

flexbox

CSS Modern Layout Module

CSS **Layout** **Mode**

Block 레이아웃 문서 배치

Inline 레이아웃 텍스트 배치

Table 레이아웃 표 배치

Positioning 레이아웃 격리된 배치

Flexbox 레이아웃 —————

Grid 레이아웃

flexbox

Flex 레이아웃을 정의하는 목적은 아이템의 폭과 높이를 모든 디스플레이 장치에서 사용할 수 있는 공간을 최적으로 채우도록 변경하는 능력입니다.

flexbox

Flexbox 레이아웃 알고리즘은 수직 방향으로 편향된 Block 레이아웃이나
수평 방향으로 편향된 Inline 레이아웃과는 대조적으로,
방향에 대해 독단적하지 않습니다.

flexbox

Block 레이아웃 모드는 UA 변경이나 수평/수직의 전환 등에 의한 방향의 변화, 크기 조정, 확대, 축소 등에 적합한 방법을 제공하고 있지 않습니다. 반면 Flexbox 레이아웃 모드는 애플리케이션 컴포넌트 및 작고 다양한 화면 레이아웃에 적합합니다. 다른 레이아웃 모드인 Grid는 큰 화면에 적합합니다.

TABLE OF CONTENTS

1	Introduction
1.1	Overview
1.2	Module interactions
2	Flex Layout Box Model and Terminology
3	Flex Containers: the 'flex' and 'inline-flex' 'display' values
4	Flex Items
4.1	Absolutely-Positioned Flex Children
4.2	Flex Item Margins and Paddings
4.3	Flex Item Z-Ordering
4.4	Collapsed Items
4.5	Implied Minimum Size of Flex Items
5	Ordering and Orientation
5.1	Flex Flow Direction: the 'flex-direction' property
5.2	Flex Line Wrapping: the 'flex-wrap' property
5.3	Flex Direction and Wrap: the 'flex-flow' shorthand
5.4	Display Order: the 'order' property
5.4.1	Reordering and Accessibility
6	Flex Lines

CSS Flexible Box Layout Module Level 1

W3C Candidate Recommendation, 26 May 2016



This version:

<http://www.w3.org/TR/2016/CR-css-flexbox-1-20160526/>

Latest published version:

<http://www.w3.org/TR/css-flexbox-1/>

Editor's Draft:

<https://drafts.csswg.org/css-flexbox/>

Previous Versions:

<http://www.w3.org/TR/2016/CR-css-flexbox-1-20160301/>

<http://www.w3.org/TR/2015/WD-css-flexbox-1-20150514/>

<http://www.w3.org/TR/2014/WD-css-flexbox-1-20140925/>

<http://www.w3.org/TR/2014/WD-css-flexbox-1-20140325/>

<http://www.w3.org/TR/2012/CR-css3-flexbox-20120918/>

<http://www.w3.org/TR/2012/WD-css3-flexbox-20120612/>

<http://www.w3.org/TR/2012/WD-css3-flexbox-20120322/>

<http://www.w3.org/TR/2011/WD-css3-flexbox-20111129/>

<http://www.w3.org/TR/2011/WD-css3-flexbox-20110322/>

<http://www.w3.org/TR/2009/WD-css3-flexbox-20090723/>

Feedback:

www-style@w3.org with subject line "[css-flexbox] ... message topic ..." ([archives](#))

Test Suite:

http://test.csswg.org/suites/css-flexbox-1_dev/nightly-unstable/

Editors:

Flexbox

CSS Modern Layout Module



Flexbox

CSS Modern Layout Module



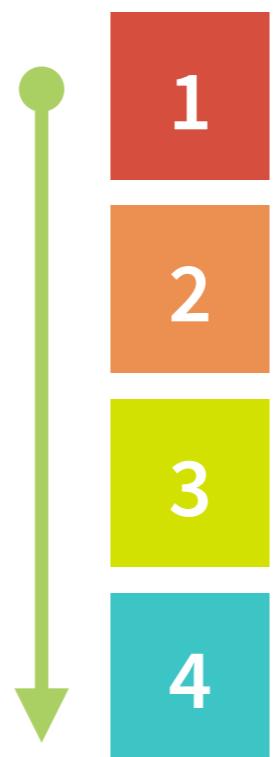
Flexbox

CSS Modern Layout Module



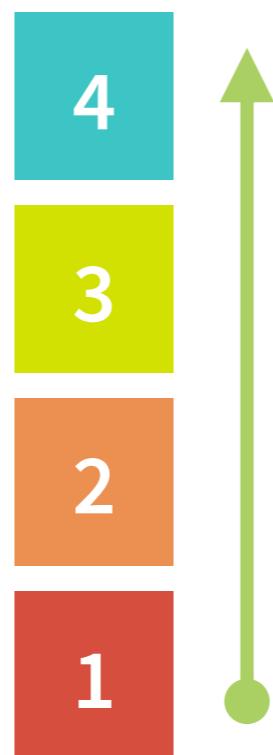
Flexbox

CSS Modern Layout Module



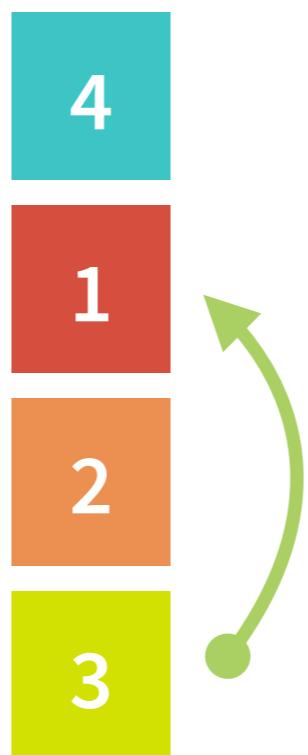
Flexbox

CSS Modern Layout Module



Flexbox

CSS Modern Layout Module



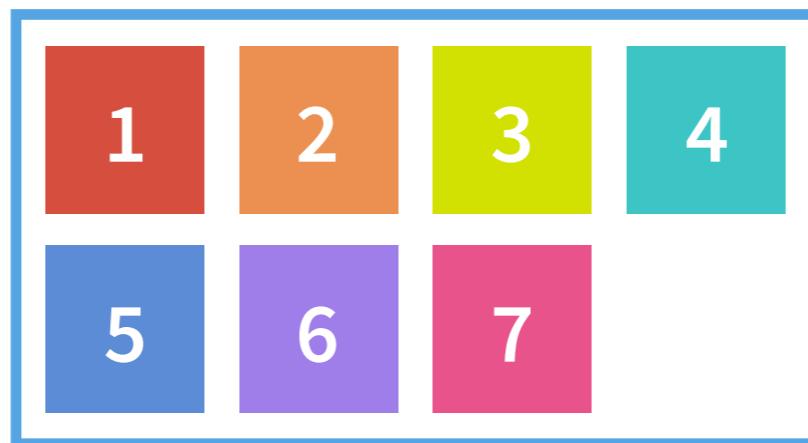
Flexbox

CSS Modern Layout Module



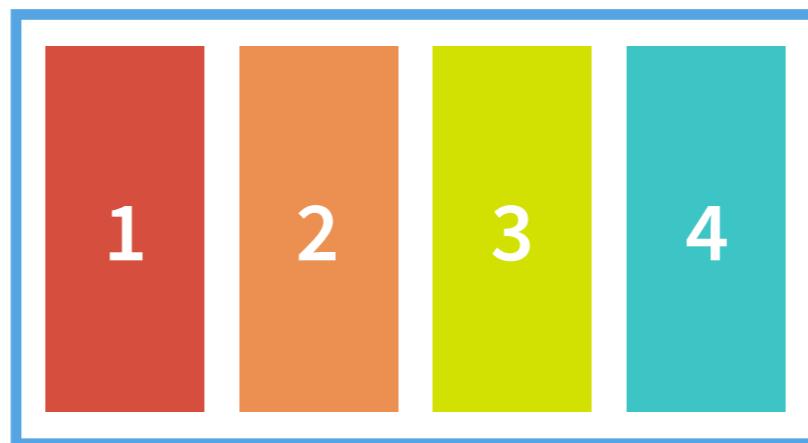
Flexbox

CSS Modern Layout Module



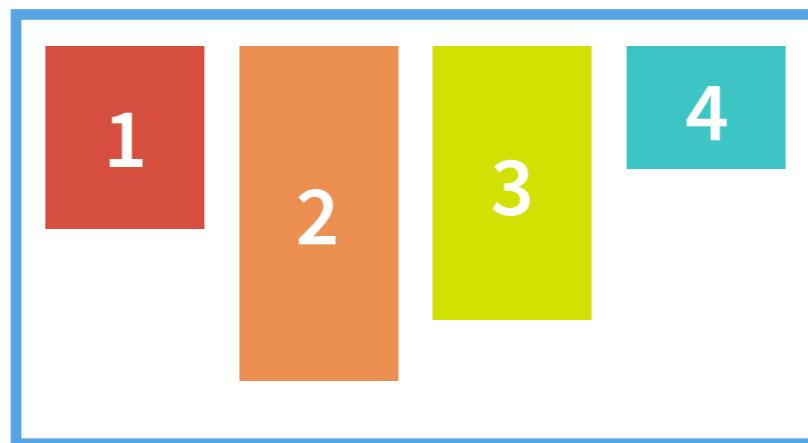
Flexbox

CSS Modern Layout Module

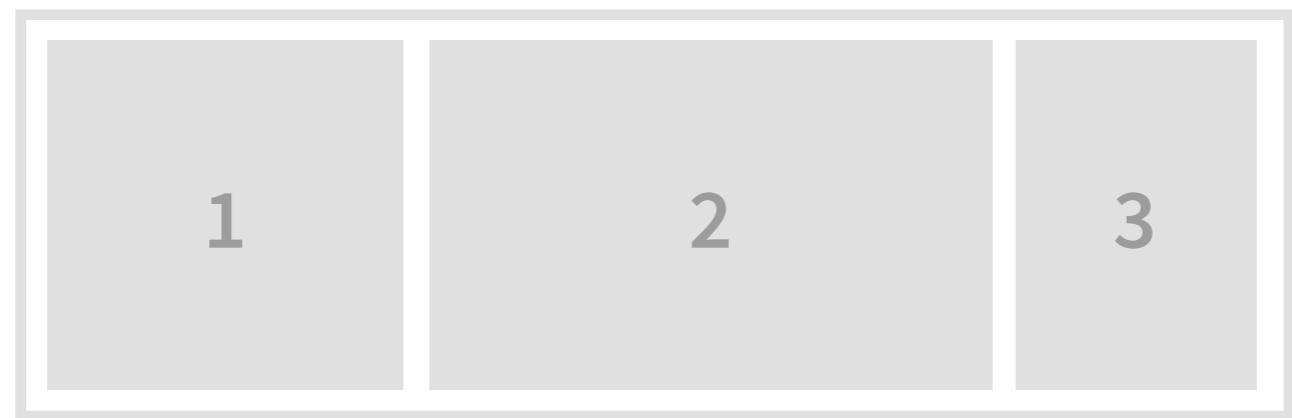


Flexbox

CSS Modern Layout Module



Flexbox Anatomy



CSS Modern Layout Module

Flex Container



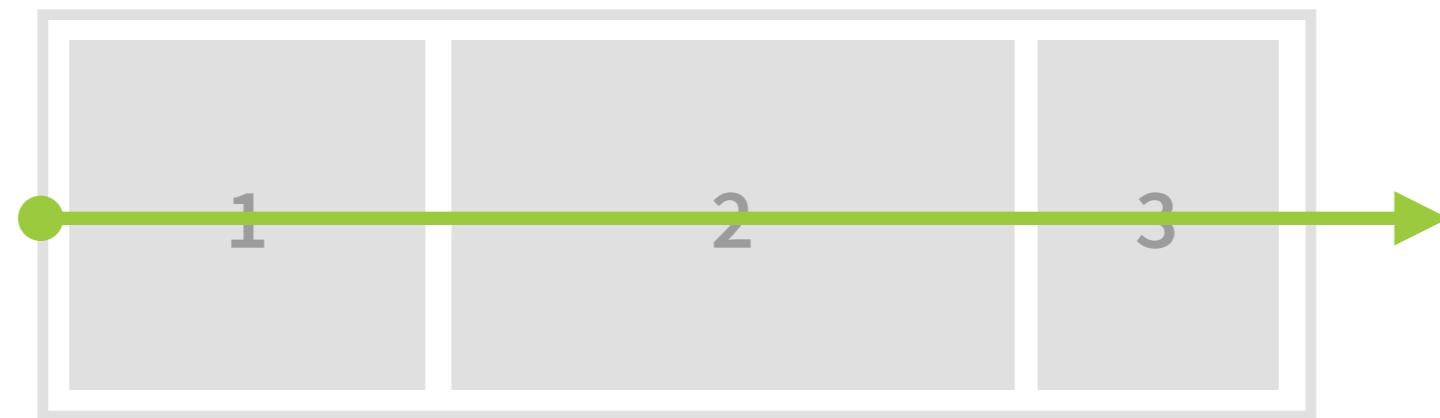
CSS Modern Layout Module

Flex Items



CSS Modern Layout Module

Main Axis



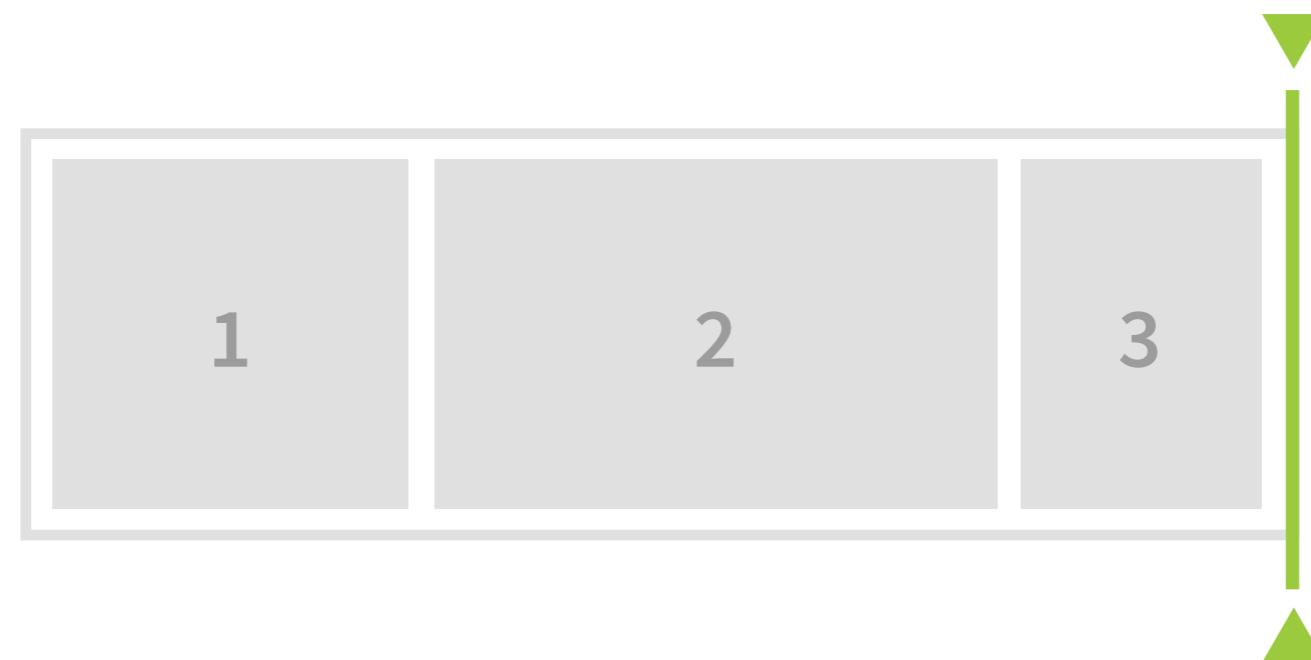
CSS Modern Layout Module

Main Start



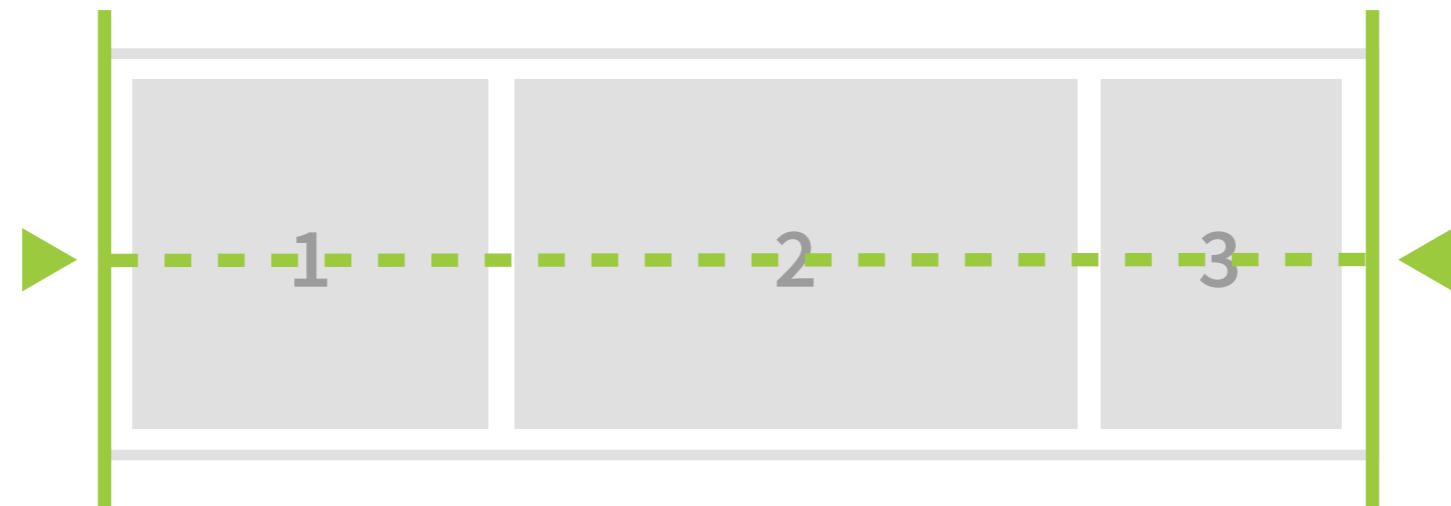
CSS Modern Layout Module

Main End



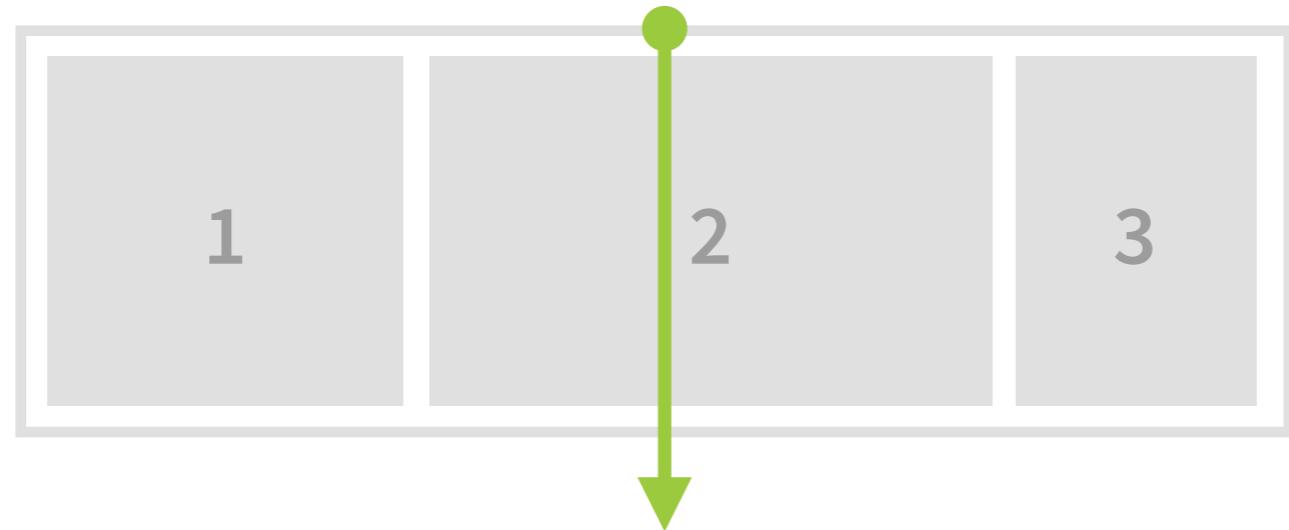
CSS Modern Layout Module

Main Size



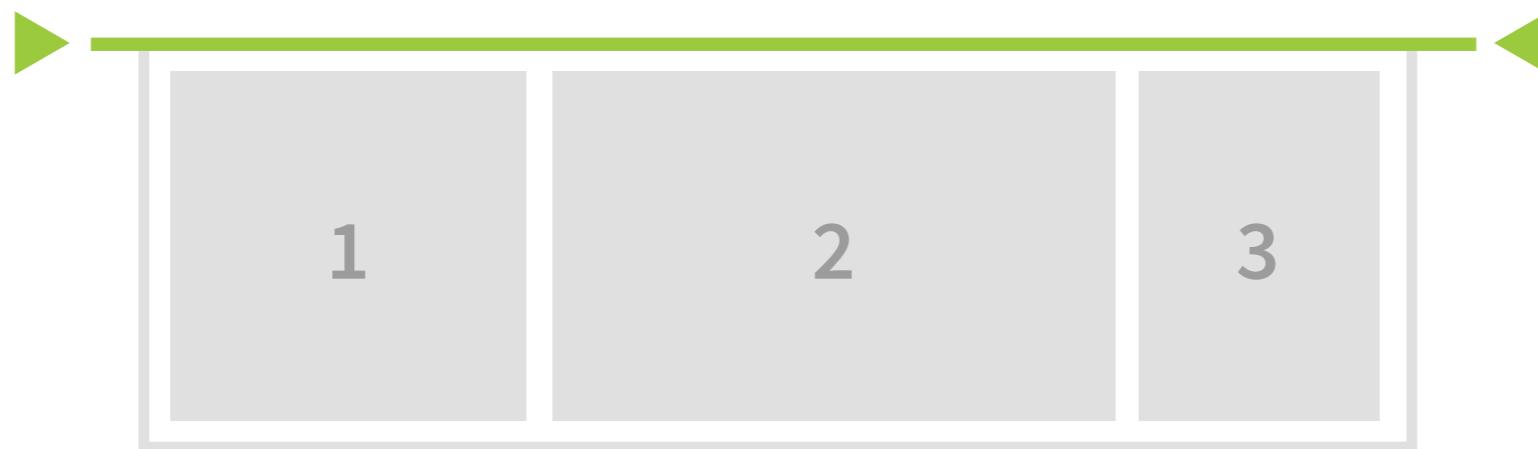
CSS Modern Layout Module

Cross Axis



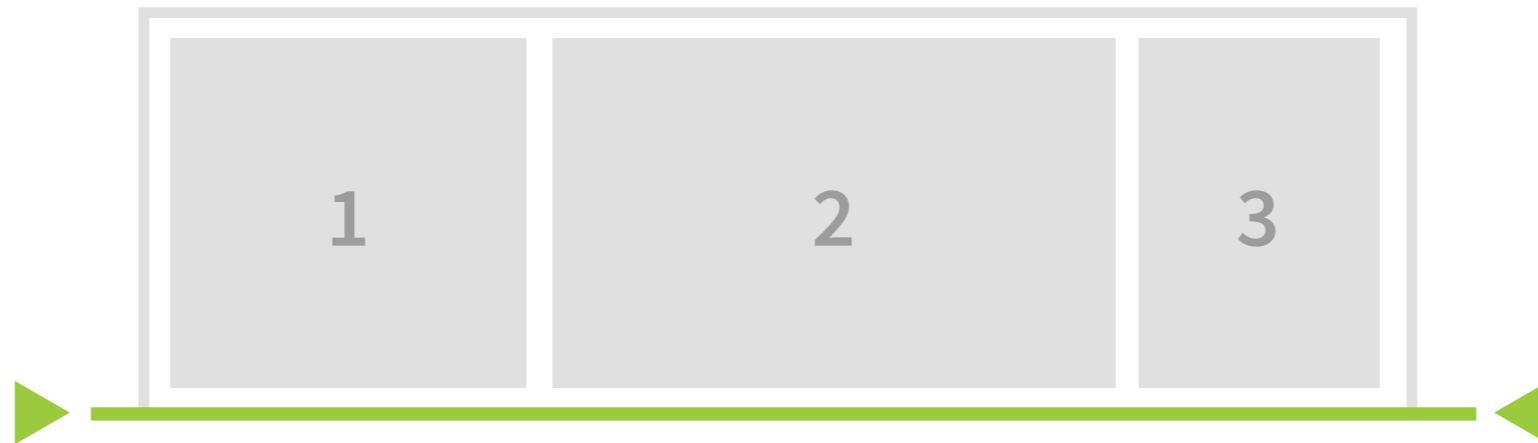
CSS Modern Layout Module

Cross Start



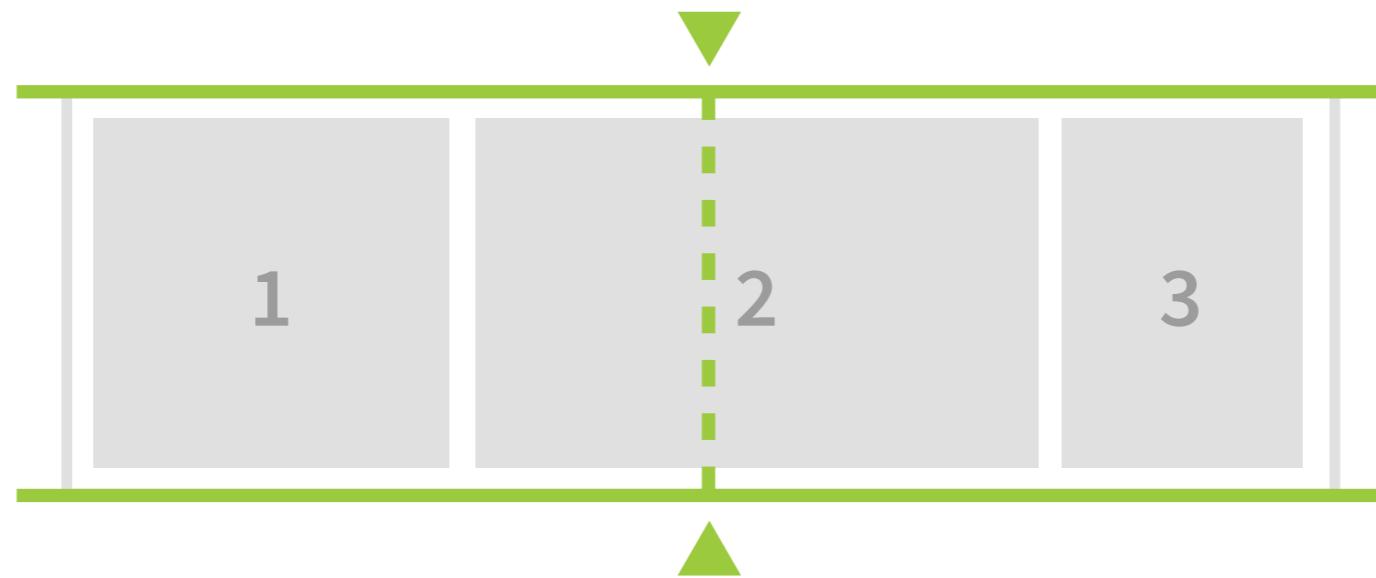
CSS Modern Layout Module

Cross End



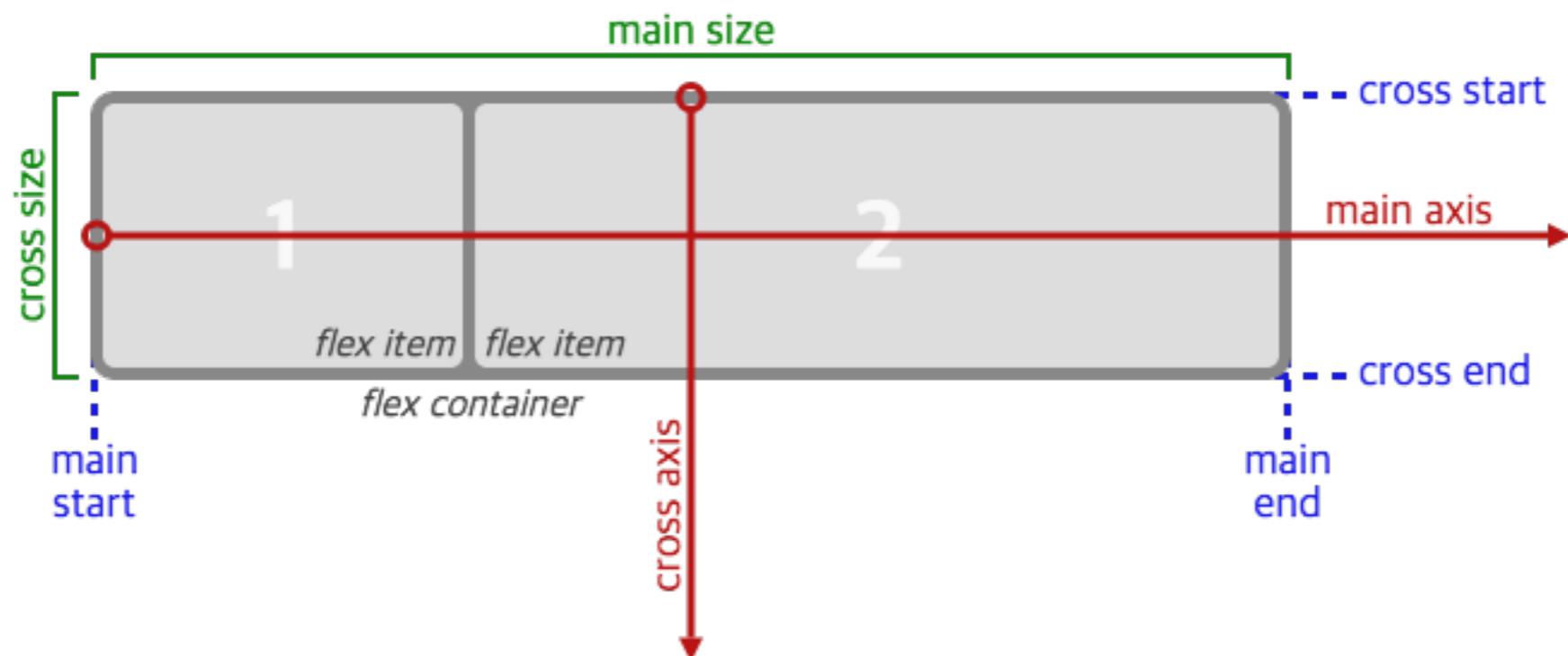
CSS Modern Layout Module

Cross Size



CSS Modern Layout Module

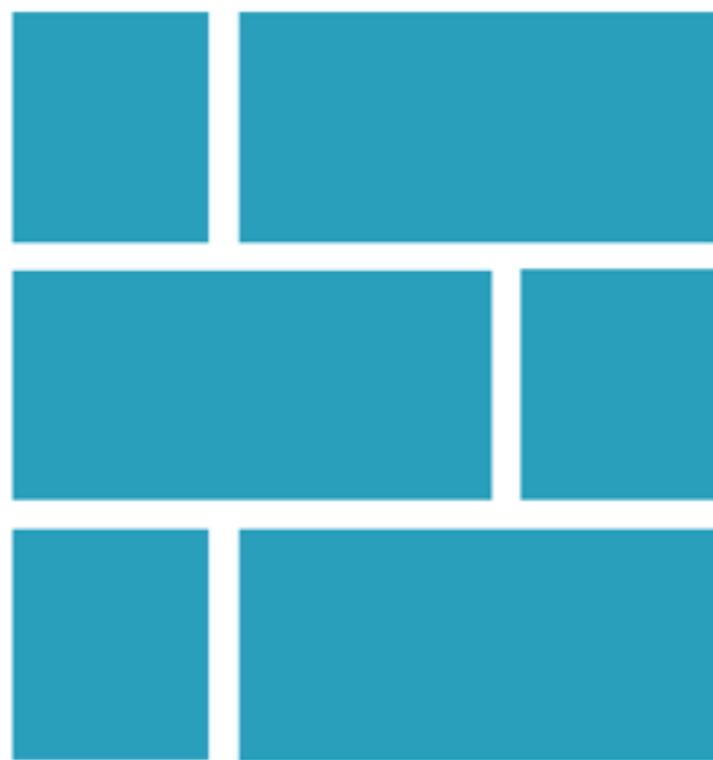
W3C Reference



CSS Modern Layout Module

Flexbox

CSS Modern Layout Module





Flex Container

display: **flex**;



CSS Modern Layout Module

Flex Container

display: **inline-flex**;

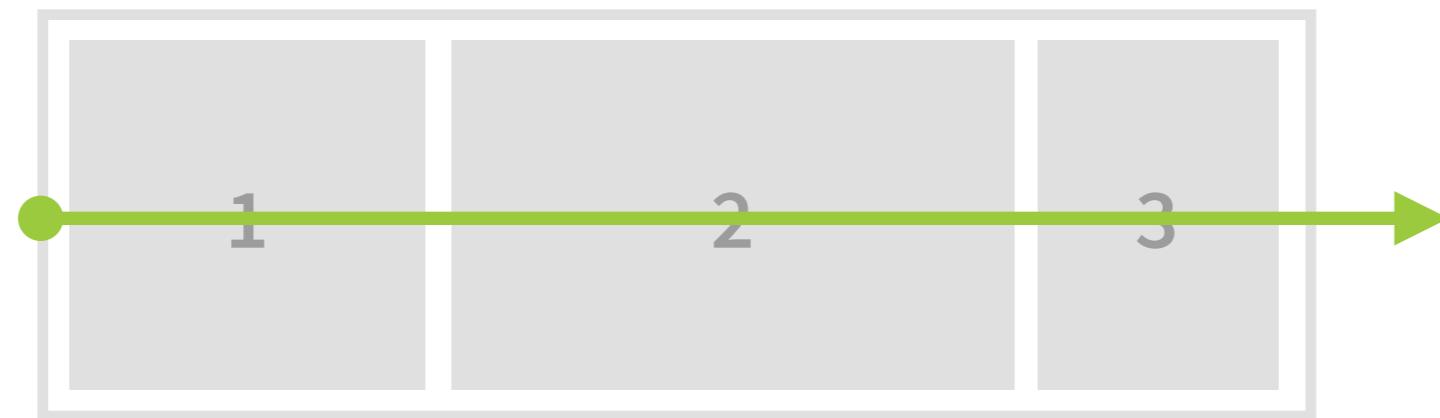


CSS Modern Layout Module

Flex Flow Direction

CSS Modern Layout Module

Main Axis



CSS Modern Layout Module

Flex Start



CSS Modern Layout Module

Flex End

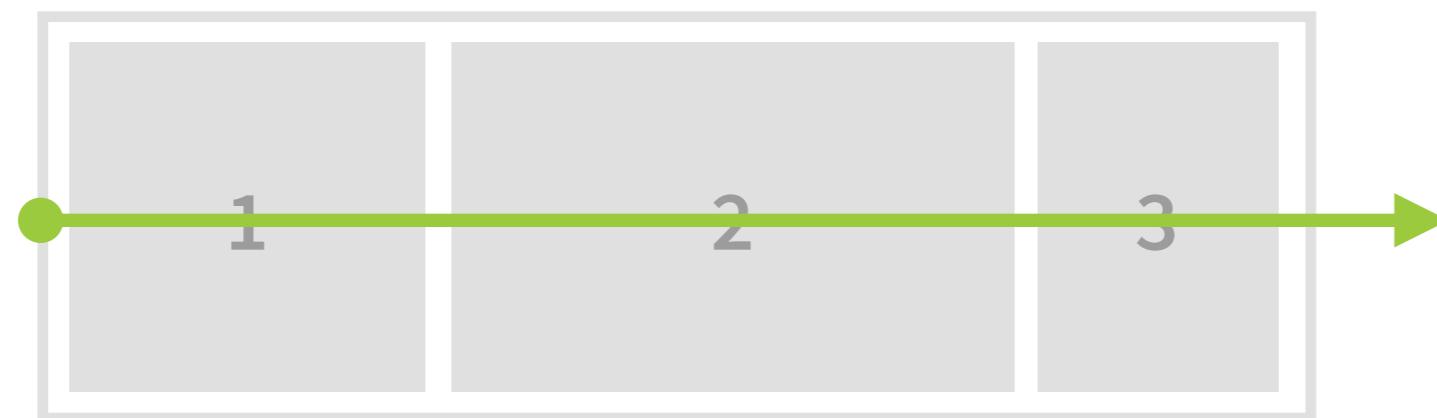


CSS Modern Layout Module



Flex Direction

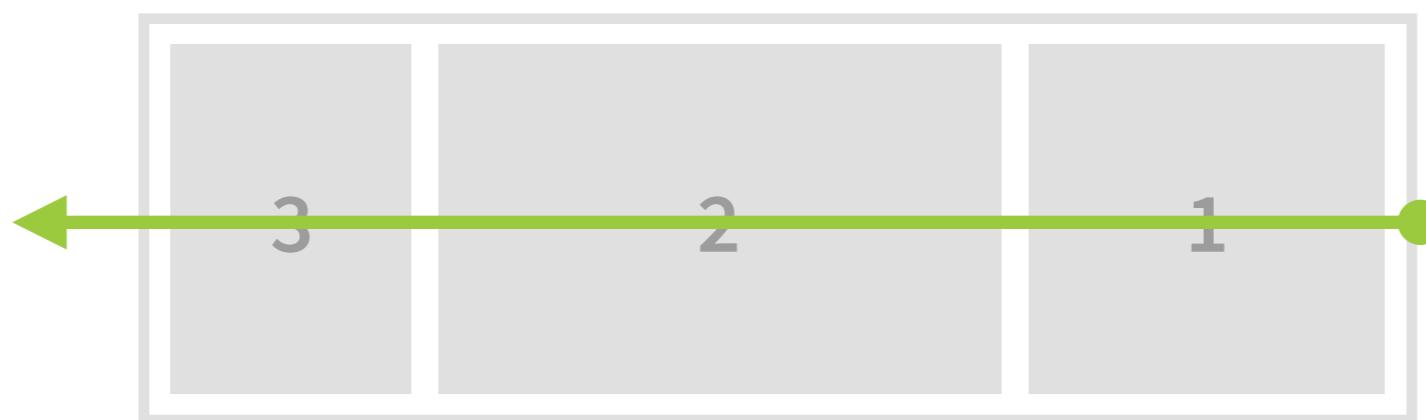
`flex-direction: row;`



CSS Modern Layout Module

Flex Direction

`flex-direction: row-reverse;`



CSS Modern Layout Module

Flex Direction

flex-direction: column;



Flex Direction

flex-direction: column-reverse;



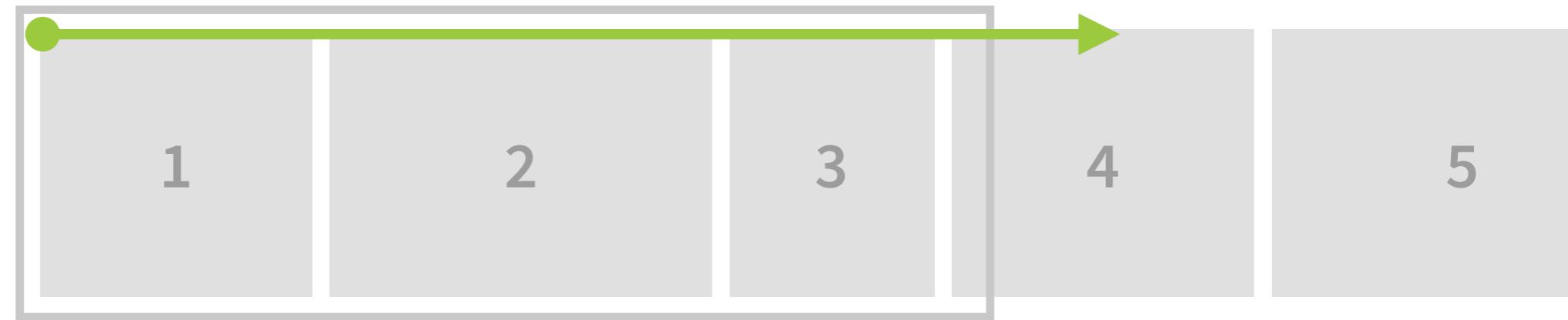
Flex Line Wrapping

CSS Modern Layout Module



Single Line Formatting

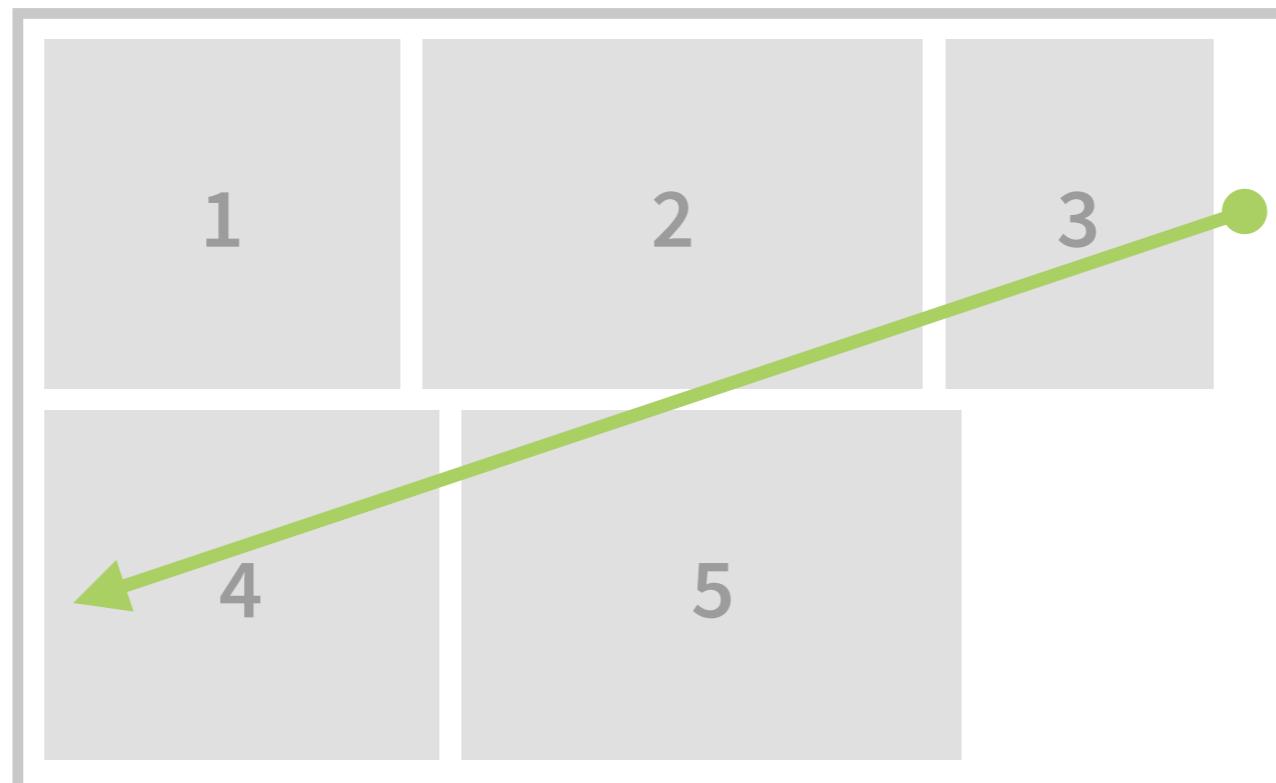
`flex-wrap: nowrap;`



CSS Modern Layout Module

Multi-Line Formatting

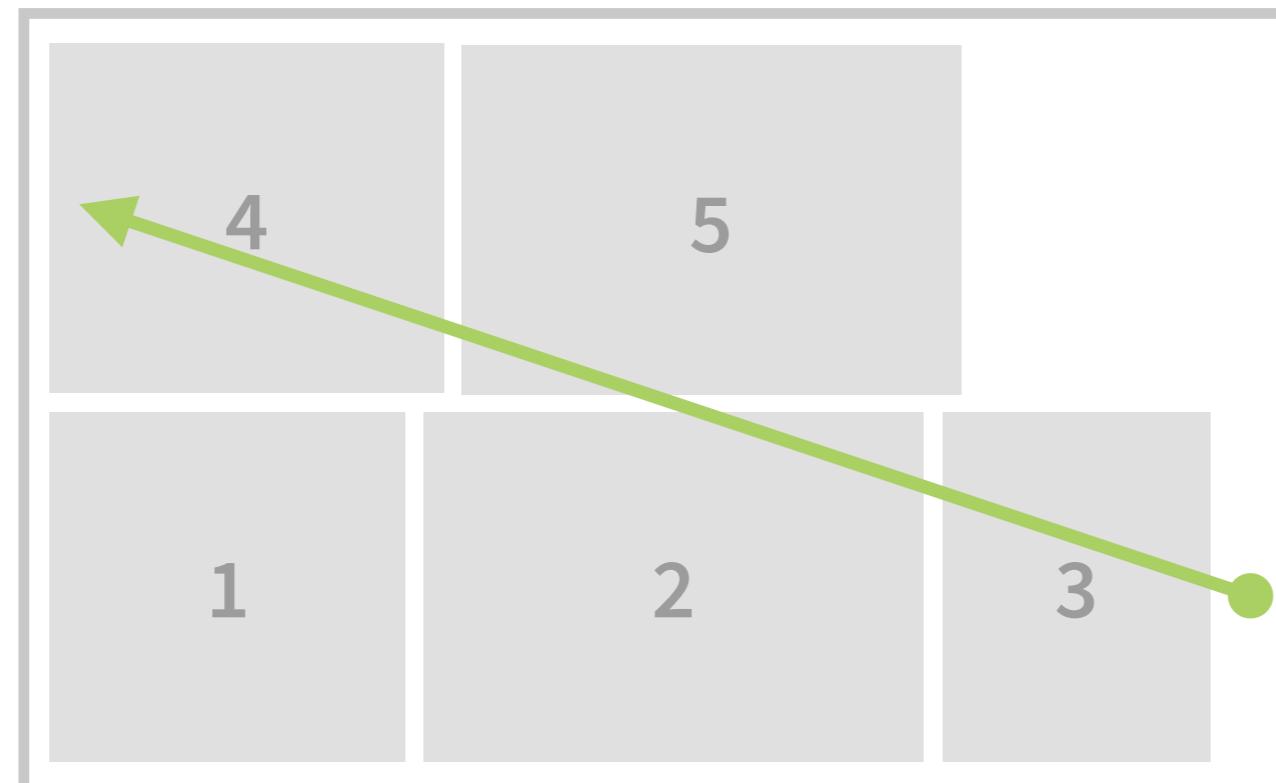
`flex-wrap: wrap;`



CSS Modern Layout Module

Multi-Line Formatting

`flex-wrap: wrap-reverse;`



CSS Modern Layout Module

Shorthand

CSS Modern Layout Module

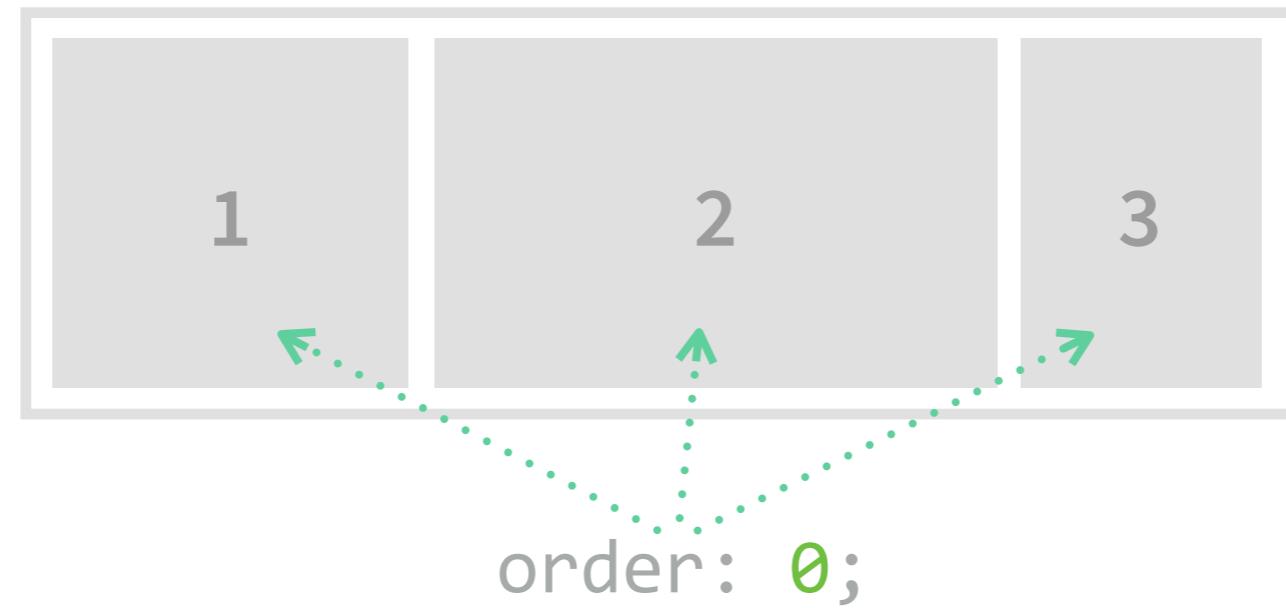
`flex-flow: <flex-direction> <flex-wrap>;`

CSS Modern Layout Module

Display Order

CSS Modern Layout Module

order



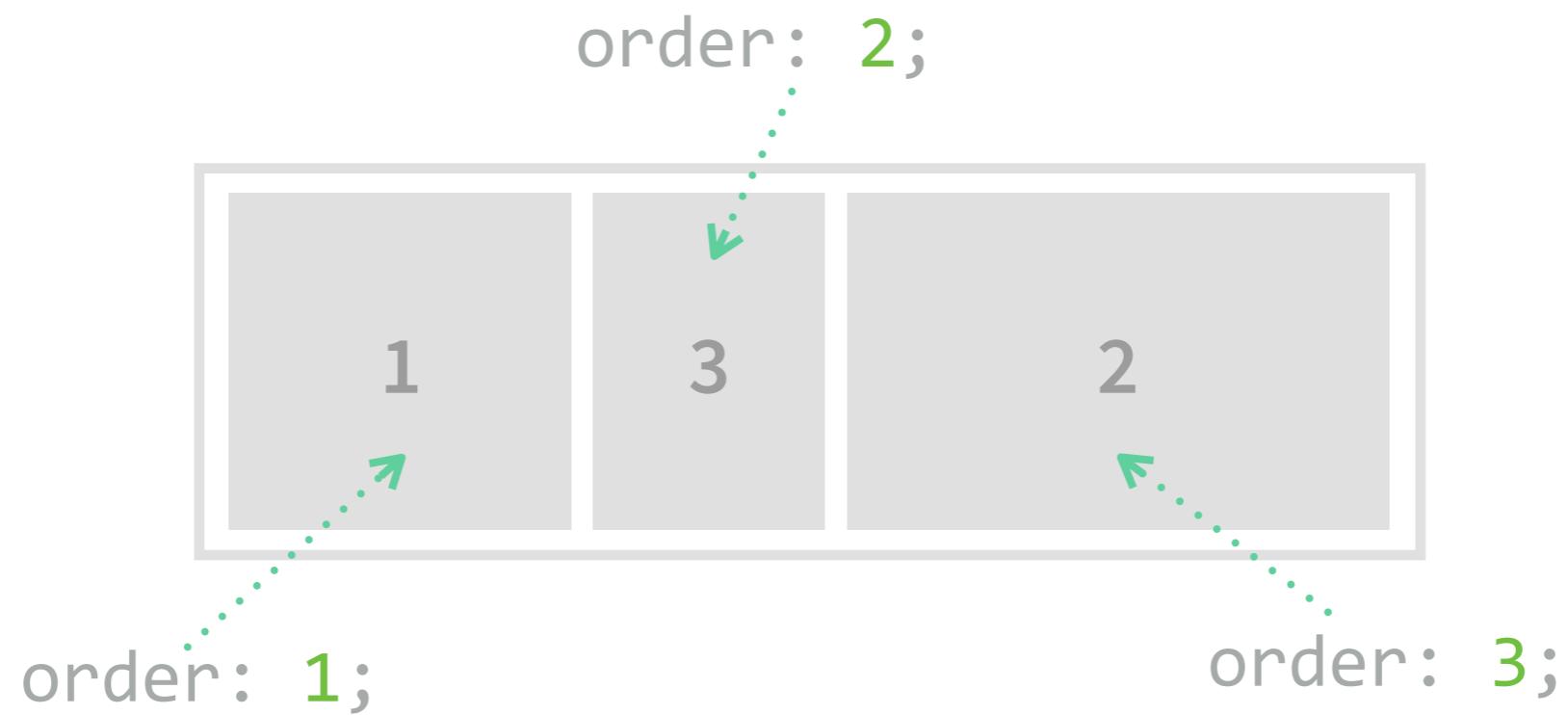
CSS Modern Layout Module

order



CSS Modern Layout Module

order



CSS Modern Layout Module

Accessibility

마크업 순서가 변경된 것은 아님



Visual Purposes

시각적으로만 순서가 변경됨



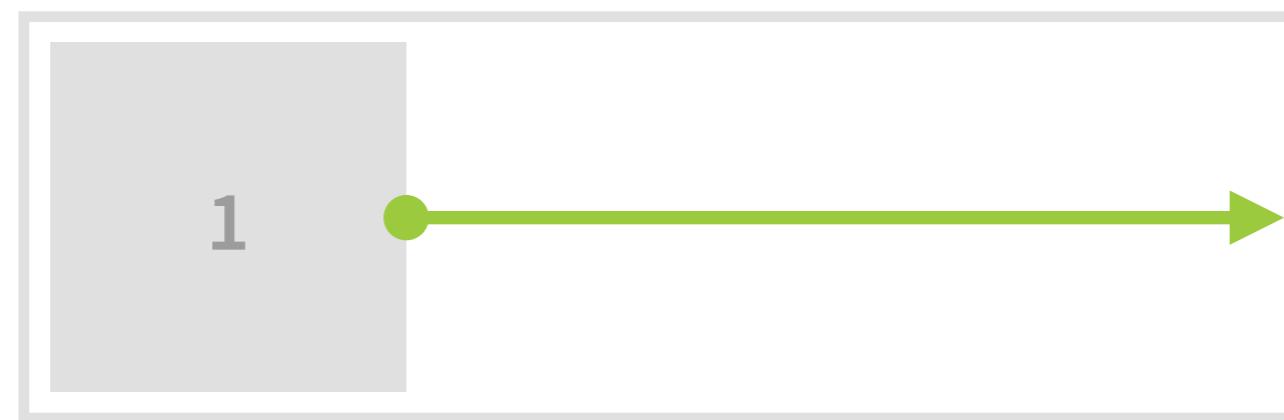
CSS Modern Layout Module

Flexibility

CSS Modern Layout Module

Flex Grow Factor

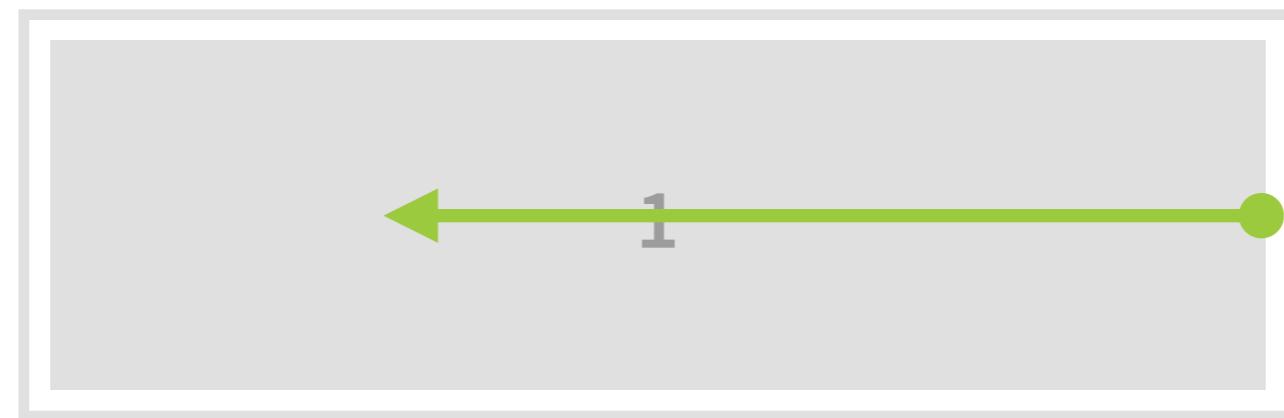
`flex-grow: 1;`



CSS Modern Layout Module

Flex Shrink Factor

`flex-shrink: 1;`



CSS Modern Layout Module

Initial Size

`flex-basis: 100px;`



CSS Modern Layout Module

Shorthand

CSS Modern Layout Module

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

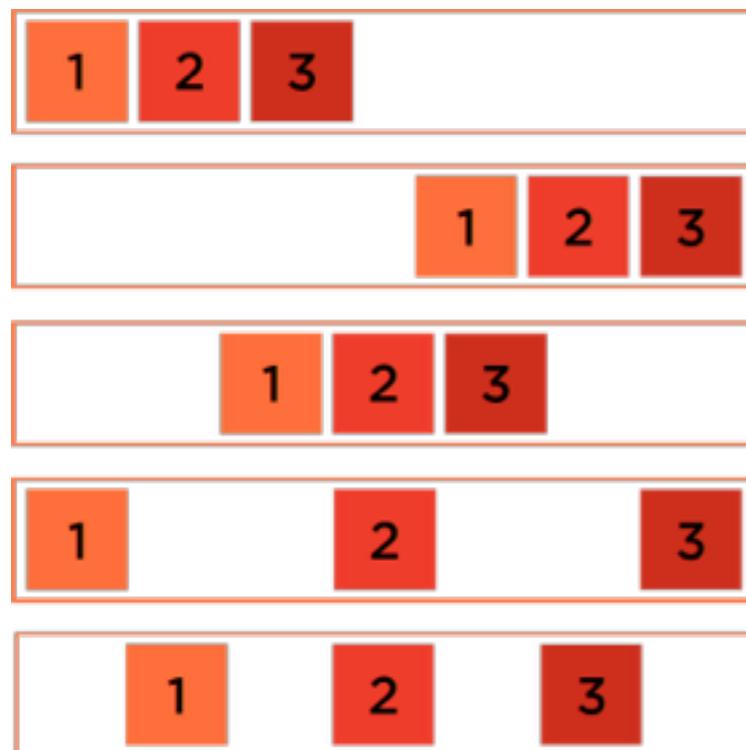
CSS Modern Layout Module

Alignment

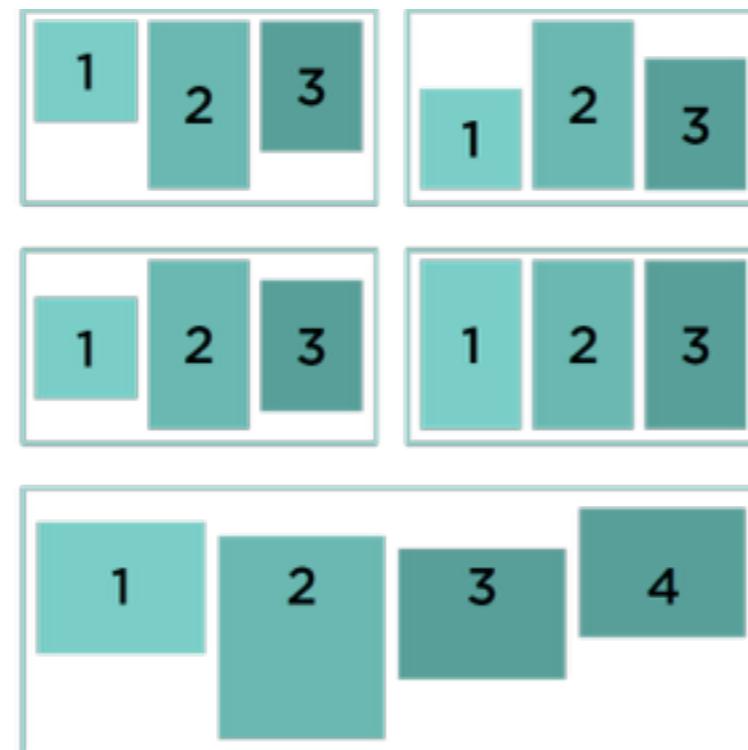
CSS Modern Layout Module

On the Container

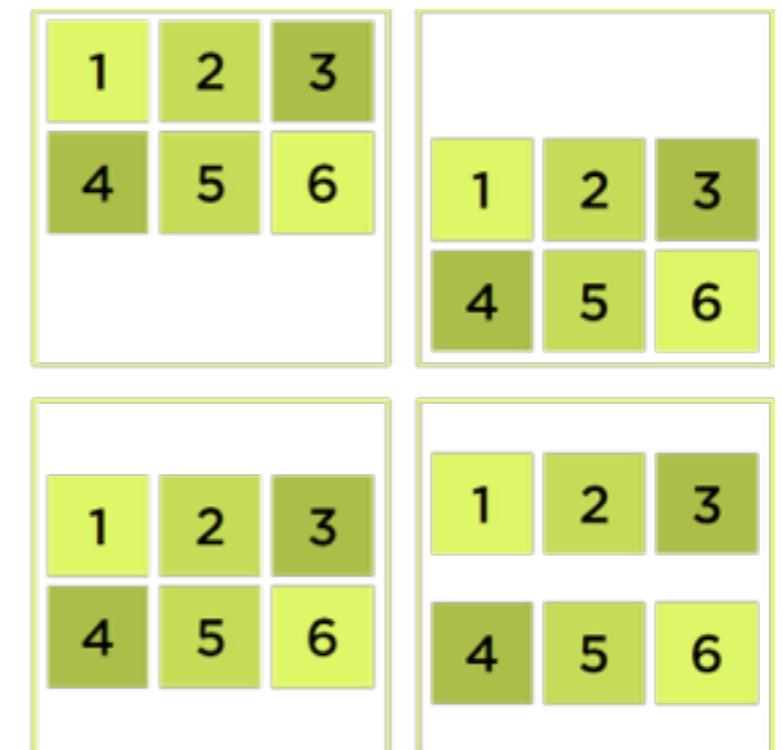
justify-content



align-items



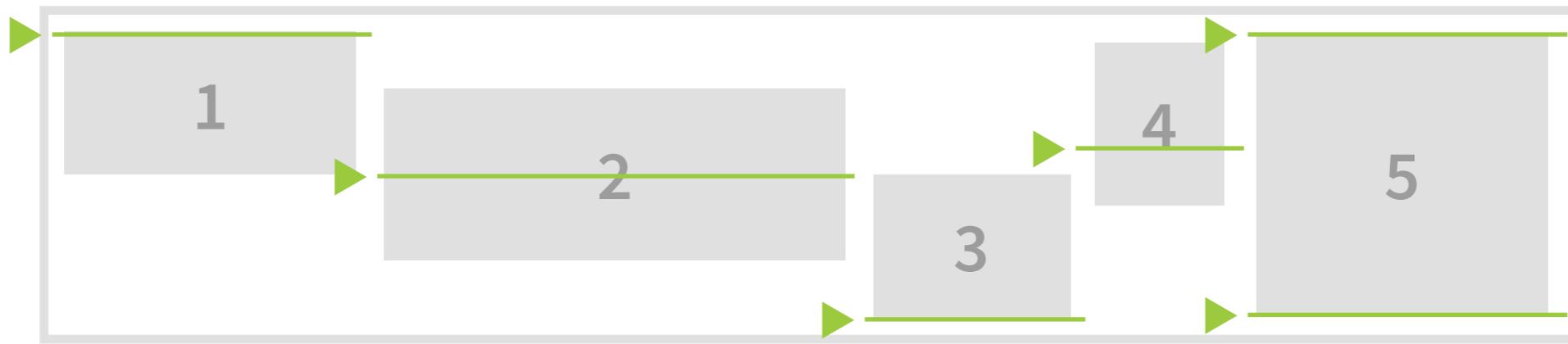
align-content



CSS Modern Layout Module

On the Item

align-self



CSS Modern Layout Module

References

CSS Modern Layout Module

Using CSS Flexible Boxes

SEE ALSO

CSS
CSS Reference
CSS Flexible Box Layout

▼ Guides
Advanced layouts with flexbox
Using CSS Flexible Boxes
Using flexbox to lay out web applications

▼ Properties
align-content
align-items
align-self
flex
flex-basis
flex-direction
flex-flow
flex-grow
flex-shrink
flex-wrap
justify-content
order

The CSS3 Flexible Box, or flexbox, is a [layout mode](#) providing for the arrangement of elements on a page such that the elements behave predictably when the page layout must accommodate different screen sizes and different display devices. For many applications, the flexible box model provides an improvement over the block model in that it does not use floats, nor do the flex container's margins collapse with the margins of its contents.

Many designers will find the flexbox model easier to use. Child elements in a flexbox can be laid out in any direction and can have flexible dimensions to adapt to the display space. Positioning child elements is thus much easier, and complex layouts can be achieved more simply and with cleaner code, as the display order of the elements is independent of their order in the source code. This independence intentionally affects only the visual rendering, leaving speech order and navigation based on the source order.

 Note: Though [CSS Flexible Boxes Layout specification](#) is at the Last Call Working Draft stage (see also the [Latest Editor draft](#)), not all browsers have implemented all features of Flexbox. That said, there is now good support across the board for flexbox. See the [compatibility table](#) on each property for up-to-date compatibility status.

Flexible Boxes Concept

The defining aspect of the flex layout is the ability to alter its items' width and/or height to best fill the available space on any display device. A flex container expands items to fill available free space, or shrinks them to prevent overflow.

The flexbox layout algorithm is direction-agnostic as opposed to the block layout, which is vertically-biased, or the inline layout, which is horizontally-biased. While the block layout works well for pages, it lacks sufficient definition to support application components that have to change orientation, resize, stretch, or shrink as the user agent changes, flips from vertical to horizontal, and so forth. Flexbox layout is most appropriate for the components of an application, and small-scale layouts, while the

IN THIS ARTICLE

[Flexible Boxes Concept](#)
[Flexible Boxes Vocabulary](#)
[Designating a Flexible Box](#)
[Flex Item Considerations](#)
[Flexible Box Properties](#)
[Properties not affecting flexible boxes](#)

[Examples](#)
[Basic flex example](#)
[Holy Grail Layout example](#)
[Playground](#)
[Things to keep in mind](#)
[Browser compatibility](#)
[See also](#)





Visual Guide to CSS3 Flexbox Properties

Code

Demo

css

flexbox

A Visual Guide to CSS3 Flexbox Properties



Dimitar Stojanov

Apr 13, 2015

Tutorials

Comments

0



Subscribe & get our FREE eBook
Build Your First Node.js App



Email Address

Free Node Book



Scotch School

Top shelf courses coming
August 30th. 50% off for early
enrollment!

Popular Scotchers



Chris Sevilleja



Nicholas Cerminara



Holly Lloyd



Ado Kukic



Dinesh Chankie

[Code Snippets](#) » [CSS](#) »

A Complete Guide to Flexbox

BY CHRIS COYIER LAST UPDATED ON MAY 18, 2016

- ▶ [Background](#)
- ▶ [Basics & Terminology](#)

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/#flexbox-background>

items 



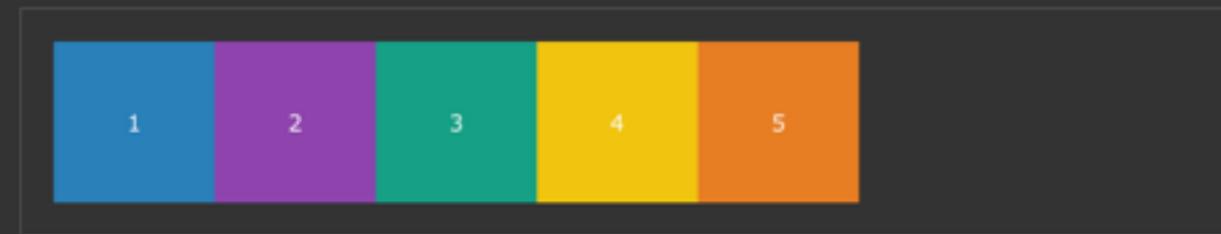
Change View

9

Properties for the flex container

FLEX-DIRECTION (property of the flex container)

- row
- row-reverse
- column
- column-reverse



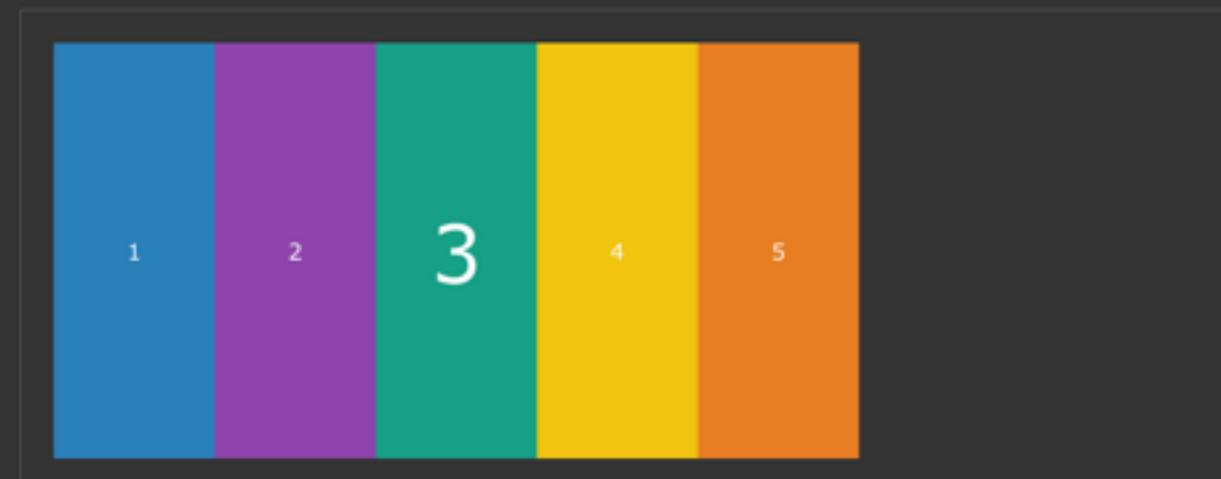
FLEX-WRAP (property of the flex container)

- nowrap
- wrap
- wrap-reverse



ALIGN-ITEMS (property of the flex container)

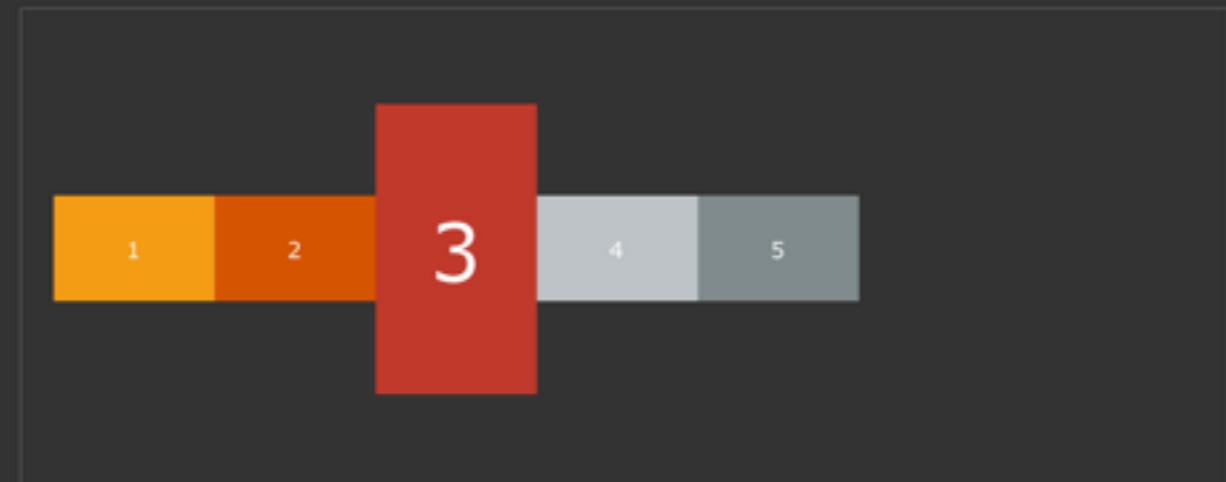
- flex-start
- flex-end
- center
- baseline
- stretch



Properties for the flex items

ALIGN-SELF (property for flex items)

- auto
- flex-start
- flex-end
- center
- baseline
- stretch



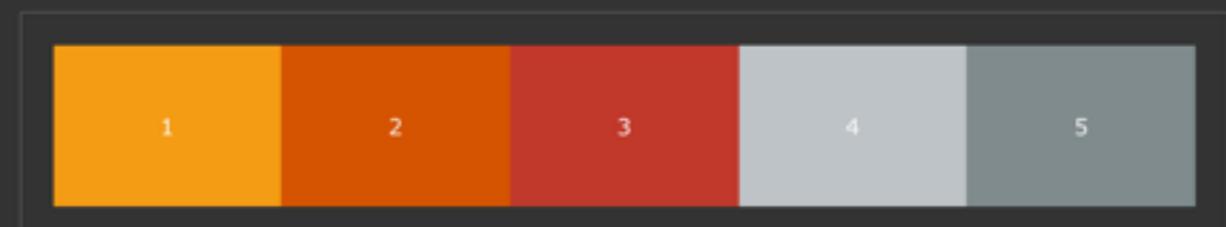
FLEX-GROW (property for flex items)

elemento 1: elemento 2: elemento 3: elemento 4: elemento 5:



FLEX-SHRINK (property for flex items)

elemento 1: elemento 2: elemento 3: elemento 4: elemento 5:



Solved by Flexbox

Cleaner, hack-free CSS

 View Project Source

 Spread the Word

Introduction

CSS has been lacking proper layout mechanisms for far too long. Transitions, animations, filters, all of these are great and useful additions to the language, but they don't address the major problems that Web developers have been complaining about for what seems like an eternity.

Finally, thanks to [Flexbox](#), we have a solution.

This site is not another CSS framework. Instead, its purpose is to showcase problems once hard or impossible to solve with CSS alone, now made trivially easy with Flexbox. And with the recent release of Internet Explorer 11 and Safari 6.1, the latest Flexbox spec is now supported in every modern browser.

Check out the demos below. View the styles in the Web inspector or dive into [the source](#) to see just how easy CSS layout will become once Flexbox becomes mainstream.

Showcase



This screenshot shows the GitHub repository page for `philipwalton / flexbugs`. The repository has 85 commits, 2 branches, 0 releases, 8 contributors, and follows the MIT license.

A community-curated list of flexbox issues and cross-browser workarounds for them.

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

Philip Walton committed on GitHub Merge pull request #171 from mattbrundage/patch-1 ... Latest commit 8889a84 on 26 Aug

LICENSE Initial commit 2 years ago

README.md Update README.md 2 months ago

README.md

Flexbugs

This repository is a community-curated list of flexbox issues and cross-browser workarounds for them. The goal is that if you're building a website using flexbox and something isn't working as you'd expect, you can find the solution here.

As the spec continues to evolve and vendors nail down their implementations, this repo will be updated with newly discovered issues and remove old issues as they're fixed or become obsolete. If you discover a bug that's not listed here, please [report it](#) so everyone else can benefit.

The bugs and their workarounds

1. Minimum content sizing of flex items not honored
2. Column flex items set to `align-items:center` overflow their container
3. `min-height` on a flex container won't apply to its flex items

This repository Search Pull requests Issues Gist + ToDo

Watch 149 Star 2,875 Fork 183

Code Issues 24 Pull requests 2 Boards Burndown Projects 0

A JavaScript polyfill for Flexbox <https://jonathantneal.github.io/flexibility/>

49 commits 3 branches 9 releases 5 contributors MIT

Branch: master New pull request Create new file Upload files Find file Clone or download

jonathantneal 2.0.1	Latest commit 9578df3 on 30 Jun
lib	Support border width in relation to border style
test	Update tests
.editorconfig	Update configurations and tests
.gitignore	2.0.0 RC
.travis.yml	2.0.0 RC
CHANGELOG.md	2.0.1
CONTRIBUTING.md	2.0.1
LICENSE.md	0.2.2
README.md	2.0.0 RC
SUPPORT.md	2.0.1
bower.json	2.0.0 RC
flexibility.js	2.0.1
package.json	2.0.1
README.md	

Flexbox Grid

A grid system based on the `flex` display property.

[Download](#)[Github](#)

Responsive

Responsive modifiers enable specifying different column sizes, offsets, alignment and distribution at xs, sm, md & lg viewport widths.



```
<div class="row">
  <div class="col-xs-12
    col-sm-8
    col-md-6
    col-lg-4">
    <div class="box">Responsive</div>
  </div>
</div>
```

[Layout Grid](#)

[Column Alignments](#)

[Column Centering](#)

[Column Ordering](#)

[Column Spacing](#)

[Max Width](#)

[Visibility](#)

[Spacing](#)

[Text Alignment](#)

[Helpers](#)

[Editor Support](#)

[Browser Support](#)

[Performance](#)

[Why Grids?](#)

Core Flex Grid

Core Flex Grid is a modern responsive mobile first CSS grid & layout library built on top of Flexbox.

Core Flex Grid is lightweight weighing at only 2kb minified and gzipped.

[Download v1.0.1](#)

[GitHub](#)

[Tweet](#)

Auto fit columns

Columns automatically evenly fill space in each flex row. The grid is a 12 column grid.

6 columns

6 columns

```
<div class="flex-row">
  <div class="col">
    <div class="demo">6 columns</div>
  </div>
  <div class="col">
    <div class="demo">6 columns</div>
  </div>
</div>
```

Responsive Columns

Responsive column classes are available. Column breakpoints are adjustable using the Sass variables or altering the CSS output. All columns are nestable. We denote classes with responsive suffixes. Core Flex Grid follows BEM like