

# Module 3 Day 4

CSS: Responsive Design 2 - Flexbox

# Today's Objectives

- Flexbox Container
  - flex-direction
  - justify-content
  - align-items
  - flex-wrap
- Flexbox Item
  - order
  - flex-grow / flex-shrink
  - flex-basis

# Flexbox - Background

## CSS Flexbox Layout Module

Before the Flexbox Layout module, there were four layout modes:

- Block, for sections in a webpage
- Inline, for text
- Table, for two-dimensional table data
- Positioned, for explicit position of an element

The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.

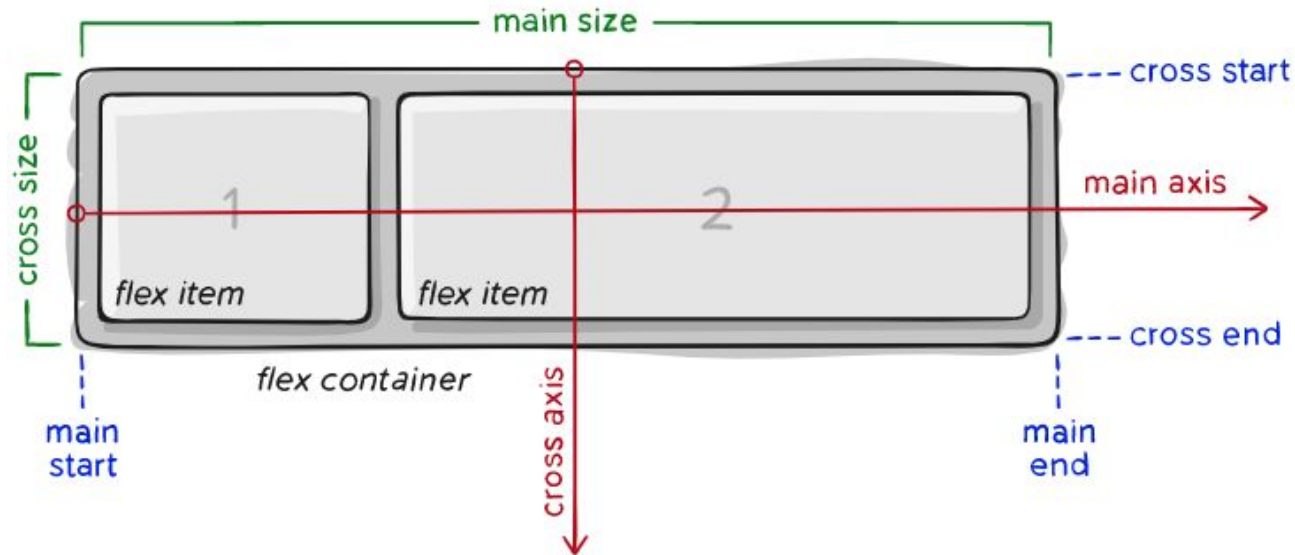
[https://www.w3schools.com/css/css3\\_flexbox.asp](https://www.w3schools.com/css/css3_flexbox.asp)

# Flexbox (Flexible Box)

Flexible Box Layout Module allows us to design a flexible, responsive layout without using float or positioning.

```
.flex-container {  
  display: flex;  
}
```

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



[CSS Flexbox \(Flexible Box\)](#)  
[A Complete Guide to Flexbox](#)

# Flexbox Container Attributes

# Flexbox: justify-content

**justify-content** tells a flex container how to layout the items within it on the main axis.

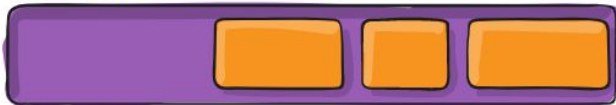
```
.flex-container {  
  display: flex;  
  justify-content: flex-start;  
}
```

[A Complete Guide to Flexbox](#)

flex-start



flex-end



center



space-between



space-around



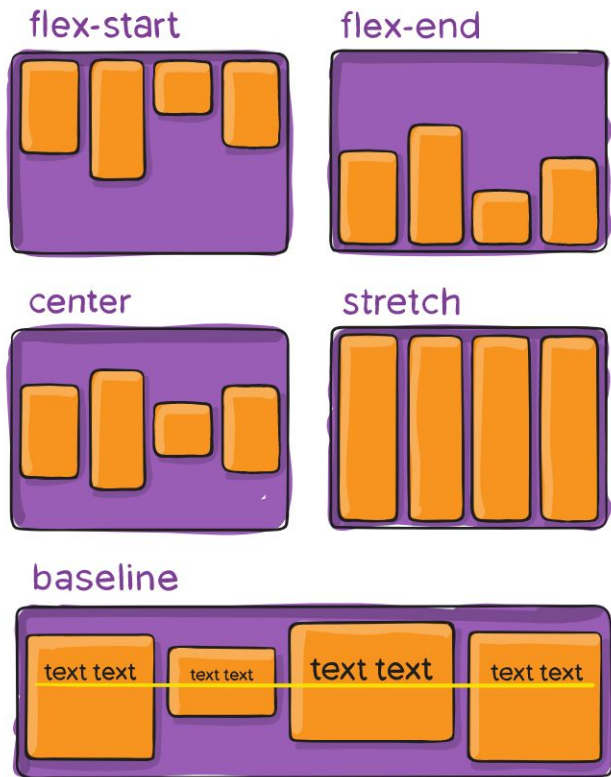
space-evenly



# Flexbox: align-items

**align-items** will layout and align items on the cross axis

```
.flex-container {  
  display: flex;  
  height: 200px;  
  align-items: flex-start;  
}
```

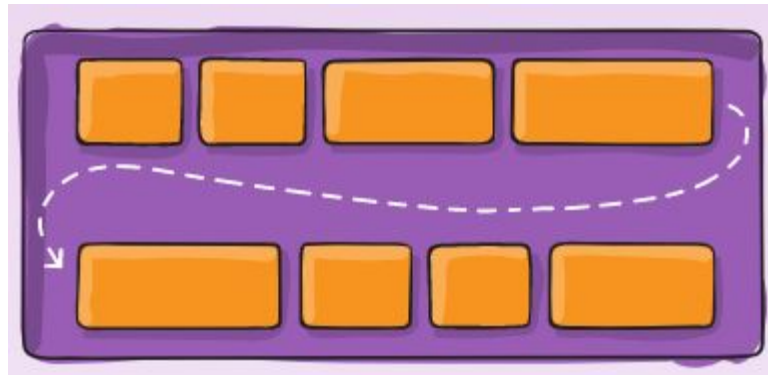


# Flexbox: flex-wrap

**flex-wrap** will tell the flex container that items may wrap.

```
.flex-container {  
  display: flex;  
  flex-wrap: wrap;  
}
```

- **nowrap** (default): all flex items will be on one line
- **wrap**: flex items will wrap onto multiple lines, from top to bottom.
- **wrap-reverse**: flex items will wrap onto multiple lines from bottom to top



[A Complete Guide to Flexbox](#)

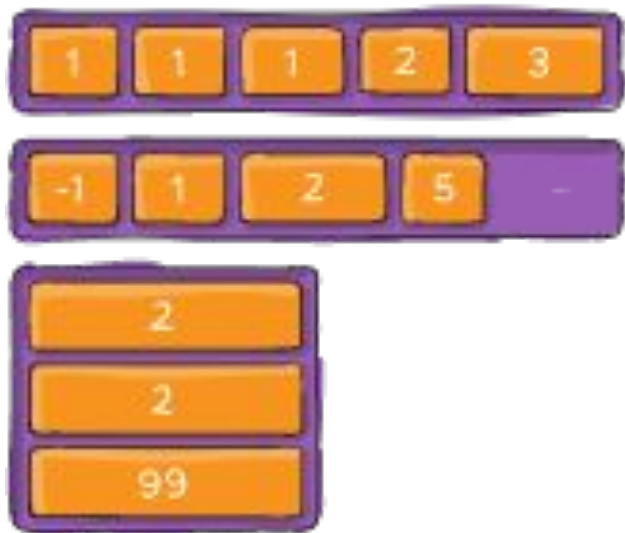


# Flexbox Item Attributes

# Flexbox: Order

**order** dictates the 0 based position of the item in the collection of container items

```
.flex-item {  
  order: 3;  
}
```



# Flexbox: Flex-Grow & Flex-Shrink

**flex-grow:** governs the ability of a flex item to grow. The value is unit-less and sets the proportion of available space inside the flex container that the item should take up.

```
.flex-item-2 {  
  flex-grow: 2;  
}
```

**flex-shrink:** has the opposite effect of **flex-grow**

[A Complete Guide to Flexbox](#)

Default is 1, space is evenly distributed



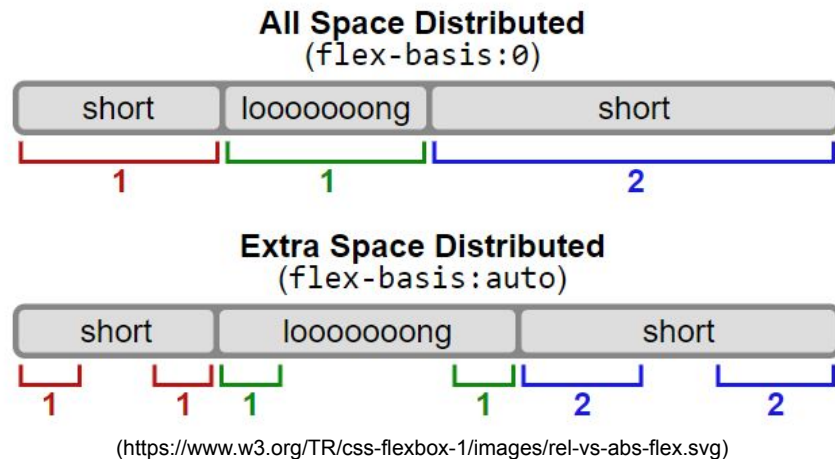
If one item has flex-grow set to 2, it will take up twice as much space as the other items (if possible)

# Flexbox: Flex-basis

## flex-basis:

- Defines the default size of an item before any remaining space is distributed.
- It is length and takes a unit of measure or a keyword.
  - **auto** defers to the element's width/height properties.
  - **content** sizes the item based on its' content\*

Note: If set to 0, the extra space around content isn't factored in. If set to auto, the extra space is distributed based on its flex-grow value.



```
.flex-item {  
  flex-basis: auto; /*← default */  
}
```

[A Complete Guide to Flexbox](#)

# What questions do you have?



[Flexbox Froggy - A game for learning CSS flexbox](#)