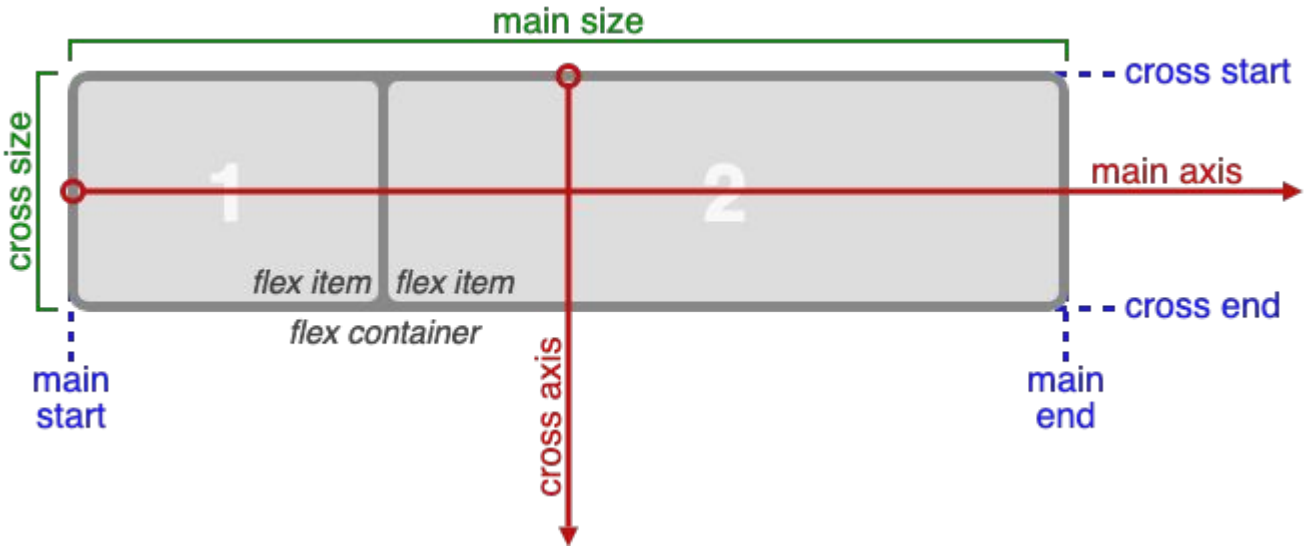
Module 3 - Lecture 4

CSS Flexbox

CSS Flexbox

https://www.w3.org/TR/css-flexbox-1/

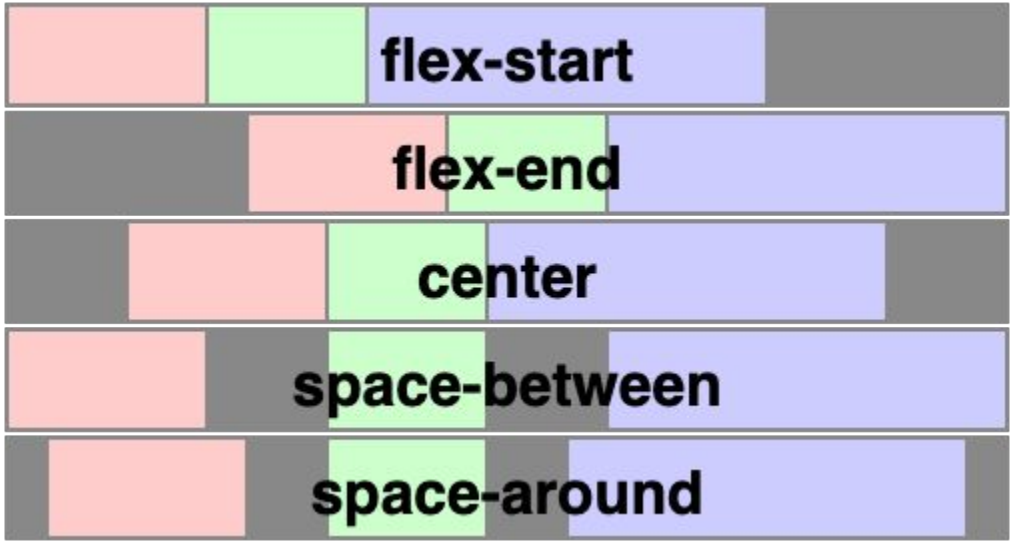


A flex container takes away some of the functionality of the block container

- For example: float and vertical alignment do not apply.

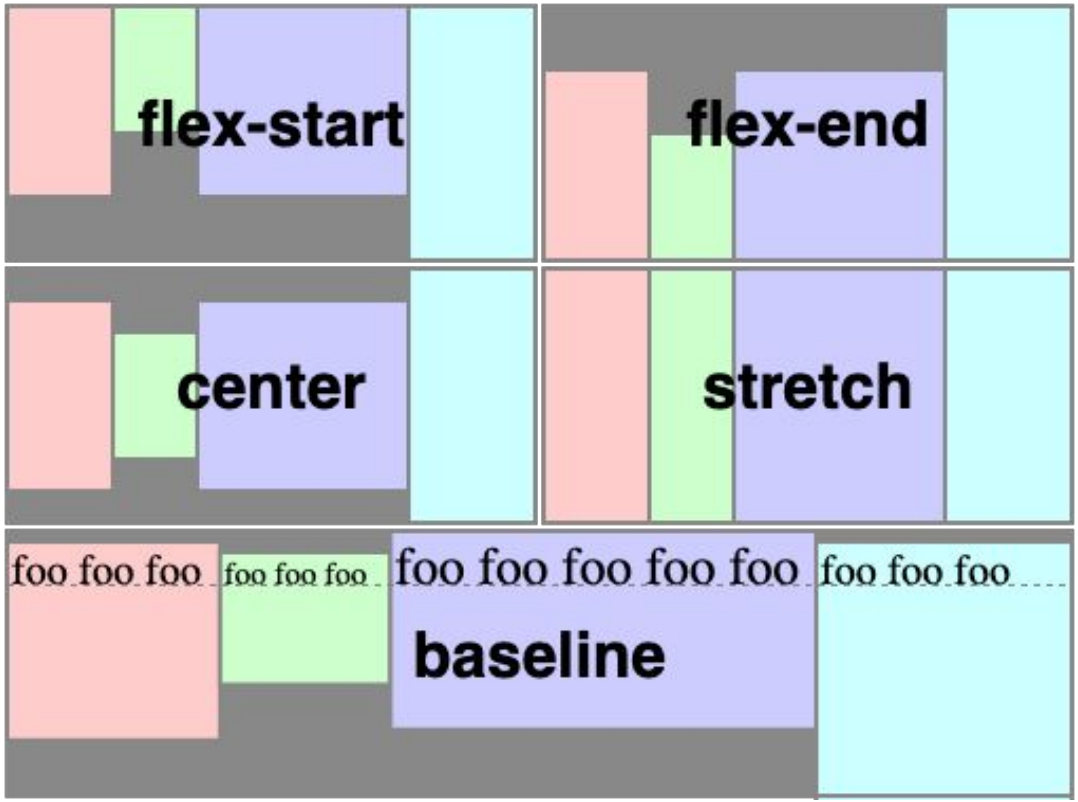
Main Axis Alignment

justify-content, justify-self
(flex-start is default)



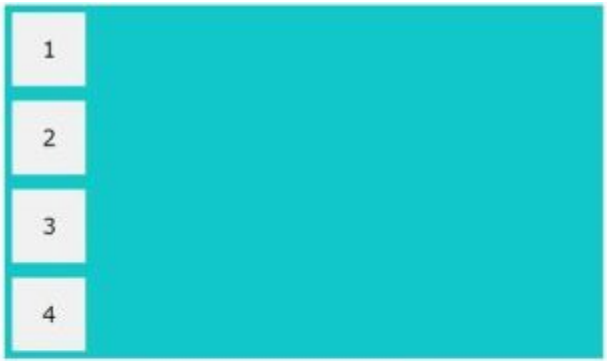
Cross Axis Alignment

align-items, align-self - (stretch is default)

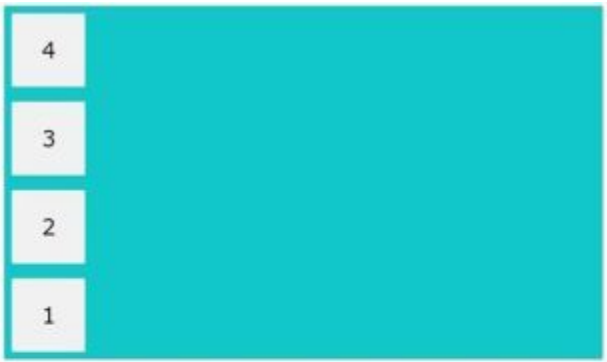


Direction of Flex Items

** Changing the direction from row to column will also change the main and cross axes



flex-direction: column



flex-direction: column-reverse

flex-direction: row

flex-direction: row-reverse

```
flex-container{
  display: flex;
  flex-direction: ____;
}
```

Customized Order of Flex Items

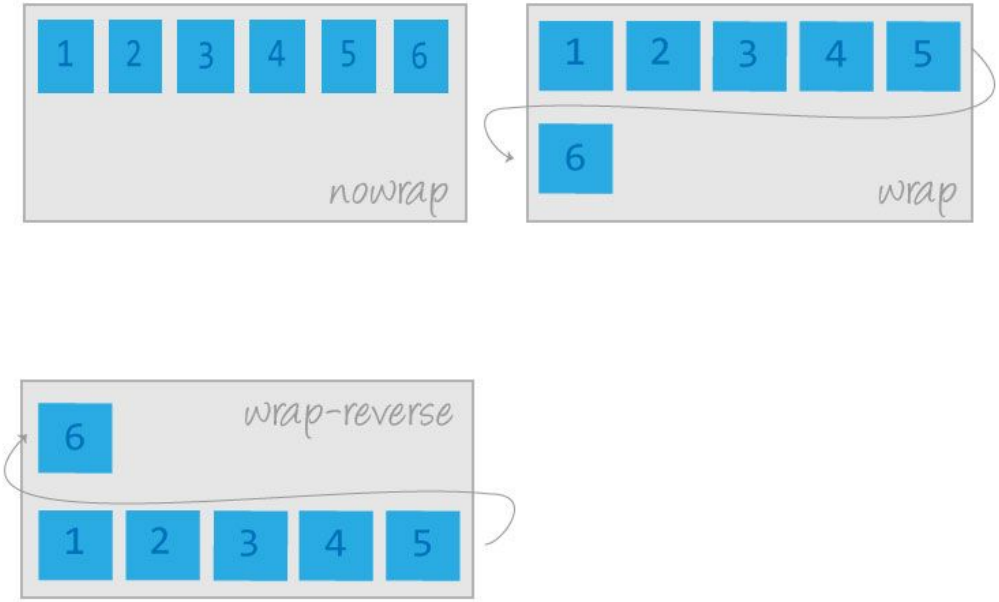
- All items have a default order of 0.
- The order property can be altered to achieve a different arrangement on the screen. It allows for positive or negative integers.

```
.items {
  display: flex;
}

.item-three {
  order: -1;
}

.item-five: {
  order: -1;
}
```

Wrapping



Flex-Wrap

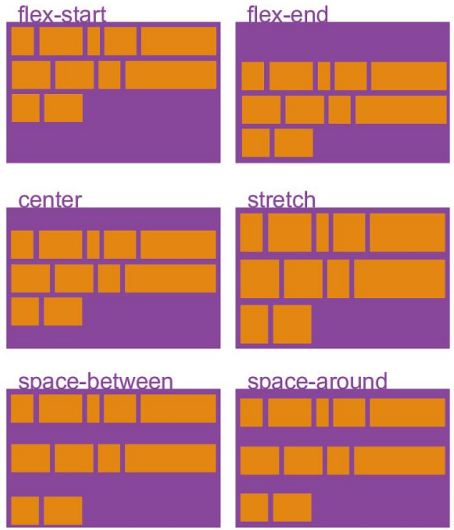
Item alignment when wrapped

align-content

Defines **line** alignment and distributes extra space along the cross axis

Has no effect when the flexbox does not wrap

Values: **flex-start**, **flex-end**, **center**, **space-between**, **space-around**, and **stretch** (default)



css-tricks.com/almanac/properties/a/align-content

Sizing Flex Items

flex-basis: The base size of the flex item.

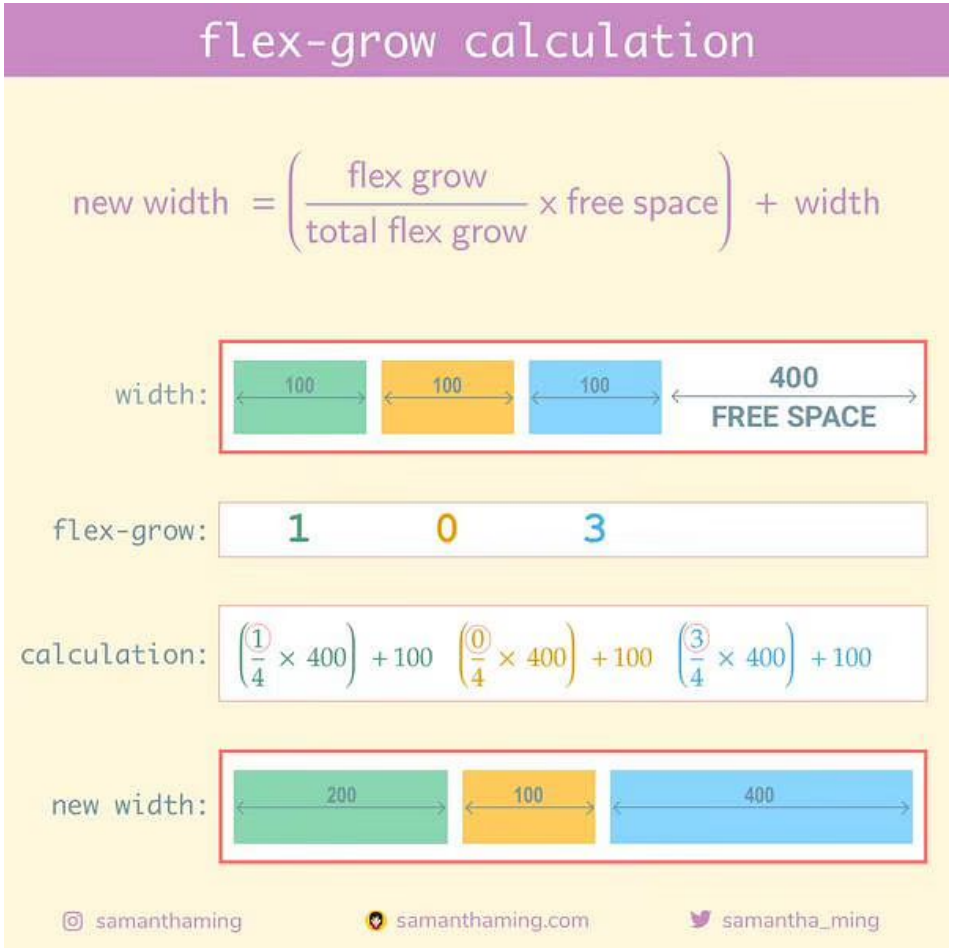
- Functionally similar to width or height, depending on the direction of the flex container, however, it will override width or height.

- Like width and height, flex-basis can be defined in absolute pixels or relative sizing using percentages or the viewport.

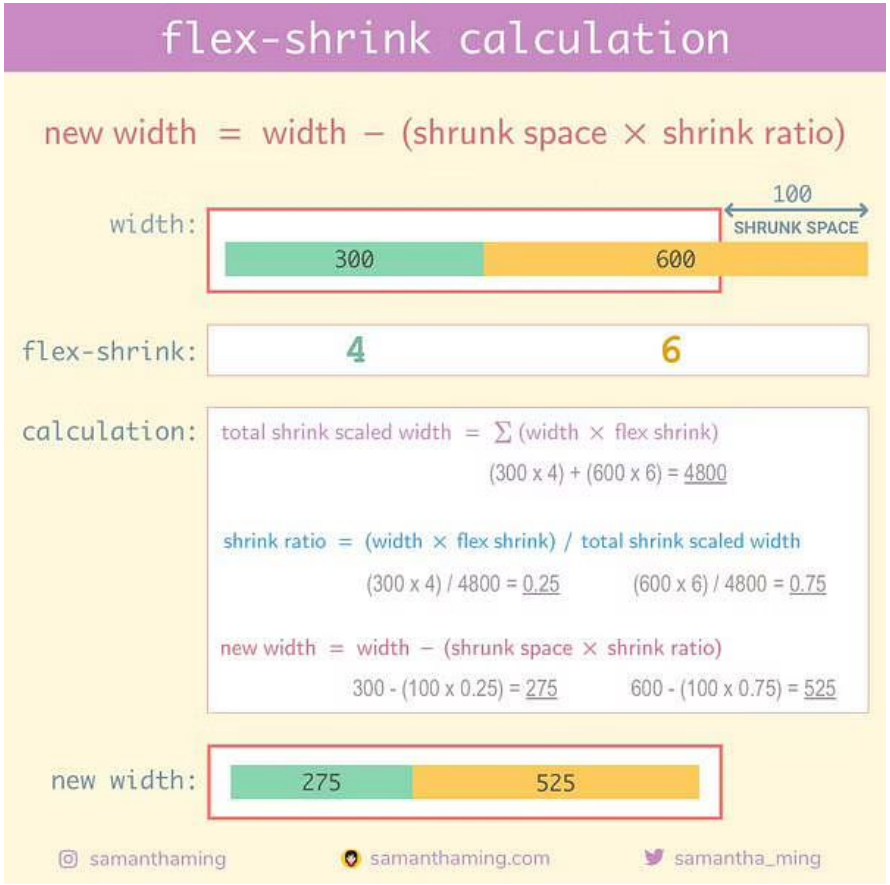
flex-grow: How much a flex item will grow relative to other flex items.

flex-shrink: How much a flex item will shrink relative to other flex items.

Sizing Flex Items



Sizing Flex Items



Images

- Images make up 60% of a webpage's size, on average.
- Use relative sizing for images to prevent them from overflowing the container.
- Use the **<picture>** element to specify different images based on media queries. This is called **art direction**.
- Use **srcset** attribute in the **<img>** element to render different images based on the device's pixel density or viewport size.
- Be choosy about which image file format to use:
  - WebP, PNG, JPEG, SVG, etc.
  - Vector vs. Raster

Resources

Flexbox

- A Complete Guide to Flexbox
- Flexbox Playground
- Flexbox Examples
- Flexbox Froggy

QUESTIONS?