Layouting

The UI Composition

‹hi/coders›

Agenda

- Box model
- Layouting
  - Different layouting techniques
- Questions

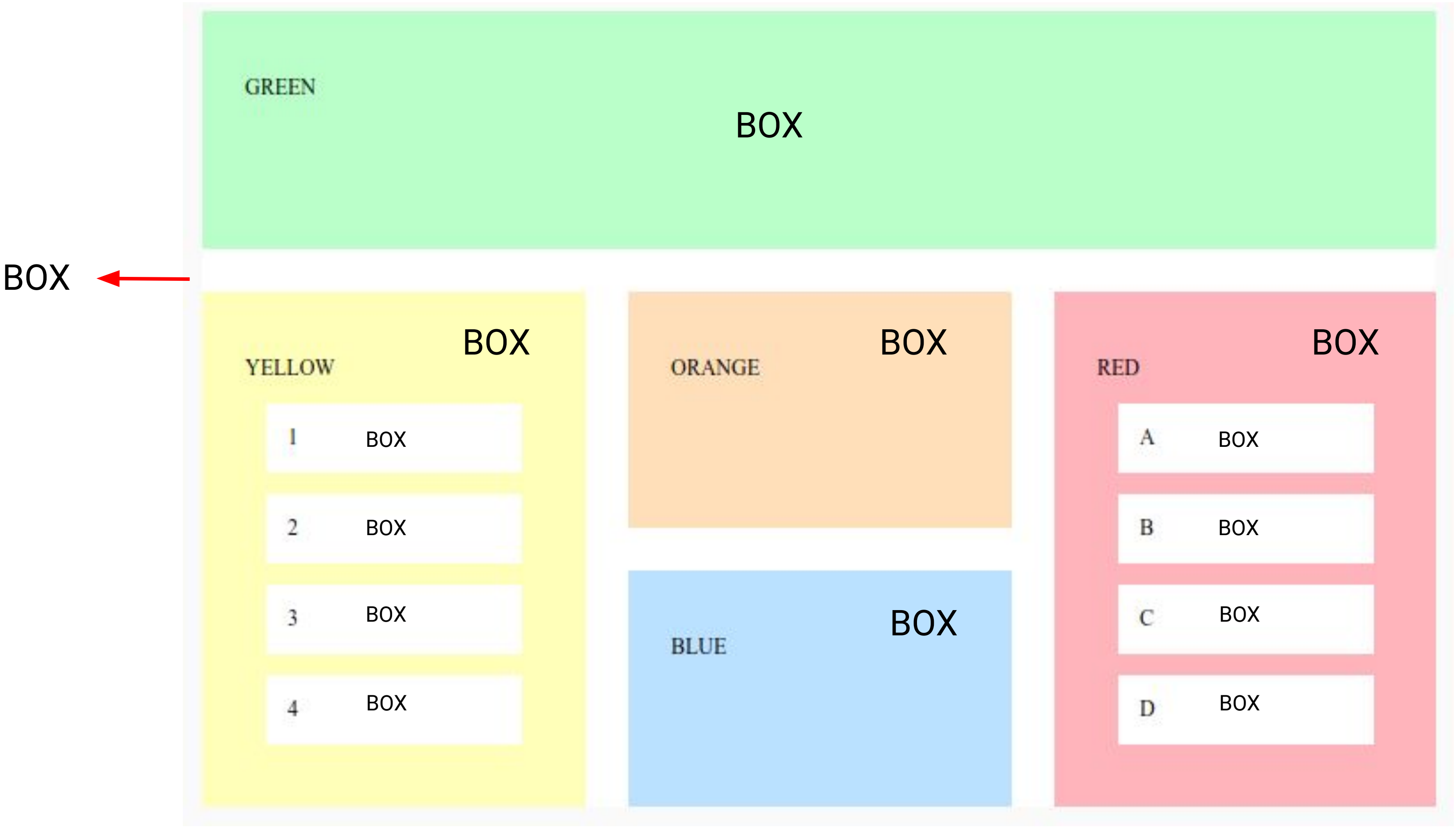
CSS

Learning Objectives

- What is box model?
- What kind of box model calculations are out there?
- What is a layout?
- CSS support for layouting.
- How to create one?

Box Model

Box Model


GREEN
BOX
BOX
YELLOW
BOX
1
BOX
2
BOX
3
BOX
4
BOX
ORANGE
BOX
BLUE
BOX
RED
BOX
A
BOX
B
BOX
C
BOX
D
BOX

Source: 9Minecraft
MeinKraft Resource Pack 1.17.1/1.16.5 - 9Minecraft.Net

Box Model

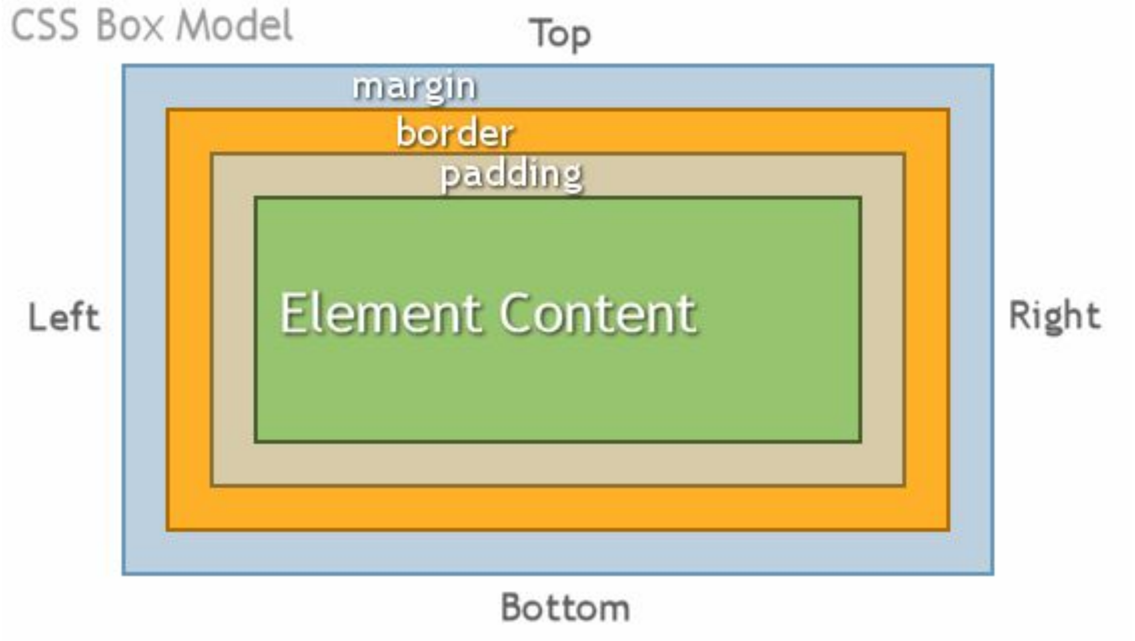
How to calculate a box?

Content Area	Padding Area	Border Area	Margin Area
width	padding-top	border-width	margin-top
min-width	padding-right	border	margin-right
max-width	padding-bottom		margin-bottom
height	padding-left		margin-left
min-height	padding		margin
max-height			

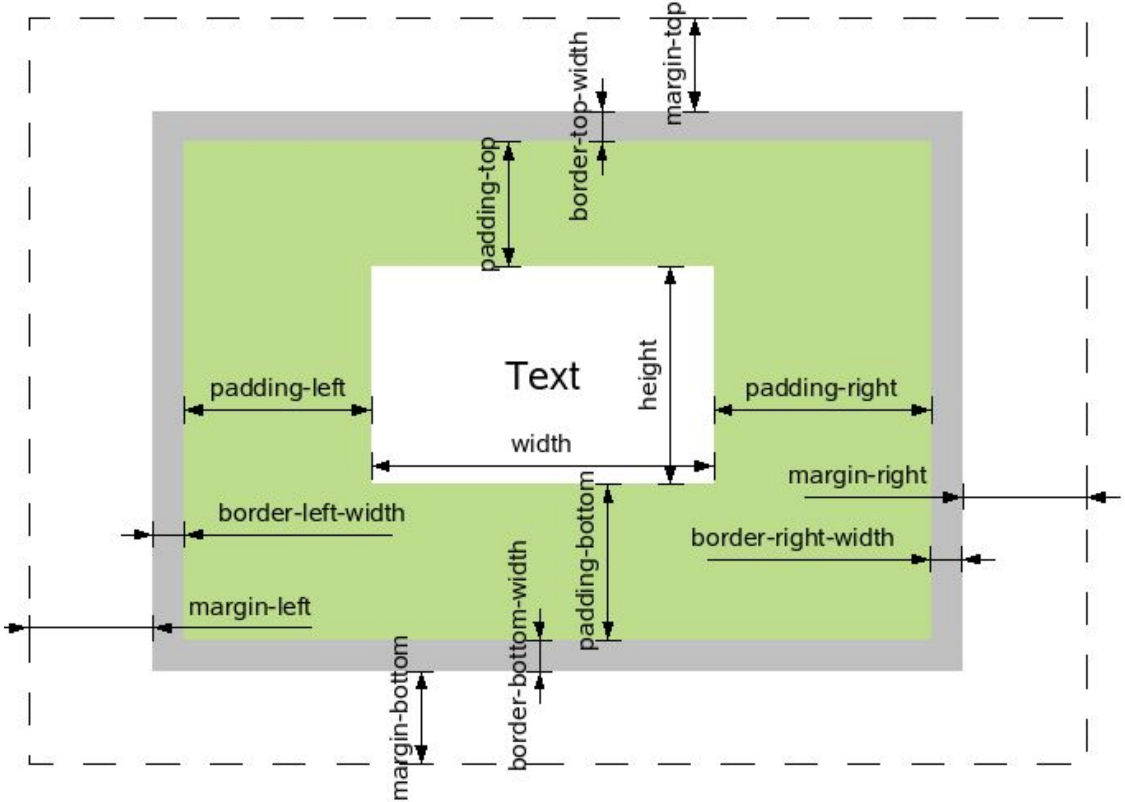
- How a box is represented?
- Inclusiveness of padding, margin, content, border
- Two different calculation method (box-sizing)
  - content-box (default), only content height and width
  - Border-box, includes content, border, padding

CSS

- padding: 5px 4px 3px 1px
- padding-left: 10px
- padding: top right bottom left
- padding: 10px 10px;
- padding: top-bottom left-right
- margin: 10px;
- border: <border-width> || <border-style> || <color>
- border-radius
- outline: ...



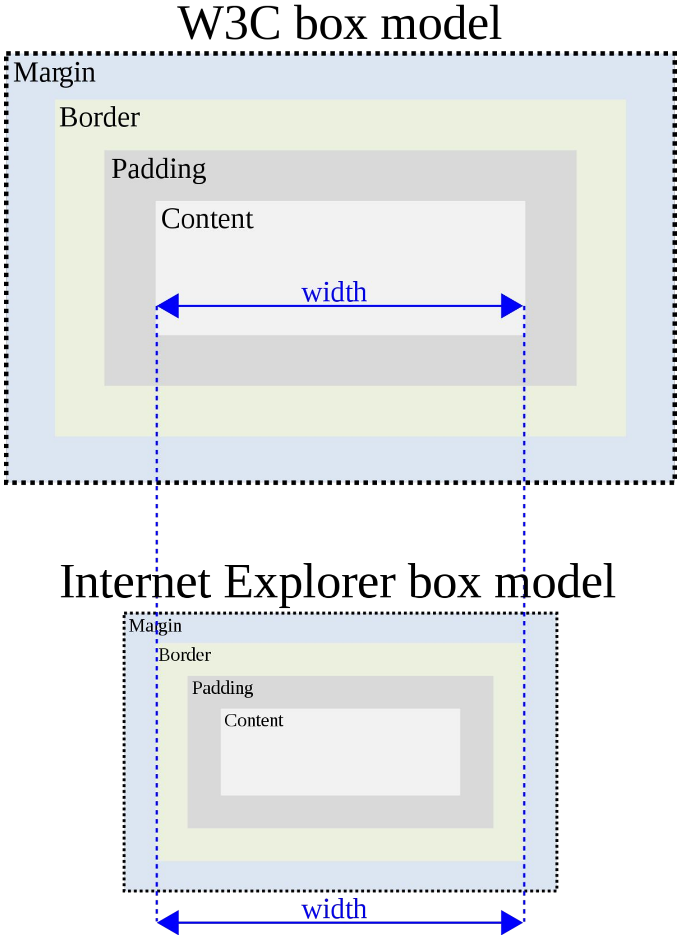
What includes a box?


margin-top
border-top-width
padding-top
Text
height
padding-left
padding-right
width
margin-right
border-left-width
border-right-width
padding-bottom
margin-left
border-bottom-width
margin-bottom

box-sizing:
content-box|border-box|initial|inherit;

content-box=> Default. The width and height properties includes only the content. Border and padding are not included.

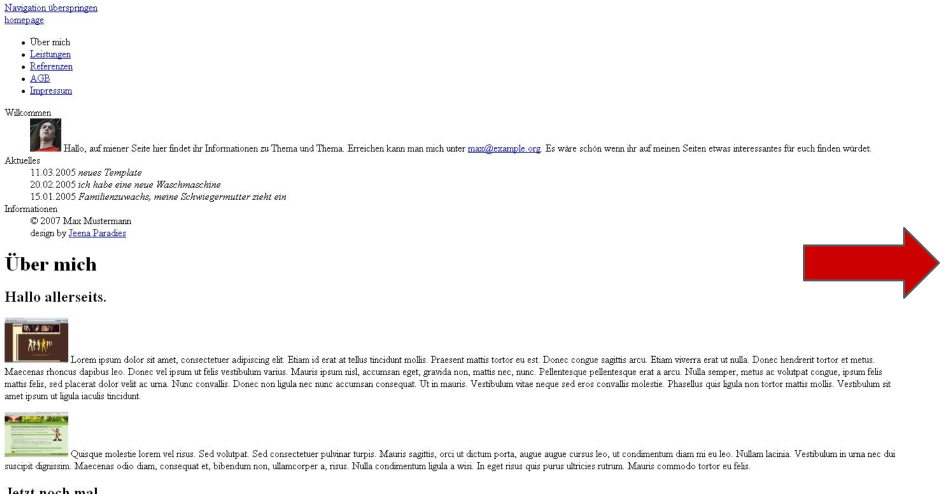
border-box=>The width and height properties includes content, padding and border.



Difference between box models

Layout

Apply a layout



Navigation überspringen
homepage

- Über mich
- Leistungen
- Referenzen
- AGB
- Impressum

Wilkommen

Hallo, auf miener Seite hier findet ihr Informationen zu Thema und Thema. Erreichen kann man mich unter max@example.org. Es wäre schön wenn ihr auf meinen Seiten etwas interessantes für euch finden würdet.

Aktuelles

11.03.2005 *neues Template*
20.02.2005 *ich habe eine neue Waschmaschine*
15.01.2005 *Familienzuwachs, meine Schwiegermutter zieht ein*

Informationen

© 2007 Max Mustermann
design by Jeena Paradies

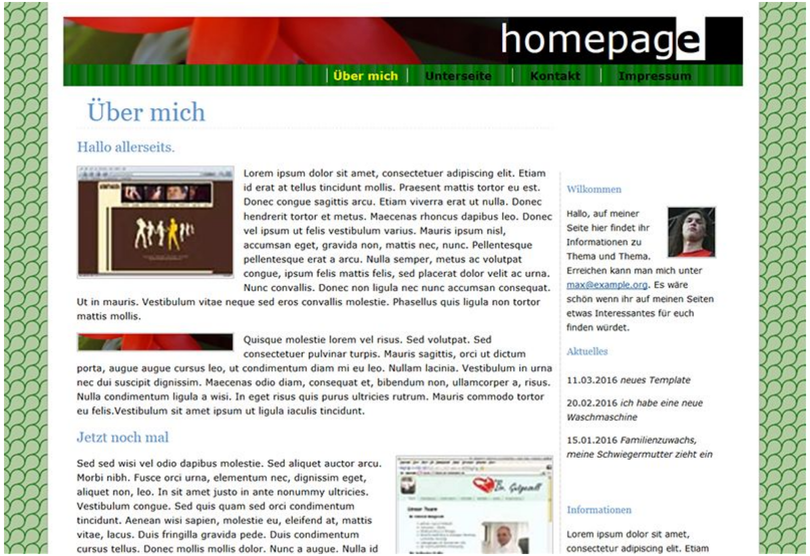
Über mich

Hallo allerseits.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam id erat at tellus tincidunt mollis. Praesent mattis tortor eu est. Donec congue sagittis arcu. Etiam viverra erat ut nulla. Donec hendrerit tortor et metus. Maecenas rhoncus dapibus leo. Donec vel ipsum ut felis vestibulum varius. Mauris ipsum nisl, accumsan eget, gravida non, mattis nec, nunc. Pellentesque pellentesque erat a arcu. Nulla semper, metus ac volutpat congue, ipsum felis mattis felis, sed placerat dolor velit ac urna. Nunc convallis. Donec non ligula nec nunc accumsan consequat. Ut in mauris. Vestibulum vitae neque sed eros convallis molestie. Phasellus quis ligula non tortor mattis mollis. Vestibulum sit amet ipsum ut ligula iaculis tincidunt.

Quisque molestie lorem vel risus. Sed volutpat. Sed consectetuer pulvinar turpis. Mauris sagittis, orci ut dictum porta, augue augue cursus leo, ut condimentum diam mi eu leo. Nullam lacinia. Vestibulum in urna nec dui suscipit dignissim. Maecenas odio diam, consequat et, bibendum non, ullamcorper a, risus. Nulla condimentum ligula a wisi. In eget risus quis purus ultricies rutrum. Mauris commodo tortor eu felis.

Jetzt noch mal



Über mich

Hallo allerseits.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam id erat at tellus tincidunt mollis. Praesent mattis tortor eu est. Donec congue sagittis arcu. Etiam viverra erat ut nulla. Donec hendrerit tortor et metus. Maecenas rhoncus dapibus leo. Donec vel ipsum ut felis vestibulum varius. Mauris ipsum nisl, accumsan eget, gravida non, mattis nec, nunc. Pellentesque pellentesque erat a arcu. Nulla semper, metus ac volutpat congue, ipsum felis mattis felis, sed placerat dolor velit ac urna. Nunc convallis. Donec non ligula nec nunc accumsan consequat. Ut in mauris. Vestibulum vitae neque sed eros convallis molestie. Phasellus quis ligula non tortor mattis mollis.

Quisque molestie lorem vel risus. Sed volutpat. Sed consectetuer pulvinar turpis. Mauris sagittis, orci ut dictum porta, augue augue cursus leo, ut condimentum diam mi eu leo. Nullam lacinia. Vestibulum in urna nec dui suscipit dignissim. Maecenas odio diam, consequat et, bibendum non, ullamcorper a, risus. Nulla condimentum ligula a wisi. In eget risus quis purus ultricies rutrum. Mauris commodo tortor eu felis.Vestibulum sit amet ipsum ut ligula iaculis tincidunt.

Jetzt noch mal

Sed sed wisi vel odio dapibus molestie. Sed aliquet auctor arcu. Morbi nibh. Fusce orci urna, elementum nec, dignissim eget, aliquet non, leo. In sit amet justo in ante nonummy ultricies. Vestibulum congue. Sed quis quam sed orci condimentum tincidunt. Aenean wisi sapien, molestie eu, eleifend at, mattis vitae, lacus. Duis fringilla gravida pede. Duis condimentum cursus tellus. Donec mollis mollis dolor. Nunc a augue. Nulla id

Wilkommen

Hallo, auf meiner Seite hier findet ihr Informationen zu Thema und Thema. Erreichen kann man mich unter max@example.org. Es wäre schön wenn ihr auf meinen Seiten etwas Interessantes für euch finden würdet.

Aktuelles

11.03.2016 *neues Template*

20.02.2016 *ich habe eine neue Waschmaschine*

15.01.2016 *Familienzuwachs, meine Schwiegermutter zieht ein*

Informationen

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam

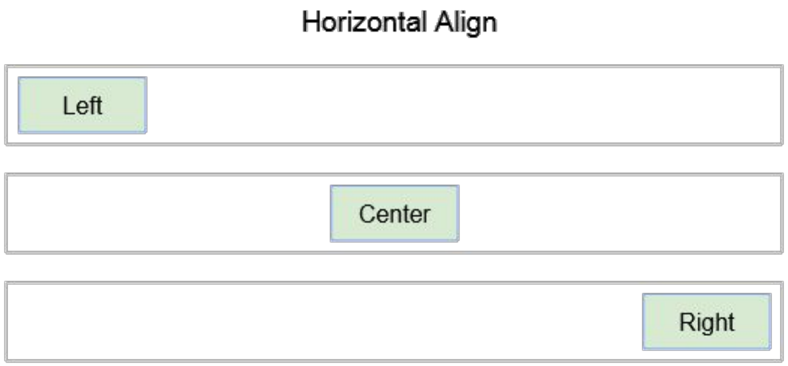
Types of Layouting

Which techniques you have!

- Centering
- Floating
- Positioning
- Using display properties
- Flex Layout
- CSS Grid
- Multi-column layout

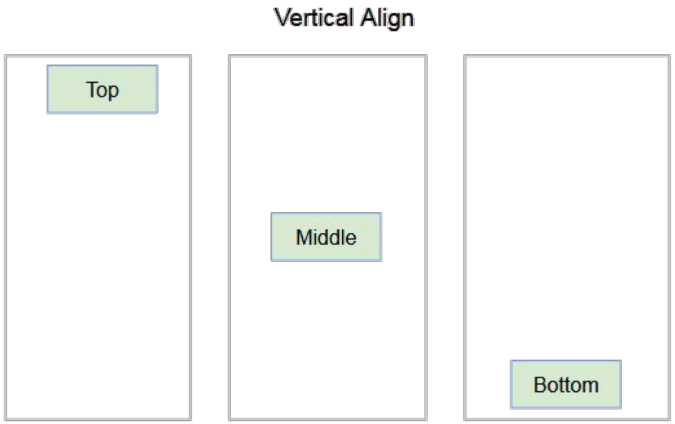
Horizontal centering

- Use auto margin
- Use text-align
- Use positioning
- Use flexbox



Vertical centering

- Use padding
- Use line-height
- Use positioning
- Use vertical-align:middle
- User transforms
- Use flexbox



Floating

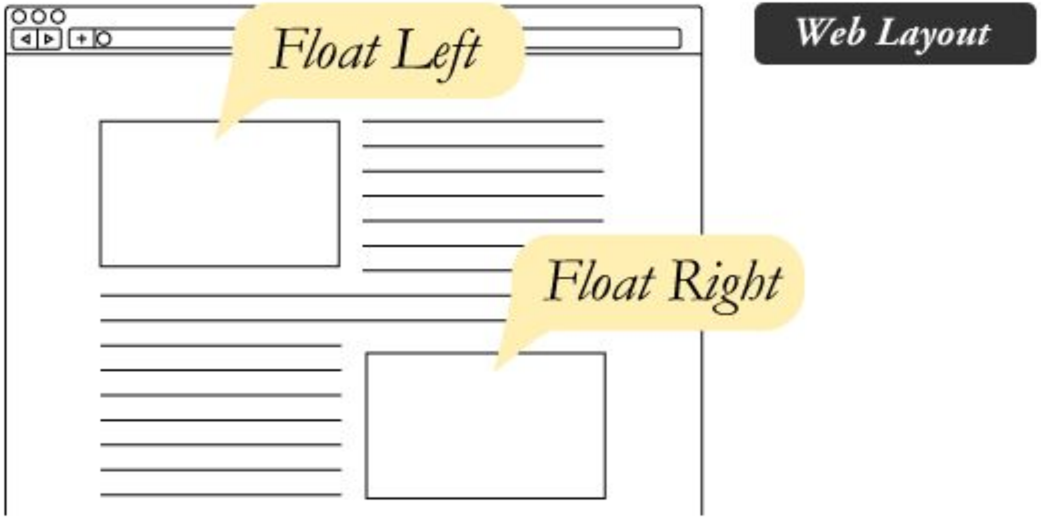
Aside from the simple example of wrapping text around images, floats can be used to create entire web layouts.

- CSS Tricks

- Default layout/flow: Positioning elements to top-left
- Float property enables changing the default behaviour
- Possible values: none, left, right
- Float can be neutralized/cleared.

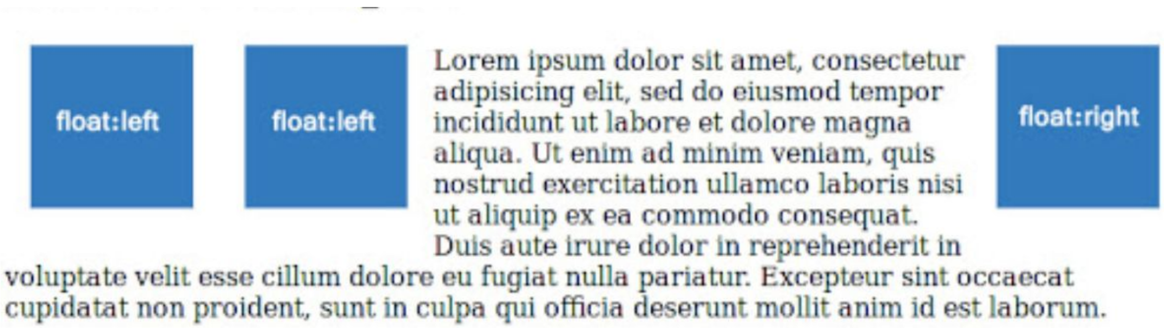
CSS

- float: left;
- float: right;
- clear: both;



float: right | left

Let's try it!



Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

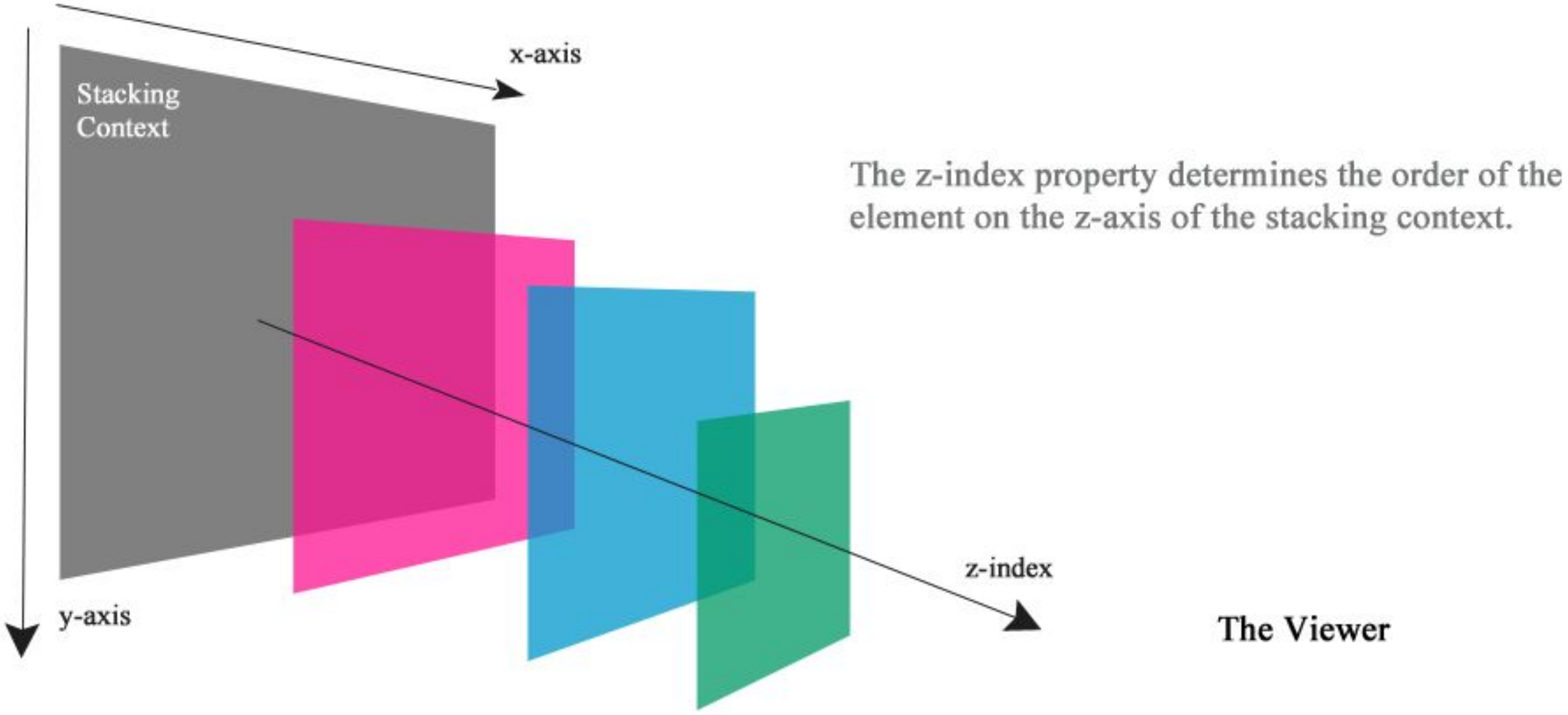
Positioning

Put the elements using a coordinate-system (x, y).

- Put elements out of order or flow
- Different types
  - relative,
  - static
  - absolute,
  - fixed,
  - sticky

CSS

- position: relative | …
- top, right, bottom, left, and z-index



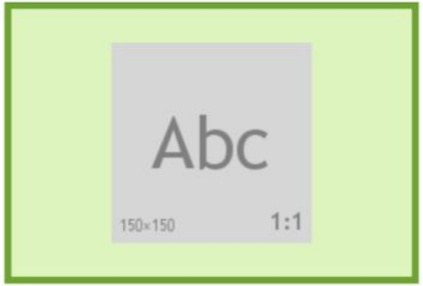
The z-index property determines the order of the element on the z-axis of the stacking context.

Dimensions

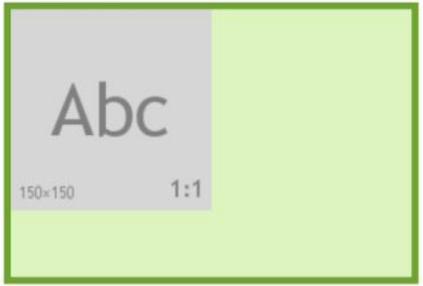
Original image

Abc
150×150
1:1

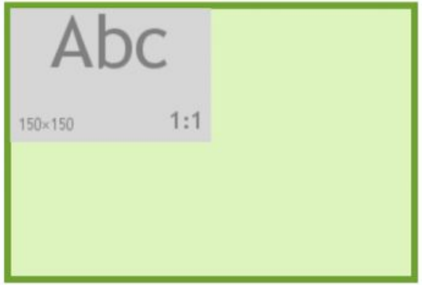
`object-position:50% 50%`



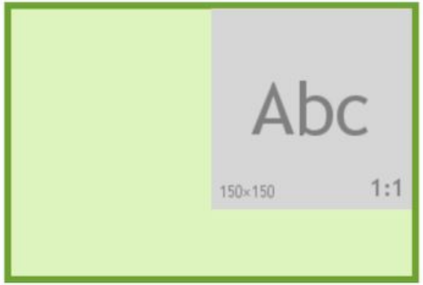
`object-position:0 0`



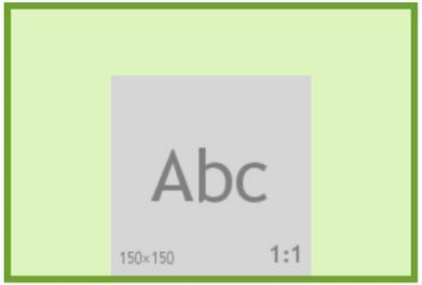
`object-position:0 -50px`

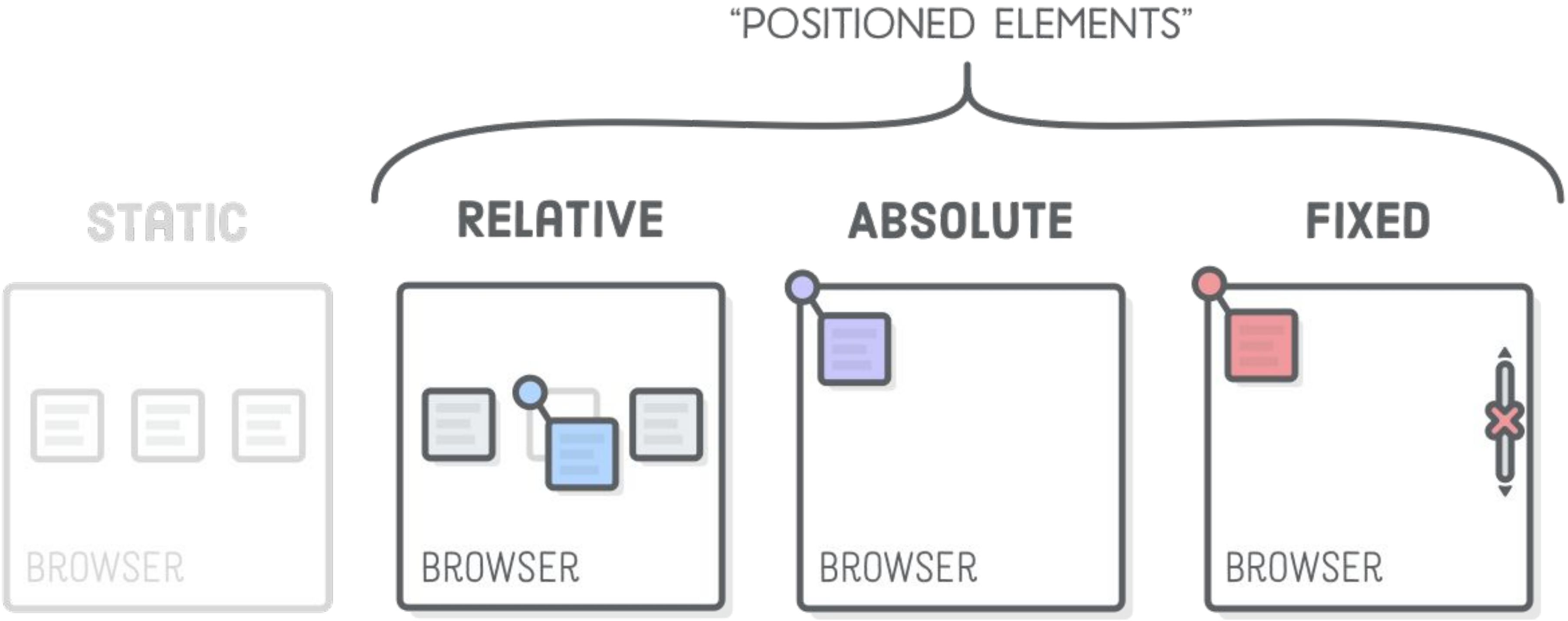


`object-position:top right`



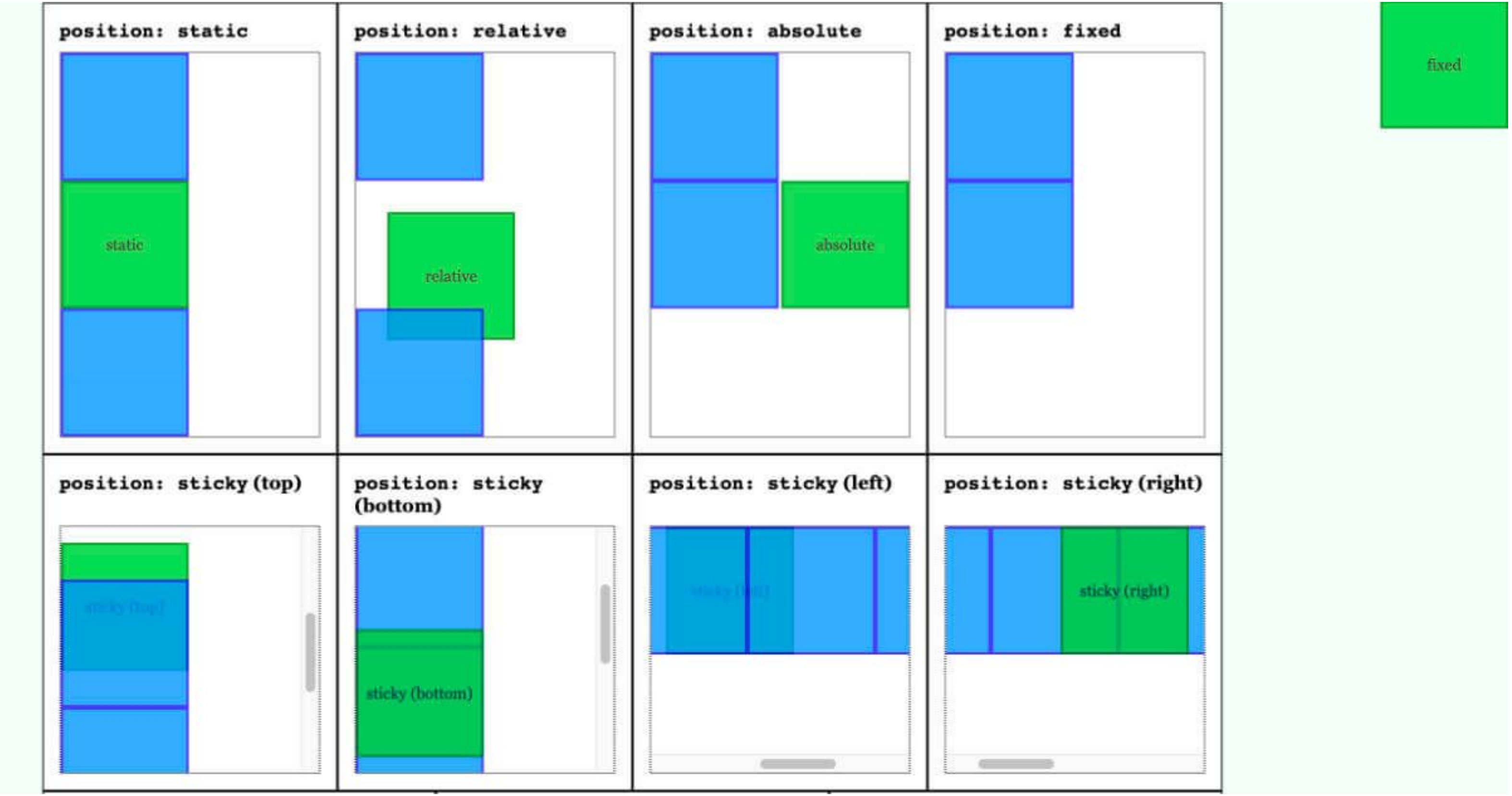
`object-position:bottom`





Types of Positioning

Source: https://internetingishard.com/html-and-css/advanced-positioning/



position: static

position: relative

position: absolute

position: fixed

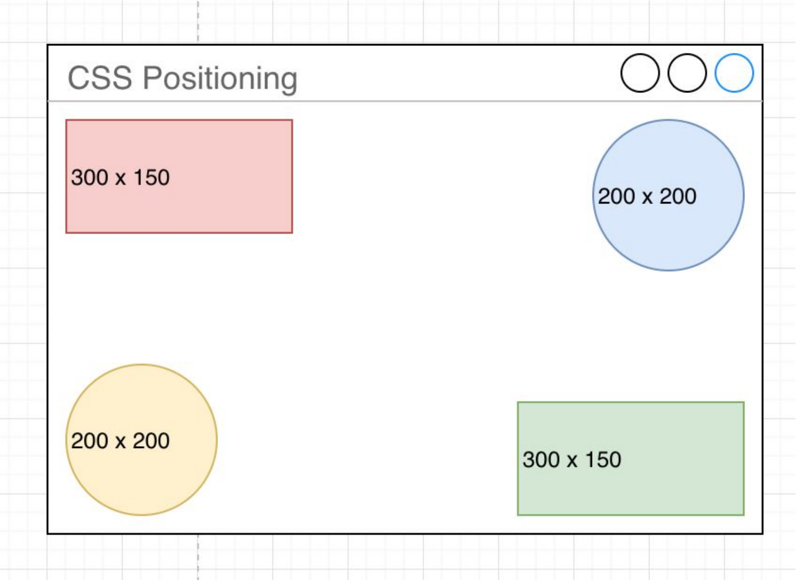
position: sticky (top)

position: sticky (bottom)

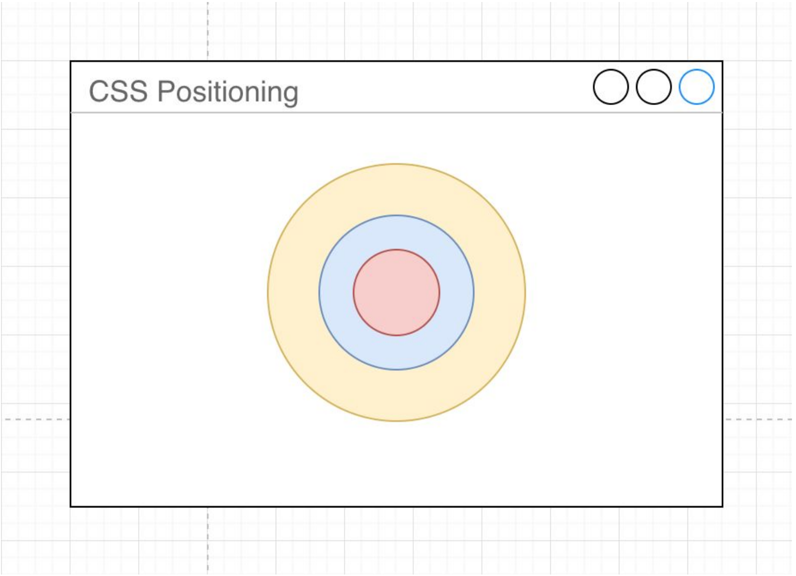
position: sticky (left)

position: sticky (right)

Let's try it!



Let's try it

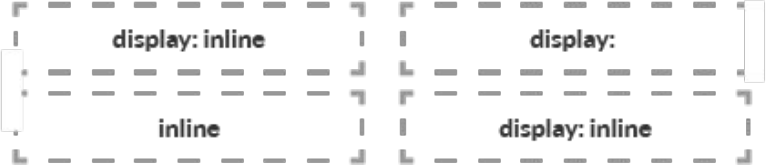


display

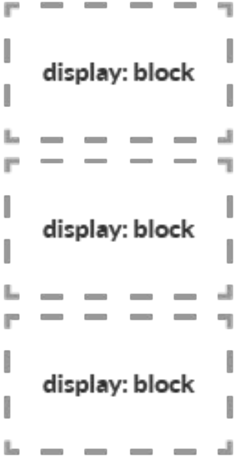
Display property

- Block, Inline, Inline-block
- Table, row, cell
- Flex
- Grid
- List

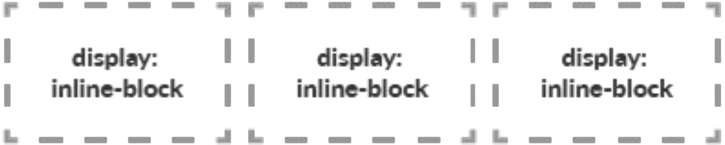
Display: inline



Display: block



Display: inline-block



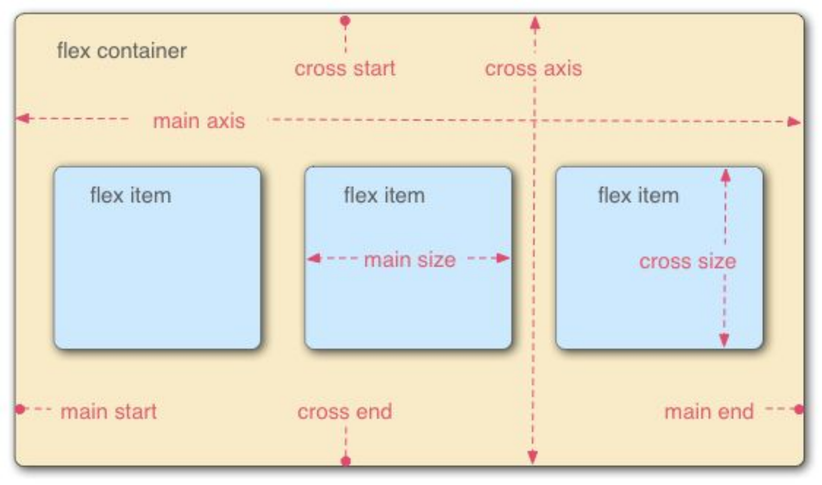
flexbox

Flexbox

- Flexbox is … ?
  - CSS Flexible Box Layout Module
- Css for Layout
- One-dimensional layouting
  - Either Column or Row, adjusting the space for elements in only one direction
- Two-dimensional layouting
  - Nesting of these boxes (horizontal inside vertical, or vertical inside horizontal) can be used to build layouts in two dimensions.
- Many settings possible
  - Compared to Box Model + Float/Clear Model

Layouting: FlexBox

- Container and Items
  - Or: parent element and child elements



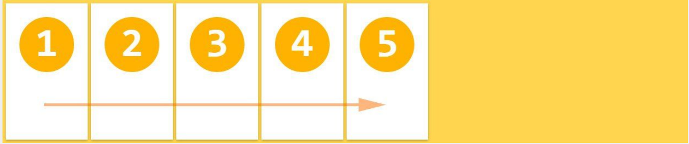
https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Flexbox

- With **display:flex**

Flexbox Container Properties

- **flex-direction**
  - With row direction the flex items are stacked in a row from left-to-right in ltr context

```
.flex-container {

  display: flex;

  flex-direction:  row;
}
```



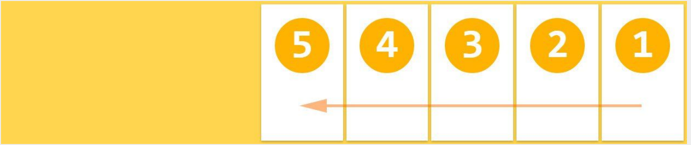
https://scotch.io/tutorials/a-visual-guide-to-css3-flexbox-properties

Default value: row

Flexbox Container Properties

- **flex-direction**
  - With row-reverse direction the flex items are stacked in a row from right-to-left in ltr context

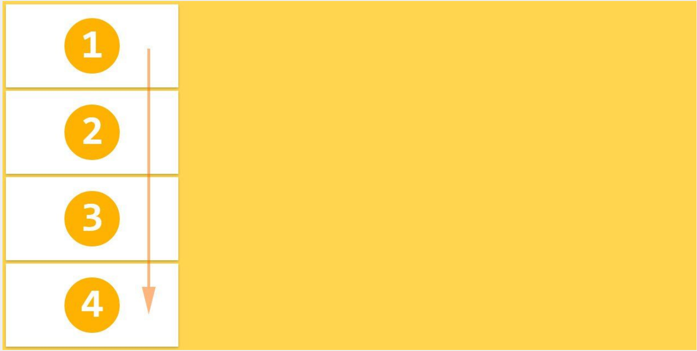
```
.flex-container {

  display: flex;

  flex-direction:  row-reverse;
}
```



Flexbox Container Properties

- **flex-direction**
  - With column direction the flex items are stacked in a column from top-to-bottom

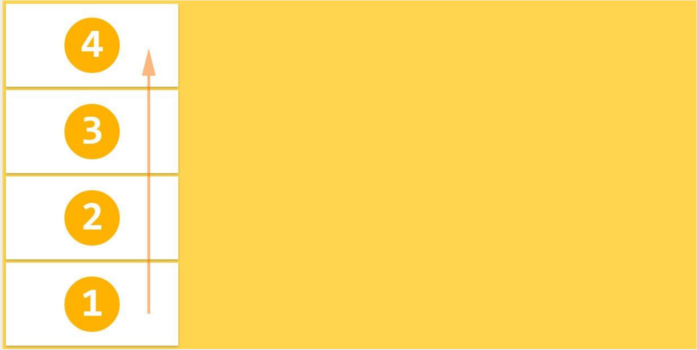
```
.flex-container {

  display: flex;

  flex-direction:  column;
}
```



Flexbox Container Properties

- **flex-direction**
  - With column-reverse direction the flex items are stacked in a column from bottom-to-top

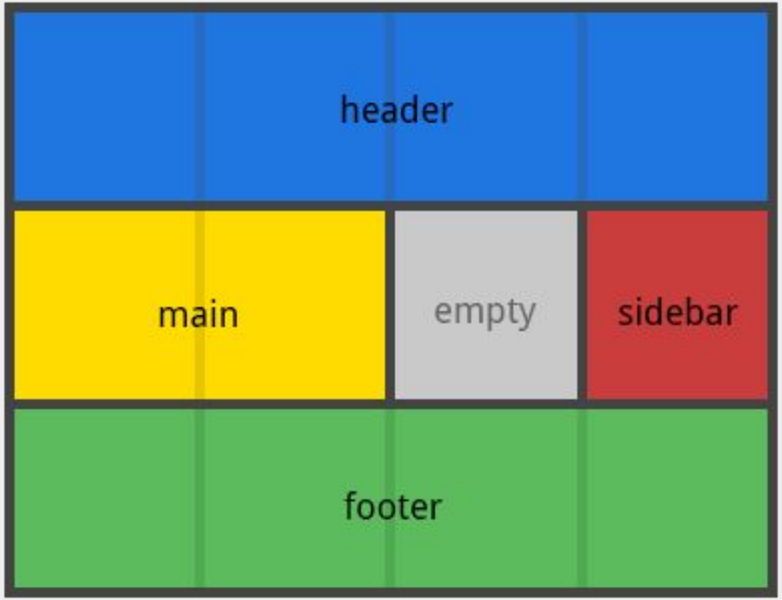
```
.flex-container {

  display: flex;

  flex-direction:
column-reverse;
}
```

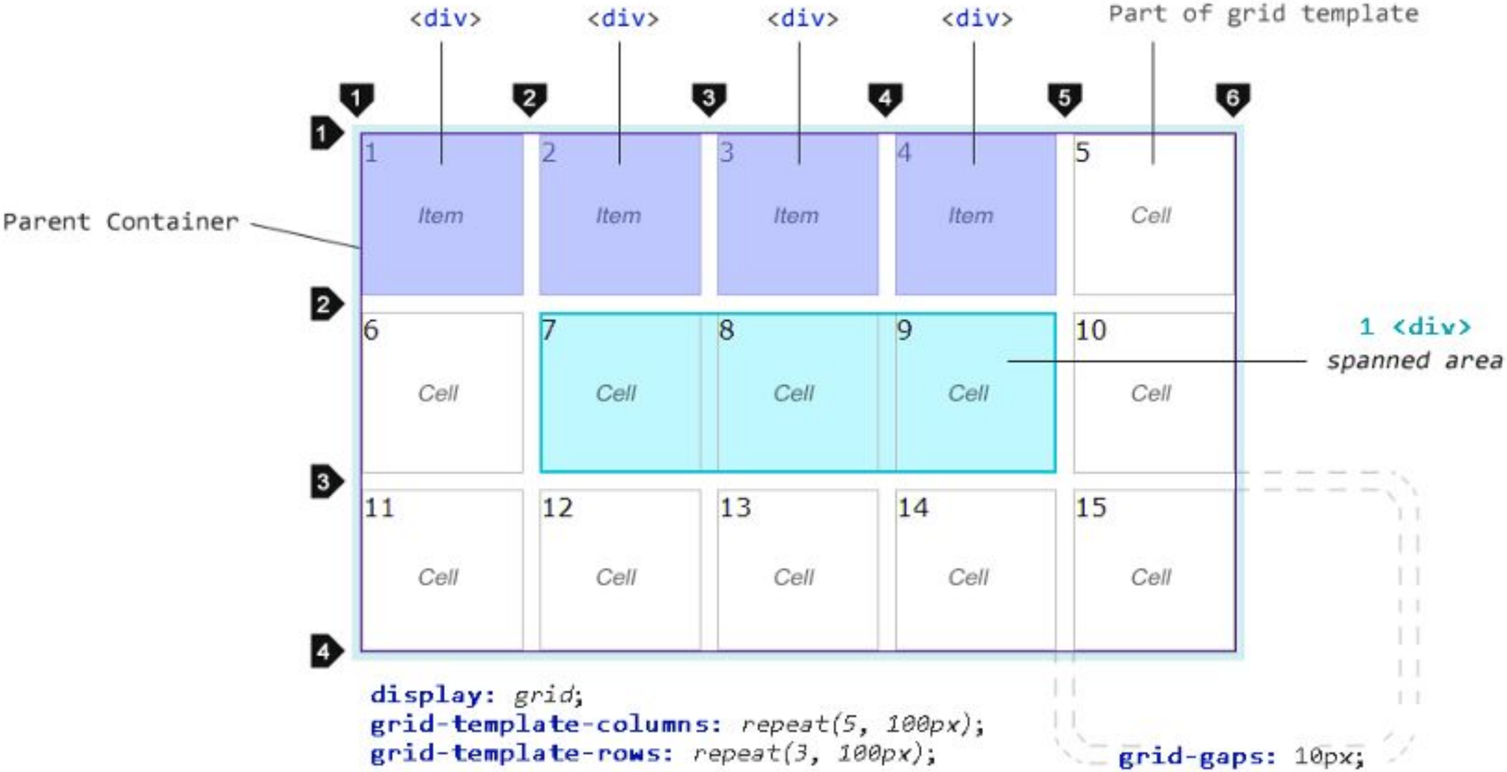


Grid

CSS Grid

Old friend “Table” for layouting, created for CSS

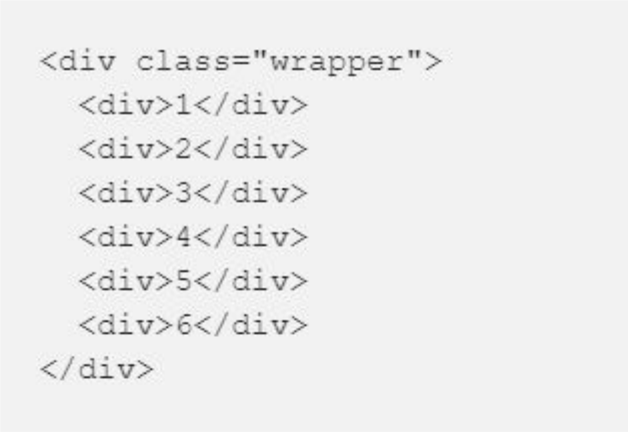




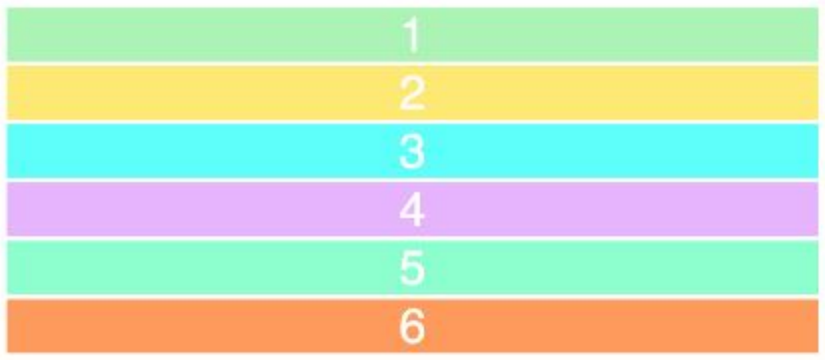
```
display: grid;
grid-template-columns: repeat(5, 100px);
grid-template-rows: repeat(3, 100px);
```

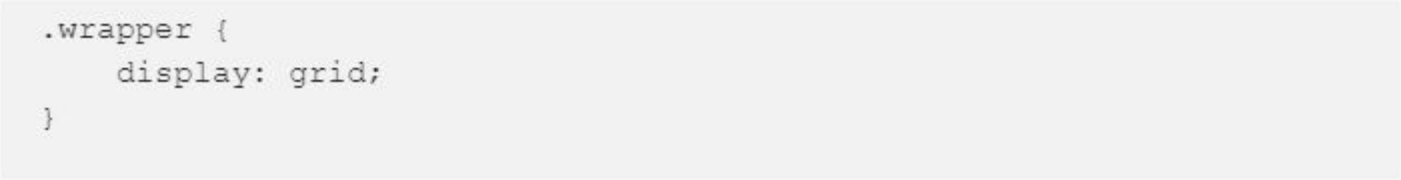
```
grid-gaps: 10px;
```

CSS Grid

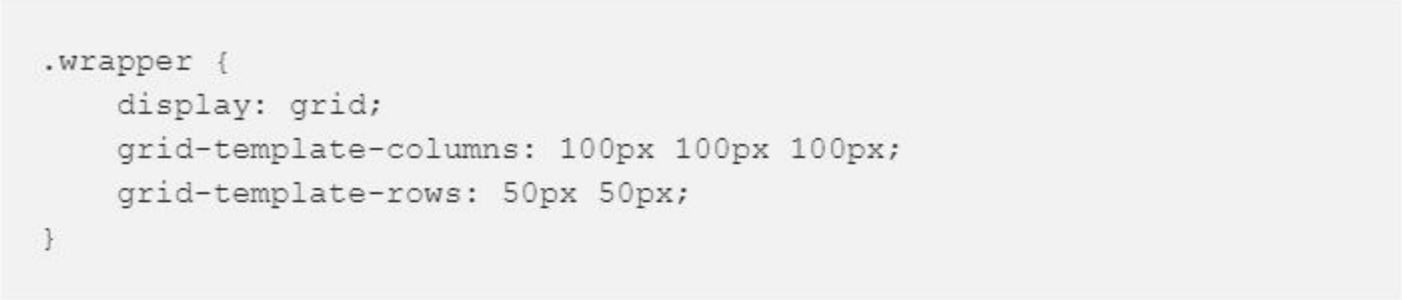


```
<div class="wrapper">
  <div>1</div>
  <div>2</div>
  <div>3</div>
  <div>4</div>
  <div>5</div>
  <div>6</div>
</div>
```

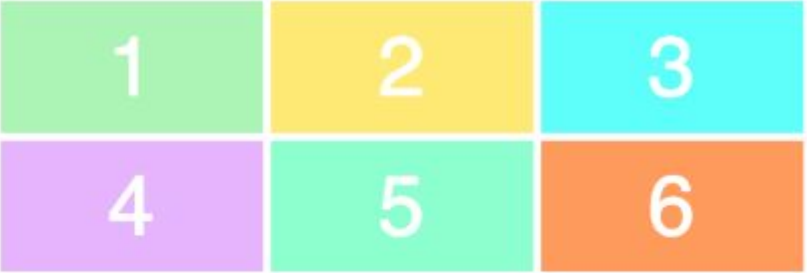


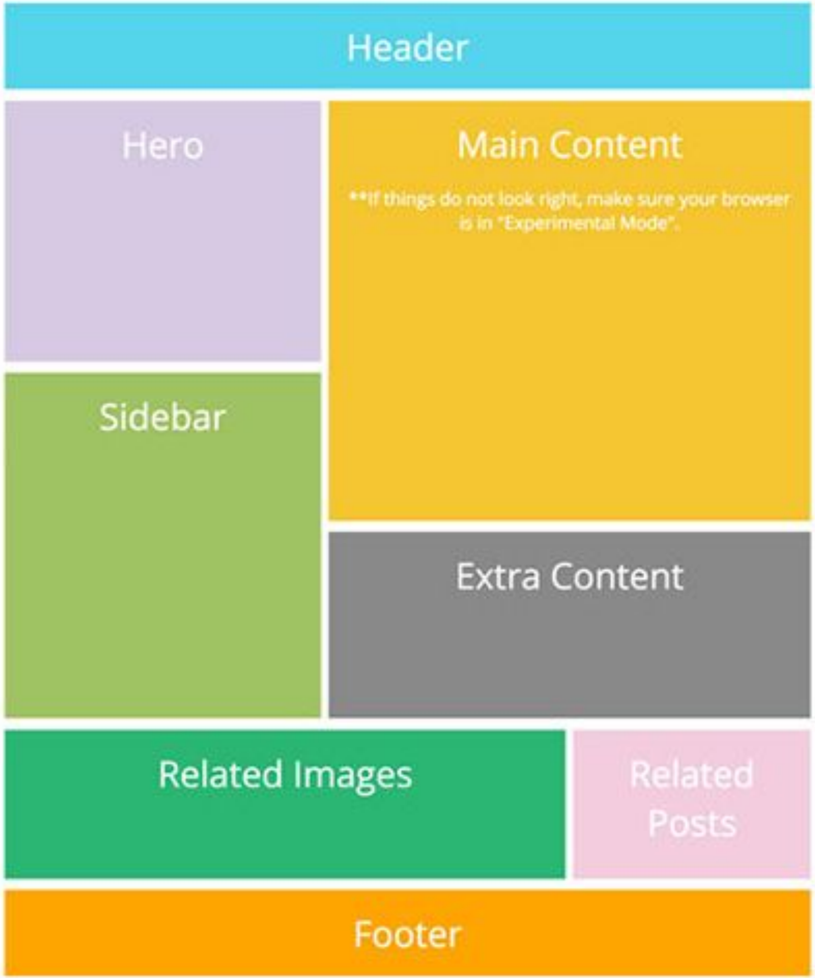


```
.wrapper {
    display: grid;
}
```



```
.wrapper {
    display: grid;
    grid-template-columns: 100px 100px 100px;
    grid-template-rows: 50px 50px;
}
```





Header

Hero

Main Content

**If things do not look right, make sure your browser is in "Experimental Mode".

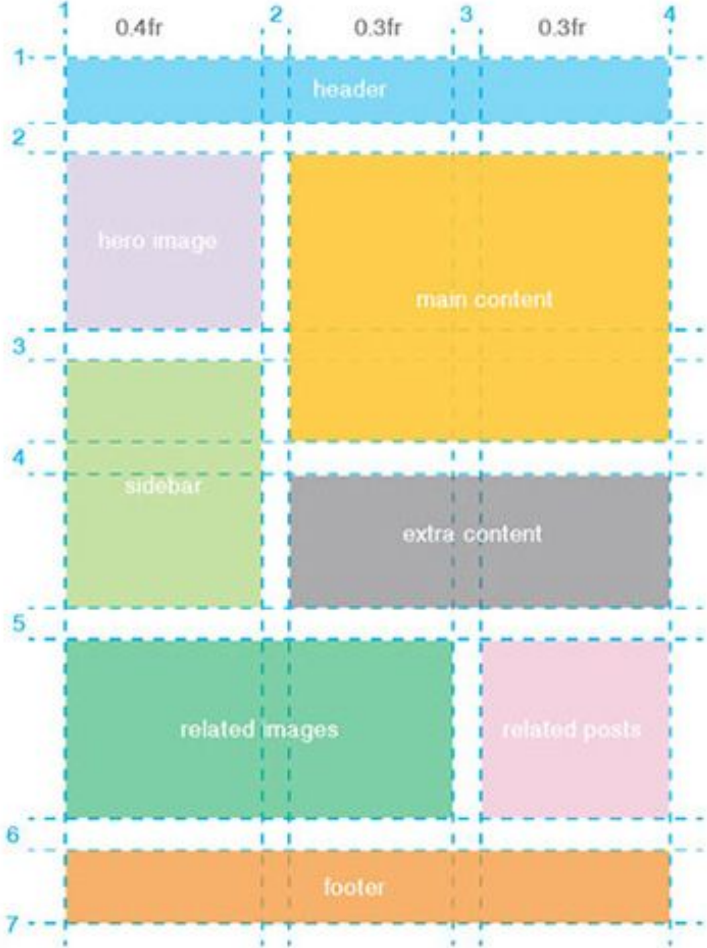
Sidebar

Extra Content

Related Images

Related Posts

Footer


1
0.4fr
2
0.3fr
3
0.3fr
4
1
2
3
4
5
6
7
header
hero image
main content
sidebar
extra content
related images
related posts
footer

How to Layout?

- First draw
- Slicing
- Use a template

Additional Resources

- https://www.w3schools.com/css/css_positioning.asp
- https://developer.mozilla.org/en-US/docs/Web/CSS/position
- https://www.w3schools.com/css/css_boxmodel.asp
- https://csslayout.io/

Questions?