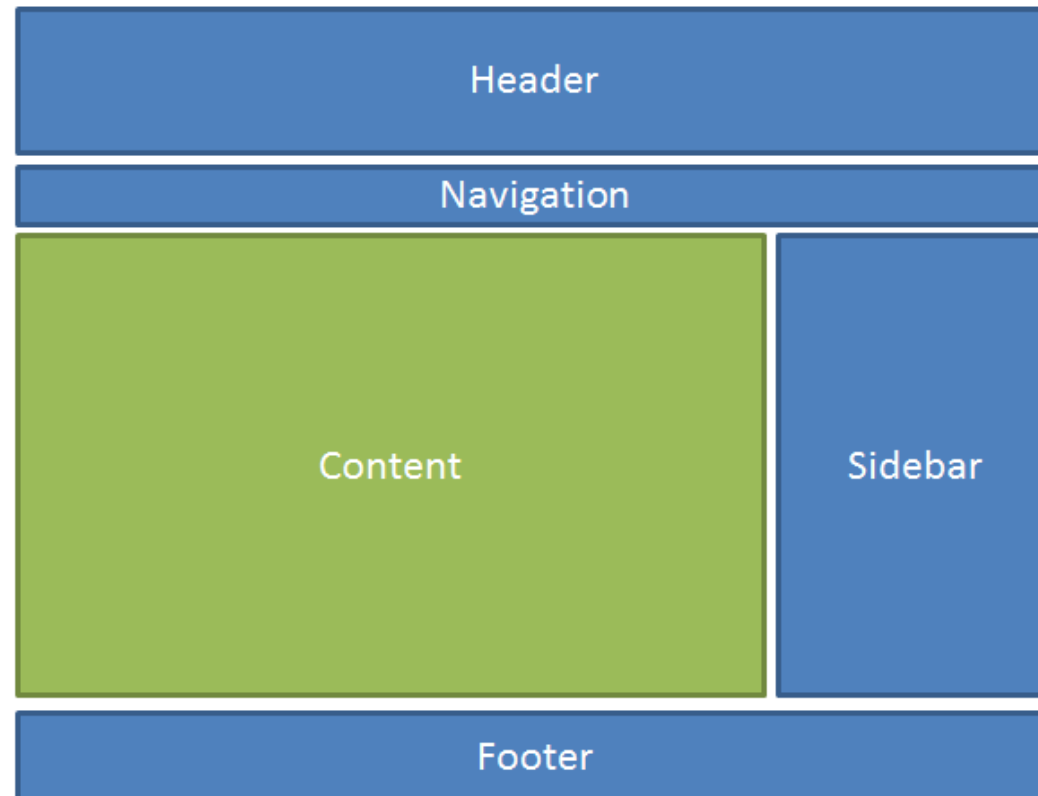


Columnar layout

A **columnar layout** is any layout where some of the content is positioned side-by-side



Columnar Layouts using CSS

- The World Wide Web: system of sharing “documents”
...remember: academic research papers!
 - No need for columnar layouts
 - Tables (for tabular data)
 - Floats (for pushing images left or right and having text wrap around)
- No concept of...
 - Banners
 - Navigation bars
 - Sidebars
 - Headers, footers, et cetera

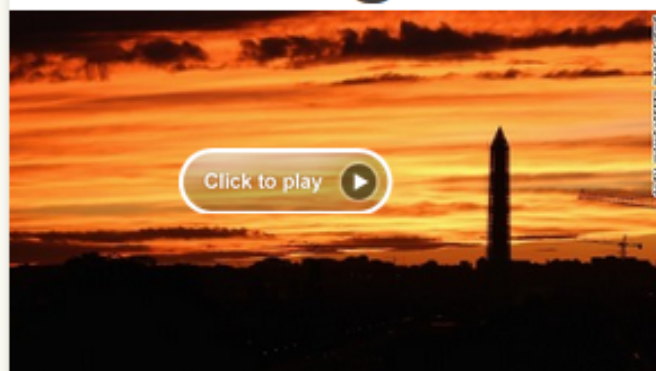


CNN TRENDS

Shutdown • House rule change • Cruise ship drowning • Banksy • 'Allah' • 'MasterChef' suicide • Madeleine McCann • CNN Heroes

DEVELOPING STORY

A 'bright' tomorrow?



Reid: 'Tremendous progress' made

Under a deal being negotiated, the government would be funded until January 15, and the debt ceiling would be increased until February 7, sources say. [FULL STORY](#)

- McConnell and Reid's bitter history
- Manchin: Here's a compromise ...
- Left behind? | What's open, closed
- **NEW** House GOP changed rules
- Poll: No winners | GOP backup plan

THE LATEST

- Family: Town attacked us after rape
- Boy drowns in cruise ship pool
- Witness: They had to pull mom away
- **NEW** Major quake hits Philippines
- Fresh lead in Madeleine McCann case
- Captured terror suspect brought to U.S.
- 'I still can't sign on to Obamacare'
- Railler: Obama, put Quran down
- Quran talk highlights GOP problem
- See crowded bridge fall into river
- Michael Douglas hid real diagnosis
- Diet powder has meth-like chemical?
- Olympian Louganis marries partner
- Live blog: Cards vs. Dodgers
- 1 Δ hshon defends Ranz Bentfaw

WOULD THE TEA PARTY LEAVE THE GOP?

Frum: Don't let the door hit you

Opinion: The tea party has contaminated the GOP brand, David Frum says. [FULL STORY](#)

More: Partisan trash talk in a PR war • Opinion: 3 lessons for tea party • Conservatives defiant • Group: Vets event 'hijacked'



READ THIS, WATCH THAT



CNN THE **EMPOWERED PATIENT** ELIZABETH COHEN
[GET TIPS & ADVICE](#)

SPONSORED BY **SonoSite FUJIFILM**

LIVE TV



Piers Morgan Live

3pm ET / 6pm PT

Unpredictable. Lively. Challenging. Fun. Home of The Piers Morgan Interview.

[Shows and Schedules](#)

Full Schedules: CNN TV - HLN TV

WEATHER



SET EDITION: U.S. | INTERNATIONAL | MEXICO | ARABIC
TV: CNN | CNN | CNN en Español | HLN

Sign up | Log in

SEARCH

POWERED BY Google

Home TV & Video CNN Trends U.S. World Politics Justice Entertainment Tech Health Living Travel Opinion iReport Money

Sports

updated 9:22 PM EDT, Mon October 14, 2013

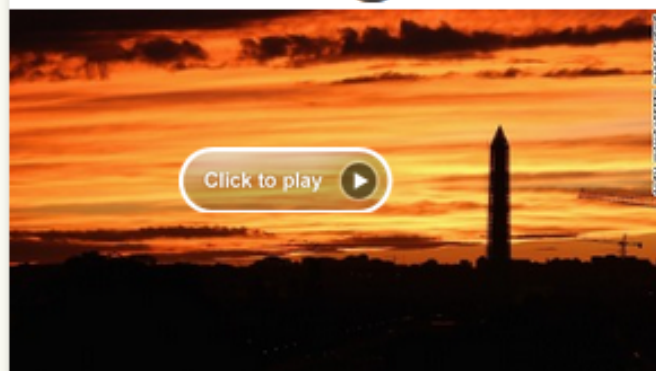
Make CNN Your homepage

CNN TRENDS

Shutdown • House rule change • Cruise ship drowning • Banksy • 'Allah' • 'MasterChef' suicide • Madeleine McCann • CNN Heroes

DEVELOPING STORY

A 'bright' tomorrow?



Reid: 'Tremendous progress' made

Under a deal being negotiated, the government would be funded until January 15, and the debt ceiling would be increased until February 7, sources say. [FULL STORY](#)

- McConnell and Reid's bitter history
- Manchin: Here's a compromise ...
- Left behind? | What's open, closed
- **NEW** House GOP changed rules
- Poll: No winners | GOP backup plan



SPONSORED BY **SonoSite**
FUJIFILM

RELATED CONTENT

LIVE TV



Piers Morgan Live

3pm ET / 6pm PT

Unpredictable. Lively. Challenging. Fun. Home of The Piers Morgan Interview.

Shows and Schedules

Full Schedules: CNN TV - HLN TV

WEATHER

THE LATEST

- Family: Town attacked us after rape
- Boy drowns in cruise ship pool
- Witness: They had to pull mom away
- **NEW** Major quake hits Philippines
- Fresh lead in Madeleine McCann case
- Captured terror suspect brought to U.S.
- 'I still can't sign on to Obamacare'
- Railler: Obama, put Quran down
- Quran talk highlights GOP problem
- See crowded bridge fall into river
- Michael Douglas hid real diagnosis
- Diet powder has meth-like chemical?
- Olympian Louganis marries partner
- Live blog: Cards vs. Dodgers
- 1 Δ hshon defends Ranz Bentley

WOULD THE TEA PARTY LEAVE THE GOP?

Frum: Don't let the door hit you

Opinion: The tea party has contaminated the GOP brand, David Frum says. [FULL STORY](#)

More: Partisan trash talk in a PR war • Opinion: 3 lessons for tea party • Conservatives defiant • Group: Vets event 'hijacked'



READ THIS, WATCH THAT



#header

#main-menu

#trends-feed

#main-story

#primary-sponsor

#live-tv

#secondary-stories

#weather

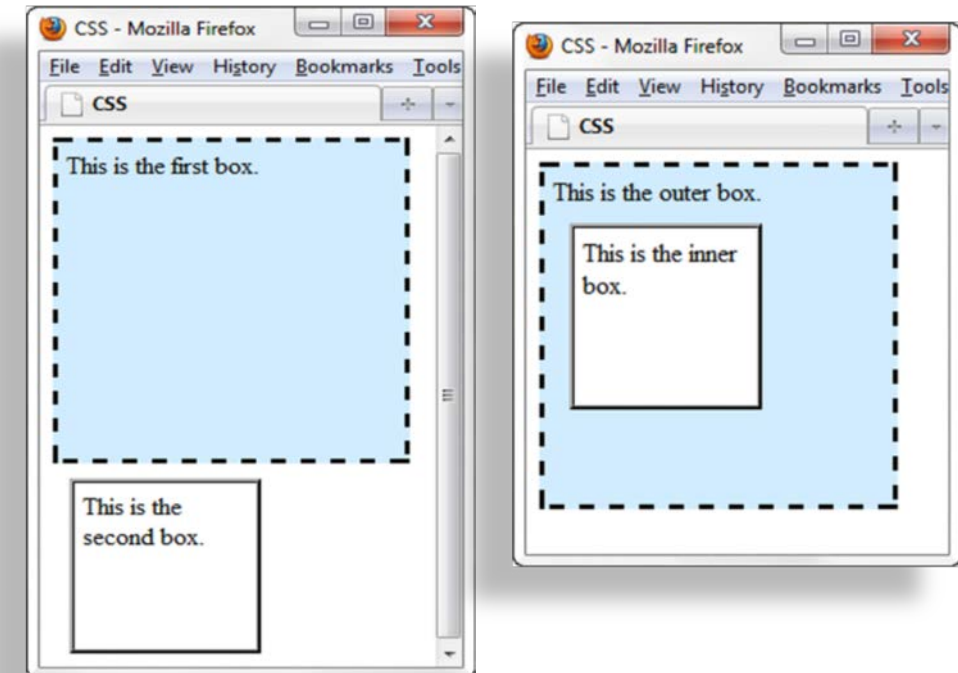
What is CSS Layout?

CSS Layout is: Altering the position of block elements in an HTML document from the natural document flow.

Natural Document Flow (*...is a layout*), then alter it with...

1. Boxes within Boxes
2. Boxes next to Boxes

CSS Layout is all about putting content in boxes, and boxes inside other boxes, and lining-up boxes side-by-side



Natural (Normal) Document Flow

This layout is simply the "Natural Document Flow" which means there is *no* columnar layout here.

(Well, technically: it's *one* column)

All blocks just line-up, one after another, top to bottom.

Layouts

Float
Inline-block
Table-cell
Flex

Lorem ipsum

This layout is simply the "Natural Document Flow" which means there is *no* "columnar layout here. All blocks just line-up, one after another, top to bottom.

Sed ut perspiciatis

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?



box 1: et iusto odio dignissimos ducimus

box 2: et iusto

box 3: odio dignissimos ducimus

box 4: dignissimos ducimus

box 5: et iusto odio dignissimos ducimus

An unordered list

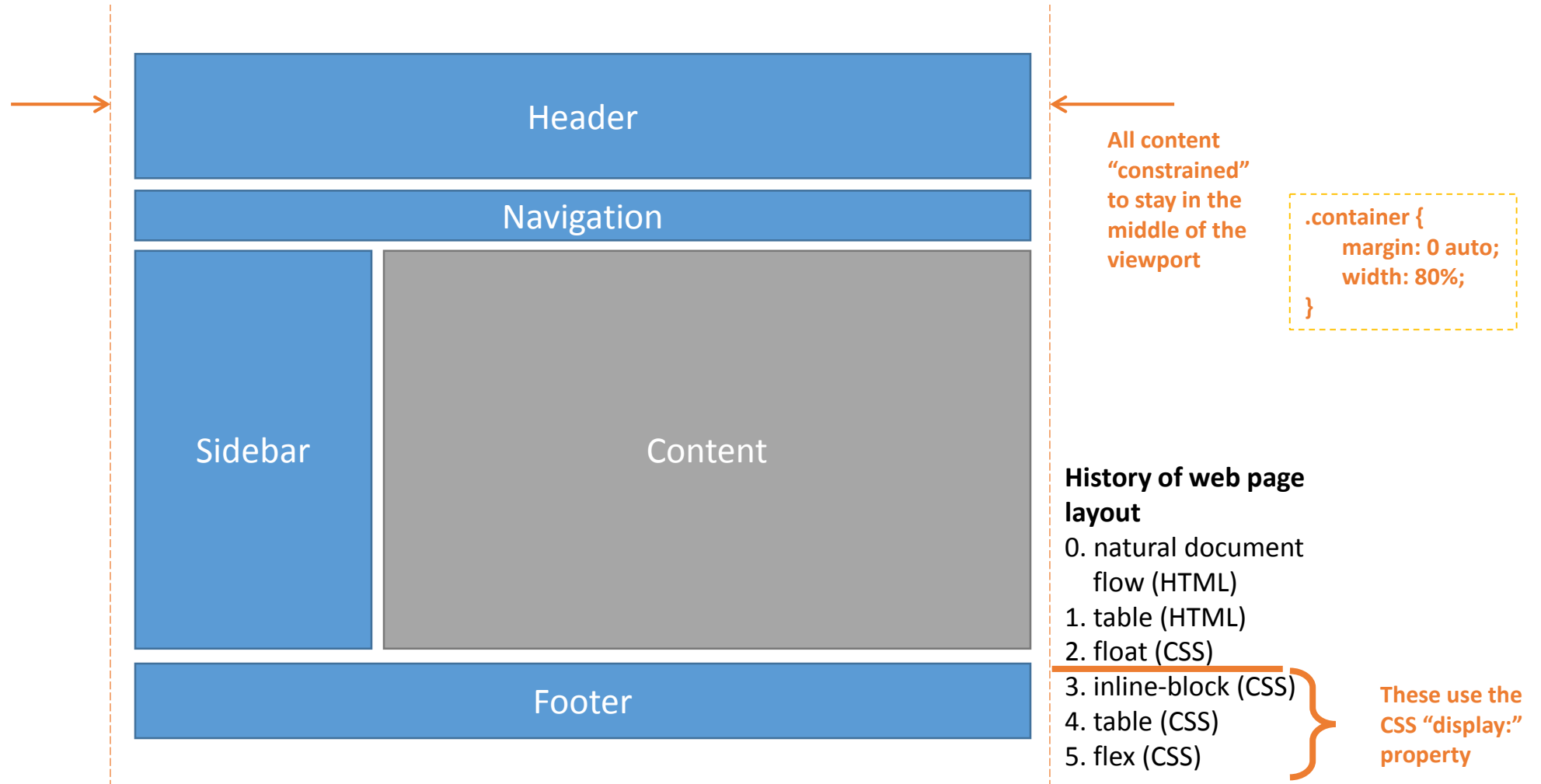
At vero eos et accusamus et iusto odio dignissimos ducimus.

- Apples
- Oranges
- Strawberries

Similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio.

A hyperlink: [Professor Kostin](#).

The Classic “C-clamp”



CSS property: display

- CSS property: **display**
 - Values:
 - block
 - inline
 - inline-block
 - table-cell
 - flex
- } very important in doing layout!

Example...

HTML:

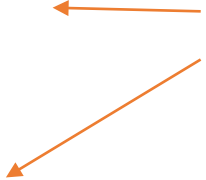
```
<article class="story">  
  Lorem ipsum...  
</article>
```

```
<aside class="story">  
  Sed imedent...  
</aside>
```

CSS:

```
.story { display: table-cell; }
```

The CSS display property

- display: **block**;
 - **Stacks-up, top-over-bottom with other elements**
 - Uses the “box model” (padding, margin, borders)
 - Is 100% wide (takes-up all the width on a line)
 - display: **inline**;
 - **Lines-up, side-by-side with other elements**
 - Ignores or does strange things to padding, margin, borders (no box model)
 - Is only as wide as it needs to be; if wider than 100%, wraps to the next line
- Defaults for all HTML elements that display content*
- 
- display: **inline-block**;
 - Uses the “box model” (like display: block) AND...
 - Is only as wide as it needs to be (like display: inline)
 - display: **table-cell**;
 - Neighboring block elements line-up side-by-side with equal height
 - Ignores some aspects of the “box model”
 - display: **flex**;
 - Neighboring block elements line-up side-by-side with equal height
 - Uses the “box model”
- Other options*

Step 1: Create Layout Blocks

...

```
<div class="container">
```

```
  <header>header</header>
```

```
  <nav>navigation</nav>
```

```
  <section>
```

```
    <article>main</article>
```

```
    <aside>sidebar</aside>
```

```
  </section>
```

```
  <footer>footer</footer>
```

```
</div><!--.container-->
```

...

Step 2: Choose a Layout Technique

At some point in your web page, you must break the normal document flow and position content side-by-side to create a columnar layout.

You can use any of the following methods (except a table):

- ~~Table layout~~ (not acceptable)
- ~~Float layout~~ (old fashioned but still widely used)

- Inline-block layout
- CSS table layout
- Flex layout



Current and best practices in the industry

Float layout

old fashioned but still widely used

Float Layout

The two large boxes (green and gray) are positioned side-by-side in a "columnar layout" by using the *float* property in CSS.

Because Float wasn't designed to do this kind of thing, it's kinda wonky. In addition to floating both boxes left, you have to apply a fix to keep the following blocks (in this case, the footer) from "falling" up.

Float Layout

Lorem ipsum

The two large boxes below (green and gray) are positioned side-by-side in a "columnar layout" by using the *float* property in CSS. Because Float wasn't designed to do this kind of thing, it's kinda wonky. In addition to floating both boxes left, you have to apply a fix to keep the following blocks (in this case, the footer) from "falling" up.

Float
Inline-block
Table-cell
Flex

Sed ut perspiciatis

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?



An unordered list

At vero eos et accusamus et iusto odio dignissimos ducimus.

- Apples
- Oranges
- Strawberries

Similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio.

The boxes below all have the float property. You can see the effect. Notice how the boxes "fall out" of this containing box. It would take one of the clearfix hacks to fix that.

box 1: et iusto odio dignissimos ducimus

box 2: et iusto

box 3: odio dignissimos ducimus

box 4: dignissimos ducimus

box 5: et iusto odio dignissimos ducimus

A hyperlink: [Professor Kostin](#).

Float Layout with “clear” (1)

HTML

```
<header>Lorem ipsum</header>

<article>
    Lorem ipsum
</article>

<aside>
    Lorem ipsum
</aside>

<footer class="clear">Lorem ipsum</div>
```

CSS

```
article, aside { float: left; }
article { width: 70%; }
aside { width: 30%; }
.clear { clear: left; }
```


Float Layout with “clear” (2)

HTML

```
<header>Lorem ipsum</header>

<section>

    <article>
        Lorem ipsum
    </article>

    <aside>
        Lorem ipsum
    </aside>

</section>

<footer>Lorem ipsum</div>
```

CSS

```
article, aside { float: left; }
article { width: 70%; }
aside { width: 30%; }
section { overflow: auto; }
```

Float Layout

If you use a float layout, you probably have to use a clear float trick so the content that comes after it doesn't do something weird.

```
/* this declaration "fix" float problems that occur BEFORE this  
element, if any */
```

```
.clear { clear: both; }
```

-- or --

```
/* Wrap content (including floated elements) in a DIV and apply  
this class to the DIV */
```

```
.clearfix { overflow: auto; }
```

Inline-block layout

Uses the “box model” (like display: block) AND...

Is only as wide as it needs to be (like display: inline)

Inline-block Layout

The two large boxes (green and gray) are positioned side-by-side in a "columnar layout" by using the `display:inline-block` property in CSS.

It works nicely, but you have to "comment out" the spaces between the HTML elements to keep a gap between the boxes from forming.

Also, notice how the heights of the two boxes are not equal. (That may or may not be important depending on your design.)

Also, you have to remember to set the CSS property `vertical-align` to "top" because the W3C thought "bottom" was better. (Whatever.)

Inline-block Layout

Lorem ipsum

The two large boxes below (green and gray) are positioned side-by-side in a "columnar layout" by using the `display:inline-block` property in CSS. It works nicely, but you have to "comment out" the spaces between the HTML elements to keep a gap between the boxes from forming. Also, notice how the heights of the two boxes are not equal. (That may or may not be important depending on your design.) Also, you have to remember to set the CSS property `vertical-align` to "top" because the W3C thought "bottom" was better. (Whatever.)

Float
Inline-block
Table-cell
Flex

Sed ut perspiciatis

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?



An unordered list

At vero eos et accusamus et iusto odio dignissimos ducimus.

- Apples
- Oranges
- Strawberries

Similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio.

The boxes below all have the display property of "inline-block". You can see the effect. Notice how the boxes have an extra space between them. It would take the "comment-out trick" to fix that.

box 1: et iusto odio dignissimos ducimus

box 2: et iusto

box 3: odio dignissimos ducimus

box 4: dignissimos ducimus

box 5: et iusto odio dignissimos ducimus

Inline Block Layout

HTML

```
<header>Lorem ipsum</header>

<article>
  Lorem ipsum
</article><!--

--><aside>
  Lorem ipsum
</aside>

<footer>Lorem ipsum</footer>
```

CSS

```
article, aside { display: inline-block; }
article { width: 70%; } //a width (any) is required
aside { width: 30%; }
```

Inline-block Layout

If you use an inline-block layout and you're counting your pixels exactly, you have to account for border widths and HTML spaces between the content for the total width to add-up right. If necessary, you have to “comment-out” space between HTML elements:

```
<article>Lorem ipsum</article><!--  
  
--><aside>Lorem ipsum</aside>
```

CSS Table layout

Neighboring block elements line-up side-by-side with equal height
Ignores some aspects of the “box model”

Table-cell Layout

The two large boxes (green and gray) are positioned side-by-side in a "columnar layout" by using the `display:table-cell` property in CSS.

It works nicely. You don't even have to set the widths (but you can).

One of the very few draw backs to using this technique is that CSS table-cell doesn't recognize the margin property, which could be a problem, based on what you're trying to accomplish with your design.

Table-cell Layout

Lorem ipsum

The two large boxes below (green and gray) are positioned side-by-side in a "columnar layout" by using the `display:table-cell` property in CSS. It works nicely. You don't even have to set the widths (but you can). One of the very few draw backs to using this technique is that CSS table-cell doesn't recognize the margin property, which could be a problem, based on what you're trying to accomplish with your design.

Float
Inline-block
Table-cell
Flex

Sed ut perspiciatis

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?



An unordered list

At vero eos et accusamus et iusto odio dignissimos ducimus.

- Apples
- Oranges
- Strawberries

Similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio.

The boxes below all have the display property of "table-cell". You can see the effect. Nice!

box 1: et
iusto odio
dignissimos
ducimus

box
2: et
iusto

box 3: odio
dignissimos
ducimus

box 4:
dignissimos
ducimus

box 5: et
iusto odio
dignissimos
ducimus

A hyperlink: [Professor Kostin](#).

CSS Table Layout

HTML

```
<header>Lorem ipsum</header>
```

```
<article>
```

```
    Lorem ipsum
```

```
</article>
```

```
<aside>
```

```
    Lorem ipsum
```

```
</aside>
```

```
<footer>Lorem ipsum</footer>
```

CSS

```
article, aside { display: table-cell; }
```

```
article { width: 70%; } //optional
```

```
aside { width: 30%; } //optional
```

CSS Table Layout

If you use a CSS table layout, remember, some margin and other normal box-model attributes don't work (or don't work normally) on the blocks.

Remember: don't confuse CSS tables with HTML tables!

Flex layout

Neighboring block elements line-up side-by-side with equal height

Uses the “box model”

New –not recognized by older browsers

Flex Layout

The two large boxes (green and gray) are positioned side-by-side in a "columnar layout" by using the `display: flex` property in CSS.

You just have to wrap all the elements you want positioned side-by-side in a block element with the flex property and it will do the rest.

There are other properties you can use to control the behavior of the "flexed" elements depending on the layout you want.

Flex Layout

Lorem ipsum

The two large boxes below (green and gray) are positioned side-by-side in a "columnar layout" by using the `display: flex` property in CSS. You just have to wrap all the elements you want positioned side-by-side in a block element with the flex property and it will do the rest. There are other properties you can use to control the behavior of the "flexed" elements depending on the layout you want.

Float
Inline-block
Table-cell
Flex

Sed ut perspiciatis

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?



An unordered list

At vero eos et accusamus et iusto odio dignissimos ducimus.

- Apples
- Oranges
- Strawberries

Similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio.

The boxes below are contained in a block element that has the display property of "flex" which automatically positions all block elements inside it side-by-side.

box 1: et
iusto odio
dignissimos
ducimus

box 2: et
iusto

box 3: odio
dignissimos
ducimus

box 4:
dignissimos
ducimus

box 5: et
iusto odio
dignissimos
ducimus

A hyperlink: [Professor Kostin.](#)

Flex Layout

HTML

```
<header>Lorem ipsum</header>

<section>

    <article>
        Lorem ipsum
    </article>

    <aside>
        Lorem ipsum
    </aside>

</section>

<footer>Lorem ipsum</div>
```

CSS

```
section { display: flex; }
article { width: 70%; } //optional
aside { width: 30%; } //optional
```

CSS Flexible Box Layout Module

- Referred to as “flex” or “flexbox” ...it’s actually a CSS display property:

`display: flex;`

...and several associated CSS properties that work with it.

- Common Properties used with flexible box layout:

- `display` `flex`
- `flex-direction` `order`
- `flex-wrap`
- `justify-content`

- Great way to easily configure multi-column page layout
- Elements contained within a flex container can be configured either horizontally or vertically in a flexible manner with flexible sizing
- Flexbox is not yet well-supported by browsers. (...maybe next year?)
Check <http://caniuse.com/#search=flex> for the current level of browser support.

More information about Flex

A popular article...

css-tricks.com/snippets/css/a-guide-to-flexbox

The screenshot shows the top of the CSS-Tricks website. The header includes the 'CSS-TRICKS' logo, a search icon, and the 'treehouse' logo with the tagline 'Grow your CSS skills. Land your dream job.' Below the header is a navigation bar with links for 'BLOG', 'VIDEOS', 'ALMANAC', 'SNIPPETS', 'FORUMS', and 'THE LODGE'. The main content area is titled 'A Complete Guide to Flexbox' and is dated 'LAST UPDATED ON: SEPTEMBER 2, 2014'. The article is divided into two main sections: 'Background' and 'Basics & Terminology'. The 'Basics & Terminology' section is further divided into two columns: 'Properties for the Parent (flex container)' and 'Properties for the Children (flex items)'. The 'Properties for the Parent' column includes a diagram of a container with three items, a definition of 'display', a CSS code snippet for setting 'display: flex', and a note about CSS columns. The 'Properties for the Children' column includes a diagram of items within a container, a definition of 'order', a diagram showing items in different orders, a CSS code snippet for setting 'order', and a definition of 'flex-grow'.

CSS-TRICKS

treehouse™ Grow your CSS skills. Land your dream job.

BLOG VIDEOS ALMANAC SNIPPETS FORUMS THE LODGE

Code Snippets » CSS »

A Complete Guide to Flexbox

LAST UPDATED ON: SEPTEMBER 2, 2014

- ▶ Background
- ▶ Basics & Terminology

container

Properties for the Parent
(flex container)

display

This defines a flex container; inline or block depending on the given value. It enables a flex context for all its direct children.

```
css
.container {
  display: flex; /* or inline-flex */
}
```

Note that CSS columns have no effect on a flex container.

flex-direction

This establishes the main-axis, thus defining the direction flex items

items

Properties for the Children
(flex items)

order

By default, flex items are laid out in the source order. However, the `order` property controls the order in which they appear in the flex container.

```
css
.item {
  order: <integer>;
}
```

flex-grow