Bootstrap

UTILITIES

IN A ROCKET

Learn front-end development at rocket speed

CONTENT

UTILITIES

LAYOUT

COMPONENTS

TEXT

COLORS

BORDERS

SIZING

SPACING

DISPLAY/
POSITION

IMAGES/
EMBEDS

MISC

TEXT: BASIC FORMATTING

WEIGHT & ITALICS

Lorem ipsum `.font-weight-bold`

Lorem ipsum `.font-weight-normal`

Lorem ipsum `.font-weight-light`

Lorem ipsum `.font-italic`

TRANSFORM

lorem ipsum `.text-lowercase`

LOREM IPSUM `.text-uppercase`

Lorem Ipsum `.text-capitalize`

TEXT: HORIZONTAL ALIGNMENT

`.text-left`

`.text-right`

`.text-center`

`.text-justify`

RESPONSIVE

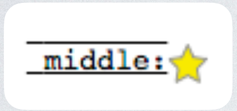
`.text-sm-left`	Left aligned text on viewports sized SM or wider.
`.text-md-right`	Right aligned text on viewports sized MD or wider.
`.text-lg-center`	Center aligned text on viewports sized LG or wider.
`.text-xl-justify`	Justified text on viewports sized XL or wider.

TEXT: VERTICAL ALIGNMENT

Vertical-align only affects inline, inline-block, inline-table, and table cell elements.

`.align-baseline`

Aligns the baseline of the element with the baseline of its parent.



`.align-middle`

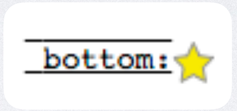
Aligns the middle of the element with the baseline plus half the x-height of the parent.

`.align-text-top`

Aligns the top of the element with the top of the parent element's font.

`.align-top`

Aligns the top of the element and its descendants with the top of the entire line.



`.align-bottom`

Aligns the bottom of the element and its descendants with the bottom of the entire line.

`.align-text-bottom`

Aligns the bottom of the element with the bottom of the parent element's font.

TEXT: WRAPPING & OVERFLOW

NO WRAP

```
<div class="text-nowrap">
  This text will overflow the parent.
</div>
```

This text will overflow the parent.

CSS
div { width: 8em; }

TRUNCATE

Requires `display: inline-block` or `display: block`.

```
<span class="d-inline-block text-truncate">
  This text will be truncated.
</span>
```

This text will be t...

CSS
span { width: 8em; }

TEXT

COLORS

BORDERS

SIZING

SPACING

DISPLAY/
POSITION

IMAGES/
EMBEDS

MISC

COLORS

text | bg -

- primary
- secondary
- success
- danger
- warning
- info
- light
- dark
- muted (Text only)
- white

Example:

```
<p class="bg-info text-white">
Hello World!</p>
```

Hello World!

TEXT

COLORS

BORDERS

SIZING

SPACING

DISPLAY/
POSITION

IMAGES/
EMBEDS

MISC

BORDERS

ADDITIVE-SUBTRACTIVE

Additive

```
.border
.border-top
.border-right
.border-bottom
.border-left
```

Subtractive

```
.border-0
.border-top-0
.border-right-0
.border-bottom-0
.border-left-0
```

COLOR

.border
.border-

- primary
- secondary
- success
- danger
- warning
- info
- light
- dark
- white

RADIUS

```
.rounded
.rounded-top
.rounded-right
.rounded-bottom
.rounded-left
.rounded-circle
.rounded-0
```

TEXT

COLORS

BORDERS

SIZING

SPACING

DISPLAY/
POSITION

IMAGES/
EMBEDS

MISC

SIZING

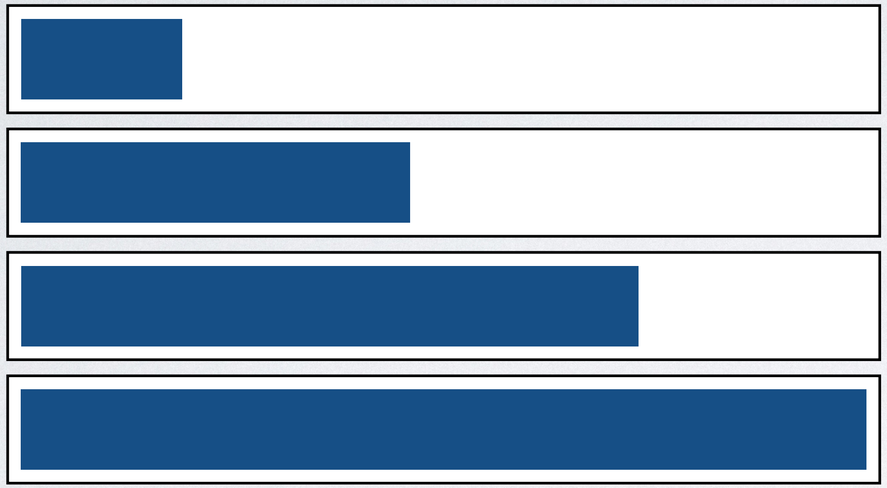
WIDTH

.w-25

.w-50

.w-75

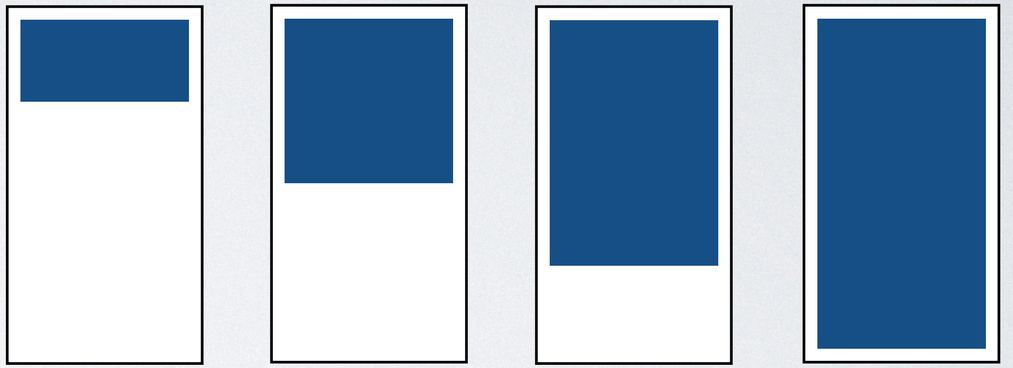
.w-100



.mw-100 max-width: 100%

HEIGHT

.h-25 .h-50 .h-75 .h-100



.mh-100 max-height: 100%

TEXT

COLORS

BORDERS

SIZING

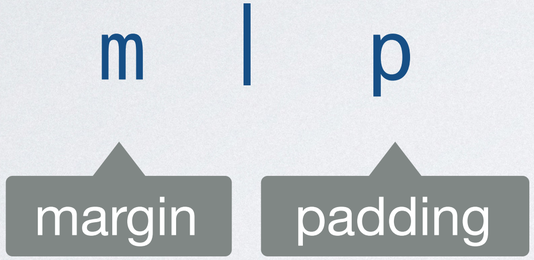
SPACING

DISPLAY/
POSITION

IMAGES/
EMBEDS

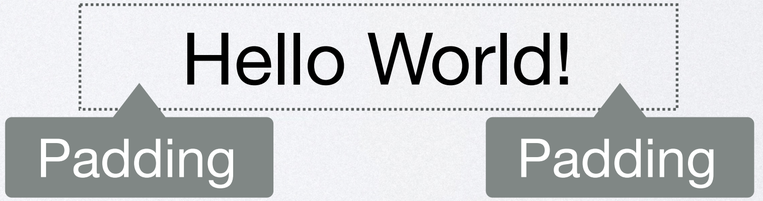
MISC

SPACING



Example:

```
<p class="px-2">
Hello World!</p>
```



.mx-auto class for horizontally centering block level fixed-width content.

```
<div class="mx-auto" style="width: 200px;">
  Hello World!
</div>
```

TEXT

COLORS

BORDERS

SIZING

SPACING

DISPLAY/ POSITION

IMAGES/ EMBEDS

MISC

DISPLAY

`.d-{breakpoint}-{value}`

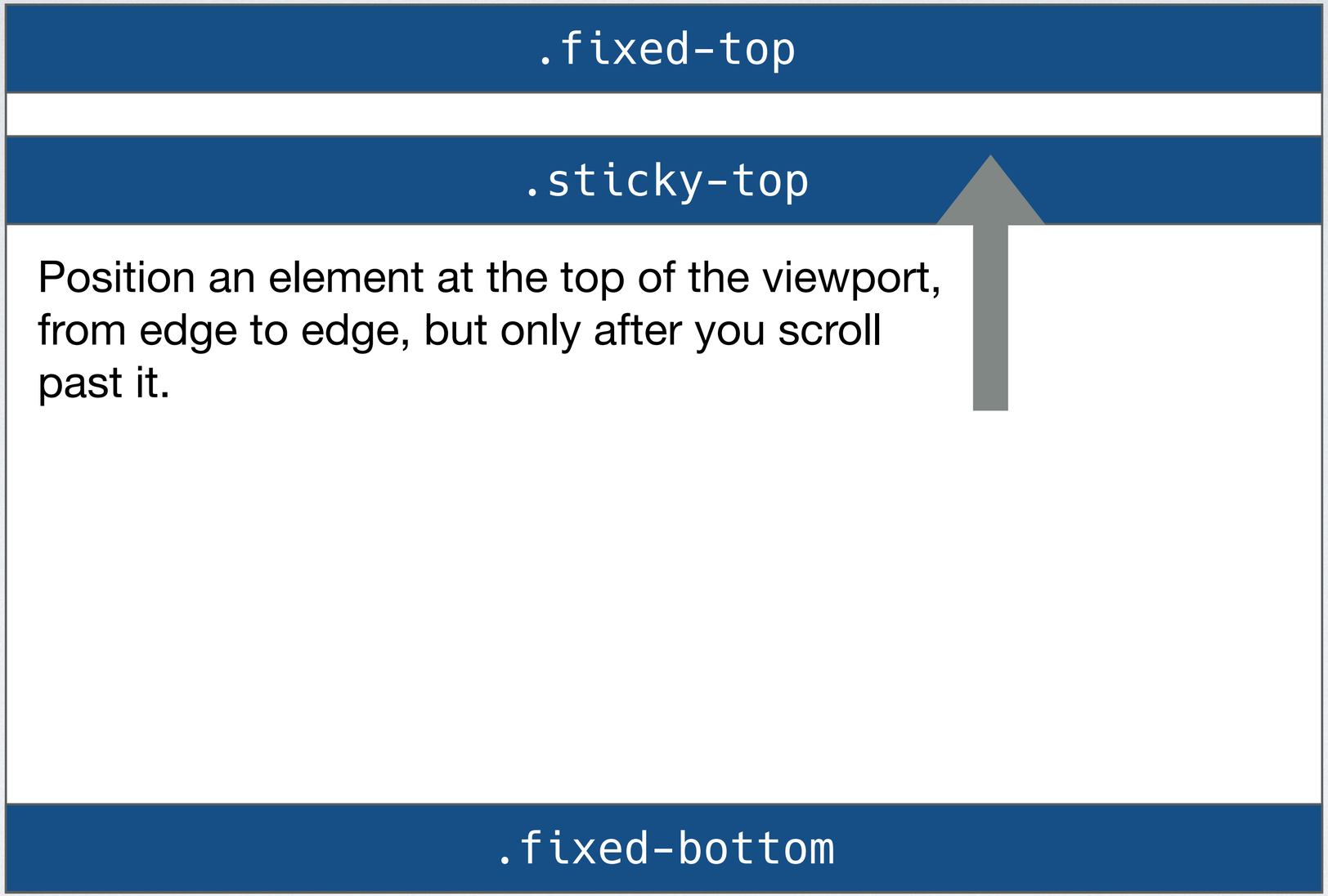
breakpoint	value
xs (default)	none
sm	inline
md	inline-block
lg	block
xl	table
	table-cell
	table-row
	flex
	inline-flex

Example:

```
.d-none .d-sm-block
```

Hidden only on xs.

POSITION: FIXED & STICKY



POSITION: FIXED & STICKY

BOOTSTRAP 3

DROPPED

Affix Plug-in

The affix plug-in toggles *position: fixed;* on and off.

BOOTSTRAP 4

OPTIONS

position: sticky

Bootstrap 4 recommends using this polyfill instead.

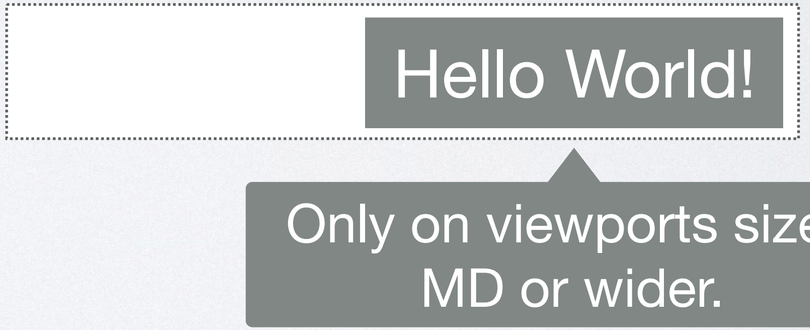
FLOATS

`.float-{breakpoint}-{value}`

breakpoint	value
sm	left
md	right
lg	none
xl	

Example:

```
<div class="float-md-right">
Hello World!</div>
```



CLEARFIX

```
<p>Ei posit malorum dolour vel, pro
ridens debutis ut.</p>

<div>
 <button class="float-left">
 Previous</button>
 <button class="float-right">
 Next</button>
</div>

<h2>Contact</h2>
```

Ei posit malorum dolour vel, pro ridens debutis ut.

Previous **Contact** Next

CLEARFIX

```
<p>Ei posit malorum dolour vel, pro
ridens debutis ut.</p>

<div class="clearfix">
 <button class="float-left">
 Previous</button>
 <button class="float-right">
 Next</button>
</div>

<h2>Contact</h2>
```

Ei posit malorum dolour vel, pro ridens debutis ut.

Previous Next

Contact

FLEX

`.d-{breakpoint}-{value}`

- sm
- md
- lg
- xl

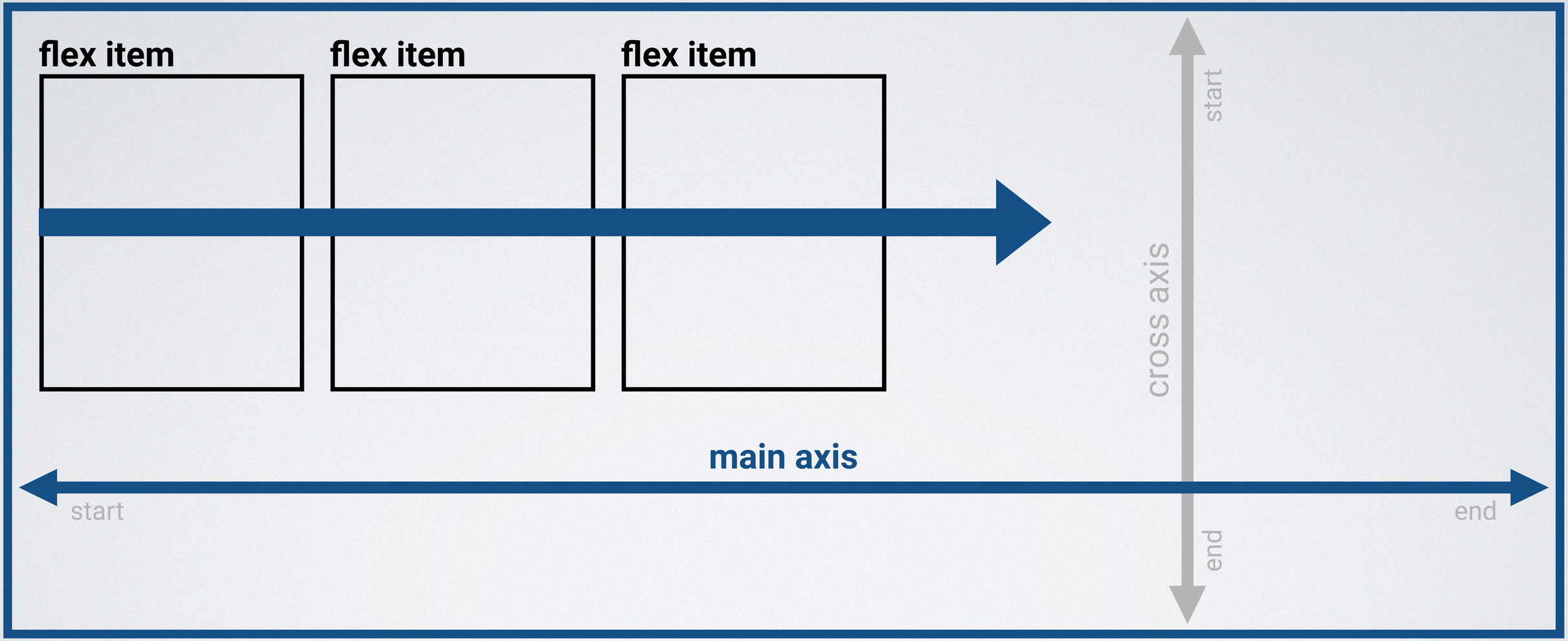
- flex
- inline-flex

Example:

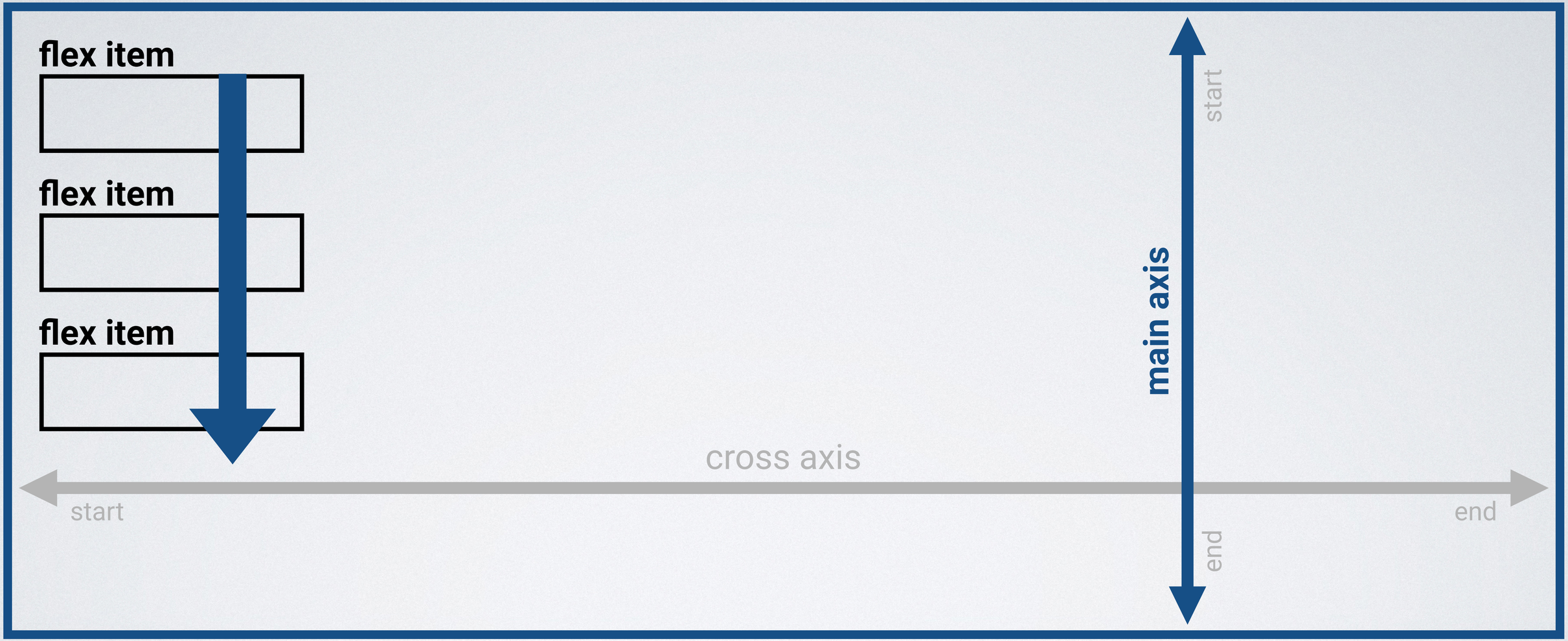
```
<div class="d-md-flex">
  I'm a flexbox container.
</div>
```

FLEX: DIRECTION

flex-direction: row



flex-direction: column



FLEX: DIRECTION

ROW

Normal

```
<div class="d-flex flex-row">
  <div>#1</div>
  <div>#2</div>
  <div>#3</div>
</div>
```

Reverse

```
<div class="d-flex flex-row-reverse">
  <div>#1</div>
  <div>#2</div>
  <div>#3</div>
</div>
```

COLUMN

Normal

```
<div class="d-flex flex-column">
  <div>#1</div>
  <div>#2</div>
  <div>#3</div>
</div>
```

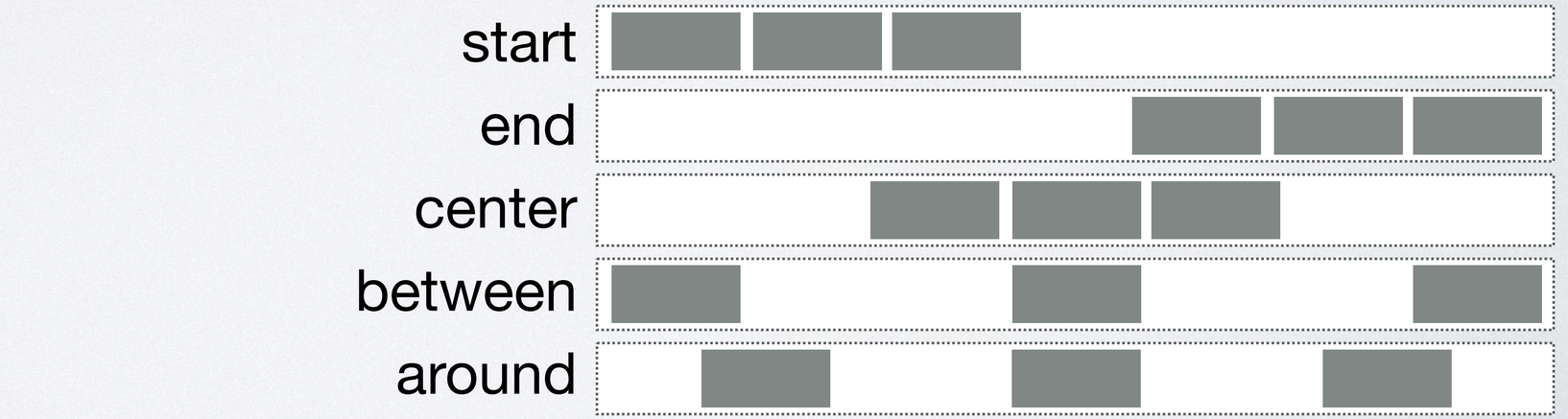
Reverse

```
<div class="d-flex flex-column-reverse">
  <div>#1</div>
  <div>#2</div>
</div>
```

FLEX: JUSTIFY CONTENT

```
.justify-content-{breakpoint}-{value}
```

breakpoint	value
sm	start
md	end
lg	center
xl	between
	around



FLEX: ALIGN ITEMS

`.align-items-{breakpoint}-{value}`

breakpoint	value
sm	start
md	end
lg	center
xl	baseline
	stretch

FLEX: ALIGN ITEMS

```
align-items:
flex-start;
```

FLEX: ALIGN ITEMS

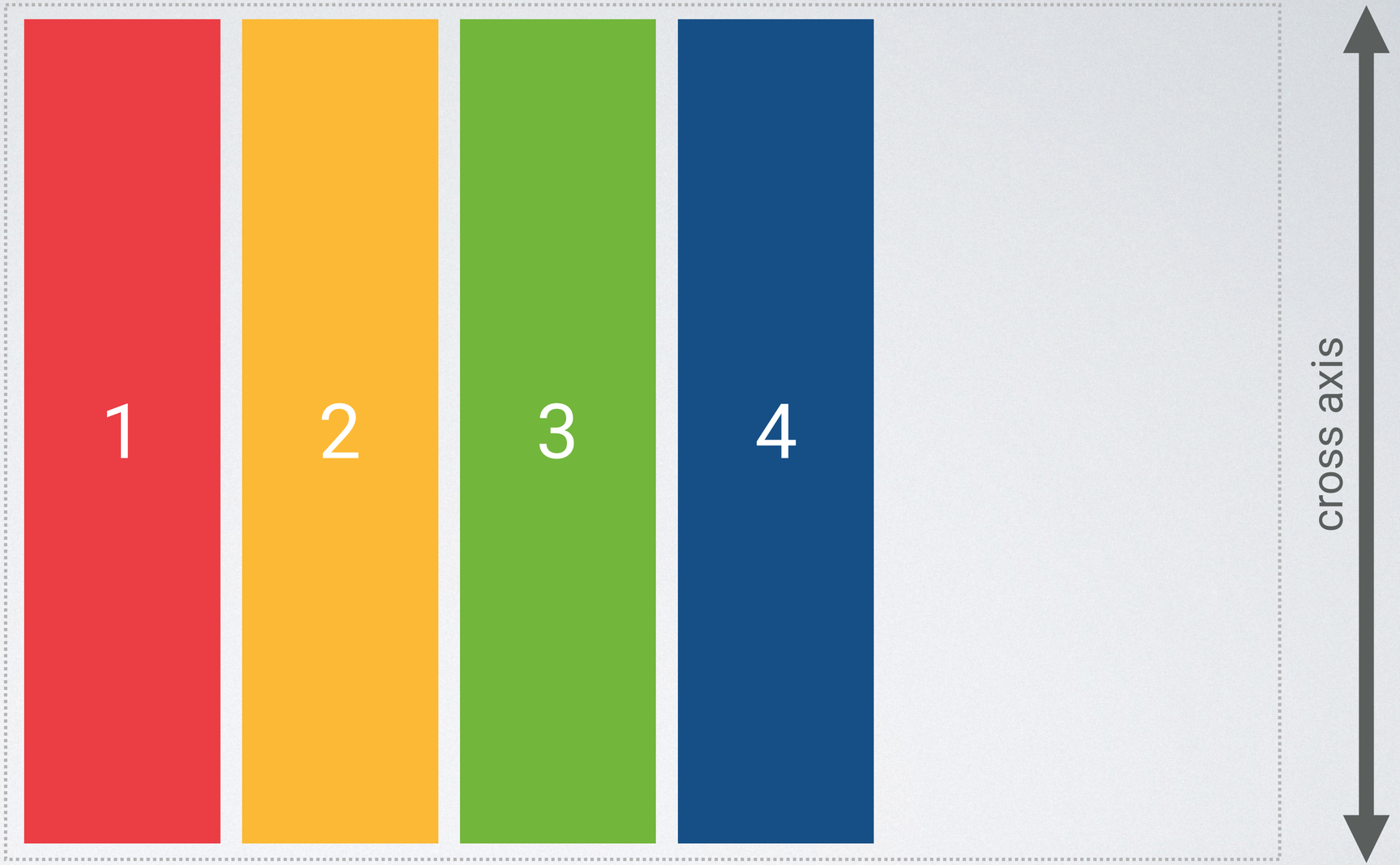
```
align-items:
flex-end;
```

FLEX: ALIGN ITEMS

```
align-items:
center;
```

FLEX: ALIGN ITEMS

```
align-items:
stretch;
```



FLEX: ALIGN ITEMS

```
align-items:
baseline;
```

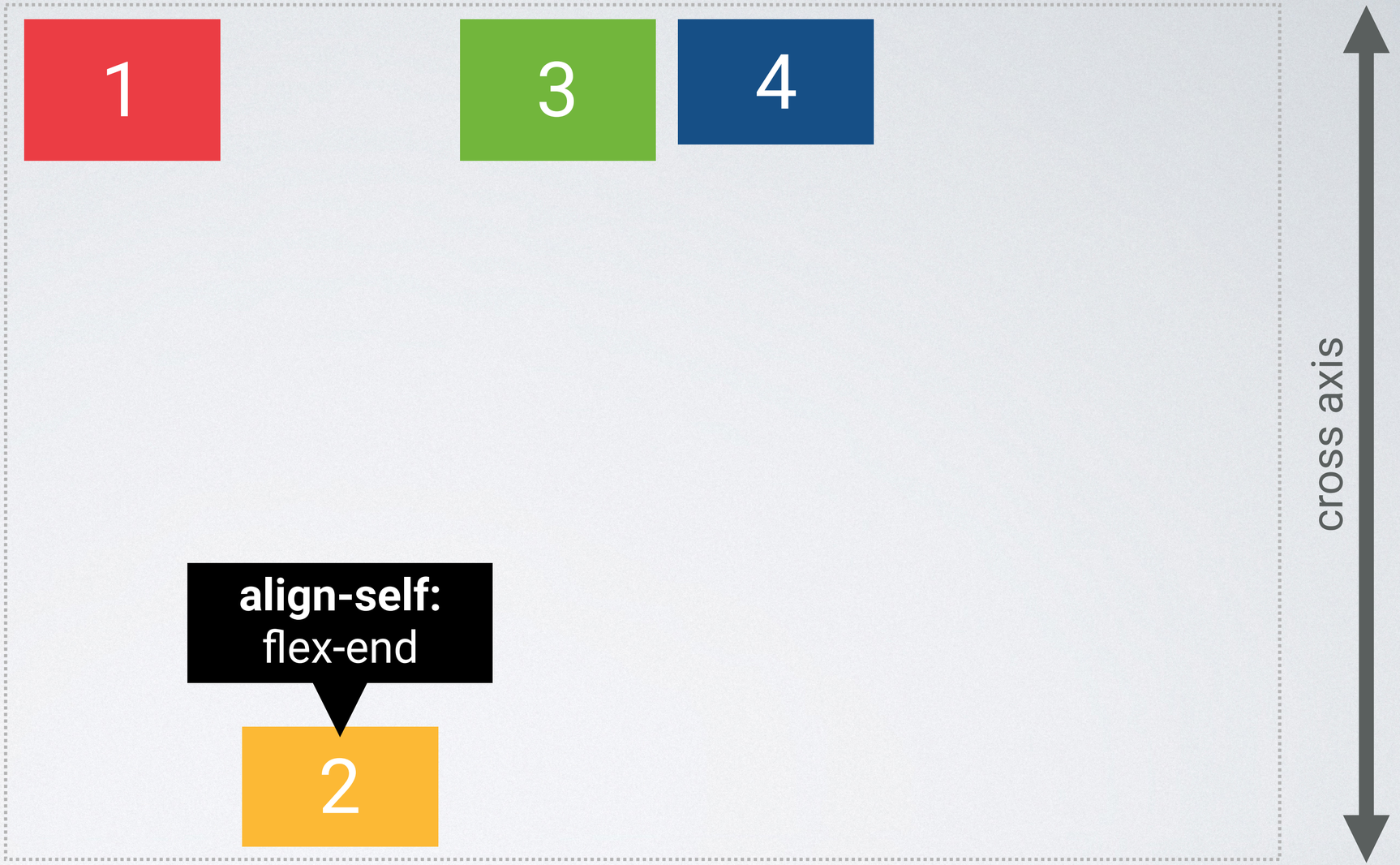
All participating flex items on the line are aligned such that their baselines align, and the item with the largest distance between its baseline and its cross-start margin edge is placed flush against the cross-start edge of the line.

FLEX: ALIGN SELF

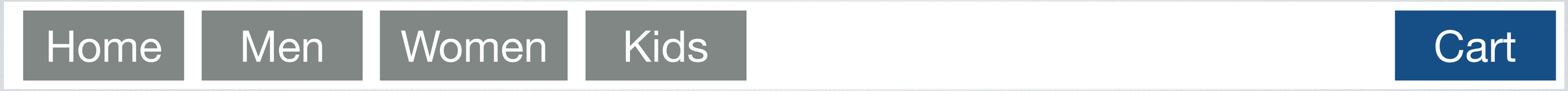
```
.align-self-{breakpoint}-{value}
```

breakpoint	value
sm	start
md	end
lg	center
xl	baseline
	stretch

FLEX: ALIGN SELF

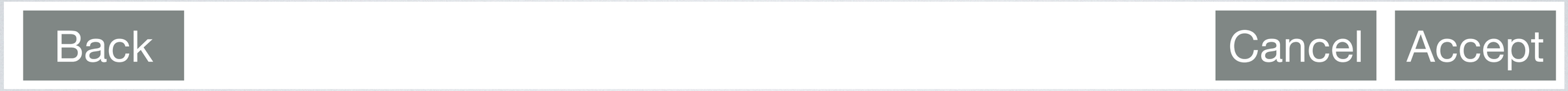


FLEX: AUTO-MARGINS



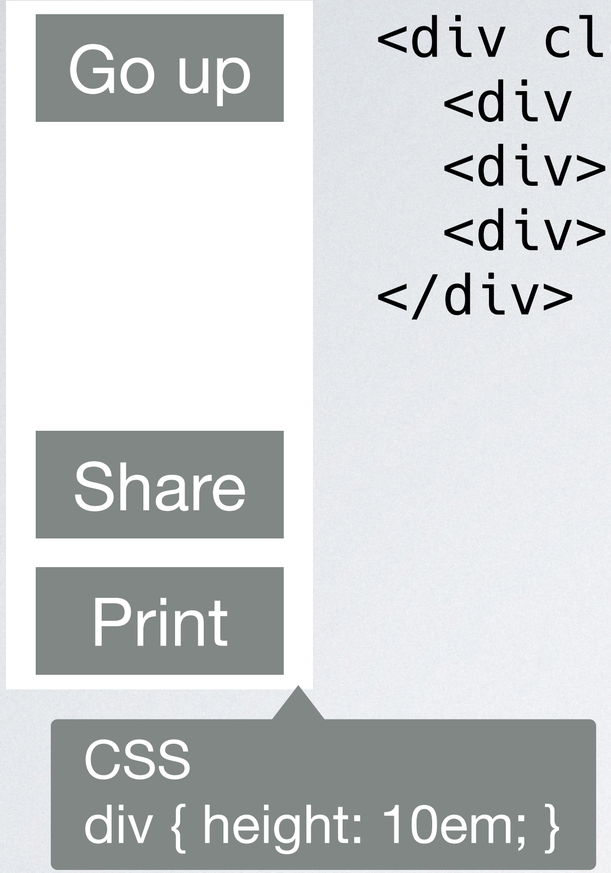

```
<div class="d-flex">
  <div>Home</div>
  <div>Men</div>
  <div>Women</div>
  <div>Kids</div>
  <div class="ml-auto">Cart</div>
</div>
```

FLEX: AUTO-MARGINS



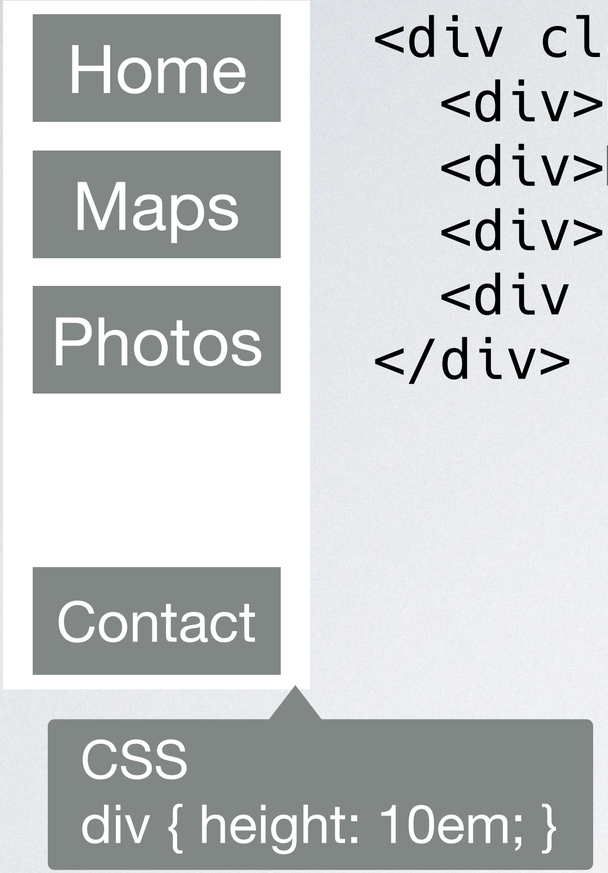

```
<div class="d-flex">
  <div class="mr-auto">Back</div>
  <div>Cancel</div>
  <div>Accept</div>
</div>
```

FLEX: AUTO-MARGINS




```
<div class="d-flex flex-column align-items-start">
  <div class="mb-auto">Go up</div>
  <div>Share</div>
  <div>Print</div>
</div>
```

FLEX: AUTO-MARGINS




```
<div class="d-flex flex-column align-items-end">
  <div>Home</div>
  <div>Maps</div>
  <div>Photos</div>
  <div class="mt-auto">Contact</div>
</div>
```

TEXT

COLORS

BORDERS

SIZING

SPACING

DISPLAY/
POSITION

**IMAGES/
EMBEDS**

MISC

IMAGE REPLACEMENT

Your company logo in the page header

HTML

```
<h1 class="text-hide my-logo">
  SurfAcademy
</h1>
```

CSS

```
.my-logo {
  background-image: url('logo.svg');
  width:  100px;
  height: 100px;
}
```

EMBED

Aspect ratios
embed-responsive-21by9
embed-responsive-16by9
embed-responsive-4by3
embed-responsive-1by1

```
<div class="embed-responsive embed-responsive-16by9">
  <iframe
    class="embed-responsive-item"
    src="https://www.youtube.com/embed/zpOULjyy-n8?rel=0" allowfullscreen>
  </iframe>
</div>
```

TEXT

COLORS

BORDERS

SIZING

SPACING

DISPLAY/
POSITION

IMAGES/
EMBEDS

MISC

SCREENREADERS

`.sr-only`

Hide an element to all devices except screen readers.

`.sr-only-focusable`

Show the element again when it’s focused (e.g. by a keyboard-only user).

Example:

```
<a class="sr-only sr-only-focusable" href="#content">
Skip to main content</a>
```

CLOSE ICON

```
<button type="button" class="close" aria-label="Close">
  <span aria-hidden="true">&times;</span>
</button>
```

TEXT

COLORS

BORDERS

SIZING

SPACING

DISPLAY/
POSITION

IMAGES/
EMBEDS

MISC

Bootstrap

UTILITIES

IN A ROCKET

Learn front-end development at rocket speed