
art 4217

Advanced Digital Applications

Lecturer

Ashwin Anandani

HTML + CSS

- First day of code!
 - Fundamentals
 - Starter exercise: Make a basic page from scratch
-

HTML + CSS (Day 2)

- Open your Codepen
 - Review
 - Practical : Using Bootstrap
 - Short Lecture : The Grid System
 - Continue Practical : Using Bootstrap, adding menus
-

HTML + CSS (Day 3)

- Review Lecture: The Grid, Bootstrap, Display
 - Practical : Use Bootstrap to make your app UI
 - Short Lecture : Getting it online
 - Practical : Put your site online and add a new folder
-

Code Basics

A little demonstration....

What We Use To Make Sites

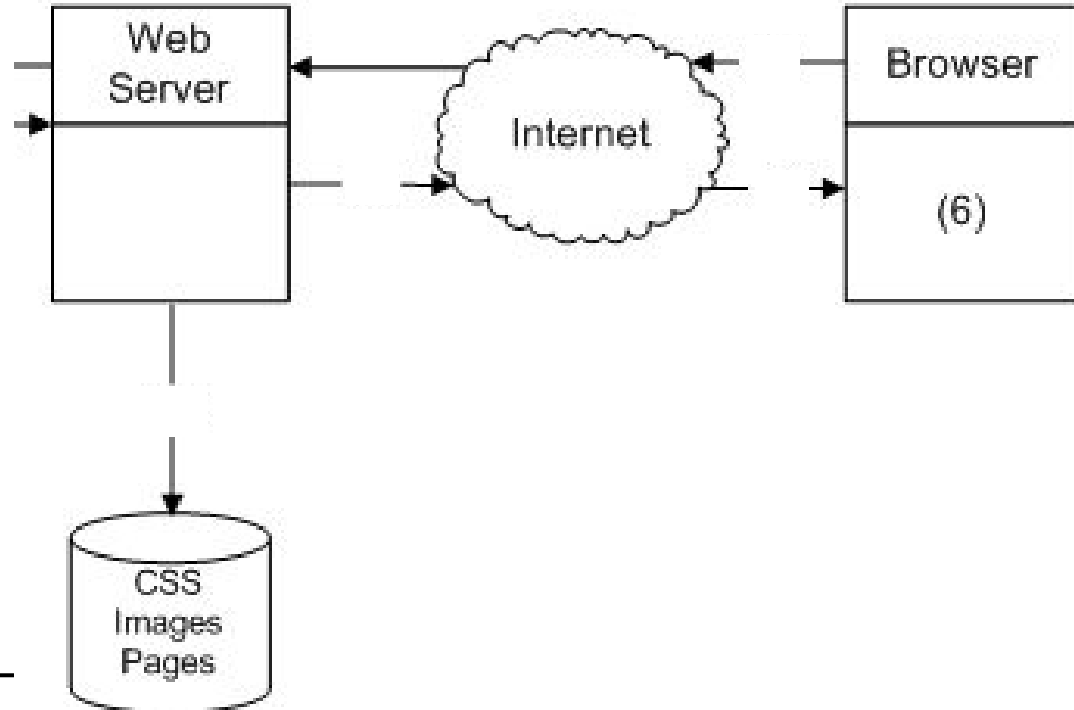
- Sites are just documents that a browser processes
- You need a code editor to write code:



Sublime Text 3

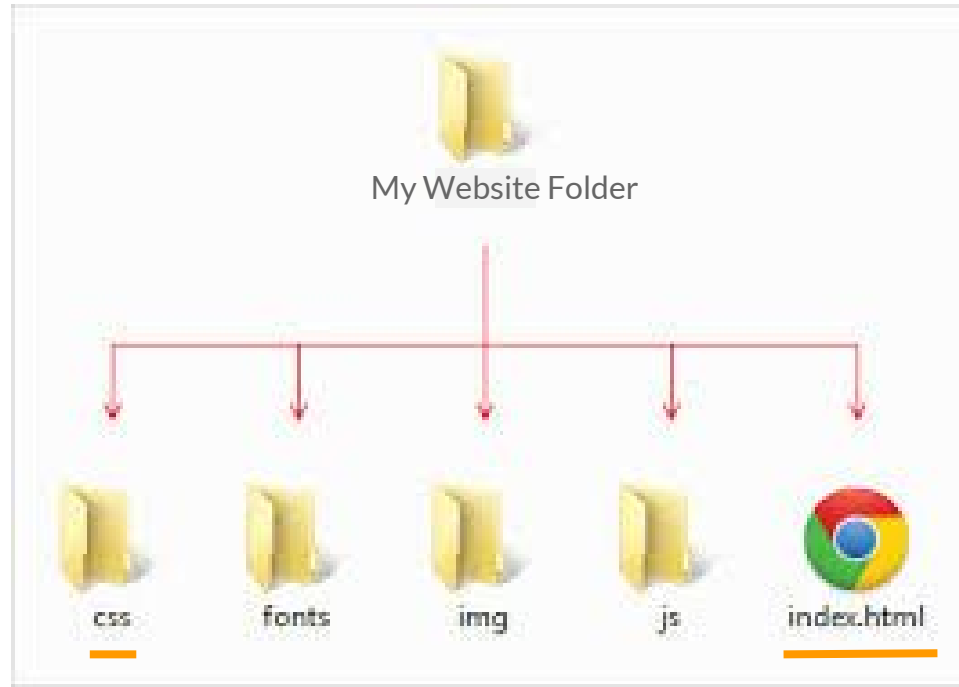
sublimetext.com/3

How sites are structured

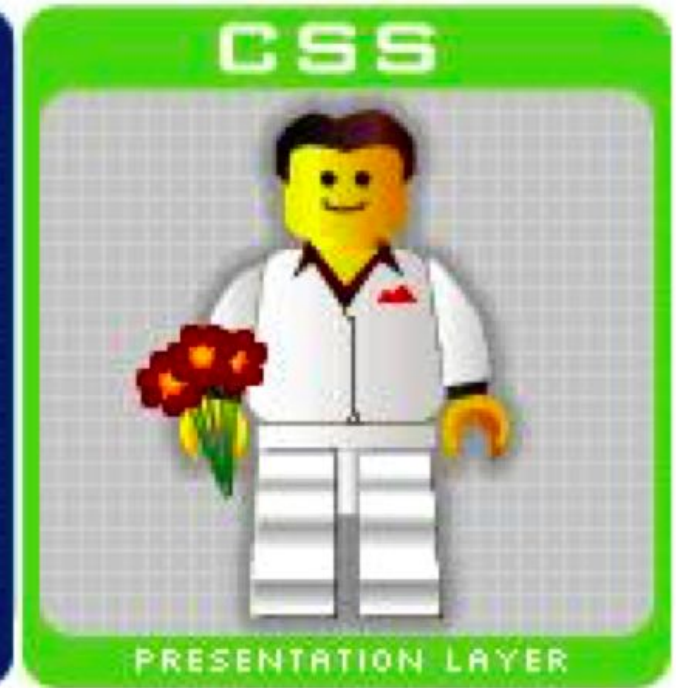
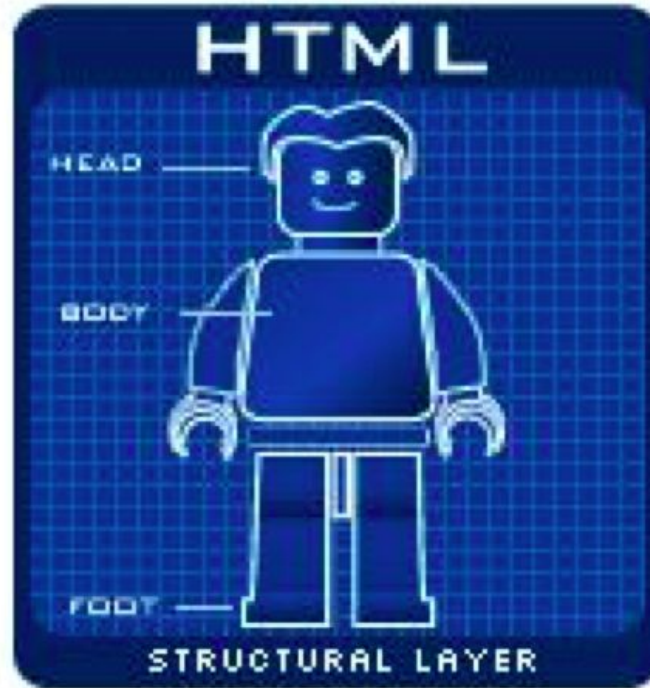


How sites are structured

On the 'Web Server'



Two Languages To Use



How HTML pages are structured

As a design

HTML

`<div id="header">`

`<div id="nav">`

`<div class="article">`

`<div class="section">`

`<div id="sidebar">`

`<div id="footer">`

An easier way to start...

www.CodePen.io/pen

Activity

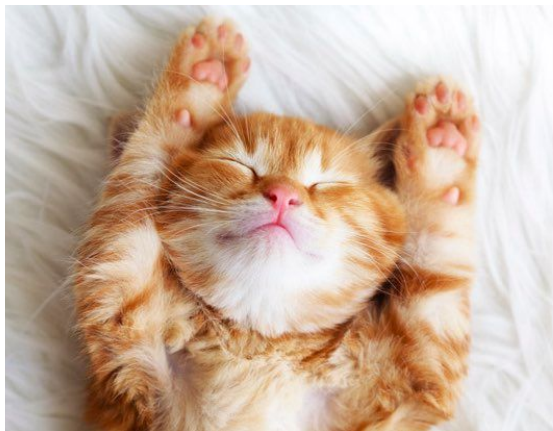
Create the same page yourself

Create The (next) Page

Everything has to look exactly the same

- Use codepen.io
 - **Modify** the styling
 - Copy down **the same text**
 - It will help you memorise
-

This is a sidebar
<div> in 35% width



Heading 2

The <p> is used for
paragraph text.

Learn from places online

- [w3schools.com/cssref](https://www.w3schools.com/cssref)
 - simplehtmlguide.com/cheatsheet.php
 - For fonts: [google.com/fonts](https://www.google.com/fonts)
 - Search Google as much as possible
-

Layout = HTML

<head> will **not** be shown on the webpage

<title> : To set the name in the browser tab

<body> : Content of the page

<div> : Divider - used to create containers for control

<h1> ... <h6> : Header tags for titles (with pre-set sizes)

<p> : Use for regular writing

<a> : Link to another site

 : For images

Layout = HTML

Example:

`<h1>`**This is a Header**`</h1>`

This is not

Styling = CSS

background: for background colours

background-image: `url(" get the link to your image ")`

color: for the colour of text

font-size: to change the size of text

margin: for the distance between other stuff

display: for how something should show (like a block or even not at all)

Styling = CSS

HTML tag property value close bracket

⎵ ⎵ ⎵ ⎵

```
a { background-color: yellow; }
```

↑ ↑

open bracket code's version of a full-stop

Styling = CSS

Example:

```
tag {  
    color: #123;  
}
```

Look up “hex
color codes”



```
.class {  
    background: #abc;  
}
```

CSS needs to talk to the HTML:
ID and **Class**

Class is for styles you want to use over and over

ID is for something really specific

```
<div class="col2" >  
<h2>HomePage</h2>  
  
```

CSS uses # to choose IDs, and . to choose **C**lasses

```
<div class="col2" >
<h2>HomePage</h2>
    
```

```
.col2 {
    float:left;
    width:46%;
    padding:0 2% 1em 2%;
    overflow:hidden;
    position:relative;
    left:-75%;
}
```

```
#floatRight {
    float:right;
}
```

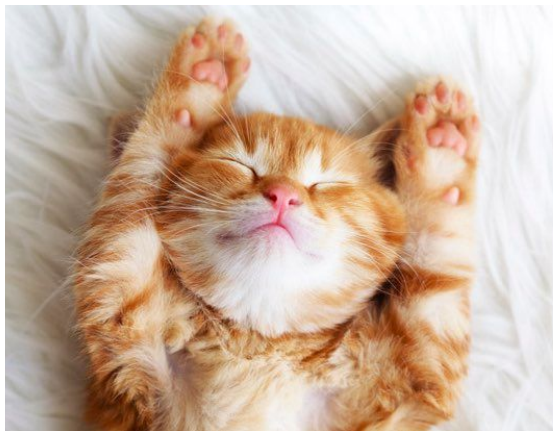
Practical II

Add the menu into your page



Menu 1 Menu 2 Menu 3 Menu 4

This is a sidebar
<div> in 35% width

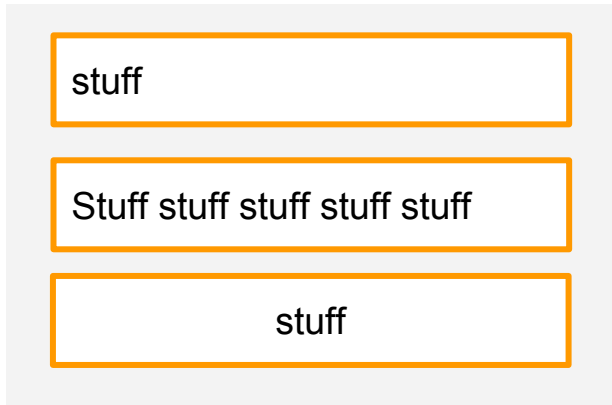


Heading 2

The <p> is used for
paragraph text.

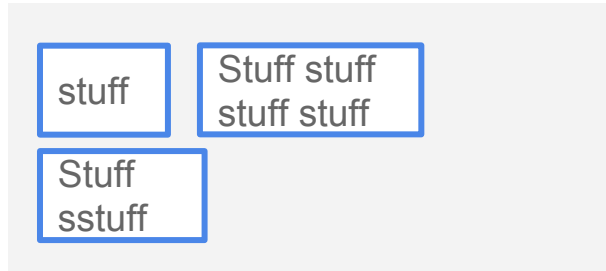
Understanding 'display'

Easy way



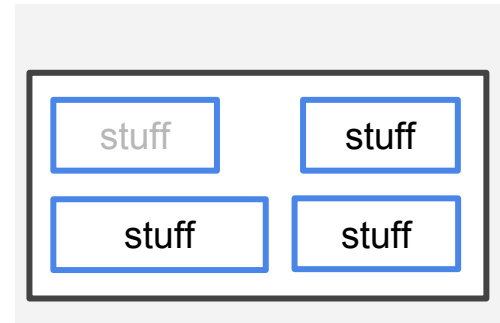
display:

block



display:

inline



display:

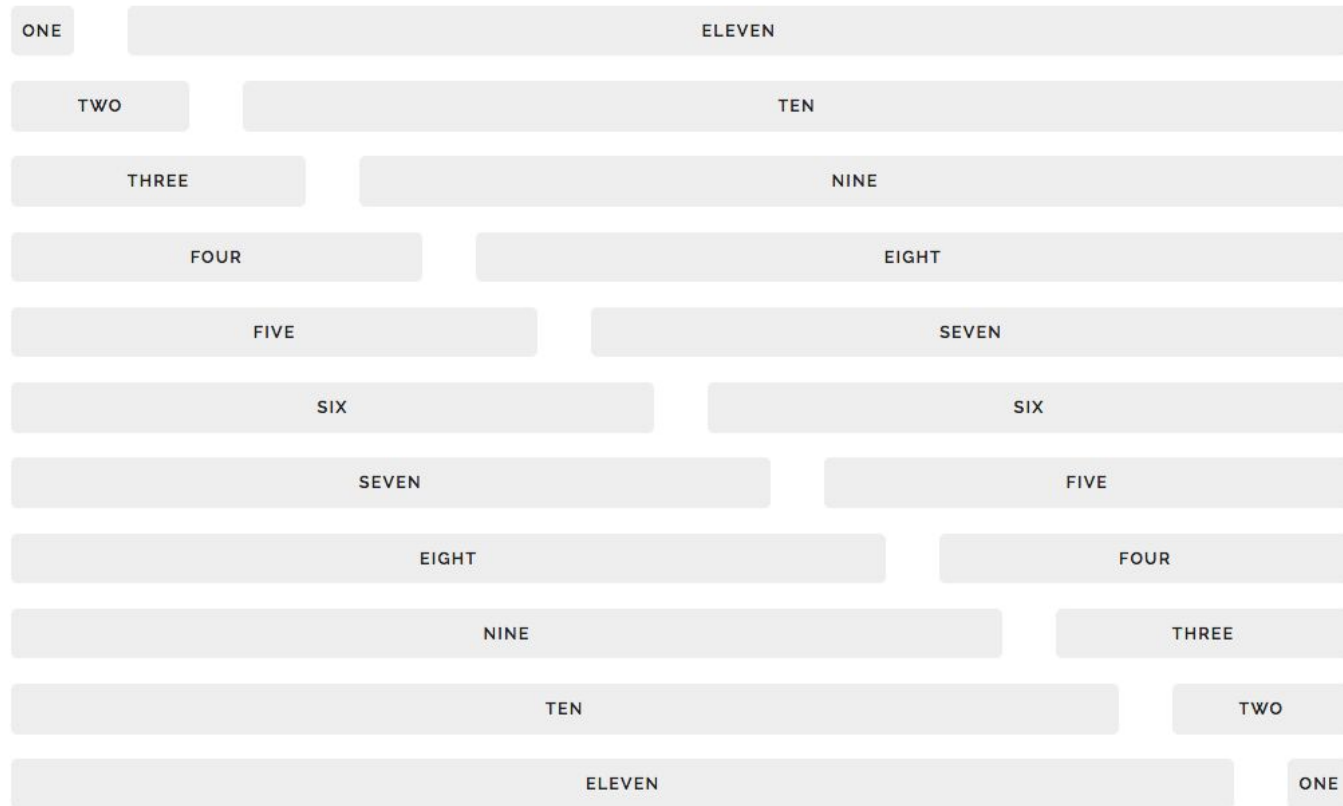
inline-block

Responsiveness

More like cheatsheet...

THE GRID

The grid is a 12-column fluid grid with a max width of 960px, that shrinks with the browser/device at smaller sizes. The max width can be changed with one line of CSS and all columns will resize accordingly. The syntax is simple and it makes coding responsive much easier. Go ahead, resize the browser.



Example: Stacked-to-horizontal

Using a single set of `.col-md-*` grid classes, you can create a basic grid system that starts out stacked on mobile devices and tablet devices (the extra small to small range) before becoming horizontal on desktop (medium) devices. Place grid columns in any `.row`.

.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1
.col-md-8								.col-md-4			
.col-md-4				.col-md-4				.col-md-4			
.col-md-6						.col-md-6					

```
<div class="row">
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
</div>
<div class="row">
  <div class="col-md-8">.col-md-8</div>
  <div class="col-md-4">.col-md-4</div>
</div>
<div class="row">
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
</div>
<div class="row">
  <div class="col-md-6">.col-md-6</div>
  <div class="col-md-6">.col-md-6</div>
</div>
```

[Copy](#)

So let's try that again...

Easier way

<http://codepen.io/ashwinbsd/pen/bqrOVy>

What's happening? **Bootstrap**

Is a pre-written grid we usually use to start with

<http://getbootstrap.com/css/#grid-intro>

Layout = HTML

<head> will **not** be shown on the webpage

<title> : To set the name in the browser tab

<body> : Content of the page

<div> : Divider - used to create containers for control

<h1> ... <h6> : Header tags for titles (with pre-set sizes)

<p> : Use for regular writing

<a> : Link to another site

 : For images

CSS needs to talk to the HTML:

ID and **Class**

Class is for styles you want to use over and over

ID is for something really specific

HTML:

```
<div class="col2" >  
  
<h2>HomePage</h2>  
  
    
```

CSS:

```
.col2 {  
    float:left;  
    width:46%;  
    padding:0 2% 1em 2%;  
    overflow:hidden;  
    position:relative;  
    left:-75%;  
}
```

Styling = CSS

background: for background colours

background-image: `url(" get the link to your image ")`

color: for the colour of text

font-size: to change the size of text

margin: for the distance between other stuff

display: for how something should show (like a block or even not at all)

Practical III

Making your App UI

Try to make your App UI

Just the opening page

- Plan it out as blocks first
 - Use `background: url(' ')`
 - Use `background-size: cover`
-

Drawing it out

Example:

HTML

```
<div id="header">
```

```
<div id="nav">
```

```
<div class="article">
```

```
<div class="section">
```

```
<div id="sidebar">
```

```
<div id="footer">
```

Learn from places online

- [w3schools.com/cssref](https://www.w3schools.com/cssref)
 - simplehtmlguide.com/cheatsheet.php
 - For fonts: [google.com/fonts](https://www.google.com/fonts)
 - Search Google as much as possible
-

Add additional pages from the UI

And link them to each other

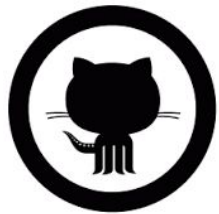
- Use ``
 - Understand how to link your CSS correctly
 - One CSS file for all HTML files
 - Use `<link ref="style.css">`
-

Getting stuff online

How to get things like images to show up at a URL

Get a free server

It'll actually be on your computer



GitHub Desktop

<http://desktop.github.com>

What's going on

- Make a folder on your computer
 - GitHub will connect it to the internet
 - You'll put your files here and be able to clone them anywhere at anytime
-

Practical III

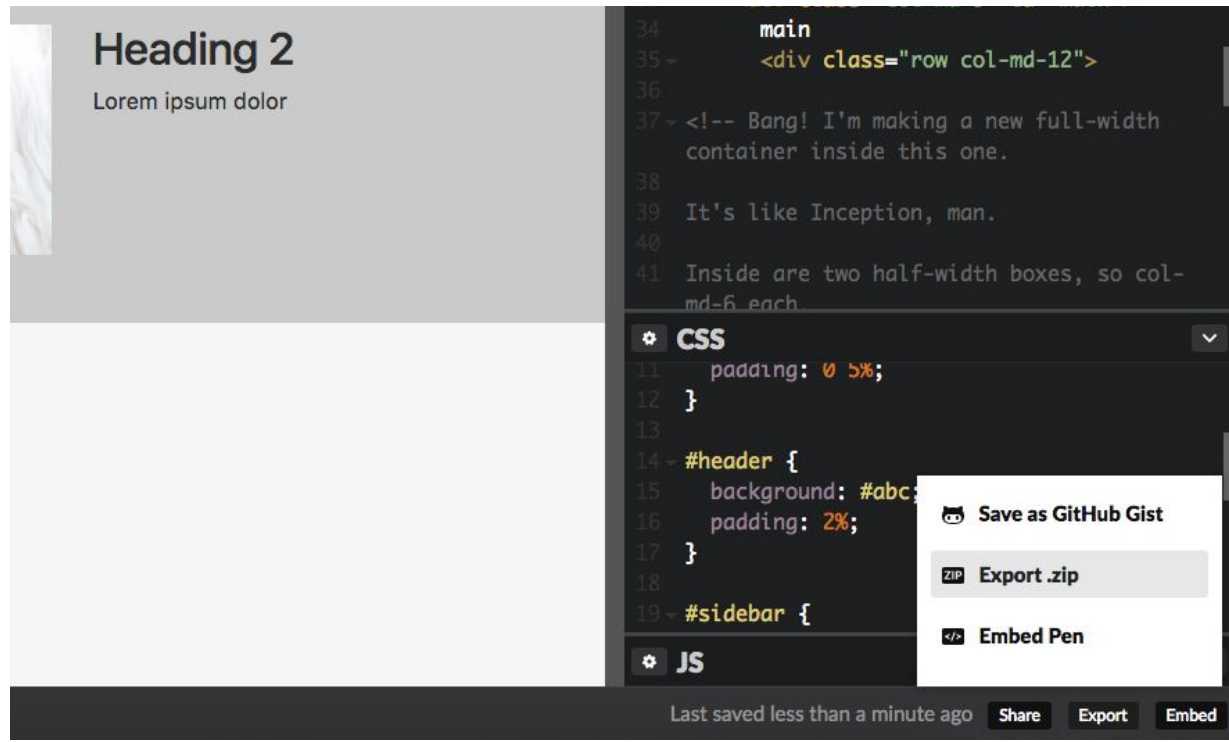
Continued

Add additional pages from the UI

And link them to each other

- Use ``
 - Understand how to link your CSS correctly
 - One CSS file for all HTML files
 - Use `<link ref="style.css">`
-

Export your code in the end



The screenshot displays a code editor interface with a light gray sidebar on the left and a dark gray main area on the right. The sidebar contains a preview of a web page with a heading and placeholder text. The main area shows code in three tabs: HTML, CSS, and JS. The HTML tab is active, showing a snippet of code with a comment. The CSS tab is also visible, showing a snippet of code. The JS tab is partially visible at the bottom. An export menu is open over the JS tab, offering options to save as a GitHub Gist, export as a ZIP file, or embed the code. At the bottom of the editor, there is a status bar indicating the last save time and buttons for sharing, exporting, and embedding.

Heading 2
Lorem ipsum dolor

```
34      main
35      <div class="row col-md-12">
36
37      <!-- Bang! I'm making a new full-width
38           container inside this one.
39      It's like Inception, man.
40
41      Inside are two half-width boxes, so col-
42           md-6 each
```

CSS

```
11      padding: 0 5%;
12  }
13
14  #header {
15      background: #abc;
16      padding: 2%;
17  }
18
19  #sidebar {
```

JS

- Save as GitHub Gist
- Export .zip
- Embed Pen

Last saved less than a minute ago [Share](#) [Export](#) [Embed](#)

This deck is now at:

[http://ashportfolio.github.io/art4217/decks/
lesson-7-9.pdf](http://ashportfolio.github.io/art4217/decks/lesson-7-9.pdf)
