The Joys of JavaScript

The Coding Bootcamp

Admin Items

Homework #2 – Questions?

Two parts to the assignment

1. Take existing Portfolio and apply Media Queries and Viewport to make mobile responsive.

2. Use Bootstrap CSS to recreate the portfolio you built in HW1. Your Bootstrap solution should minimize use of media queries.

Today's Class!

Objectives

In today's class we'll be introducing:

- JavaScript Definitions
- JavaScript Basics:
  - Variables
  - Logging, Alerting, Prompting
  - Arrays
- If/Else Statements
- Array Assignments
- The Concept of For-Loops

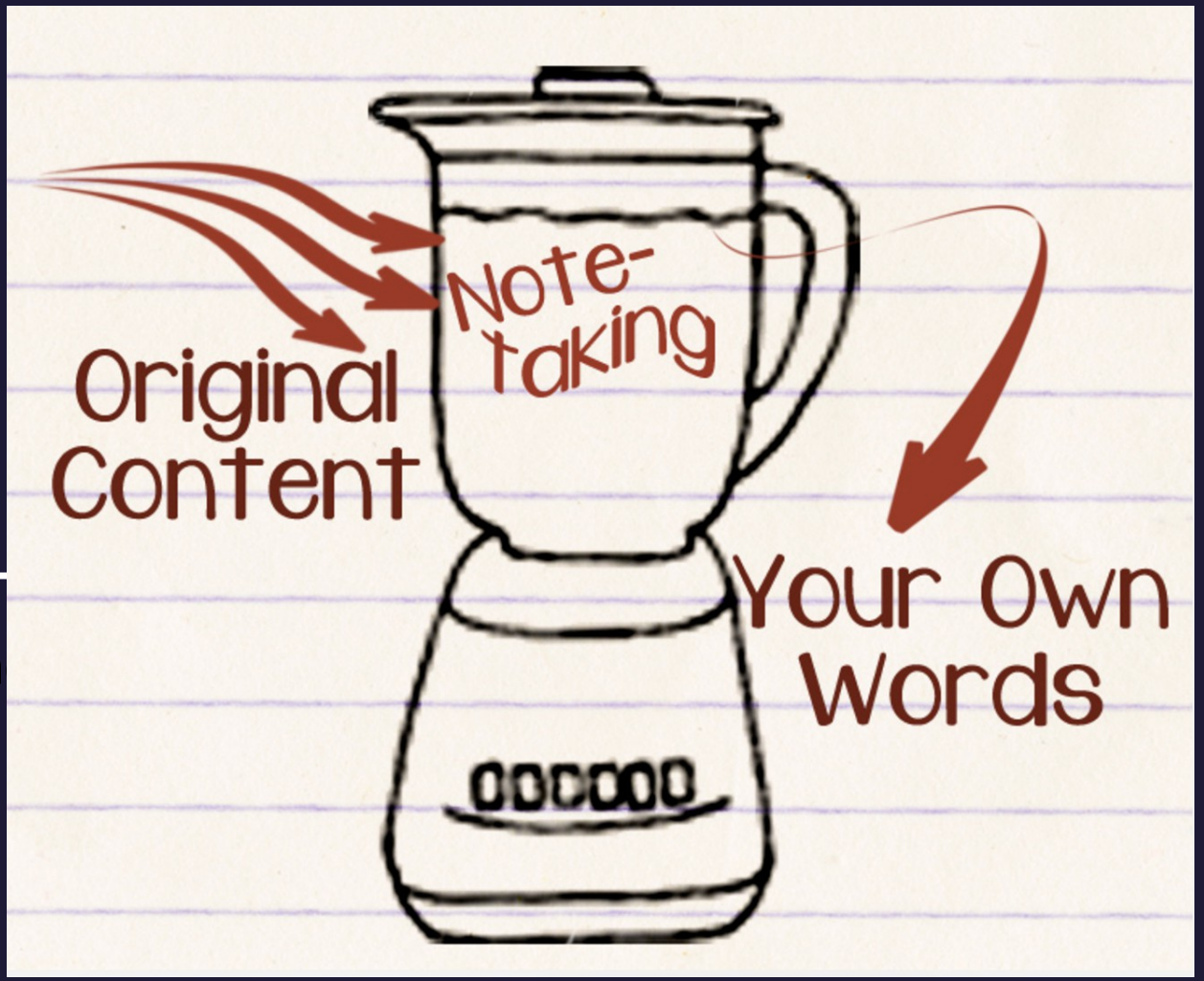
OMG JavaScript!

Prepare to become true coders.

How to Learn JavaScript

Your Brain on JavaScript…

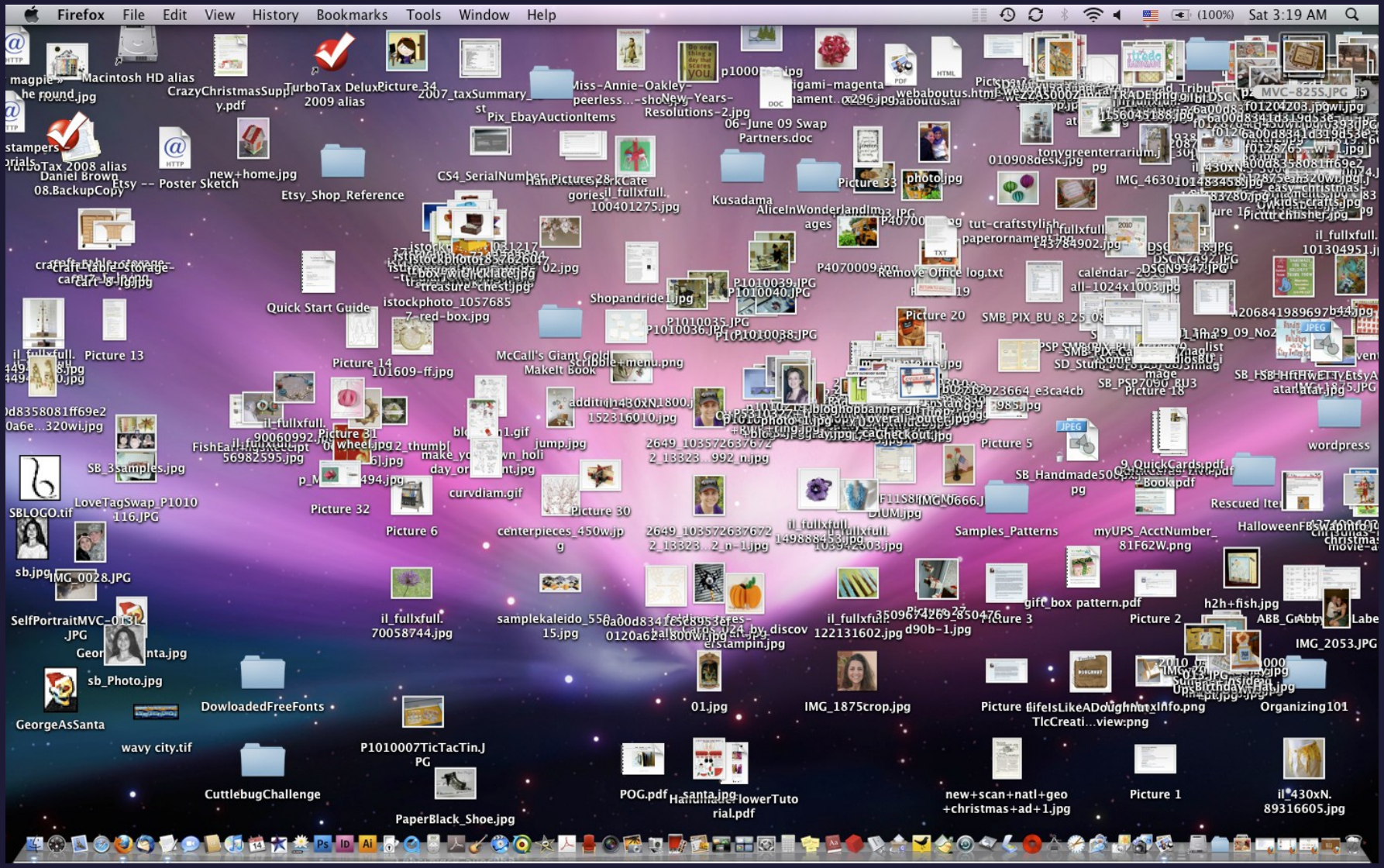
Time to Take Notes…



Th

Taking notes is important for retention and comprehension!

And Keep Organized!!!



Overall Tips

- **Review Immediately:** We'll be building upon these concepts quickly. The firmer your grasp now, the better off you'll be.

- ***Re-do* the exercises in class:** Don't just re-read! Actually spend the time to re-do them from scratch on your own.

- **Get Help:** Come to office hours. Ask conceptual questions. Ask specific questions. Just keep asking questions!

- **Don't be Afraid:** You will get this. It will take time, but you will get this. Just keep at it. Patience will pay off.

Warmup Activity

> YOUR TURN!!

Code Dissection:

1. Download the file sent to you via slack.

2. Open it in Chrome and observe what happens.

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

p.s. *We haven't covered JavaScript before, but a big part of being a developer is learning on the fly!*

MAJOR p.s. *When downloading any code going forward, be sure to hit "Download". If you copy and paste directly from Slack, your code will not work!*

What is JavaScript?

JavaScript Definitions

- **JavaScript** is the third of the three fundamental programming languages of the modern web (along with HTML, CSS).

- **JavaScript** allows developers to create **dynamic** web applications capable of taking in user inputs, changing what's displayed to users, animating elements, and much more.

Variables

Basic Variables

- Variables are the nouns of programming.
- They are “things” (Numbers, Strings, Booleans, etc.).
- They are composed of variable names and values.

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

INSTRUCTOR DEMO!

Variable Assignment

The Coding Bootcamp

Demo Time

Instructor: Demo

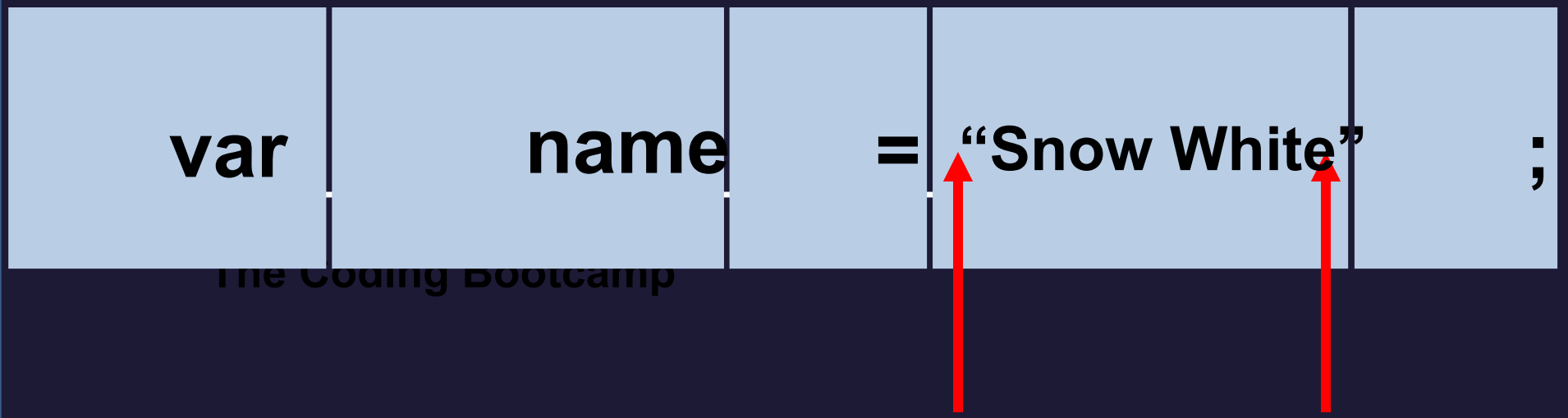
(BasicVariablesDemo | 02-BasicVariablesDemo)

The Coding Bootcamp

Basic Variables (Syntax)

Var Keyword	Variable name	Assignment	Value	Termination
var	name	=	“Snow White”	;

Basic Variables (Syntax)



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

> YOUR TURN!!

Code Creation:

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

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

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

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

Logs, Prints, Alerts

Demo Time

Instructor: Demo

(ConsoleDemoInstructor.html | 04-ConsoleLogDemo)

The Coding Bootcamp

Console.log

- console.log is a quick expression used to print content to the debugger.

- It is a very useful tool to use during development and debugging.

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

// The console.log() method is used to display data in 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);
```

Sad Little Bug…

Hey Class!

How do you comfort a JavaScript bug?

Hey Class!

How do you comfort a JavaScript bug?

The Coding Bo

You “console” it.

Sad Little Bug…

Don’t worry!

*It was a **hilarious** joke… that will make sense in a few weeks.*

The Coding Bootcamp

> YOUR TURN!!

Code Creation:

1. Using the file sent to you as a guide, modify the code so that is uses console.log instead of alerts to display messages.

2. Then open the file in the browser and open up chrome Developer tools -> Console to confirm the changes worked.

3. With a partner, discuss the different between using alert.

```
alert("Welcome: " + name);

alert("Pizzas cost $5 each");

alert("Your total is: $" + totalCost);

alert("Still Hungry: " + isHungry);
```

INSTRUCTOR DEMO!

Alerts, Prompts, Confirms

The Coding Bootcamp

Demo Time

Instructor: Demo

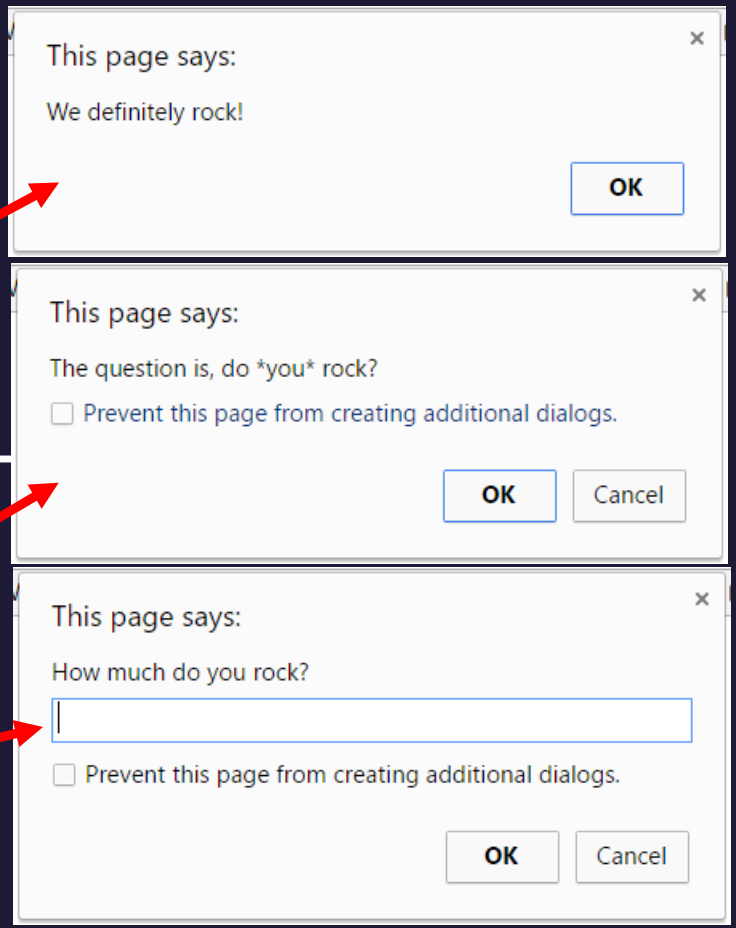
(PromptDemo.html | 06-PromptDemo)

The Coding Bootcamp

Alerts, Prompts, Confirms

- Alerts, Confirms, and Prompts will create a popup box 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?");
```



> YOUR TURN!!

Code Creation:

Write JavaScript code that does the following:

- Using a confirm, ask the user: “Do you like _____?” and store their response in a variable.
- Using a prompt, ask the user: “What kind of _____? do you like?” and store their response in a variable.
- Alert both variables to the screen.

Document Write

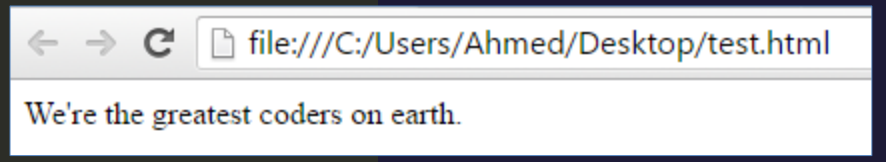
Writing to HTML

- We can also use JavaScript to directly write to the HTML page itself using **document.write( ).**

- Later we will go over *much* more advanced approaches for writing HTML using JavaScript and jQuery.

```
1  <!DOCTYPE html>
2  <html lang="en-us">
3    <head>
4      <meta charset="UTF-8">
5      <title>Document Write</title>
6    </head>
7    <body>
8
9      <script type="text/javascript">
10
11       document.write("We're the greatest coders on earth.");
12
13     </script>
14
15   </body>
16 </html>
```

Test.html (chro



Test.html (sublime)

If/Else Statements

Demo Time

Instructor: Demo

(conditionaldemo.html | 08-ConditionalDemo)

The Coding Bootcamp

If/Else Statements

- 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 {
  document.write("You don't like sushi or ginger tea.");
}
```

> YOUR TURN!!

Code Creation:

- Create a website (from scratch) that asks users if they eat steak.
- If they respond with “yes”, write the following to the page: “Here’s a Steak Sandwich!”.
- If they respond with “no”, write 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!”
- **Hint:** You will need to use document.write( ) from the last activity.

> YOUR TURN!!

Code Dissection:

1. Open the file sent to you in Sublime.

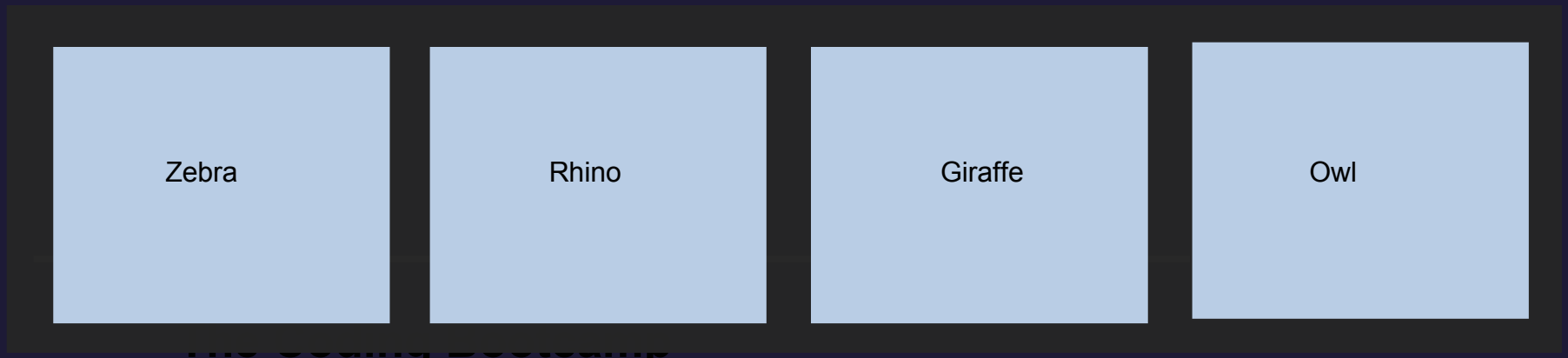
2. With a partner, go through and predict what the result of each "conditional" statement will be (i.e. will the "if" or the "else" be triggered).

3. 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



Index 0 Index 1 Index 2 Index 3

The Zoo Pen... Coded

Array Name: zooAnimals

The Coding Bootcamp

Coded in JavaScript using an Array

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

Basic Arrays

- Arrays 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];
```

Basic Arrays Indices

- To recover the value at any specific index you 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]);
```

Demo Time

Instructor: Demo

(ArraysDemo.html | 11-ArraysDemo)

The Coding Bootcamp

> YOUR TURN!!

Class Code Dissection:

1. With a partner, take a few moments to look over the following code.

2. Above each console.log() write a comment “predicting” what you think the output will be.

Challenge Activity?

> YOUR TURN!!

Code Creation (Challenge):

- Create a website that accomplishes the following:

1. Create an array of your favorite bands.
2. With a prompt, ask the user's favorite band.
3. If it's one of your favorites, alert: "YEAH I LOVE THEM!".
4. If it's not, alert: "Nah. They're pretty lame.".
5. **Hint:** You will need to research how to use .indexOf()
6. **Hint:** You will need to research how to use .toLowerCase()

Code Dissection: Basic JS

1. Re-examine the file sent to you during yesterday's class.

2. See if you can better understand how it works – after having gone through today's class.

3. Prepare to share once the time is up.

Activity: 15-CoolPeopleArray | **Suggested Time:** 5 min

Code Creation: Array Logging (If Needed)

1. Follow the instructions provided in the file to console.log each of the names in the “coolPeople” variable.

2. Hint: You should be repeating the same line 6 times.

3. Be prepared to share once time is up.

Activity: 16-ArraySetting | **Suggested Time:** 7 min

Code Creation: Array Setting

1. Follow the instructions in the file provided to convert each item in the array to lower case.

2. Make sure to only add in lines of code where instructed.

3. **Hint:** You will need to use the method .toLowerCase(). Research if you don't remember how to use it.

4. Be prepared to share once time is up.

For Loops

Back to The Zoo Pen

Array Name: zooAnimals

The Coding Bootcamp

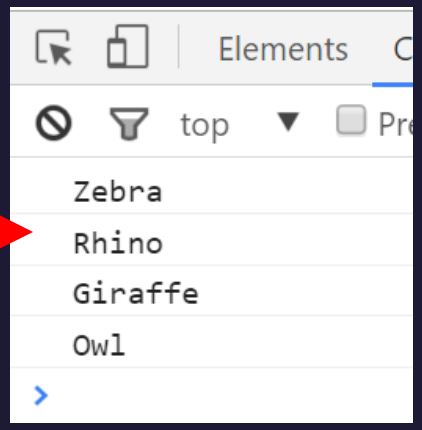
```
// Our array of zoo animals.
var zooAnimals = ["Zebra", "Rhino", "Giraffe", "Owl"];
```

Back to The Zoo Pen (Logging)

Array Name: zooAnimals

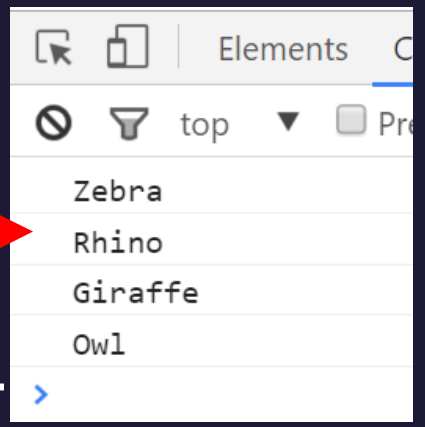
The Coding Bootcamp

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

console.log(zooAnimals[0]);
console.log(zooAnimals[1]);
console.log(zooAnimals[2]);
console.log(zooAnimals[3]);
```



Don’t Repeat Yourself (DRY)

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

console.log(zooAnimals[0]);
console.log(zooAnimals[1]);
console.log(zooAnimals[2]);
console.log(zooAnimals[3]);
```



The Coding Bootcamp

Repeated Code!

Let’s be more efficient

Activity: 17-MyFirstLoop | **Suggested Time:** 5 min

Code Creation: For Loop Dissection

1. With a partner, spend a few moments trying to dissect the code sent to you.

2. Try to explain to one another what is happening with each line of code.

3. Feel free to do research if you are stumped. As a hint, look into the phrase: “For-Loop”.

4. Be prepared to share when time is up.

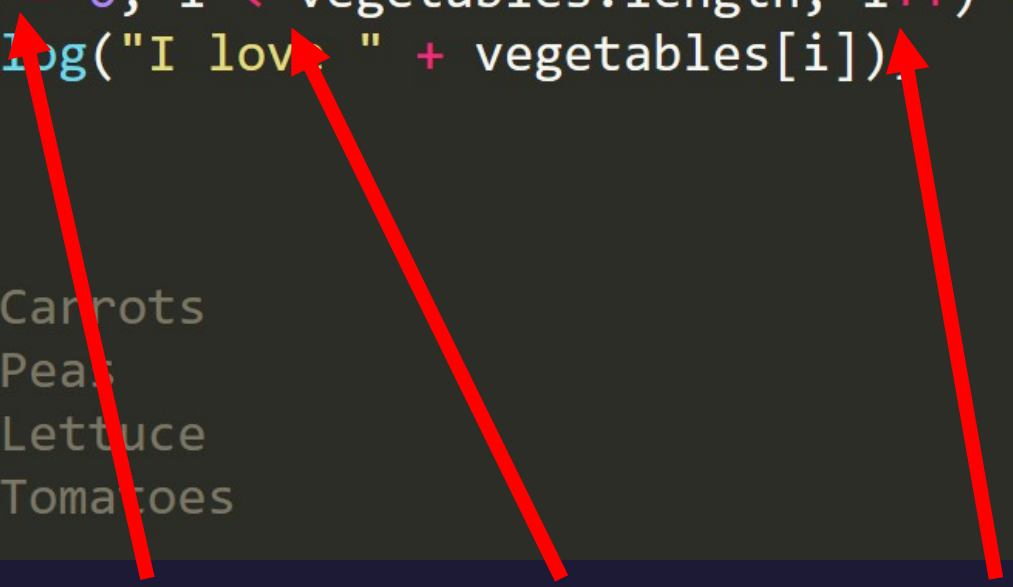
Enter the For-Loop

- For loops are critical in programming.
- We use for loops to run repeated blocks of code over a set period.
- Each for loop is composed of a:
  - Variable declaration or counter (iterator)
  - A loop condition
  - An iteration (addition)

```
// Start with an Array.
var vegetables = ["Carrots", "Peas", "Lettuce", "Tomatoes"];

// Loops through each index of the Array.
for (var i = 0; i < vegetables.length; i++) {
  console.log("I love " + vegetables[i]);
}
```

Enter the For-Loop

```
// Start with an Array.
var vegetables = ["Carrots", "Peas", "Lettuce", "Tomatoes"];

// Loops through each index of the Array.
for (var i = 0; i < vegetables.length; i++) {
  console.log("I love " + vegetables[i]);
}

// Logs:
// I love Carrots
// I love Peas
// I love Lettuce
// I love Tomatoes
```



Iterator. *Condition.* *Increment.*

Enter the For-Loop

```
// Start with an Array.
var vegetables = ["Carrots", "Peas", "Lettuce", "Tomatoes"];

// Loops through each index of the Array.
for (var i = 0; i < vegetables.length; i++) {
  console.log("I love " + vegetables[i]);
}

// Logs:
// I love Carrots
// I love Peas
// I love Lettuce
// I love Tomatoes
```

Code between the { } gets repeated each time the iterator is smaller than the condition. *(i.e. in this case i < 4)*

Enter the For-Loop

```
// Start with an Array.
var vegetables = ["Carrots", "Peas", "Lettuce", "Tomatoes"];

// Loops through each index of the Array.
for (var i = 0; i < vegetables.length; i++) {
  console.log("I love " + vegetables[i]);
}

// Logs:
// I love Carrots
// I love Peas
// I love Lettuce
// I love Tomatoes
```

Running the code "loops" through and prints each element in the array.

Run That Loop

```
// Start with an Array.
var vegetables = ["Carrots", "Peas", "Lettuce", "Tomatoes"];

// Loops through each index of the Array.
for (var i = 0; i < vegetables.length; i++) {
  console.log("I love " + vegetables[i]);
}
```

When i = 0 … console.log("I love Carrots")

The Co ng Bootcamp

Run That Loop

```
// Start with an Array.
var vegetables = ["Carrots", "Peas", "Lettuce", "Tomatoes"];

// Loops through each index of the Array.
for (var i = 0; i < vegetables.length; i++) {
  console.log("I love " + vegetables[i]);
}
```

When i = 1 … console.log("I love Peas")

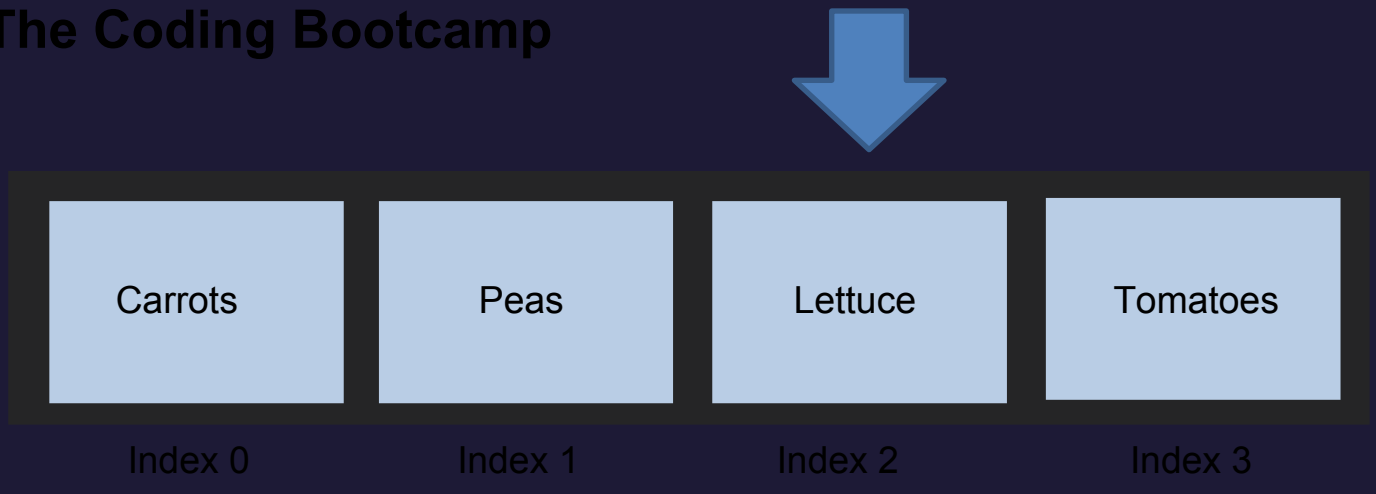
The Coding Bootcar

Run That Loop

```
// Start with an Array.
var vegetables = ["Carrots", "Peas", "Lettuce", "Tomatoes"];

// Loops through each index of the Array.
for (var i = 0; i < vegetables.length; i++) {
  console.log("I love " + vegetables[i]);
}
```

When i = 2 … console.log("I love Lettuce")

The Coding Bootcamp

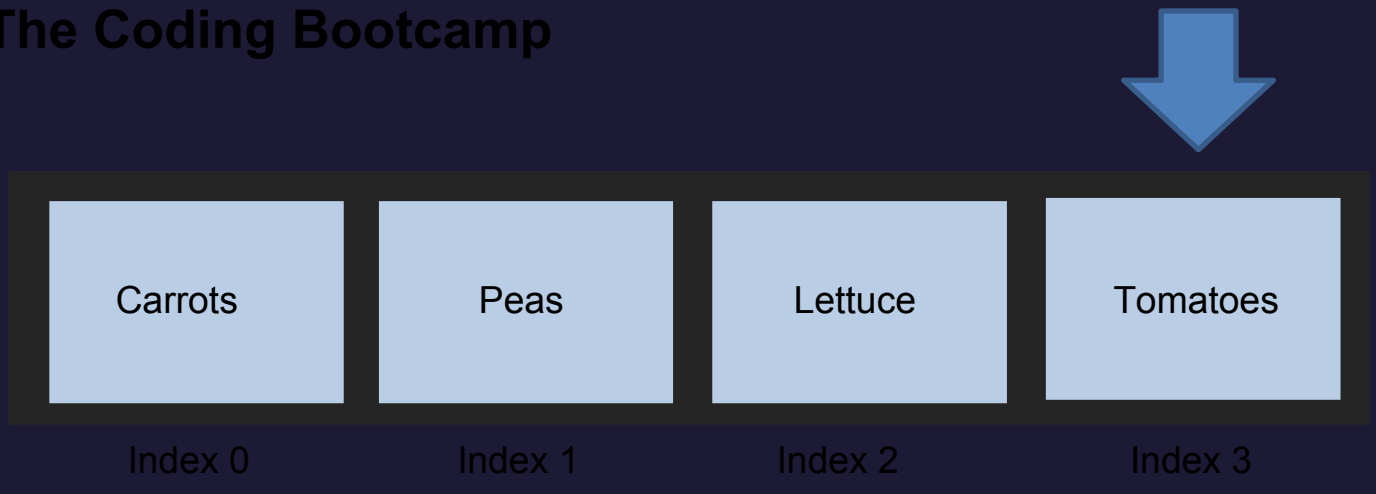


Run That Loop

```
// Start with an Array.
var vegetables = ["Carrots", "Peas", "Lettuce", "Tomatoes"];

// Loops through each index of the Array.
for (var i = 0; i < vegetables.length; i++) {
  console.log("I love " + vegetables[i]);
}
```

When i = 3 … console.log("I love Tomatoes")

The Coding Bootcamp



Code Creation: For-Loop Zoo

1. Spend a few moments, re-writing the code below using a for-loop.

2. If you need help, use the code from the previous example as a guide.

3. Then try to explain to the person next to you how your code works.

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

console.log(zooAnimals[0]);
console.log(zooAnimals[1]);
console.log(zooAnimals[2]);
console.log(zooAnimals[3]);
```

Code Creation: Hard Loop (Time Permitting)

- Starting from scratch, write code that loops through the following array:

```
// This is our starting myFarm array.
var myFarm = ["chickens", "pigs", "cows", "horses", "ostriches"];
```

- And console.logs the name of each animal on the farm.

- Then using the .charAt() method (research it) check if the first letter in the animal's name begins with a "c" or "o". If it does, create an alert saying: "Starts with c or an o!"

Homework #3

Questions?

> YOUR TURN!!

Code Dissection (Re-examined, Time-permitting):

1. Re-examine the file sent to you at the start of class.

2. See if you can better understand how it works – after having gone through today's class.