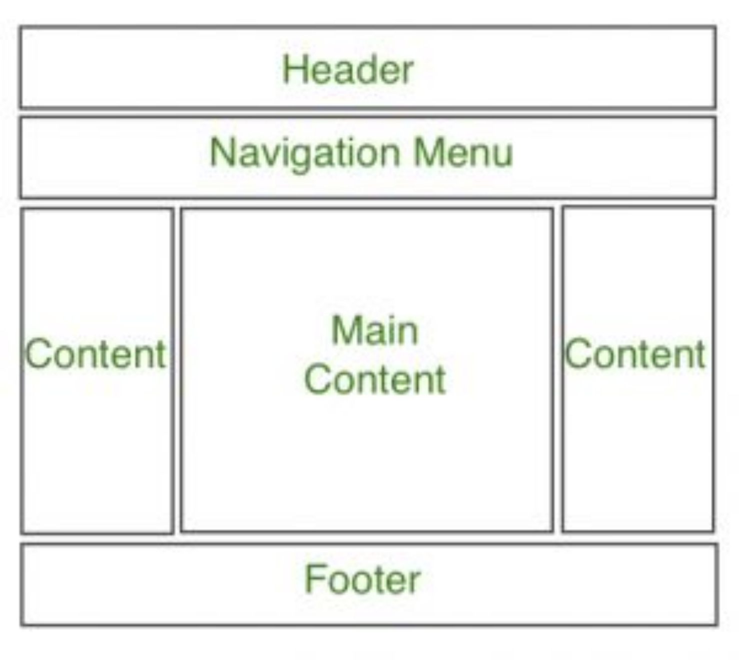
Aim: How Can We Use Divs To Organize Our Web Pages?

Do Now: What websites can you think of that have a layout similar to the one depicted here?



Header

Navigation Menu

Content

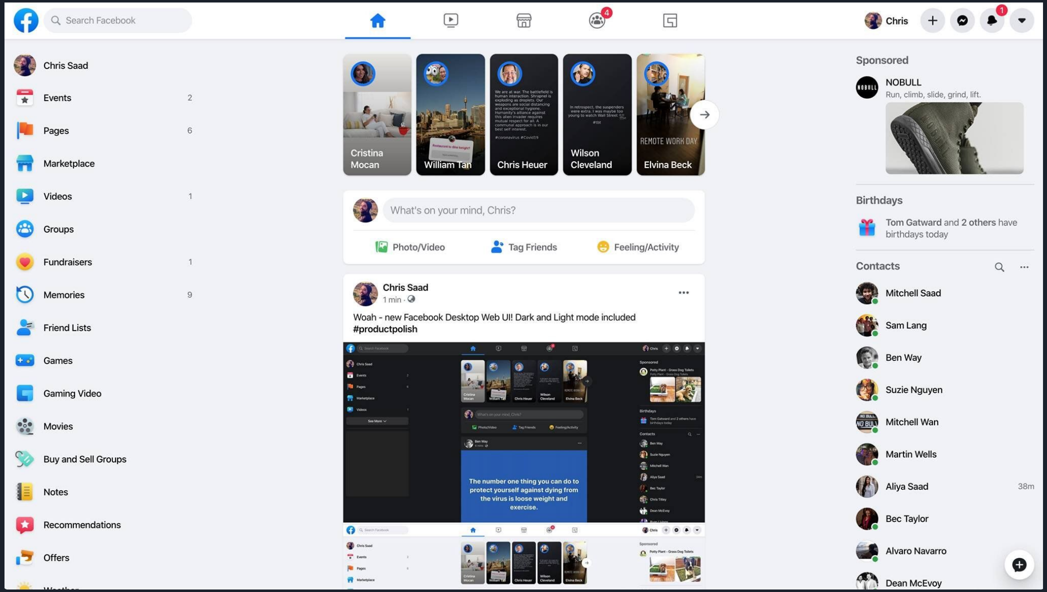
Main Content

Content

Footer

Discussing Website Layouts

- Websites are rarely just a single, continuous column of text, images, or other content
- Instead, websites are often divided up into different sections
- For example, on Facebook the middle column shows posts on your timeline, the left column shows the different parts of the site you can navigate to, and the right column shows the friends you can message



Chris Saad
1 min ·

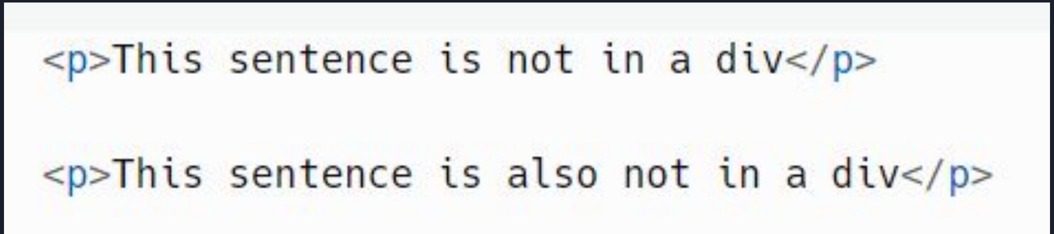
Woah - new Facebook Desktop Web UI! Dark and Light mode included #productpolish

The number one thing you can do to protect yourself against dying from the virus is loose weight and exercise.

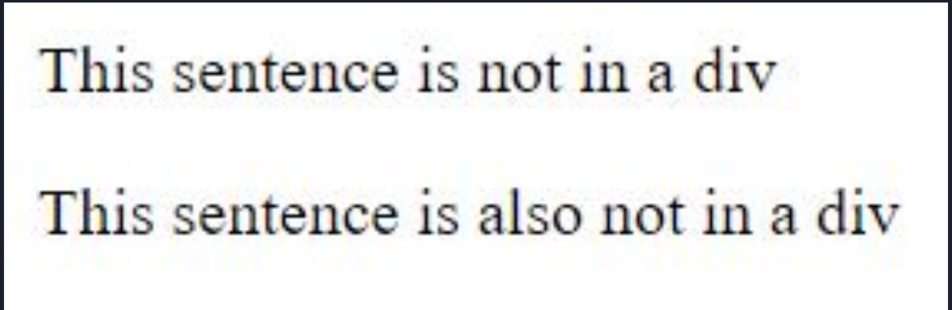
The <div> tag

- There are a wide range of reasons why we might want to divide our web pages into different parts, or group similar or related things on a web page together
- Whatever the reason, we can do so using the <div> tag!
- The <div> tag goes in the body portion of out HTML document and its main purpose is to act as a container
- We can put nearly any content in between our <div> start and end tags, including other <div> tags
- However, it is important to note that the <div> tag by itself does nothing and wouldn’t even be noticeable that it was used on a web page
- The true strength of the <div> tag lies in how it interacts with CSS

No CSS Or Divs



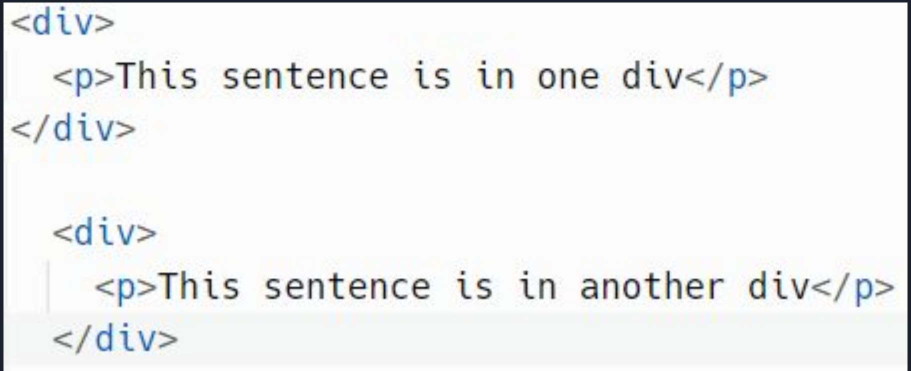
```
<p>This sentence is not in a div</p>

<p>This sentence is also not in a div</p>
```



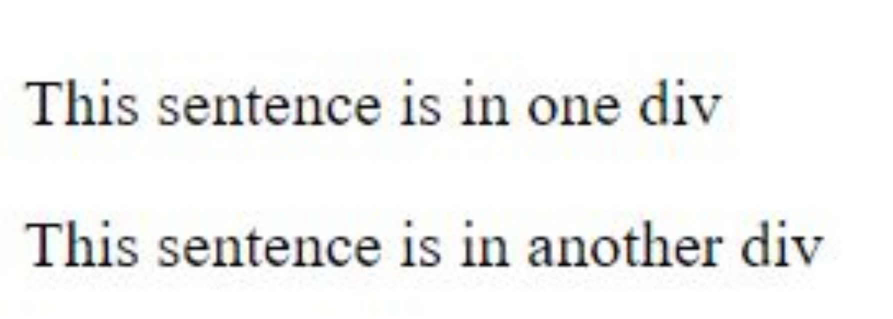
This sentence is not in a div

This sentence is also not in a div

Divs without CSS



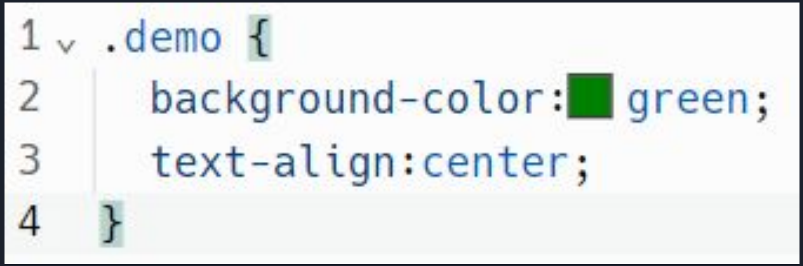
```
<div>
  <p>This sentence is in one div</p>
</div>

  <div>
    <p>This sentence is in another div</p>
  </div>
```



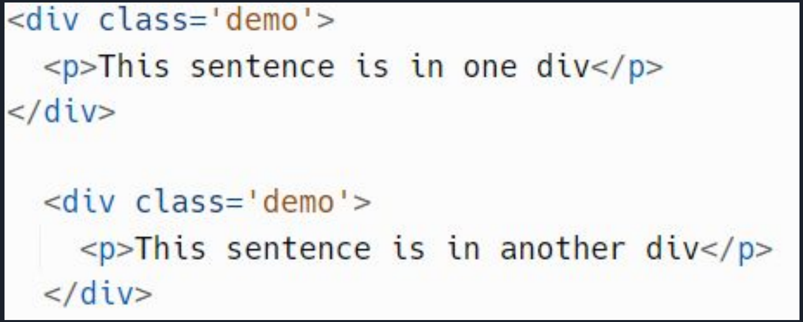
This sentence is in one div

This sentence is in another div

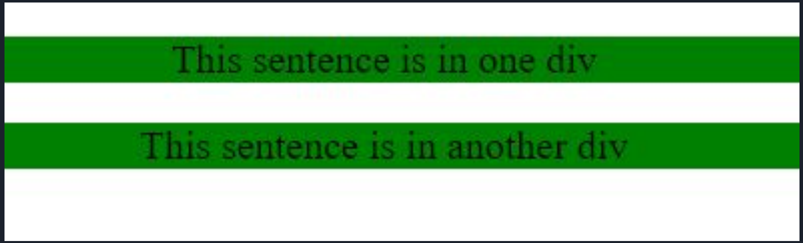
Divs with CSS



```css
.demo {
    background-color: green;
    text-align:center;
}
```



```html
<div class='demo'>
  <p>This sentence is in one div</p>
</div>

  <div class='demo'>
    <p>This sentence is in another div</p>
  </div>
```

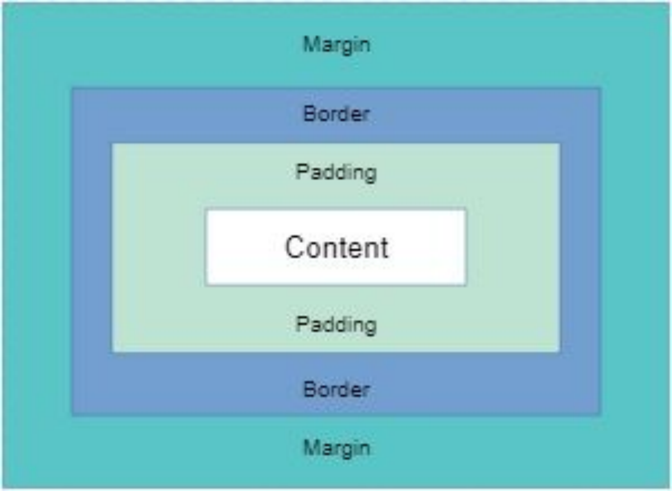


This sentence is in one div

This sentence is in another div

The Box Model

- As previously mentioned, the strength of the <div> tag lies in how it can be used with CSS
- The Box Model is an important facet of CSS and can really help control the size, shape, and color of our divs
- Nearly every HTML element on a web page is actually made up of 4 concentric rectangles, as depicted below



The Box Model: Content

- Content is the innermost box of the Box Model
- Although we haven’t referred to it by name before, this is the box that we’ve been dealing with in class so far
- The content box is where the content of our HTML elements live, such as images for the <img> tag, text for the <p> tag, and the various things that might go inside a <div> tag
- When we set the width and/or height of an element using CSS, we’re actually only setting the size of the content box

The Box Model: Padding

- Padding lies in between the Content and Border boxes of the Box Model
- As the name suggests, padding is just extra space around the Content box
- Notably, if we set a background-color for an HTML element, the padding will be the same color
- We can change how much padding an element has by using the padding property
- Alternatively, we can specify how much padding to use on each side by using the padding-top, padding-left, padding-bottom, and padding-right properties
- Below are both valid ways to add padding to an element

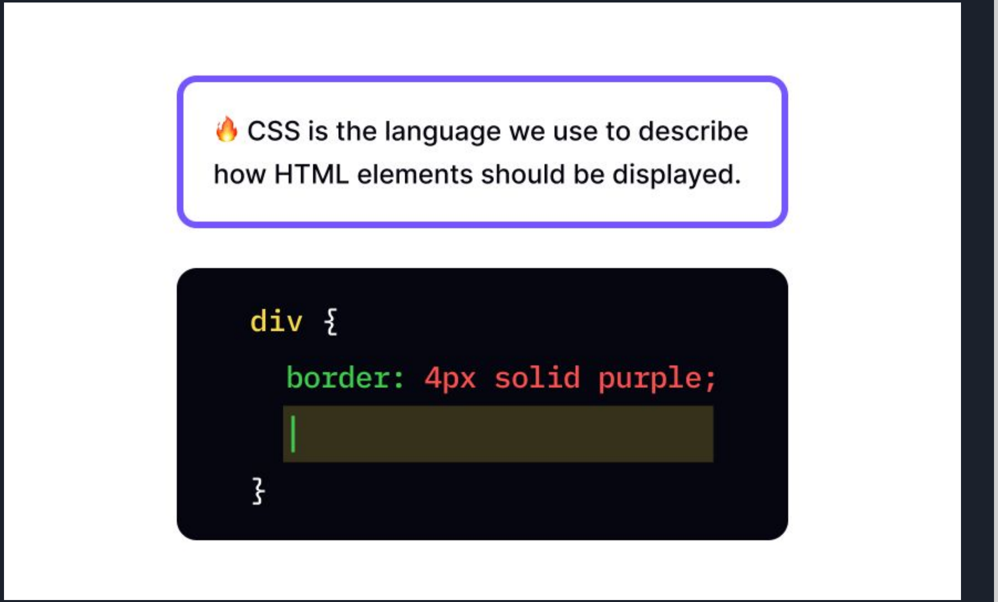
```
div {

    padding: 5px;

}
```

```
div {
    padding-top:5px;
    padding-left:10px;
    padding-right:5px;
}
```

The Box Model: Border

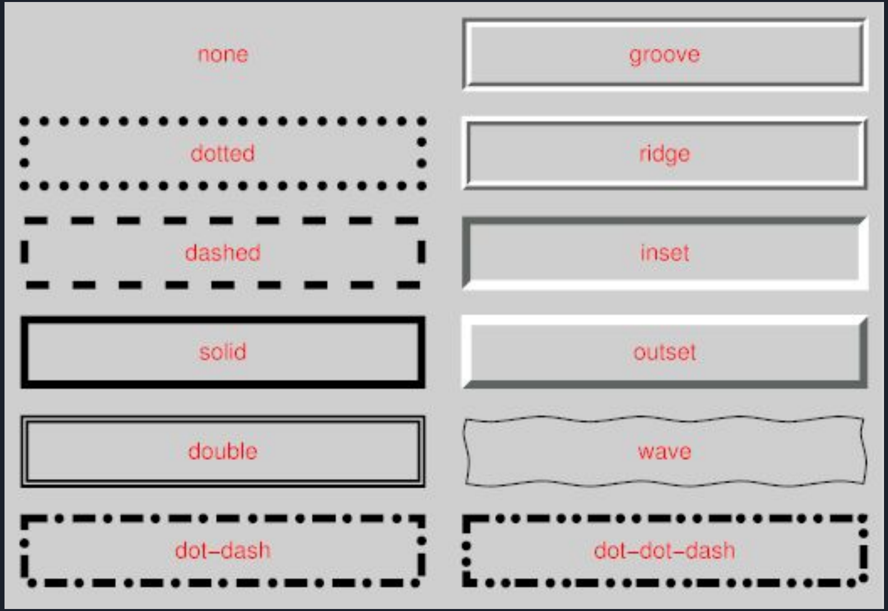
- Border lies in between the Padding and Margin boxes of the Box Model
- The Border box is the outermost visible box of the box model
- Unlike the Padding box, not only can we specify how thick the border should be, we can also set the color and style of the border
- The border property takes 3 values, size, style, and color
- The value for size works similarly to padding and we can also use color names or hex codes for the color value
- Some of the different styles of border we can use are shown on the next slide
- Also, similar to padding, we can set each side of the border independently using border-top, border-left, border-bottom, or border-right

The Box Model: Border



🔥 CSS is the language we use to describe how HTML elements should be displayed.

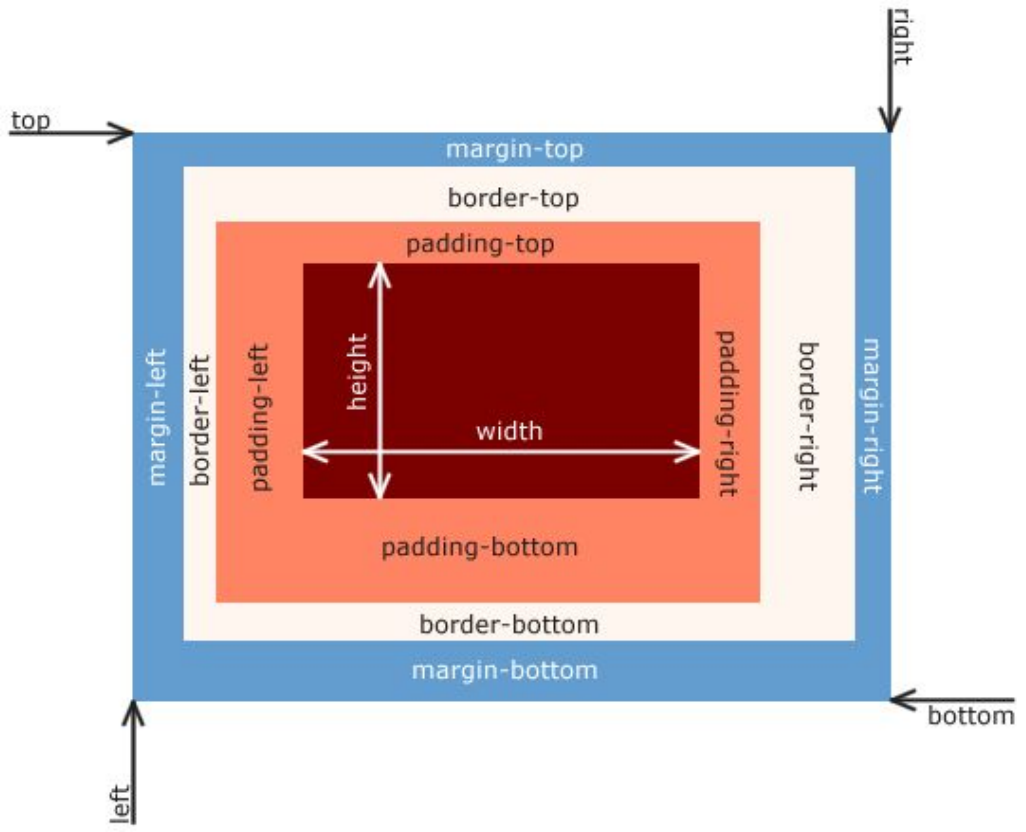
```
div {
  border: 4px solid purple;
  
}
```



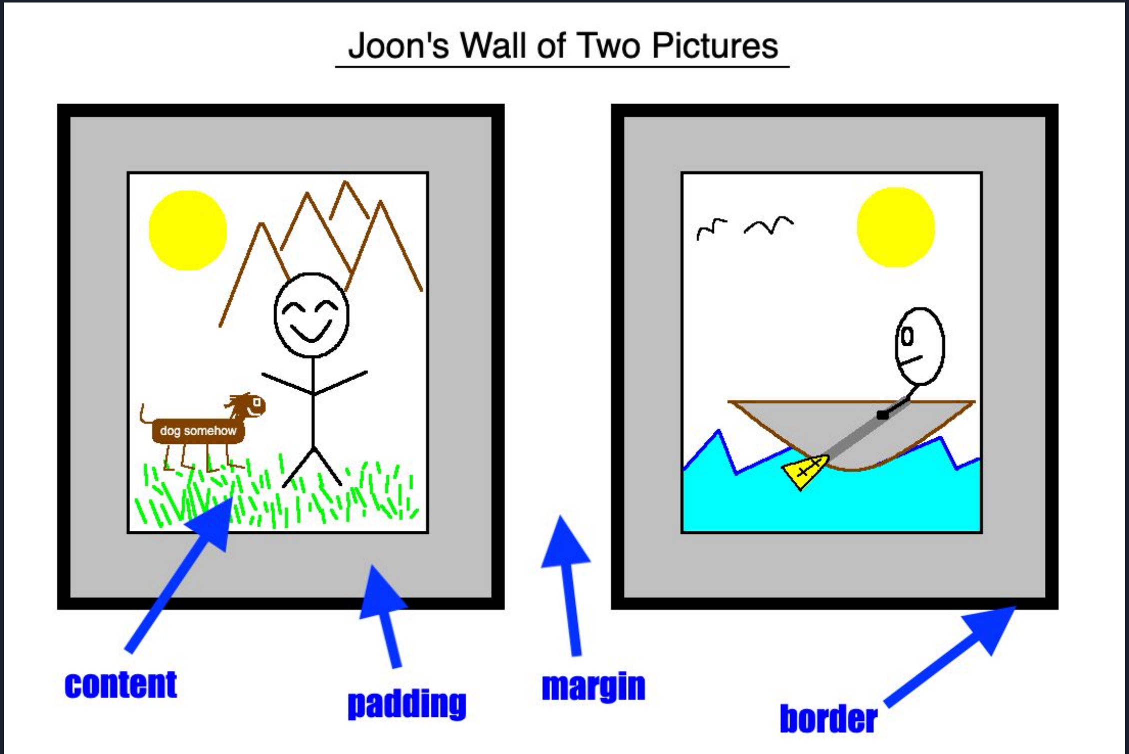
The Box Model: Margin

- The margin is the outermost box of the box model
- Unlike the other 3 parts of the Box Model, the margin is invisible
- In essence, the margin controls how much space you want between your HTML elements, like needing to keep a 6ft distance around others
- We can use the margin property to control how much space we should have around an HTML element
- Also, similar to padding and border, we can set each side of the margin independently by using margin-top, margin-left, margin-bottom, or margin-right

Box Model and Corresponding CSS properties



Example of parts of the Box Model



The Display Property

- The Box Model can help us control the size, shape, and color of a div (or other HTML elements)
- However, we also need to consider how we should multiple divs should be arranged on a page
- Theoretically, we could give each div position:absolute and position them on a web page manually, but that is not always the best choice and often harder than the alternative
- Instead, we can use the display CSS property to have a bit more control over how multiple HTML elements are arranged
- There are many possible values for the display property, but for today's lesson we will discuss only three
  - display: block
  - display: inline
  - display: inline-block

display: block

- When an HTML element has display:block, it stretches to take up as much width as possible
- In addition, elements with a block value for display will always start on a new line
- Most of the HTML tags that we’ve learned about so far actually have a value of block by default, including the <div> tag
- display: block is useful if you want to arrange your HTML elements vertically

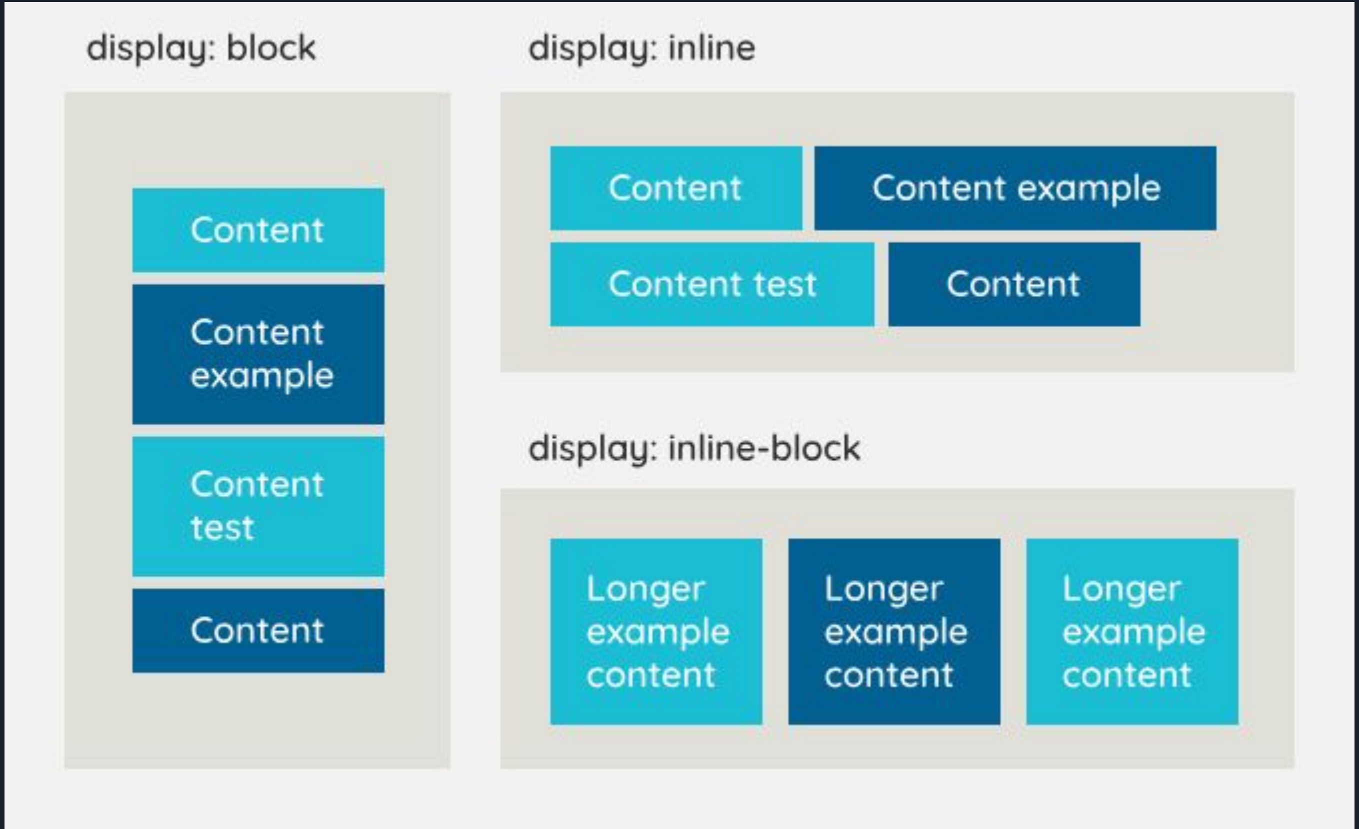
display: inline

- When an HTML element has display:inline, it only takes up as much space as needed
- In addition, elements with an inline-value for display do not start on a new line
- The <span> tag and <a> tag are examples of tags that have a value of inline by default
- display: inline is useful if you want to arrange your HTML elements horizontally, but there is another option as well…

display: inline-block

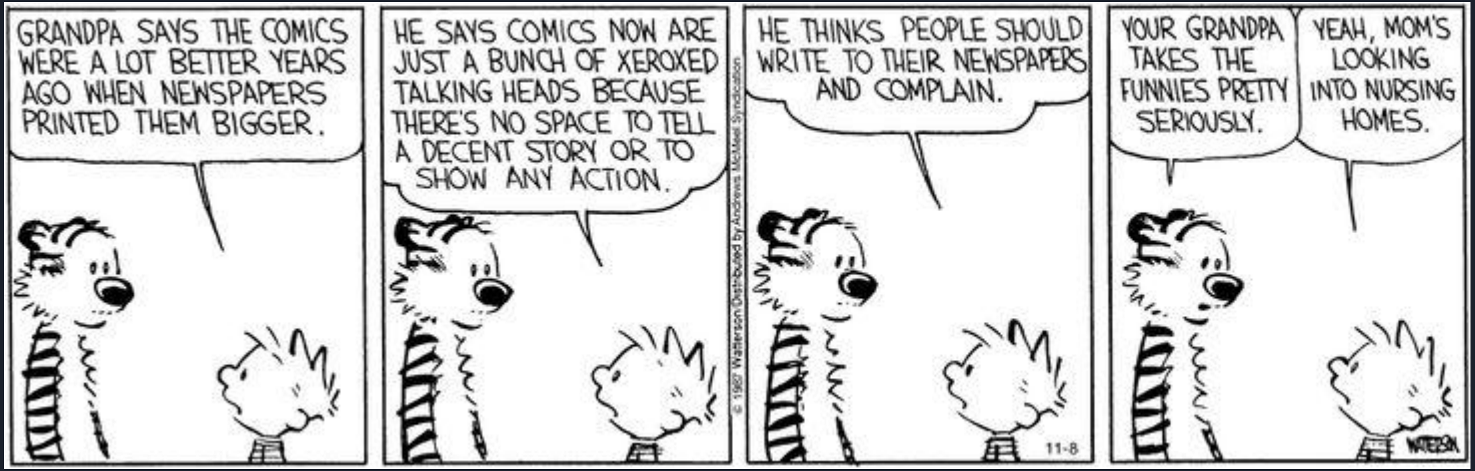
- An HTML element with display: inline-block behaves similarly to inline elements
- However, while inline elements only take up as much space as needed, inline-block elements can have their size set by the width and height property
- display: inline-block is useful if you want to arrange your HTML elements horizontally and have more control over their appearance

The Display Property Visualized

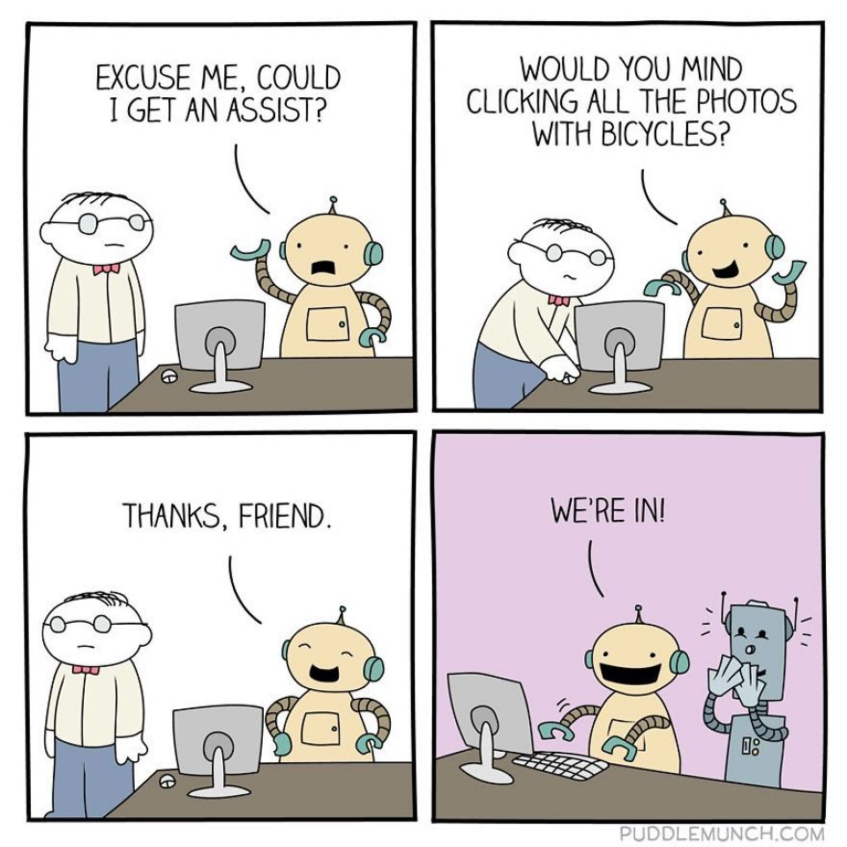


When designing your comic, keep in mind that they can be…

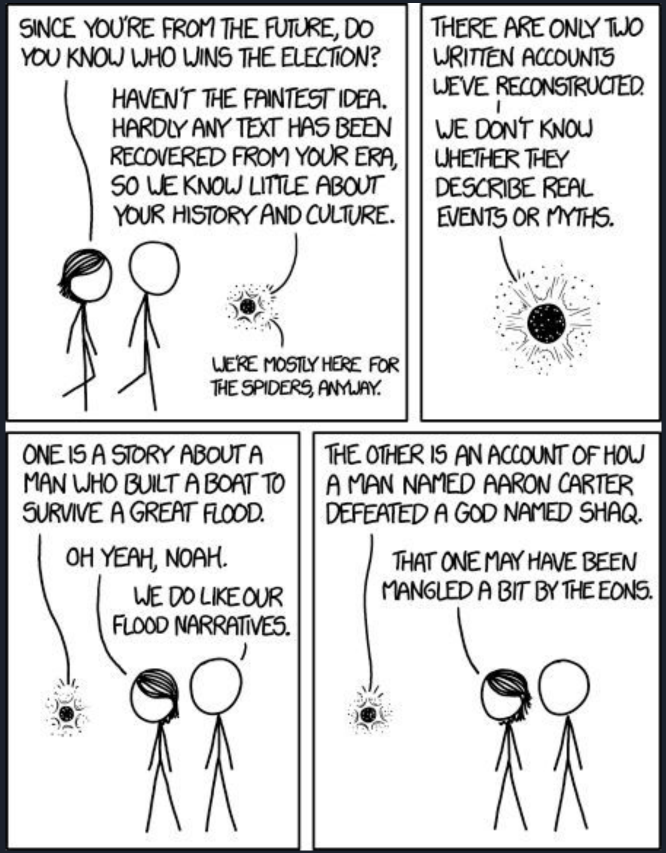
Horizontal


GRANDPA SAYS THE COMICS WERE A LOT BETTER YEARS AGO WHEN NEWSPAPERS PRINTED THEM BIGGER.
HE SAYS COMICS NOW ARE JUST A BUNCH OF XEROXED TALKING HEADS BECAUSE THERE'S NO SPACE TO TELL A DECENT STORY OR TO SHOW ANY ACTION.
© 1987 Watterson/Distributed by Andrews McMeel Syndication
HE THINKS PEOPLE SHOULD WRITE TO THEIR NEWSPAPERS AND COMPLAIN.
11-8
YOUR GRANDPA TAKES THE FUNNIES PRETTY SERIOUSLY.
YEAH, MOM'S LOOKING INTO NURSING HOMES.
WATTERSON

2x2


EXCUSE ME, COULD I GET AN ASSIST?
WOULD YOU MIND CLICKING ALL THE PHOTOS WITH BICYCLES?
THANKS, FRIEND.
WE'RE IN!
PUDDLEMUNCH.COM

Or Even Unequal in Size


SINCE YOU'RE FROM THE FUTURE, DO YOU KNOW WHO WINS THE ELECTION?
HAVEN'T THE FAINTEST IDEA. HARDLY ANY TEXT HAS BEEN RECOVERED FROM YOUR ERA, SO WE KNOW LITTLE ABOUT YOUR HISTORY AND CULTURE.
WE'RE MOSTLY HERE FOR THE SPIDERS, ANYWAY.
THERE ARE ONLY TWO WRITTEN ACCOUNTS WE'VE RECONSTRUCTED.
WE DON'T KNOW WHETHER THEY DESCRIBE REAL EVENTS OR MYTHS.
ONE IS A STORY ABOUT A MAN WHO BUILT A BOAT TO SURVIVE A GREAT FLOOD.
OH YEAH, NOAH.
WE DO LIKE OUR FLOOD NARRATIVES.
THE OTHER IS AN ACCOUNT OF HOW A MAN NAMED AARON CARTER DEFEATED A GOD NAMED SHAQ.
THAT ONE MAY HAVE BEEN MANGLED A BIT BY THE EONS.