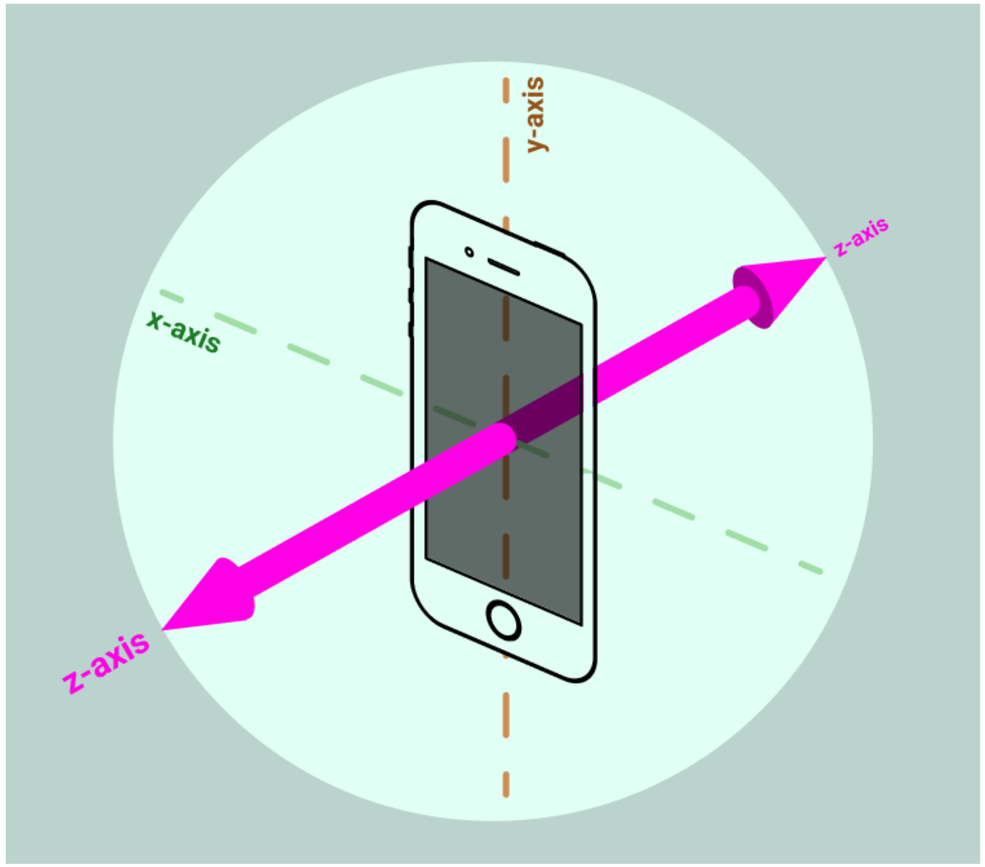
Positioning: axes



X-Axis

- **Direction:** Runs left to right
- **CSS:** margin-left, padding-right, left, right, …

Y-Axis

- **Direction:** Run top to bottom
- **CSS:** margin-top, padding-bottom, top, bottom, …

Z-Axis

- **Direction:** Runs in and out of the screen (towards or way from the viewer)
- **CSS:** z-index, translateZ(), …

Positioning: recap

Static: The element is positioned according to the normal flow of the document.

- **Use Case:** Basic document flow without any specific positioning

Relative: Relative to the elements normal position. Setting top, right, bottom, or left will move it from its normal position.

- **Z-Axis:** `z-index` can be applied, creating a stacking context.
- **Use Case:** Minor adjustments to the position without affecting the layout of other elements.

Absolute: Relative to its nearest positioned parent (non-static).

- **Z-Axis:** `z-index` can be applied, creating a stacking context.
- **Use Case:** Precise placement of elements, such as in overlays, tooltips, and modals.

Fixed: The element is positioned relative to the viewport and does not move when the page is scrolled.

- **Z-Axis:** `z-index` can be applied, creating a stacking context.
- **Use Case:** Fixed headers, footers, or any element that should stay visible during scroll.

Sticky: The element toggles between relative and fixed positioning depending on the scroll position.
It is positioned relative until it crosses a specified threshold, then it becomes fixed.

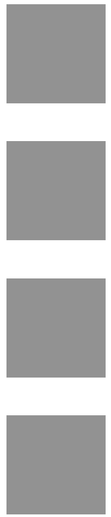
- **Z-Axis:** `z-index` can be applied, creating a stacking context.
- **Use Case:** Sticky headers that remain at the top of the viewport while scrolling past a certain point.

Flexible Box Layout (FlexBox)

Used for laying out elements in one direction

Row

Column

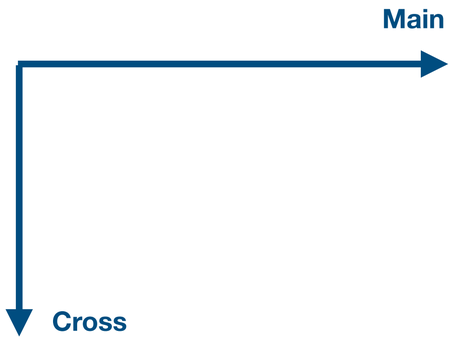


FlexBox: axes

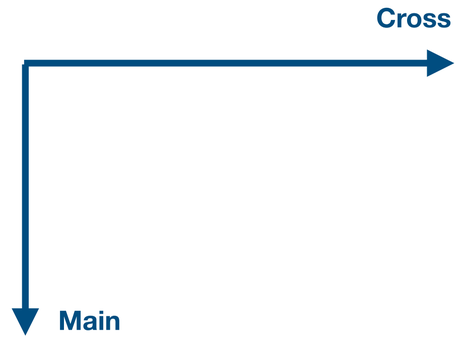
Main (primary)

Cross (secondary)

Direction: row



Direction: column



FlexBox: aligning items

- **justify-content** (along the **main** axis)
- **align-items** (along the **cross** axis)
- **align-content**

FlexBox: sizing items

- **flex-basis** (the initial size of a flex item)
- **flex-grow** (the growth factor)
- **flex-shrink** (the shrink factor)
- **flex** (shorthand which combines the 3 above)

Item properties

Remark: should be applied on flex items, not containers

Media Queries

To provide different styles for different devices depending on their features, such as:

- Screensize
- Orientation

Responsive design

We can use it to make webpages that look great on mobile phones, tablets, laptops and desktop computers

- Desktop first
- Mobile first

Responsive design

- Desktop first
- Mobile first

Transformations

- **rotate()** - rotating elements
- **scale()** - making elements larger or smaller
- **skew()** - tilting element to the left or the right
- **translate()** - positioning on the screen