

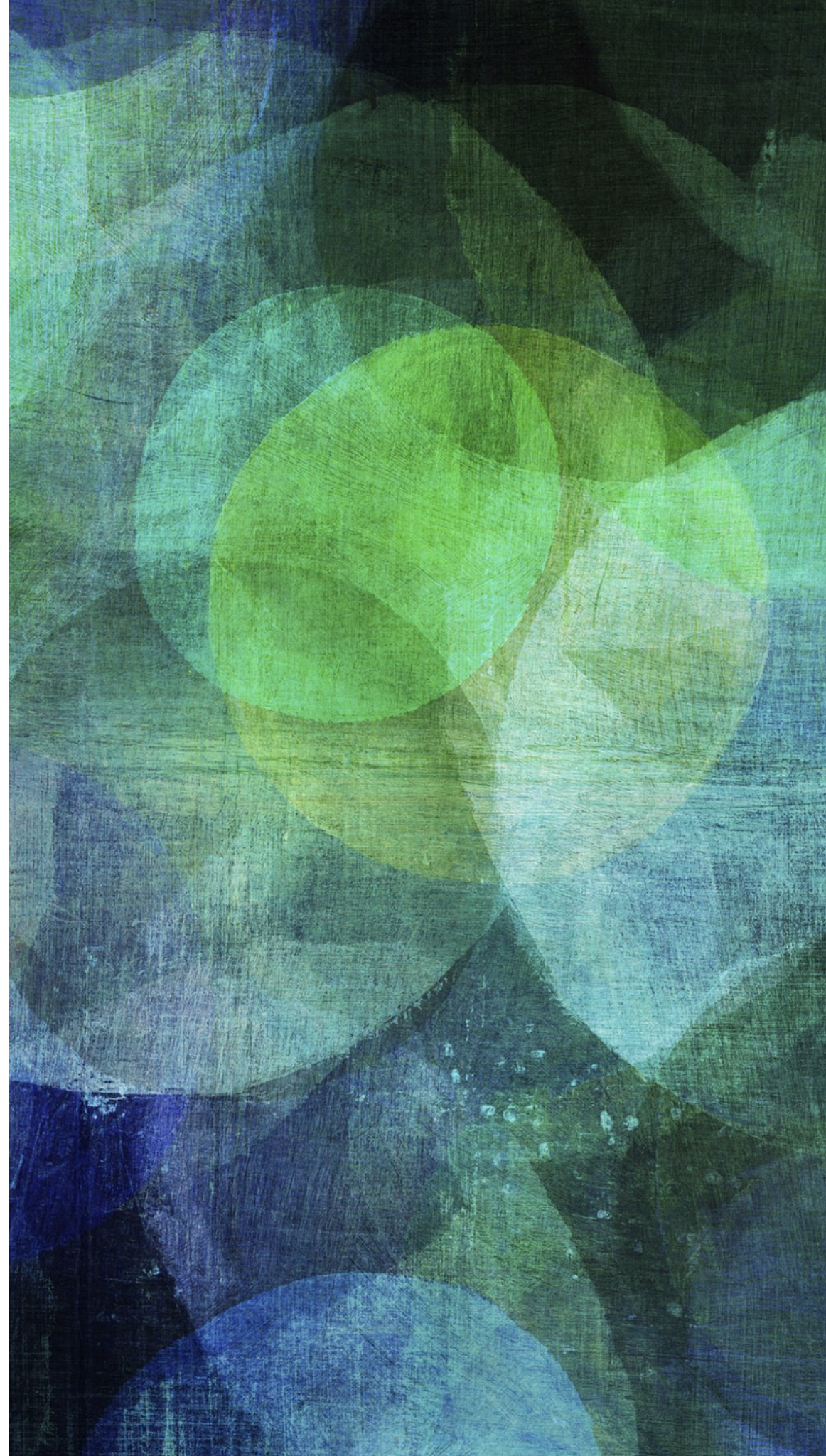
# HTML + CSS

---

*ScottyLabs WDW*

# OVERVIEW

.....  
*What are HTML and CSS?*  
*How can I use them?*





# WHAT ARE HTML AND CSS?

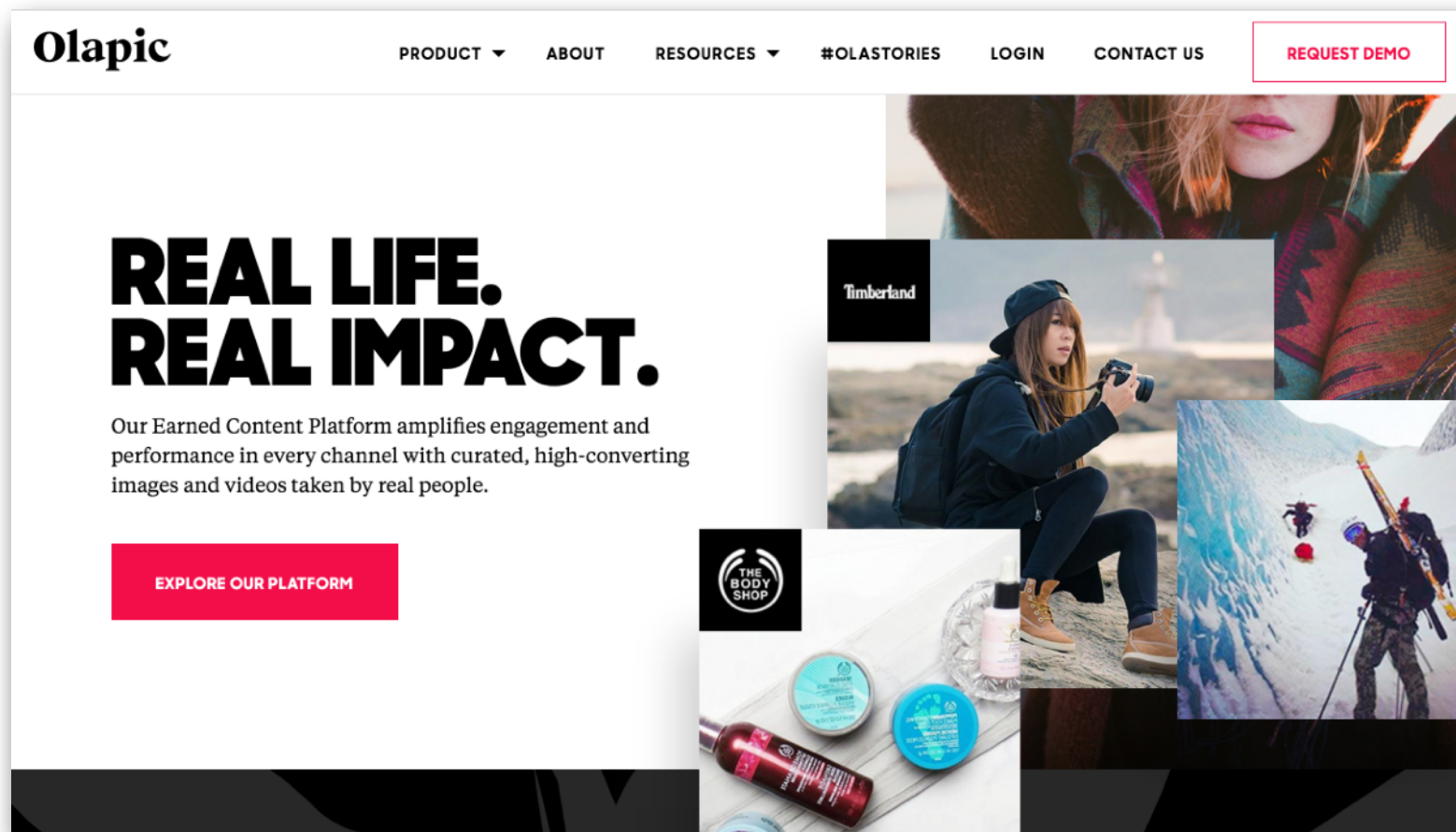
---

- HTML - HyperText Markup Language
  - Specifies webpage content hierarchy
  - Describes *rough* layout of content
- CSS - Cascading Style Sheets
  - Tells the browser how the content should look when displayed
  - Allows for fine-tuned control of layout
  - Responsible for “responsive” designs

# WHAT ARE HTML AND CSS? (TLDR)

---

**HTML = CONTENT**  
**CSS = STYLE**



15-122 Principles of Imperative Computation, Fall 2014		
<a href="#">Home</a> <a href="#">Schedule</a> <a href="#">Assignments</a> <a href="#">Resources</a> <a href="#">Staff</a> <a href="#">C0 Language</a> <a href="#">Piazza</a> <a href="#">Autolab</a> <a href="#">Quizzes</a>		
Schedule		
DATE	TOPIC	OTHER
M Aug 25	Lab 0: <a href="#">Starting out with C0</a>	
T Aug 26	<a href="#">Overview</a> (code)	
R Aug 28	<a href="#">Contracts</a> (slides)	
F Aug 29	Recitation 0: <a href="#">C0 and Contracts</a>	Quiz 0
M Sep 1	Labor day — optional C0 at CMU laptop setup session, 4-6pm, GHC 4401	
T Sep 2	<a href="#">Integers</a> (Slides)	Written 1 Due
R Sep 4	<a href="#">Arrays</a>	Programming 1 (Scavenger Hunt) Due
F Sep 5	Recitation 1: <a href="#">C0's basic types</a>	Quiz 1
M Sep 8	Lab 2: <a href="#">Ints and Arrays</a>	Written 2 Due
T Sep 9	<a href="#">Search</a> (arrayutil.c0 handout) (Code)	
R Sep 11	<a href="#">Sorting</a> (Code)	Programming 2 (Pixels) Due
F Sep 12	Recitation 2: <a href="#">Big-O Notation</a>	Quiz 2
M Sep 15	Lab 3: <a href="#">Timing and Testing</a>	Written 3 Due
T Sep 16	<a href="#">Binary Search</a> (Code)	
R Sep 18	<a href="#">Quicksort</a> (Code)	Programming 3 (Images) Due
F Sep 19	Recitation 3: <a href="#">Other Sorting Algorithms</a>	Quiz 3
M Sep 22	Lab 4: <a href="#">Processing Arrays of Strings</a>	Written 4 Due
T Sep 23	<a href="#">Data Structures</a> (Code)	
R Sep 25	<a href="#">Stacks and Queues</a>	Programming 4 (DosLingos) Due
F Sep 26	Recitation 4: <a href="#">Stacks, queues, recursion</a> (Solutions)	Quiz 4
M Sep 29	Lab 5:	Written 5 Due



With just *HTML* and *CSS*, we can build a complete **Static Website**.



We cannot build *Dynamic Websites* that adapt to different data.

# WITHOUT CSS STYLING

.....

## Sergey Brin's Home Page

Ph.D. student in Computer Science at Stanford - [sergey@cs.stanford.edu](mailto:sergey@cs.stanford.edu)

### Research

Currently I am at [Google](#).

In fall '98 I taught [CS 349](#).

### Data Mining

A major research interest is data mining and I run a meeting group here at Stanford. For more information take a look at the [MIDAS](#) home page or see the [datamine maling list achive](#). Here are some recent publications:

- **Extracting Patterns and Relations from the World Wide Web**

by Sergey Brin.

We demonstrate a technique for extracting relations from the WWW based on the duality of patterns and relations. We experiment with it by extracting a relations of [books](#). WebDB Workshop at EDBT '98 ([postscript](#)).

- **Dynamic Data Mining: A New Architecture for Data with High Dimensionality**

by Sergey Brin and Lawrence Page.

We describe a new architecture for data mining (sorry not yet available online). It makes use of some of the dynamic itemset counting technology described below.  
Work in progress. ([postscript](#))

- [Scalable Techniques for Mining Causal Structures](#)

by Craig Silverstein, Sergey Brin, Rajeev Motwani, and Jeff Ullman.

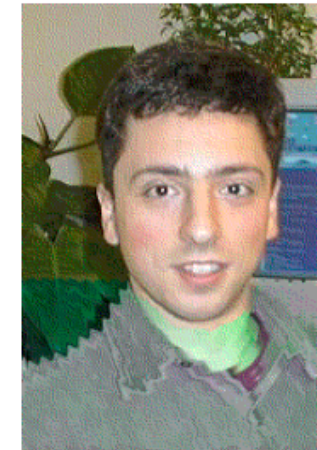
We address mining for causality, not just correlation in data.  
To appear in VLDB '98. ([abstract](#), [gzipped ps](#))

- [Dynamic Itemset Counting and Implication Rules for Market Basket Data](#)

by Sergey Brin, Rajeev Motwani, Jeffrey D. Ullman and Shalom Tsur.

We present and algorithm for counting large itemsets faster than previous algorithms. We rely on partial results to guide the mining process.  
Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 255-264, Tuscon, Arizona, May 13-15 1997. ([html](#), [postscript](#), [gzipped ps](#), [bibtex](#))

- [Beyond Market Baskets: Generalizing Association Rules to Correlations](#)





# WITH CSS STYLING

.....

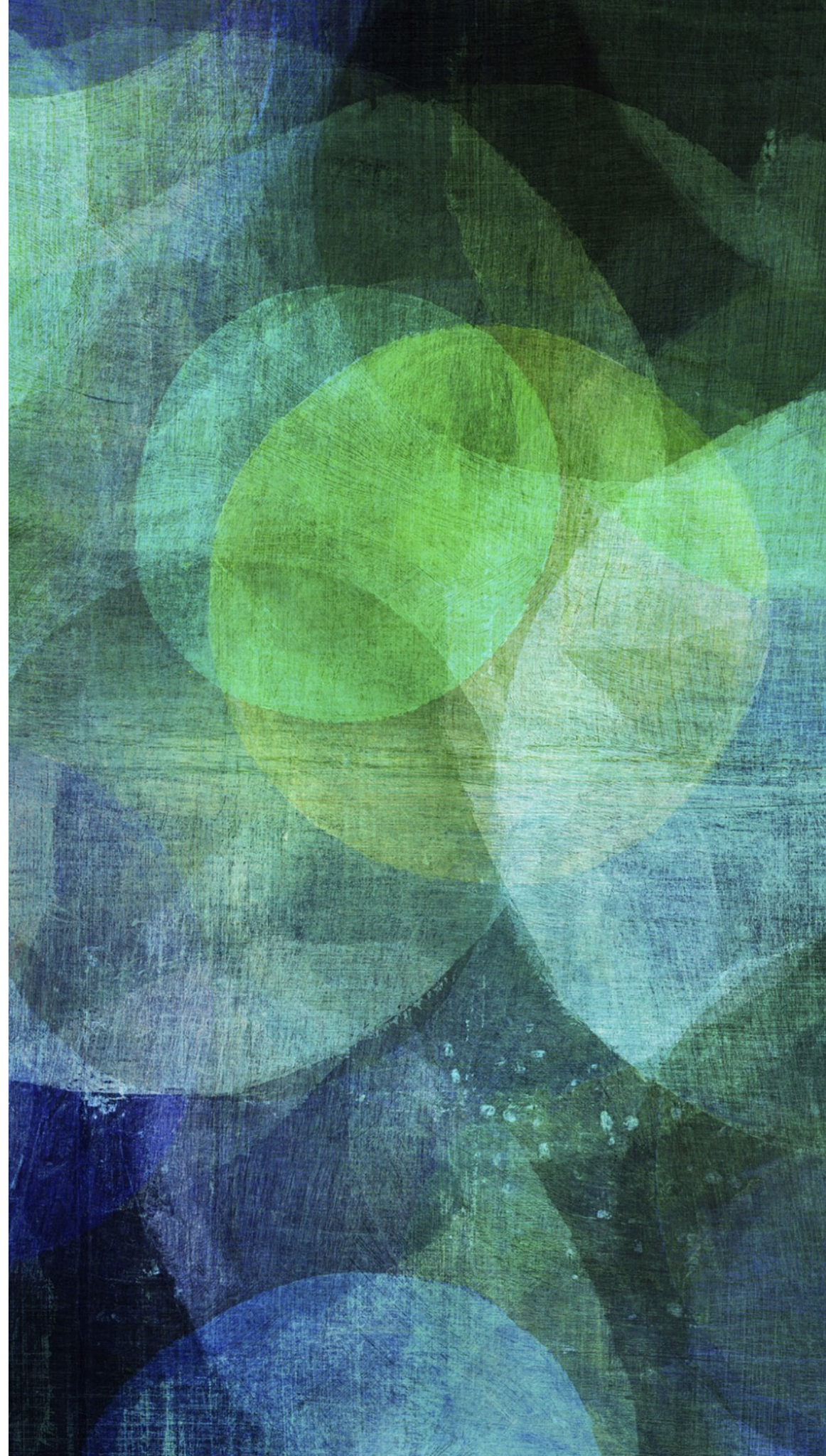




# HTML TAGS

---

*Components to compose  
webpages.*





# THE DOM

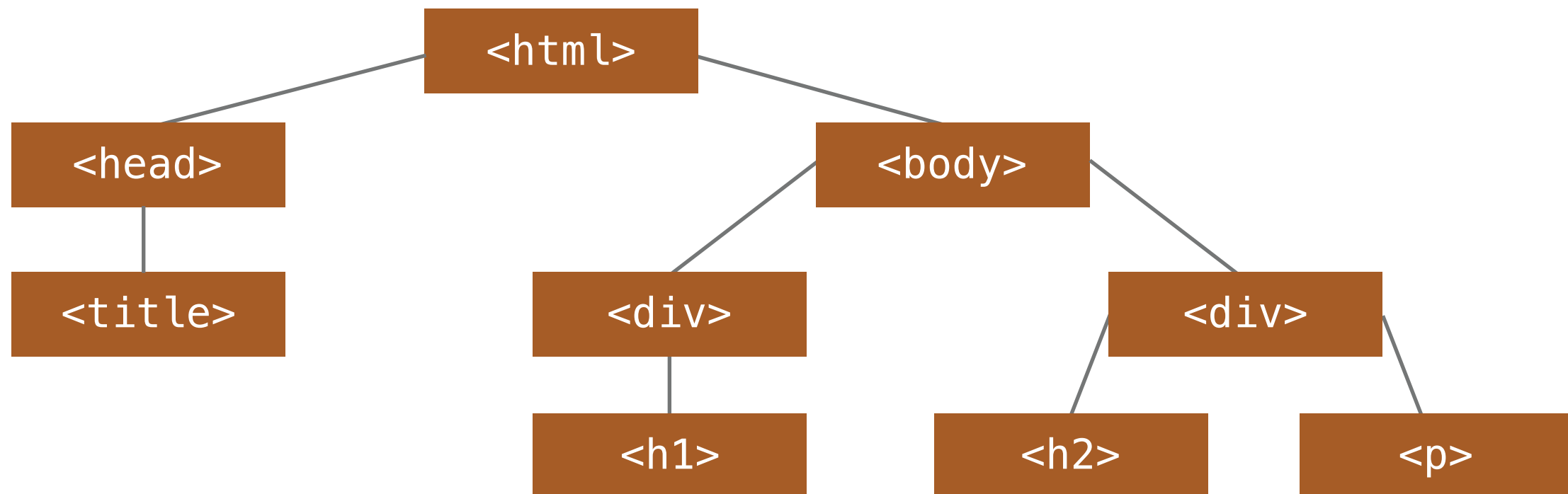
---

## DOCUMENT OBJECT MODEL

Known as the “DOM” the Document Object Model is a convention that describes the hierarchal organization of page “elements”. It can be visualized as a tree, with the usual *parent*, *child* and *sibling* terminology used to describe element relationships. The DOM is specified using “tags”.

# THE DOM

---



- Each HTML element is an object, with its own style properties
- The DOM describes the inheritance structure of CSS
- The DOM also defines how to manipulate HTML elements dynamically (check this out in the Javascript presentation!)



# FOLLOWING ALONG

---

*[skrulcik.github.io/WDW\\_HTMLCSS](https://skrulcik.github.io/WDW_HTMLCSS)*

# DIV TAGS

---

```
<div> Content </div>
```

- Organizes content areas into logical blocks
- Guides the browser laying out the page to bundle groups of tags together
- Very useful for more complex CSS layouts

# DIV TAG EXAMPLE

---

## Harry Bovik

Contact: skrulcik@gmail.com

### Work Experience

#### Software Engineer - *Carnegie Mellon*

Researching stuff that has to do with computers. Somehow related to *science*. Gotta love that scienc  
And those *Computers*. Was awarded many awards for award-worthy accomplishments.

#### Software Development Intern

Worked on **really important** stuff. More stuff about jobs and crap. Responsibility. **Big Data.**  
**Responsive Design. Agile Development.** Buzzwordified description of basic tasks.

Note how the most important words are bolded by using the *strong* tag.

### Mad Skillz Yo

Nunchuck skillz

Bow hunting skillz

Computer hacking Skillz



# HEADER TAGS

---

```
<h1>, <h2>, . . . , <h6>
```

Used to denote prominent text such as:

- Title
- Subtitle
- Section Header

The number indicates prominence, with 1 being most prominent and 6 least prominent.

# HEADER TAG EXAMPLE

---

h1

→ **Harry Bovik**

Contact: [skrulcik@gmail.com](mailto:skrulcik@gmail.com)

---

h2

→ **Work Experience**

**Software Engineer - *Carnegie Mellon***

Researching stuff that has to do with computers. Somehow related to *science*. Gotta love that scienc  
And those *Computers*. Was awarded many awards for award-worthy accomplishments.

h3

→ **Software Development Intern**

Worked on **really important** stuff. More stuff about jobs and crap. Responsibilty. **Big Data.**  
**Responsive Design. Agile Development.** Buzzwordified description of basic tasks.

Note how the most important words are bolded by using the *strong* tag.

**Mad *Skillz* Yo**

Nunchuck skillz

Bow hunting skillz

Computer hacking Skillz

# PARAGRAPH TAGS

---

```
<p></p>
```

- Standard way to denote a large block of text
- **<p>** tags add spacing between each other, if you want text with no spacing, it must be within the same **<p>** tag



# P TAG EXAMPLE

---

## Harry Bovik



Contact: skrulcik@gmail.com

---

## Work Experience

### Software Engineer - *Carnegie Mellon*



Researching stuff that has to do with computers. Somehow related to *science*. Gotta love that science And those *Computers*. Was awarded many awards for award-worthy accomplishments.

### Software Development Intern



Worked on **really important** stuff. More stuff about jobs and crap. Responsibility. **Big Data. Responsive Design. Agile Development.** Buzzwordified description of basic tasks.

Note how the most important words are bolded by using the *strong* tag.

## Mad Skillz Yo

Nunchuck skillz

Bow hunting skillz

Computer hacking Skillz

# IN-LINE TAGS

---

```
<strong></strong>
```

Makes text bold

```
<em></em>
```

Emphasizes text (usually italics)

# IN-LINE TAG EXAMPLES

---

em




## Work Experience

Software Engineer - *Carnegie Mellon*

Researching stuff that has to do with computers. Somehow related to *science*. Gotta love that science. And those *Computers*. Was awarded many awards for award-worthy accomplishments.

## Software Development Intern

Worked on **really important** stuff. More stuff about jobs and crap. Responsibility. **Big Data**. **Responsive Design**. **Agile Development**. Buzzwordified description of basic tasks.



strong



# ANCHOR TAGS (LINKS)

---

```
<a href="https://google.com">Google</a>
```

- Defines a link to another page
- **href** - Relative or absolute url to link to page
  - Relative: **./other-pages/blog-post-2.html**
  - Absolute: **https://google.com**

# LIST TAGS

---

```
<ul></ul>
```

Un-ordered (bulleted) list

```
<ol></ol>
```

Ordered (numbered) list

```
<li></li>
```

“List Item” - these tags go around each line/object in the list

# TABLE TAGS (1/2)

---

```
<table></table>
```

Defines the beginning and end of a table element.

```
<thead>  
</thead>
```

Contains the header row

```
<tbody>  
</tbody>
```

Contains data rows

## TABLE TAGS (2/2)

---

```
<tr></tr>
```

“Table Row” - Marks the beginning and end of a row of data

```
<th></th>
```

Table header entry

```
<td></td>
```

Table data entry

# COMMENTS

---

```
<!-- Comment Text -->
```

HTML is mostly self-documenting, so only use comments where they are really needed.



# IMAGE TAGS

---

```

```

- Places an image in your webpage
- **src** - Relative or absolute url for an image resource
  - Relative: **./img/my-image.jpg**
  - Absolute: **http://imgur.com/r/Harambe/Avyjs05**

# SUMMARY OF TAGS

---

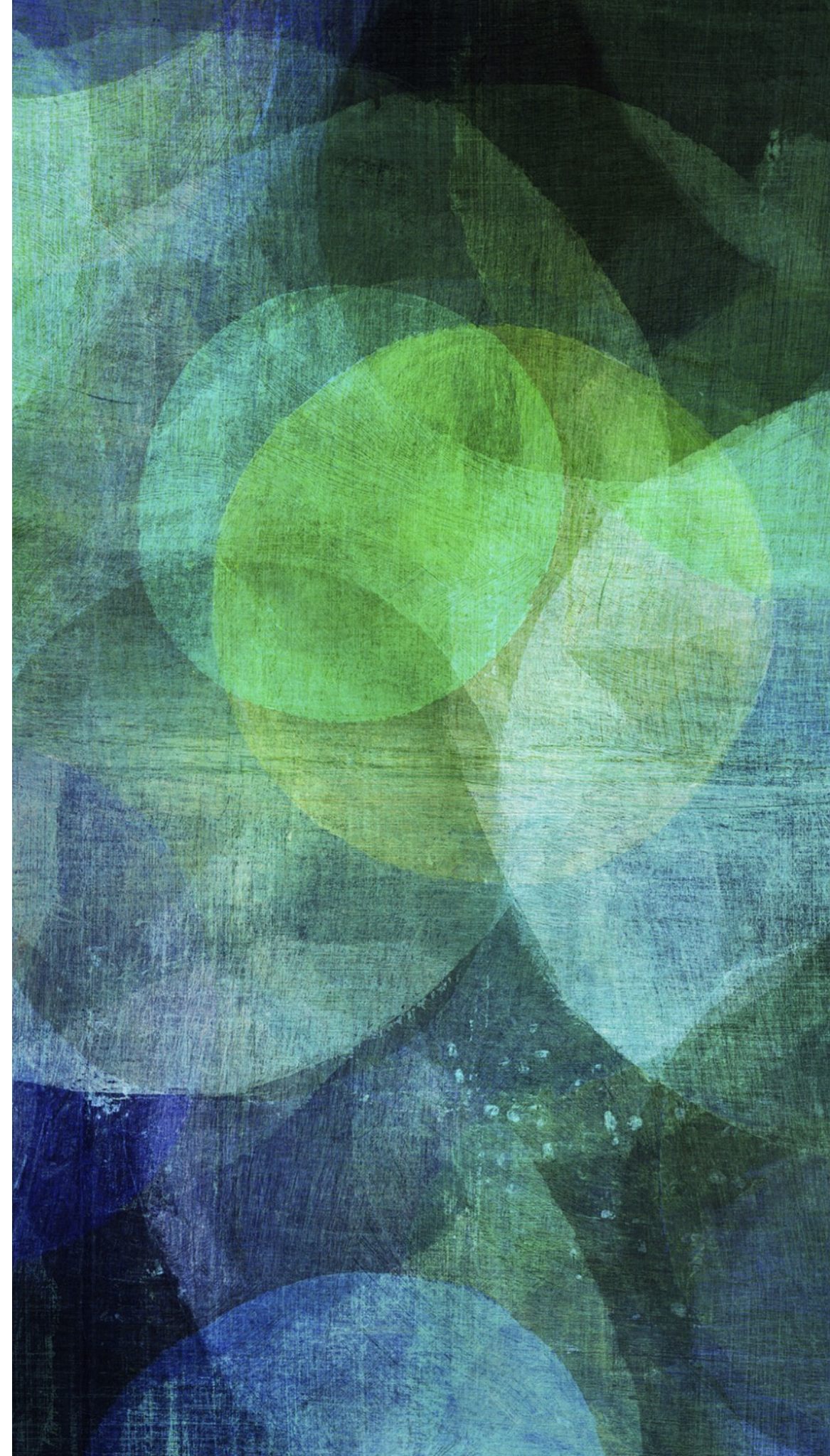
Tag	Purpose
<code>div</code>	Organize and group similar content
<code>h[1,6]</code>	Make titles/headers prominent
<code>p</code>	Organize body text
<code>strong</code>	Bold key words
<code>em</code>	Emphasize (italicize) key words
<code>ul/ol</code>	Bulleted list/Numbered List
<code>a</code>	Link to pages or sections
<code>table</code>	Make a table
<code>img</code>	Image



# CSS PROPERTIES

---

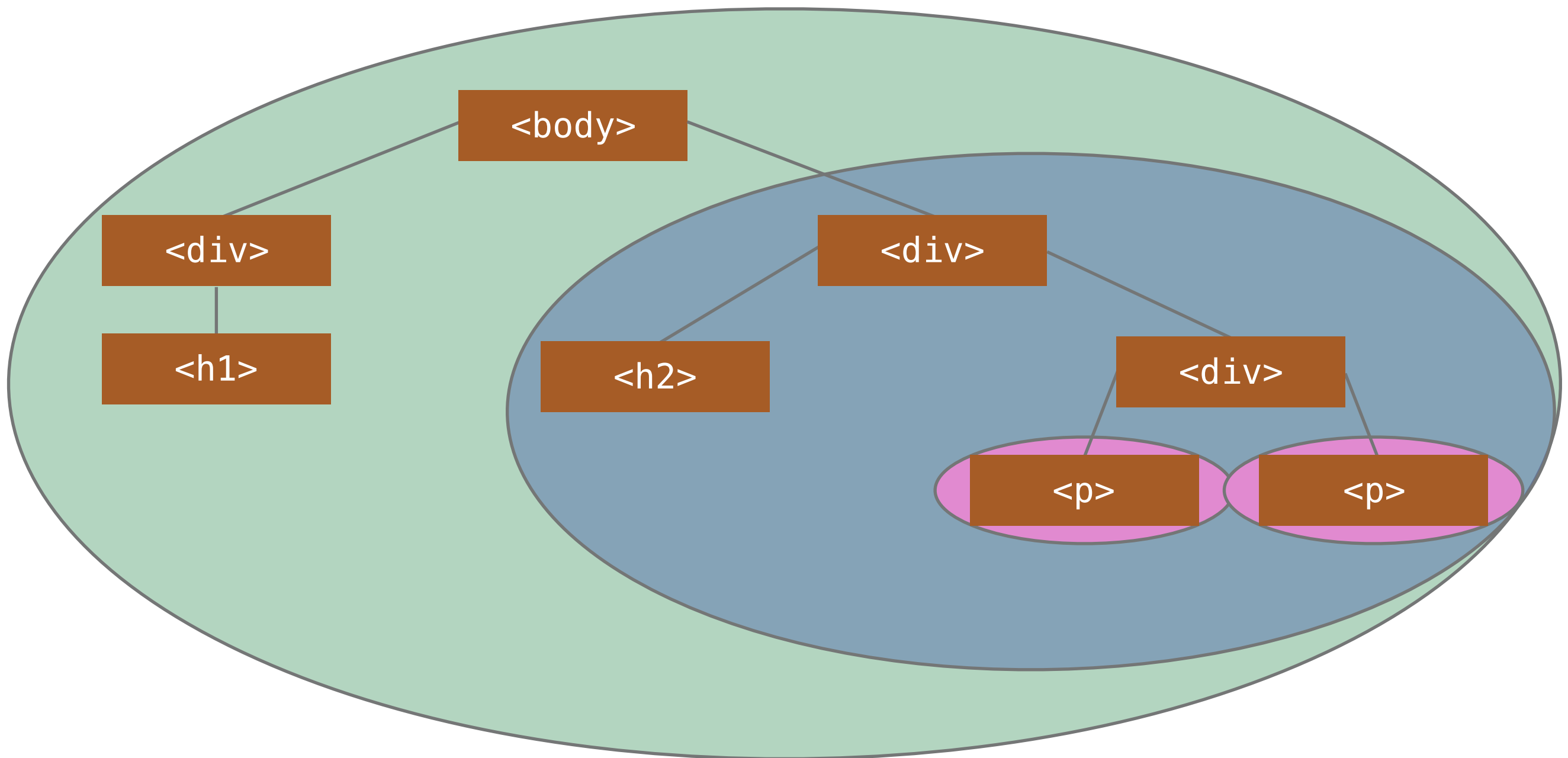
*Making HTML look less  
poopy.*





# C FOR CASCADING

---



# READING CSS CODE

---

## Selector

Defines which part of the DOM tree is affected.

```
1  h1, h2 {  
2      font-family: 'Roboto', sans-serif;  
3      background-color: #E3E4E5;  
4      color: #0F0F0F;  
5      display: block;  
6      padding-right: 1em;  
7  }
```

## Property

Properties control color, fonts, layouts, etc.

## Value

Can be a size, hex color, or pre-defined value



# CSS SELECTORS

---

`div`

You can use any tag (div,h1,p, etc.) as a selector.  
It selects ALL elements of that tag.

`#my-id`

Denoted by octothorpe/hashtag. Selects one specific element by id (this is preferred to "style=" in HTML)

# CSS SELECTORS

---

```
.my-class
```

Denoted by leading period. Selects ALL elements of a given class.

```
<div class="my-class"></div>
```

Adding a class in HTML

# COMPOUND SELECTORS

---

```
selector1 selector2
```

Space Separated: Applies to all elements of **selector2** that are descendants of **selector1**

```
selector1 > selector2
```

> Separated: Applies to all elements of **selector2** that are *direct* children of **selector1**

# COMPOUND SELECTORS

---

```
selector1, selector2
```

Comma Separated: Applies style to all elements satisfying any of the selectors.

```
div.my-class.otherclass
```

Multiple classes: Applies to all elements that have all of the combined classes.



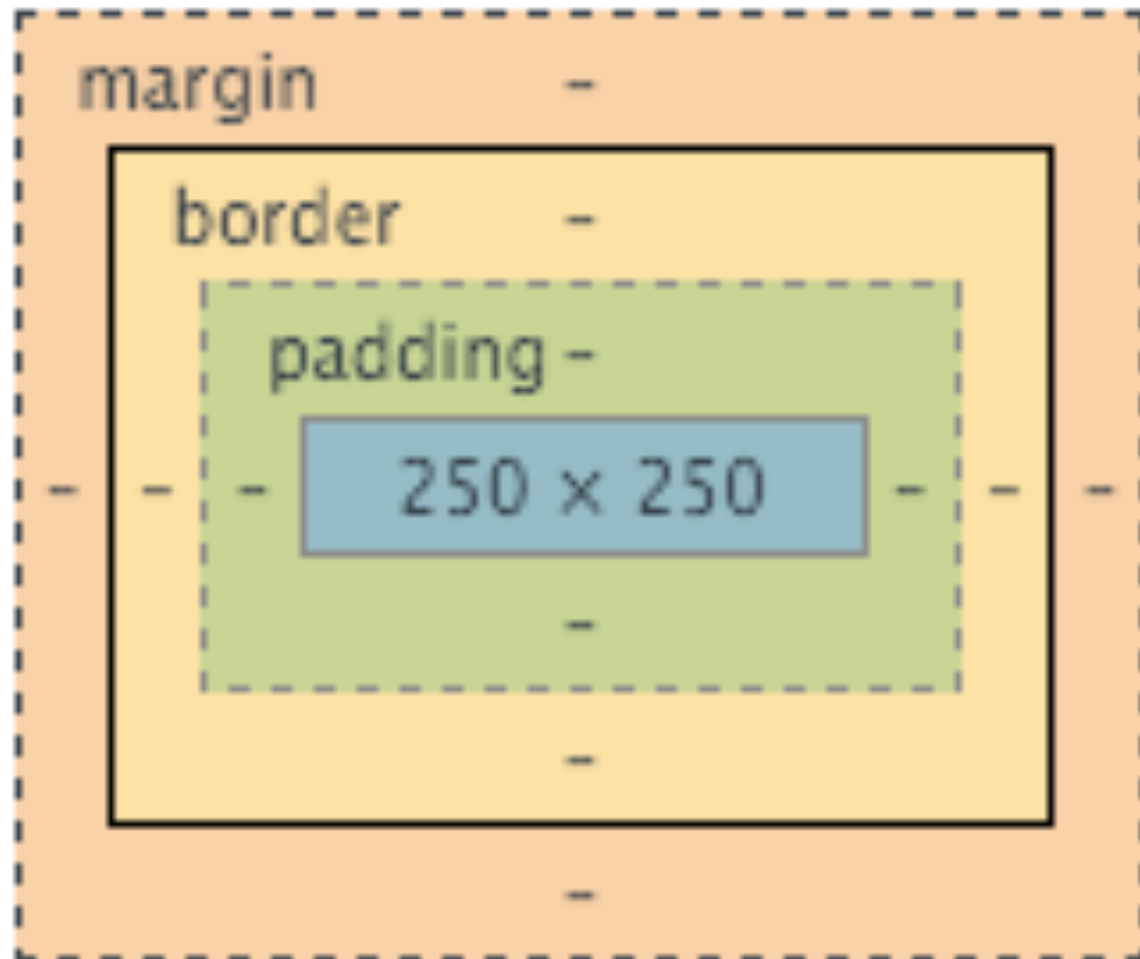
# CSS FONTS

---

- Changing fonts completely changes the look and feel
- Want to make a bad website look good quickly? Just change it to a nicer font!
- Google Fonts
  - Massive Open Source font collection
  - Free for commercial use 👍
  - Loads from their servers, which are definitely faster than whatever you are using

# THE BOX MODEL

.....



- Describes multiple CSS properties in one simple diagram
- Can be used to define spacing and layout for a DOM element
- Editable box model available via Google Developer Tools

# LAYOUT PROPERTIES

---

- **margin** - Amount space between elements at the same level in the DOM tree
- **border** - Defines the line surrounding element
- **padding** - Size of inset space between an object border and its content
- **width/height** - Exactly what you would expect
- These properties take scalar values either in the following formats: "**<num>px**" for pixels, "**<num>%**" for percent of width, "**<num>em**" for size based on regular font size.

# CSS LAYOUT EXAMPLE

.....

width: 60%;



## Harry Q. Bovik

Email: [skrulcik@gmail.com](mailto:skrulcik@gmail.com)

Phone: (123) 321-4567

Harry is the man. You should hire him because he is great for lots of reasons.

Moreover, he's the bees knees.

## Work Experience

### Software Engineer - *Carnegie Watermelon University*

Researching stuff that has to do with **computers**. Somehow related to science. Gotta love that **science**. And those Computers. Was awarded many **awards** for award-worthy accomplishments. [Website](#)

### Software Development Intern - *ACME Software*

Worked on really **important stuff**. More stuff about jobs and crap. Responsibility. **Big Data**. Agile Blah Blah Blah. Buzzwordified description of basic tasks. [Project Page](#)

## Mad Skillz Yo

- Nunchuck Skills
- Bow Hunting Skills
- Computer Hacking Skills
- You know, skills.

## Programming Languages

Language	Proficiency
C	Skilled



# CSS LAYOUT EXAMPLE

---

margin-bottom: 1.5em;

↓

↑

↓

↑

↓

↑

## Harry Q. Bovik

Email: [skrulcik@gmail.com](mailto:skrulcik@gmail.com)

Phone: (123) 321-4567

Harry is the man. You should hire him because he is great for lots of reasons.

Moreover, he's the bees knees.

## Work Experience

**Software Engineer - Carnegie Watermelon University**

Researching stuff that has to do with **computers**. Somehow related to science. Gotta love that **science**. And those Computers. Was awarded many **awards** for award-worthy accomplishments.[Website](#)

**Software Development Intern - ACME Software**

Worked on really **important stuff**. More stuff about jobs and crap. Responsibility. **Big Data**. Agile Blah Blah Blah. Buzzwordified description of basic tasks. [Project Page](#)

## Mad Skillz Yo

- Nunchuck Skills
- Bow Hunting Skills
- Computer Hacking Skills
- You know, skills.

## Programming Languages

Language	Proficiency
C	Skilled

# CSS COLOR

---

```
color: #046380;
```

- CSS supports Hex colors (**#FFFFFF** = white, **#FF0000** = red) and built-in colors like black, teal, and red
- Properties to define color:
  - **color** - specifies foreground (text) color
  - **background-color** - specifies background color
  - **border-color** - used instead of **border** property if you want to specify color of border only

# CSS SHADOW

---

```
box-shadow: 5px 5px 5px #888888;
```

Order of arguments:

1. Horizontal Offset
2. Vertical Offset
3. Blur Radius (Optional)
4. Color (Optional)

The **text-shadow** property has the same options.

# CSS FLOAT PROPERTY

---

```
float: right;
```

- Allows text and other elements to wrap around the floated object (think the images in a news article)
- Float can be **none**, **left**, **right** or **inherit**
- Using floating, and the **display** property, we can put multiple divs on the same line

# CSS DISPLAY PROPERTY

---

```
display: block;
```

- Defines how the browser tries to arrange the elements on the page
- Over 20 possible values, some basics:
  - **inline**: think `<em>` or `<strong>` tag
  - **block**: put in its own vertical area, like `<p>` tag
  - **flex**: designed for dynamic sized blocks to occupy the same vertical area

# RESPONSIVE DESIGN – CSS MEDIA QUERIES

---

```
@media (max-width: 600px) {}
```

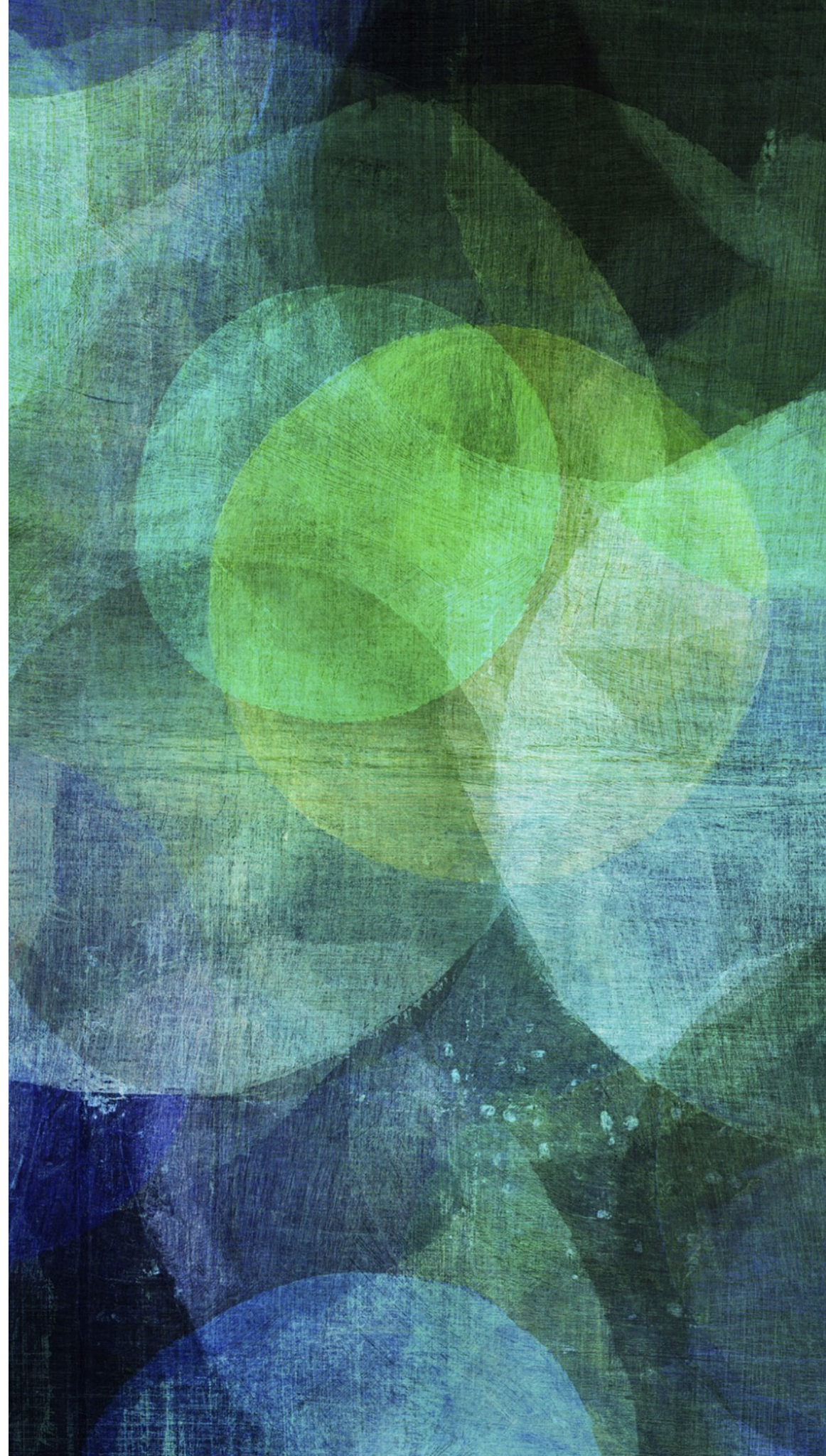
- Media queries allow us to change CSS based on properties of the device and browser
- Often used to make “mobile-optimized” versions of sites
- Common Queries:
  - **max-width, min-width, width**
  - **max-aspect-ratio, min-aspect-ratio, aspect-ratio**



# RESOURCES

---

*Where to go next!*





# HTML TEMPLATE

---

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>WDW Template</title>
5     <link rel="stylesheet" href="style.css">
6   </head>
7   <body>
8     <!-- Content Here -->
9   </body>
10 </html>
11
```

- DOCTYPE tells the browser how to parse the file
- html tags surround all HTML in the document
- The head section is where you set the title, and load stylesheets (more on this later)
- The body section is where the actual content of the page is written
- Download this from the Github repo

# WHERE TO LEARN MORE

---

<https://scottylabs.org/wdw/html/resources>