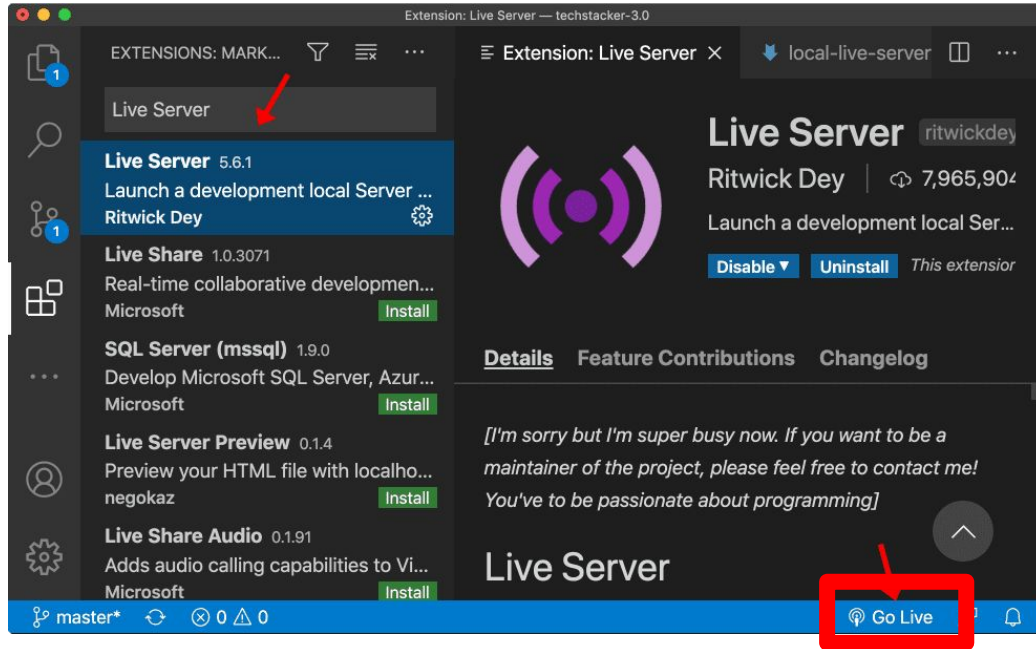


# Setup



1. Download VS Code
2. Add the Live Server extension
3. Clone/copy all the files/code from here:

<https://github.com/zeligsonbrett/Web-Development-Workshop>

Click here to open a live server which will display your html file that is open!!

---

# Front-end Web Development Workshop

Brett Zeligson

---

---

# Topics we'll be Covering Today

- What is front-end web development?
- Front-end vs. backend web development
- HTML
- CSS
- JavaScript

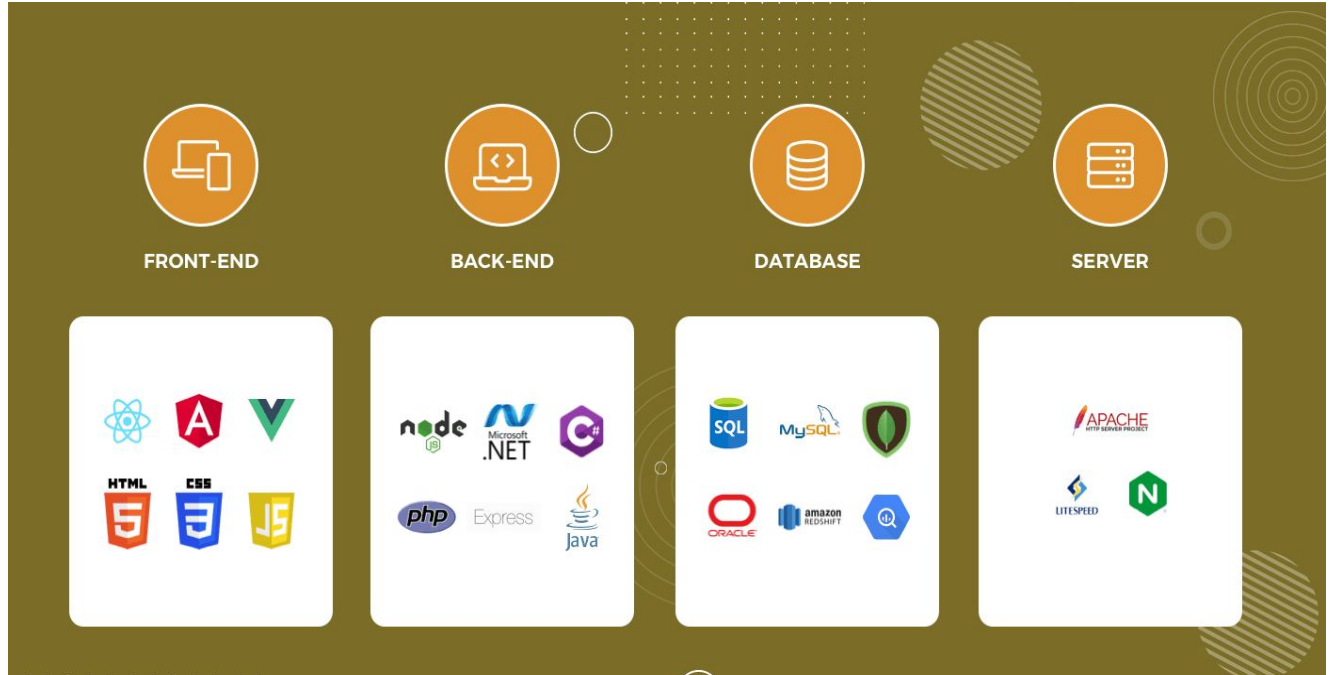
---

# What is Front-end Web Development

- Front-end web development is the development of the **graphical user interface** of a website, through the use of **HTML, CSS, and JavaScript**, so that users can view and interact with that website.
- Aka the stuff the user sees



# Front-end vs. Backend WebDev



—

# HTML

---

# What is HTML?

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="UTF-8">
5      <title>Title goes here</title>
6    </head>
7    <body>
8
9    </body>
10 </html>
```

- **Hypertext Markup Language (HTML)** is the standard markup language for creating web pages and web applications.
- Web browsers receive HTML documents from a web server or from local storage and **render the documents into multimedia web pages**. HTML describes the structure of a web page semantically

# Tags

Tag	Description
<html> ... </html>	Declares the Web page to be written in HTML
<head> ... </head>	Delimits the page's head
<title> ... </title>	Defines the title (not displayed on the page)
<body> ... </body>	Delimits the page's body
<h <i>n</i> > ... </h <i>n</i> >	Delimits a level <i>n</i> heading
<b> ... </b>	Set ... in boldface
<i> ... </i>	Set ... in italics
<center> ... </center>	Center ... on the page horizontally
<ul> ... </ul>	Brackets an unordered (bulleted) list
<ol> ... </ol>	Brackets a numbered list
<li> ... </li>	Brackets an item in an ordered or numbered list
 	Forces a line break here
<p>	Starts a paragraph
<hr>	Inserts a horizontal rule
	Displays an image here
<a href="..."> ... </a>	Defines a hyperlink



—

# CSS

---

# What is CSS?

```
17 span.ts * {
18   cursor: pointer;
19   font-size: .9em;
20   font-family: Menlo, Monaco, Consolas, "Courier New", monospace;
21 }
22 p > span.ts {
23   background-color: rgba(250, 250, 210, 0.29);
24   di
25 }
26   direction
27   display
28   /* An In combination with 'float' and 'position', determines the type of box or ... (i)
29   span.t flex-direction
30   co padding
```

- **Cascading Style Sheets (CSS)** is a style sheet language used for describing the presentation of a document written in a markup language like HTML.
- CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.
- Why is it cascading?

# CSS Selectors

JULIA EVANS  
@bork

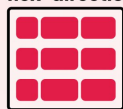
## a few CSS selectors

<code>div</code> matches div elements <div>	<code>#welcome</code> # matches the id <div id="welcome">	<code>div .button</code> match any .button that's a child of a div
<code>.button</code> . matches the class <a class="button">	<code>div.button</code> match divs with class "button" <div class="button">	<code>div &gt; .button</code> match any .button that's a <u>direct</u> child of a div
<code>a:hover</code> matches a elements that the cursor is hovering over	<code>ul li:first-child</code> match the first item of a list. there's last-child too.	<code>a[href^="http"]</code> match links where the href attribute starts with "http" (external links)
<code>:checked</code> matches if a checkbox or radio button is checked	<code>tr:nth-child(odd)</code> match every other row of a table (make stripes!)	<code>div:not(#header)</code> match all divs except the one with id "header"

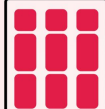
- CSS Selectors are patterns of elements and other terms that tell the browser **which HTML elements should be selected to have the CSS property values** inside the rule applied to them
- A more robust selectors list (scroll down):  
[Adam Marsden/css-cheat-sheet: CSS Cheat Sheet - A reference for CSS goodness. \(github.com\)](#)

## CSS Flexbox

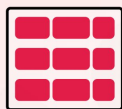
### flex-direction



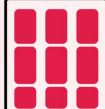
row



column

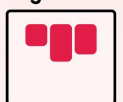


row-reverse



column-reverse

### align-items



flex-start



center



flex-end



stretch

### justify-content



flex-start



center



flex-end



space-between



space-around



space-evenly

### align-content



flex-start



center



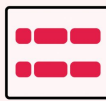
flex-end



stretch



space-between



space-around

# Flexbox

- **Flexbox** is a CSS web layout model that allows responsive elements with a container to be automatically arranged depending on viewport size.
  - Basically helps with making elements in your website responsive to viewport changes.

# CSS Cheat Sheet

brought to you by pxleyes.com

## Selectors

<code>div</code>	all DIV tags
<code>div, span</code>	all DIV tags and all SPAN tags
<code>div span</code>	all SPAN tags inside DIVs
<code>#content</code>	element with ID "content"
<code>.box</code>	all elements with CLASS "box"
<code>ul#box</code>	UL tag with ID "box"
<code>span.box</code>	all SPAN tags with CLASS "box"
<code>*</code>	all elements
<code>#box *</code>	all elements inside #box
<code>a:link, a:active,</code>	links in normal state, in clicked state,
<code>a:visited</code>	and in visited state
<code>a:hover</code>	link with mouse over it
<code>div &gt; span</code>	all SPANs one-level deep in a DIV

## Box Model



## Positioning

<code>position</code>	places elements on screen, e.g. absolute, fixed, relative
<code>float</code>	stacks elements horizontally in a particular direction, e.g. left
<code>top, left, right, bottom</code>	specifies the offsets used in absolute, fixed, and relative positions, e.g. top:10px;left:10px
<code>display</code>	sets how the element is placed in the doc flow, e.g. block, inline, none
<code>z-index</code>	sets the stacking order of elements, e.g. z-index of 1 is below z-index of 2
<code>overflow</code>	sets what happens to content outside of container, e.g. auto, hidden

## Text

<code>font-family</code>	font used, e.g. Helvetica, Arial
<code>font-size</code>	text size, e.g. 60px, 3em
<code>color</code>	text color, e.g. #000, #abcdef
<code>font-weight</code>	how bold the text is, e.g. bold
<code>font-style</code>	what style the text is, e.g. italic
<code>text-decoration</code>	sets a variety of effects on text, e.g. underline, overline, none
<code>text-align</code>	how text is aligned, e.g. center
<code>line-height</code>	spacing between lines, e.g. 2em
<code>letter-spacing</code>	spacing between letters, e.g. 5px
<code>text-indent</code>	indent of the first line, e.g. 2em
<code>text-transform</code>	applies formatting to text, e.g. uppercase, lowercase, capitalize
<code>vertical-align</code>	align relative to baseline, e.g. text-top

## Borders and Lists

<code>border</code>	sets border style for all borders, in the format: border: (solid, dashed, dotted, double) (width) (color), e.g. border: solid 1px #000
<code>border-top</code>	sets border style for a specific border (same property syntax used for padding and margin, e.g. margin-left)
<code>border-bottom</code>	
<code>border-left</code>	
<code>border-right</code>	
<code>list-style-type</code>	sets style of bullets, e.g. square
<code>list-style</code>	sets how text wraps when bulleted, e.g. outside, inside
<code>position</code>	sets an image for a bullet, e.g. list-style-image:url(bullet.png)

## Everything Else

<code>background</code>	sets background of an element, in the format: background: (color) (image) (repeat) (position), e.g. background: #000 url(bg.png) repeat-x top left
<code>cursor</code>	sets shape of cursor, e.g. pointer
<code>outline</code>	a border drawn around an element that doesn't affect the box model
<code>border-collapse</code>	sets how borders within tables behave, e.g. collapse
<code>clear</code>	sets on what side a new line starts in relation to nearby floated elements, e.g. left, right, both

Always write `<!doctype html>` in your files!

# JavaScript

---

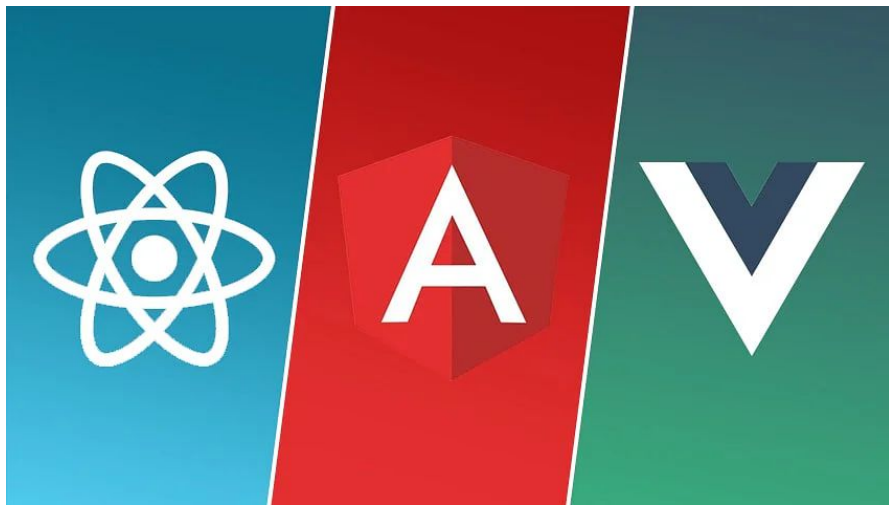
# What is JavaScript?

```
6   });
7   const callAfterSomeTime = (callback, time) => afterSomeTime(time) then(callback);
8
9   callAfterSomeTime(() => console.log('Hello after 1500ms'), 1500);
10
11  const getData = async (url) => fetch(url);
12
13  document
14    .querySelector('#submit')
15    .addEventListener('click', function() {
16      const name = document.querySelector('#name').value;
17
18      // send to backend
19      const user = await fetch(`/users?name=${name}`);
20      const posts = await fetch(`/posts?userId=${user.id}`);
21      const comments = await fetch(`/comments?post=${posts[0].id}`);
```

- **JavaScript (JS)** is a programming language used in both front and backend web development to add functionality to otherwise static pages.

---

# What are React, Angular, and Vue?



- JavaScript libraries for building dynamic user interfaces.
- **React.js:** Enables you to put HTML/CSS into components that you design.
  - Basically it allows you to create custom HTML tags that you can use to modularize and simplify your code.
- Each of these libraries are well-documented so there are resources to learn them.



---

# Questions?

---

---

# Thank you!

Additional Questions: [zeligson@princeton.edu](mailto:zeligson@princeton.edu)

---