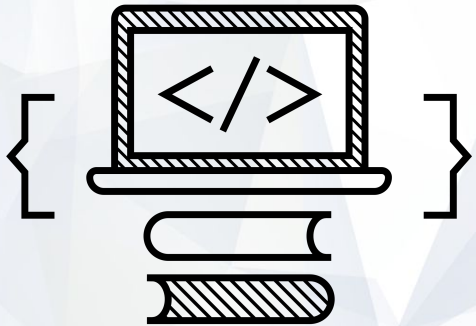
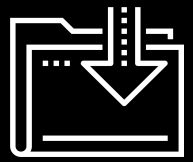
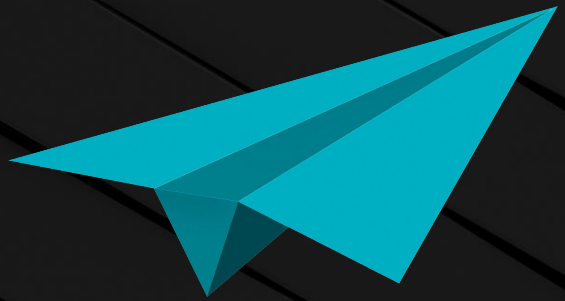


Intro to JavaScript

Skills Bootcamp in Front-End Web Development

Lesson 4.1





Office Hours

30 Minutes


WELCOME

Today's Objectives

By the end of class today, you will:

Gain a preliminary grasp of JavaScript definitions and of basic syntax.

Utilize variables, logging, arrays, and if-else statements to create simple JavaScript applications.

Prepare to become true coders!

JS

How to Learn JavaScript

Your Brain on JavaScript


Take
notes!


Keep organized!

Learning JavaScript

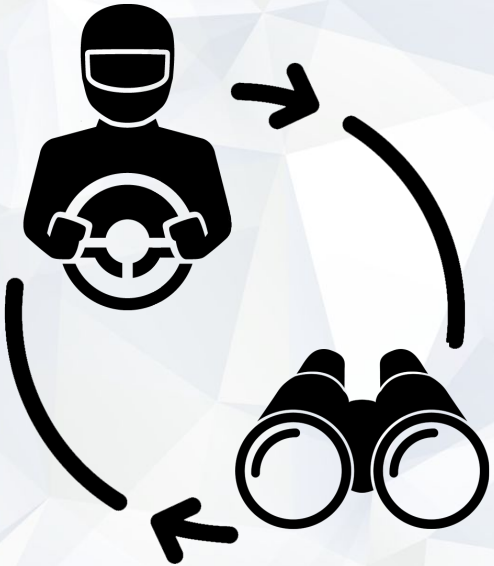
Follow these general tips:

Redo class activities on your own.

Review classwork immediately.

Come to office hours and keep asking questions.

Do not fear—you will get this!



Pair Programming Activity:

Code Dissection

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

Suggested Time:

10 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!

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. 

Variables

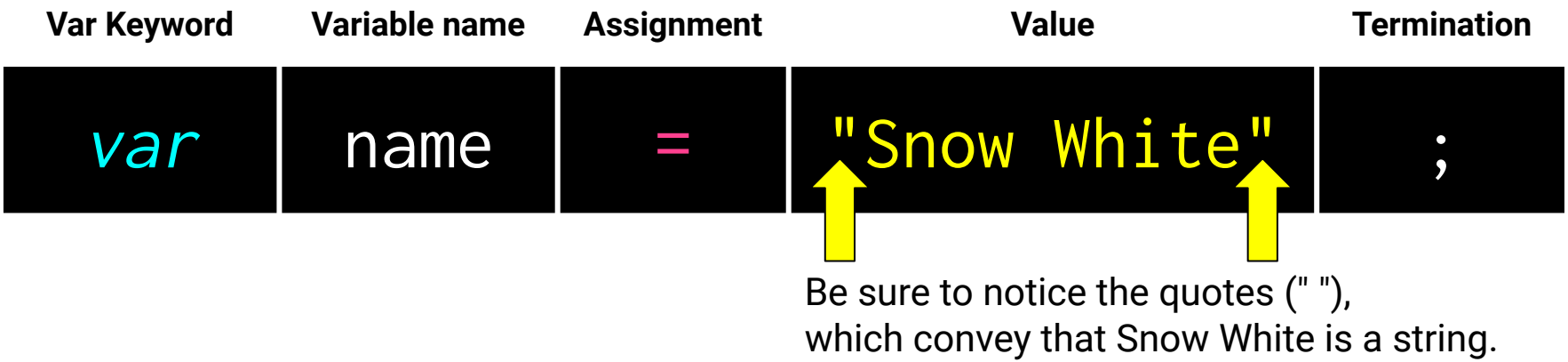
Variables

- The *nouns* of programming
- Numbers, strings, Booleans, etc.
- Made up of a **name** and a **value**

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

Variable Basics: Syntax

Variable Basics: Syntax



Instructor Demonstration

Variables

Activity: Variable

In this activity, you'll 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 completed the activity, you should see a series of pop-up windows with text inside.

04 If you successfully completed the activity, you should see a series of pop-up windows with text inside.

Time's Up! Let's Review.

Questions?

Instructor Demonstration

Console Log

Console.log

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

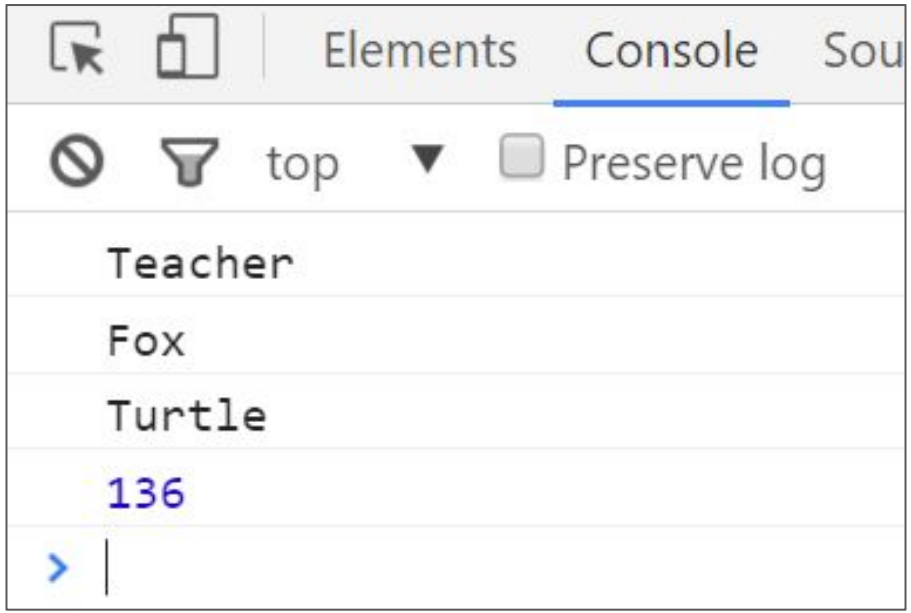
```
var quick = "Fox";
var slow = "Turtle";
var numbers = 121;

// 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);
```

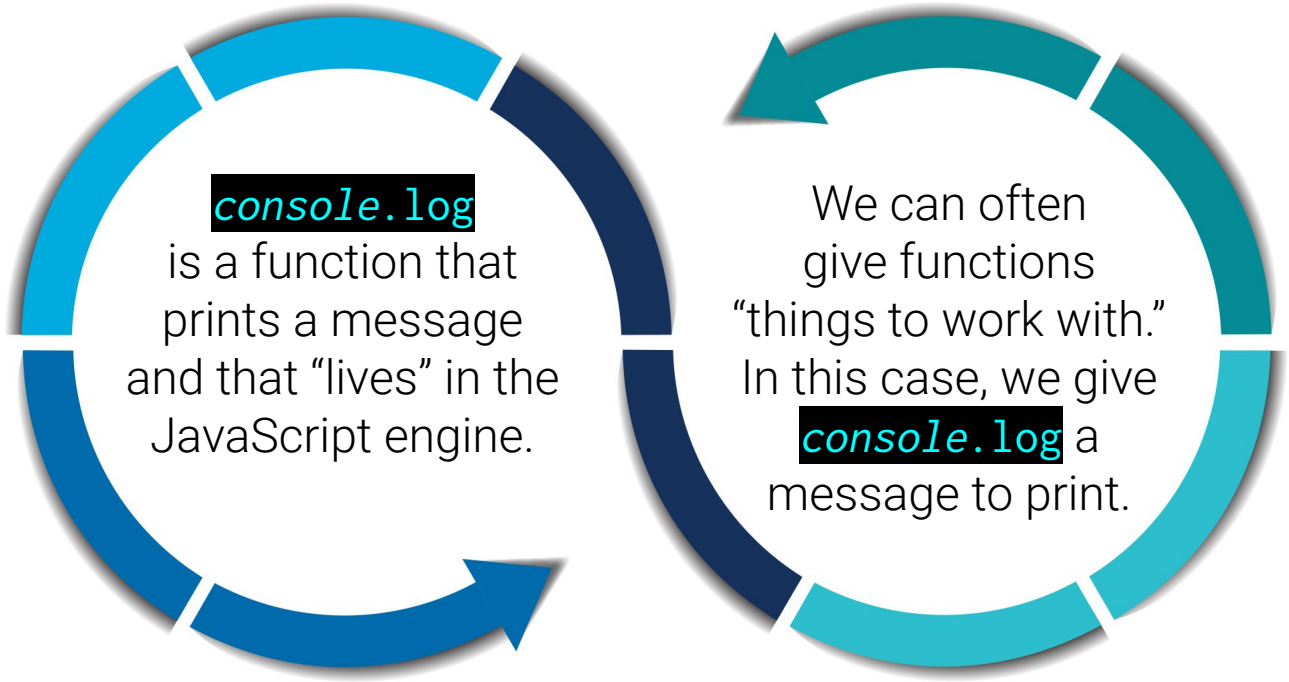
Console.log

In Google Inspector, the line `console.log` outputs to:



A **function** is “code that does a thing and lives somewhere else.”

Console.log


`console.log`
is a function that prints a message and that “lives” in the JavaScript engine.
We can often give functions “things to work with.” In this case, we give `console.log` a message to print.

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

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

You `"console"` it!

Activity: Console Log

In this activity, you'll modify the code so that it uses `console.log` instead of alerts to display messages.

(Instructions sent via Slack)

Suggested Time:

5 Minutes

Activity: Console Log

01 Using the file sent to you as a guide, modify the code so that is uses `console.log` instead of alerts 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);
```

Time's Up! Let's Review.

Questions?


Break

Instructor Demonstration

Alerts, Prompts, Confirms

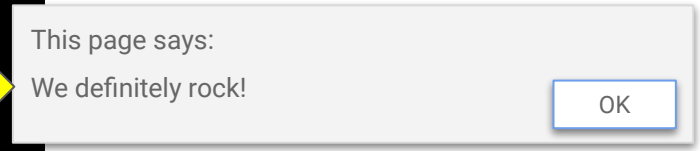
Alerts, Prompts, Confirms

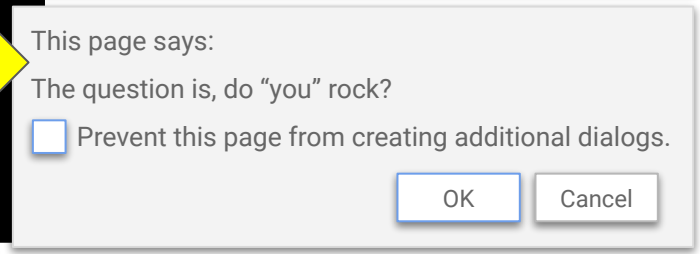
`alerts`, `prompts`, and `confirms` create a pop-up in the browser when run. These are also useful for development and debugging.

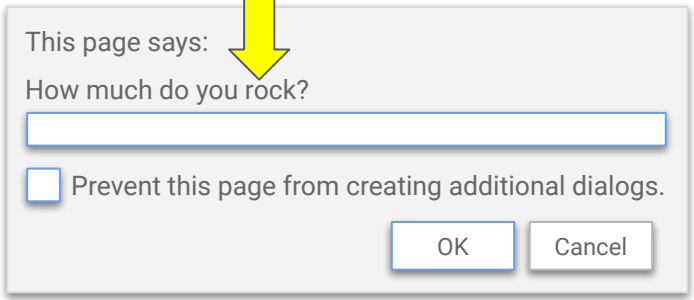
```
// Alert
alert("We definitely rock!");

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

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







Alerts, Prompts, Confirms

`prompt` and `confirm` create data that can be stored in a variable directly.

`alert`, `prompt`, and `confirm` are ***functions***.

Functions that “give you something back” are said to ***return*** something.

`prompt` and `confirm` ***return*** whatever the user enters in the text prompt.

```
// Notice that the confirm and prompt take in variables but the alert doesn't
Var doYouRock = confirm ("The question is, do *you* rock?");
Var howMuchRock = prompt ("How much do you rock?");
```

Activity: Alert

In this activity, you'll write JavaScript code.

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

03 `alert` both variables to the screen.

Time's Up! Let's Review.

Questions?

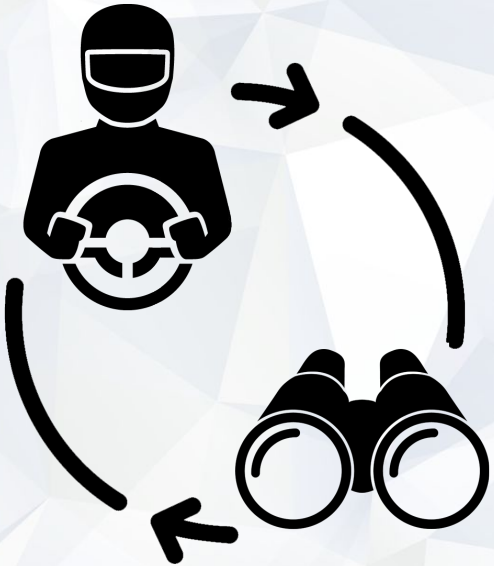
Instructor Demonstration

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.");
}
```



Pair Programming Activity:

`if-else` Part 1

With a partner, you'll 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, lert the following: *“No sake for you!”*

Time's Up! Let's Review.

Questions?

Time to Code

`if-else` Part 2

Suggested Time:

10 Minutes

10 Minutes

Activity: `if-else` Part 2

Do this activity as a class.

01 Open the file sent to you in your code editor.

02 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?).

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

Arrays

The Zoo Pen

Array name: zooAnimals

The Zoo Pen: Coded

Array name: zooAnimals

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", true];
```

Arrays: Indexes

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

Time to Code

Class Code Dissection

Suggested Time:

10 Minutes

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.

Hint: Comments are the grayed lines that begin with //. These lines are ignored by JavaScript, and they allow you to explain your code. Commenting on your code is an extremely useful habit to get into, as it allows other developers to more easily read your code. It will also help you better understand your own applications when you look back at them.

Time's Up! Let's Review.

Questions?

Challenge #1

Favorite Band Array

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 need to research how to use `.toLowerCase()`.

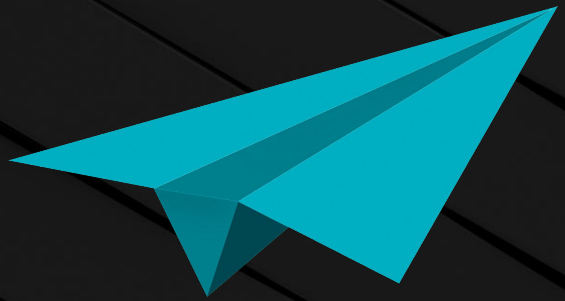
Challenge #2

Code Dissection (Re-Examined)

Questions?


RECAP


The End



Office Hours

30 Minutes