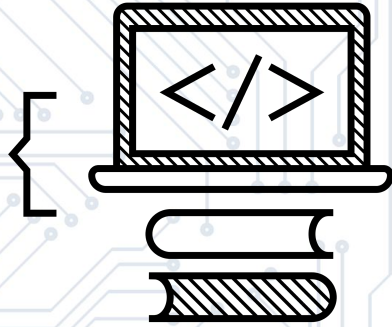


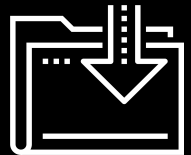
It's Saturday, Let's have some fun!!!





Bootstrap Layouts and Responsive Design

Web Development
Lesson 2.3



Recap

What we've learned

01

HTML + CSS and how to **Git** all the Things

02

How to use **Bootstrap CSS Framework** to create layouts, components, and styling

03

Next Steps:

We will learn to use **view port & media queries** for mobile responsive websites and then we begin working with **JavaScript**

Mobile Responsiveness

Media Queries



Media queries define how CSS styles are applied in relation to the characteristics of the device viewport.



Through media queries, we can change the way a webpage is displayed based on width, height, orientation, and even media type (screen, paper, braille, etc.).



Bootstrap uses media queries extensively to help you create mobile-responsive websites.



But remember! Media queries must be declared last.



Instructor Demonstration

Media Queries

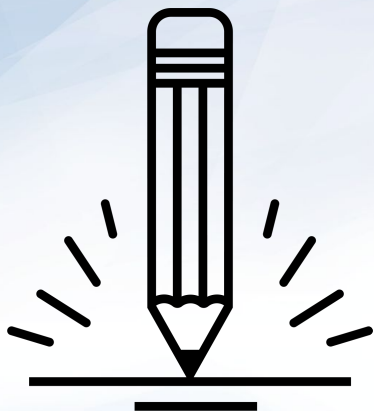
Activity: Explain Media Queries

Explain to your partner what the following CSS code does:

```
@media screen and (max-width: 768px) {  
  body {  
    background-color: #333;  
  }  
  
  .wrapper {  
    width: 600px;  
  }  
}
```

Suggested Time: 2 Minutes





Activity:

Build Media Query

(Follow the steps sent to you via Slack)

Suggested Time:
10 minutes



Viewport



Viewport refers to the display being used to view the website.



By default, mobile devices with small-screen, high-resolution viewports can render content at a typical desktop screen ratio (scaled down at the smaller screen size).



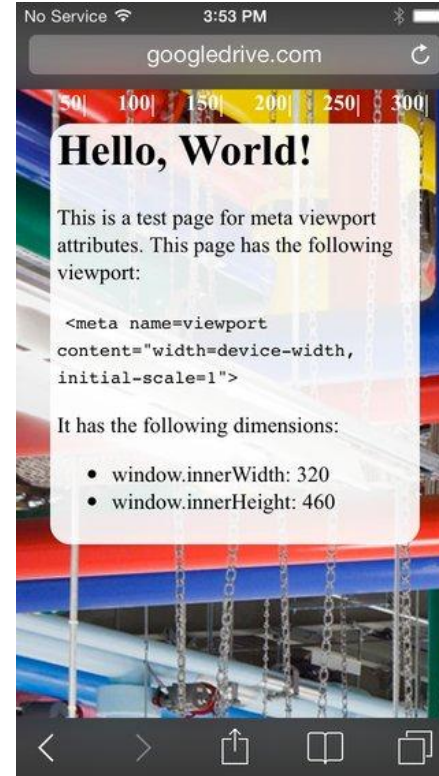
Use a `meta` tag to specify that a webpage should scale its content to the width of its current viewport.

Viewport

Without Viewport content scaling



With Viewport content scaling



Viewport Inclusion

Just include this line in the head

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

And then include a variety of @media queries for various sized screens.

```
@media screen and (max-width: 768px) {
```

```
@media screen and (max-width: 480px) {
```



Activity: Viewport

Spend the next few moments on the site viewportemulator.com and visit your favorite websites using various viewports.

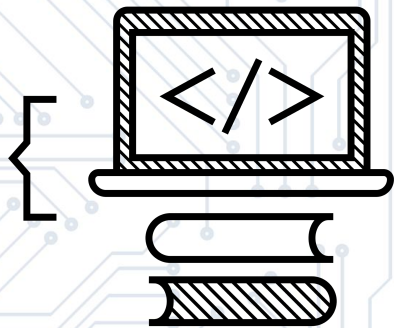
Notice how the sites' displays change.

Suggested Time:
5 minutes



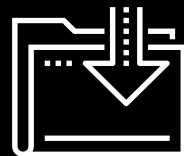
HTML Validator

<https://validator.w3.org/>



The Joys of JavaScript

Web Development
Lesson 3.1



Today's Class

Class Objectives

In today's class, we will introduce:



JavaScript Definitions



JavaScript Basics:



Variables



Logging, alerting, prompting



Arrays



If/else statements

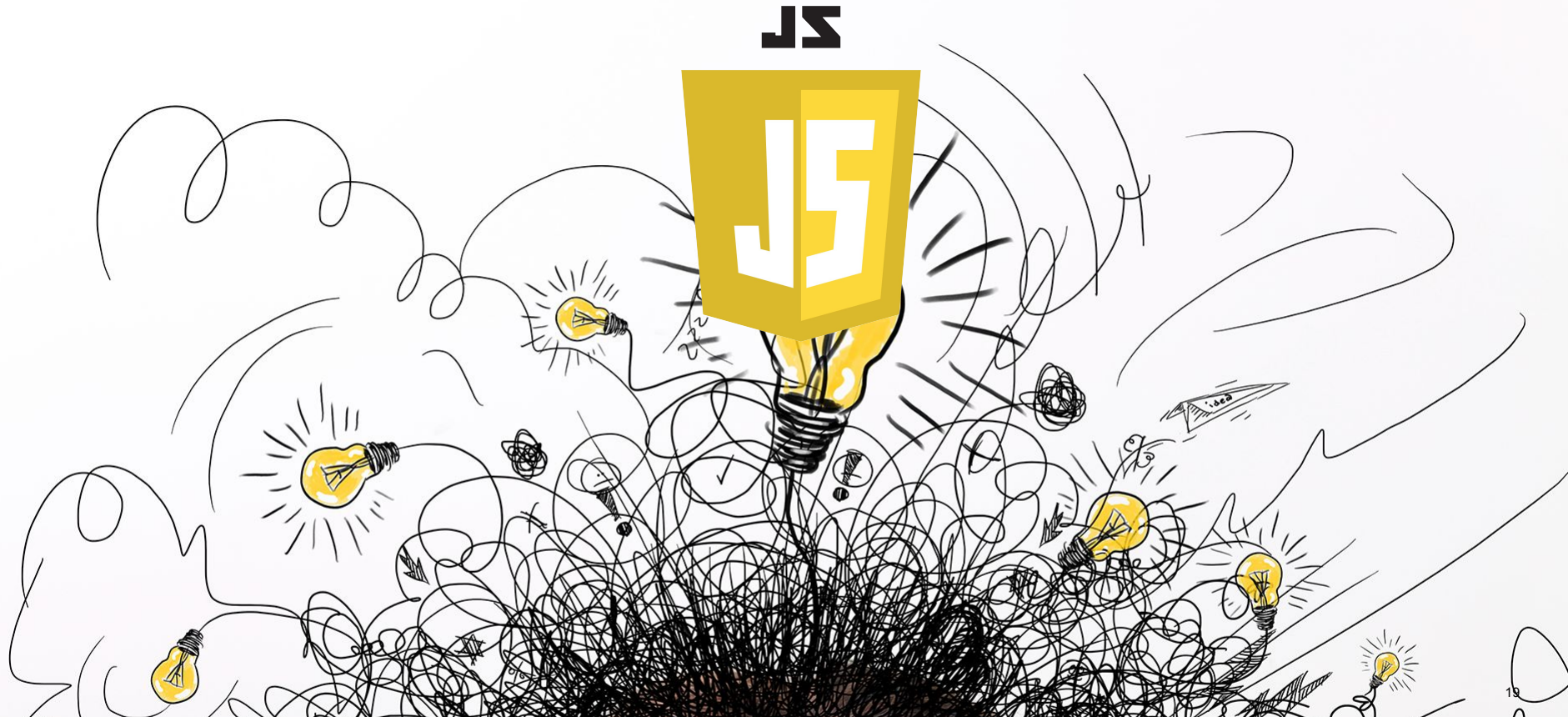
JavaScript

Prepare to become
true coders!



How to Learn JavaScript

Your Brain on JavaScript





Time to Take Notes!



And Stay Organized!

Learning JavaScript

Follow these general tips:



Review classwork immediately.



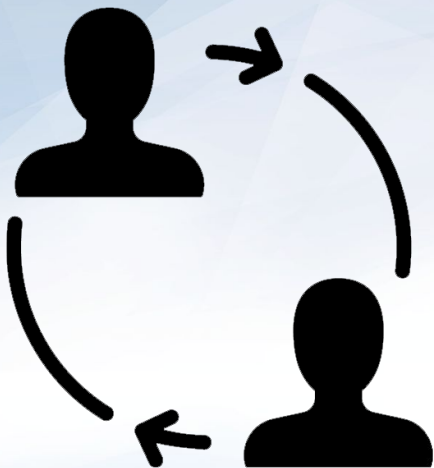
Redo class activities at home.



Come to office hours and keep asking questions.



Do not fear—you will get this!



Partner Activity: Code Dissection

A big part of being a developer is learning on the fly!

Instructions sent via Slack

Suggested Time:
7 minutes



Partner Activity: Code Dissection

01

Download the file sent to you via Slack.

02

Open it in Chrome and observe what happens.

03

With a partner, try to explain how the code connects to the events that happen on the page.



When downloading code from Slack, make sure you choose **Download**.
If you copy and paste directly from Slack, your code will not work!




Suggested Time: 7 minutes



What Is JavaScript?

JavaScript Definition

JavaScript is one of the three fundamental programming languages of the modern web (the others are HTML and CSS).

HTML	CSS	JavaScript
Used to write content.	Used to format content.	Used to create dynamic web applications that take in user inputs, change what's displayed to users, animate elements, and much more.
HTML 	CSS 	JS 

Variables

Variables



The *nouns* of programming



Numbers, strings, Booleans, etc.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Data_structures



Made up of a **name** and a **value**

```
var name = "Snow White";  
var dwarfCount = 7;  
var isSleeping = true;
```

Variable Basics: Syntax

Var keyword	Variable name	Assignment	Value	Termination
<i>var</i>	name	=	"Snow White"	;

Variable Basics: Syntax

Var Keyword	Variable name	Assignment	Value	Termination
<i>var</i>	name	=	"Snow White"	;

Be sure to notice the quotes (""),
which convey that Snow White is a string.



Instructor Demonstration

Variables



Activity: Variables

In this activity, you will fill in the missing JavaScript code to create variables.

Instructions sent via Slack

Suggested Time:
10 minutes



Activity: Variables

01

Using the instructions in the file sent to you, fill in the missing JavaScript code to create variables.

02

When you are done, open the file in Chrome and check the output.

03

If you successfully complete the activity, you will see a series of pop-up windows with text inside.

04

Finally, look at the rest of the code and try to figure out why the text displayed the way it did.

Suggested Time: 10 minutes





Time's Up! Let's Review.

Console Log



Instructor Demonstration

Console Log

Console.log

`console.log` is a quick expression that prints content to the debugger—very useful during development and debugging!

```
var quick = "Fox";
var slow = "Turtle";
var numbers = 121;

// The console.log() method is used to display data in the the browser's console.
// We can log strings, variables, and even equations.

console.log("Teacher");
console.log(quick);
console.log(slow);
console.log(numbers + 15);
```

How do you comfort a **JavaScript bug**?



How do you comfort a **JavaScript bug**?
You **"console"** it!





Activity: Console Log

Instructions sent via Slack

Suggested Time:
7 minutes



Activity: Console Log

01

Using the file sent to you as a guide, modify the code so that it uses `console.log` instead of `alert` to display messages.

02

Then open the file in the browser and open up Chrome Developer Tools -> Console to confirm that the changes worked.

03

With a partner, discuss the difference between `console.log` and `alert`.

```
alert("Welcome: " + name);  
alert("Pizzas cost $5 each");  
alert("Your total is: $" + totalCost);  
alert("Still Hungry: " + isHungry);
```

Suggested Time: 7 minutes





A close-up photograph of a computer keyboard. The central focus is a large, white, rectangular key with rounded corners. On this key, there is a dark blue icon of a coffee cup with three wavy lines above it representing steam. Below the icon, the word "Break" is printed in a dark blue, serif font. The key is set against a background of other keyboard keys, which are slightly out of focus. To the left, a key with double quotation marks is visible. Above the main key, there are keys with forward and backward slashes. To the right, a key with a vertical line and a diagonal slash is visible. The lighting is soft and even, highlighting the texture of the keys.

Break

Alerts, Prompts, Confirms

Alerts, Prompts, Confirms

Alerts, prompts, and confirms create a popup in the browser when run. These are also useful for development and debugging.

```
// Alert
alert("We definitely rock!");

// Confirm
var doYouRock = confirm("The question is, do *you* rock?");

// Prompt
var howMuchRock = prompt("How much do you rock?");
```

This page says:
We definitely rock!

OK

This page says:
The question is, do "you" rock?

☐ Prevent this page from creating additional dialogs.

OK

Cancel

This page says:
How much do you rock?

☐ Prevent this page from creating additional dialogs.

OK

Cancel



Instructor Demonstration

Alerts, Prompts, Confirms



Activity: Alerts

Instructions sent via Slack

Suggested Time:
15 minutes



Activity: Alerts

Write JavaScript code that does the following:

01

Using a `confirm`, ask the user “Do you like ____?” and store their response in a variable.

02

Using a `prompt`, ask the user: “What kind of ____ do you like?” and store their response in a variable.

03

`alert` both variables to the screen.

Suggested Time: 10 minutes





Time's Up! Let's Review.

If/Else Statements



Instructor Demonstration

Conditionals

If/Else Statements Are Critical

Each statement is composed of an if, else-if, or else (keyword), a condition, and the resulting code in curly brackets {}.

```
// If the user likes sushi (confirmSushi === true), we run the following block of code.
if (confirmSushi) {
  alert("You like " + sushiType + "!");
}
// If the user likes ginger tea (confirmGingerTea === true), we run the following block of code.
else if (confirmGingerTea) {
  alert("You like ginger tea!!");
}
// If neither of the previous condition were true, we run the following block of code.
else {
  alert("You don't like sushi or ginger tea.");
}
```



Partner Activity: If/Else Part 1

With a partner you will create a website (from scratch) that asks users if they eat steak.

Suggested Time:
15 minutes



Partner Activity: If/Else Part 1

01

With a partner, create a website (from scratch) that asks users if they eat steak.

02

If they respond with yes, alert the following to the page: *"Here's a Steak Sandwich!"*.

03

If they respond with no alert the following to the page: *"Here's a Tofu Stir-Fry!"*.



Bonus: Ask what the user's birth year is. If they are under 21, alert the following:
"No sake for you!"

Suggested Time: 15 minutes





Time's Up! Let's Review.



Activity: If/Else Part 2

As a class, we will go through and predict what the result of a conditional statement will be.

Instructions sent via Slack.

Suggested Time:
10 minutes



Activity: If/Else Part 2

Do this activity as a class.



Open the file sent to you in your code editor.



As a class, go through and predict what the result of each conditional statement will be (i.e., will the “if” or the “else” be triggered).



Then run the program to check if you are right. Note any that you got wrong and ask about it in class.

Suggested Time: 10 minutes



Arrays

The Zoo Pen

Array Name: zooAnimals

Zebra

Index 0

Rhino

Index 1

Giraffe

Index 2

Owl

Index 3

The Zoo Pen: Coded

Array Name: zooAnimals

Zebra

Index 0

Rhino

Index 1

Giraffe

Index 2

Owl

Index 3

Coded in JavaScript using an array:

```
// Our array of zoo animals.  
var zooAnimals = ["Zebra", "Rhino", "Giraffe", "Owl"];
```

Arrays



Arrays are a type of variable that are *collections*.



These collections can be made up of strings, numbers, Booleans, other arrays, objects ... anything.



Each element of the array is marked by an index. Indexes always start with 0.

```
var nickCharacters = ["Tommy", "Doug", "Oblina"];
```

```
var diceNumbers = [1, 2, 3, 4, 5, 6,];
```

```
var mixedArray = ["Zoo", 12, "Carrot", 3];
```

Arrays: Indices

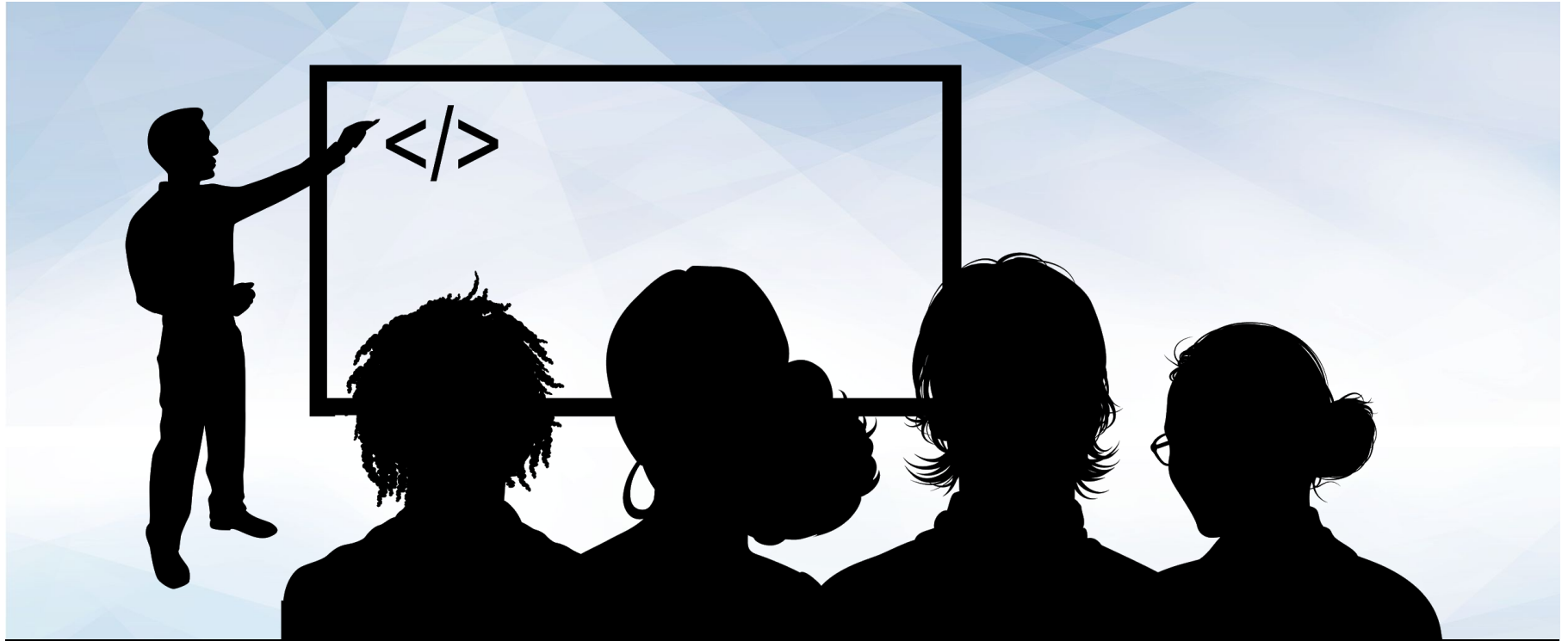


To recover the value at any specific index, include the name of the array with a square bracket `[]` and inside the bracket is the element's index.



You can easily grab the number of elements in the array using the method `array.length`.

```
// Our array of zoo animals.  
var zooAnimals = ["Zebra", "Rhino", "Giraffe", "Owl"];  
  
// Prints 4 to the console because there are 4 items in our zooAnimals array.  
console.log(zooAnimals.length);  
  
// Prints Rhino to the console. Remember, the first item in an array has an index position of 0!  
console.log(zooAnimals[1]);  
  
// Prints undefined...because the last index ("Owl") is 3.  
console.log(zooAnimals[4]);
```



Instructor Demonstration

Arrays

Partner Activity: Code Dissection

With a partner, take a few moments to look over the following code (sent via Slack).

Above each `console.log()`, write a comment predicting what you think the output will be.

Suggested Time: 7 minutes



Recap

What we've learned

01

HTML + CSS and how to **Git** all the Things

02

How to use **Bootstrap CSS Framework** to create layouts, components, and styling

03

Next Steps:

We will learn to use **view port & media queries** for mobile responsive websites and then we begin working with **JavaScript**

Class Objectives

In today's class, we will introduce:



JavaScript Definitions



JavaScript Basics:



Variables



Logging, alerting, prompting



Arrays



If/else statements



Questions?

Homework #3



Challenge: Favorite Band Array

In this challenge, you will create an array of your favorite bands.

Suggested Time:
15 minutes



Challenge: Favorite Band Array

Create a website that accomplishes the following:

01

Create an array of your favorite bands.

02

With a prompt, ask the user's favorite band.

03

If it's one of your favorites, alert: *"YEAH, I LOVE THEM!"*

04

If it's not, alert: *"Nah. They're pretty lame."*




Hint: You will need to research how to use `.indexOf()`

Hint: You will also need to research how to use `.toLowerCase()`

Suggested Time: 15 minutes



A black silhouette of a person standing on a jagged mountain peak, holding a flag aloft. A dashed line representing a path leads up the mountain. The background is a light blue geometric pattern.

Challenge: Code Dissection (Re-examined)

Re-examine the file sent to you at the start of class.
See if you can better understand how it works after
having completed today's class.

Suggested Time:
12 minutes

