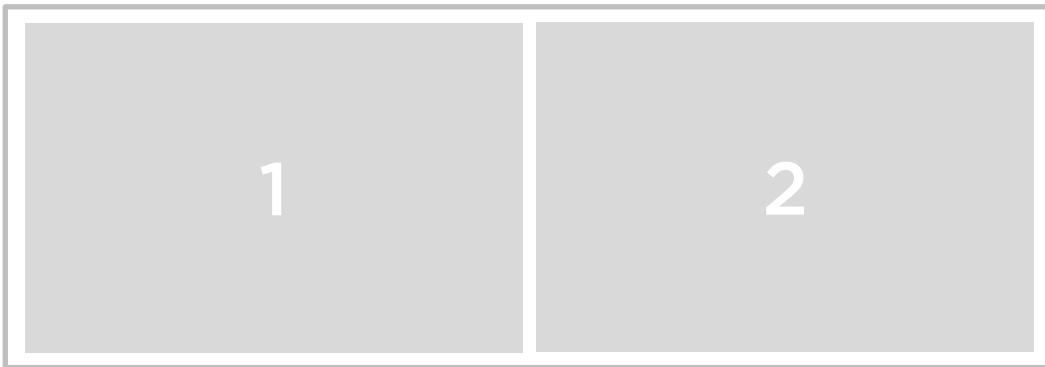
Float  
Width



Float

Width

Clear



Float  
Width  
Clear



Super Easy!



I want to  
be aligned  
to the  
bottom

Me too!

It seems  
impossible

Flexbox Was  
Made For Layout!

Float Was Not!

**Header**

**Navigation**

**Main Content**

**Sidebar**

**Footer**

Got an Unknown  
Number of Items?

# Summary

---

# Flexbox



Flexible layouts

Distribute extra space

Align content

Parent > child relationship

Main Axis vs. Cross Axis

Flow content horizontally or vertically

Align items on the container or individually

Stretch, shrink, & grow

Change display order

Real world examples