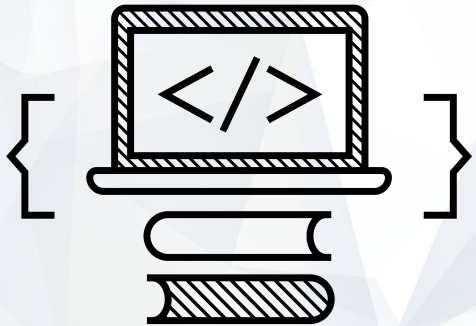
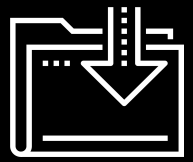
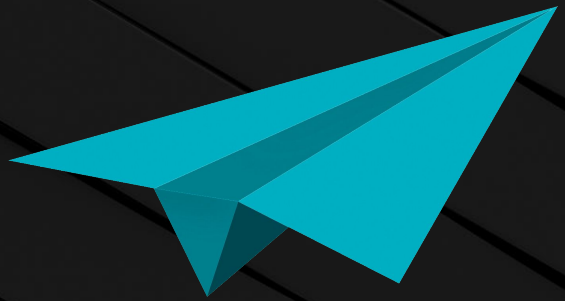


Arrays and Iteration

Skills Bootcamp in Front-End Web Development

Lesson 4.2





Office Hours

30 Minutes


WELCOME

Today's Objectives

By the end of class today, you will:

- Define the syntax and uses of a `for` loop.
- Demonstrate how to use a `for` loop.
- Utilize conditional logic within their functions.
- Construct a rock-paper-scissors game utilizing `for` loops, arrays, and conditional logic.
- Understand the use and syntax of JavaScript's `for` loops.
- Use `for` loops, arrays, and conditional logic to create a rock-paper-scissors game.


Basics
Recap

What is JavaScript?
And what is it used for?

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

JavaScript Definitions

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.

What is a variable?
And how do we declare one?

Variable Basics

Variables are the “nouns” of programming.

They are “things” (numbers, strings, Booleans, etc.).

A variable is composed of a variable name and a value.

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

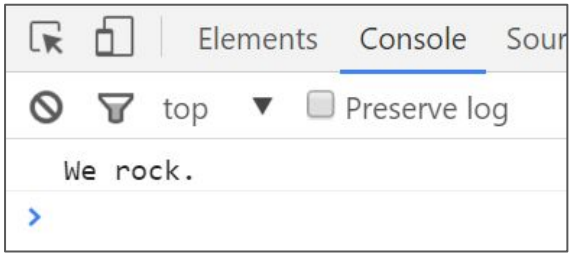
What is meant by `console.log`?
And how does it differ from an alert, prompt, or confirm?

Console.log vs. JavaScript Pop-Up Boxes

`console.log` displays discreetly to the debugger.

`alert` displays a pop-up message to the user.

```
console.log("We rock.");
```



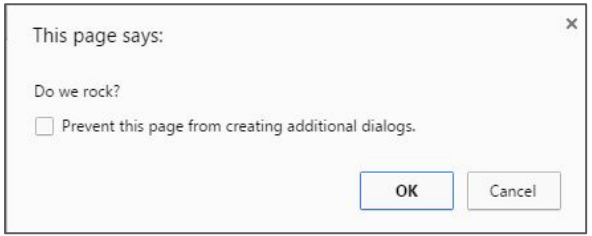

```
alert("We Rock.");
```

Console.log vs. JavaScript Pop-Up Boxes

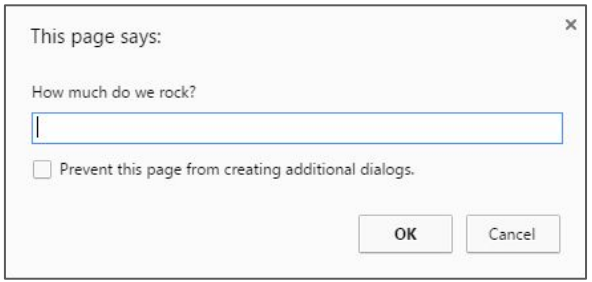
`confirm` displays a true/false pop-up.

`prompt` displays a pop-up with a text-box input.

```
confirm("Do we rock?");
```




```
prompt("How much do we rock?");
```



How do we check conditions?

`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.");
```

What is an array?

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];
```

Activity: Basic JS Dissection

In this activity, you'll re-examine the file from the last class.

(Instructions sent via Slack)

Suggested Time:

3 Minutes

Activity: Basic JS Dissection

Re-examine the file sent to you during the last class.

See if you now better understand how it works.

Prepare to share when time is up.

Time's Up! Let's Review.

Questions?

Time to Code

Array Logging

Suggested Time:

5 Minutes

Activity: Array Logging

Follow the instructions provided in the file to `console.log` each of the names in the `coolPeople` variable.

Hint: You should be repeating the same line six times.

Be prepared to share when time is up.

Time's Up! Let's Review.

Questions?

Time to Code

Array Setting

Suggested Time:

5 Minutes

Activity: Array Setting

- Follow the instructions in the file provided to convert each item in the array to lowercase.
- Make sure to only add in lines of code where instructed.
- **Hint:** You will need to use the method `.toLowerCase()`. Research if you don't remember how to use it.
- Be prepared to share when time is up.

Back to the Zoo Pen

Array name: zooAnimals

Coded in JavaScript using an array.

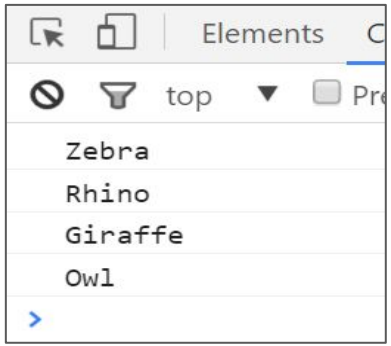
```
// Our array of zoo animals.
var zooAnimals = ["Zebra", "Rhino", "Giraffe", "Owl"];
```

Back to the Zoo Pen

Array name: zooAnimals

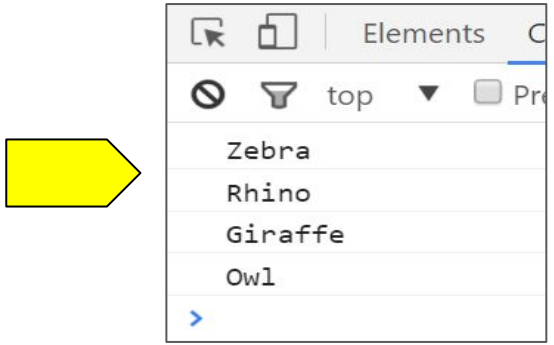
Zebra	Rhino	Giraffe	Owl
Index 0	**Index 1**	**Index 2**	**Index 3**

```
// 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]);
```



What's Wrong Here?

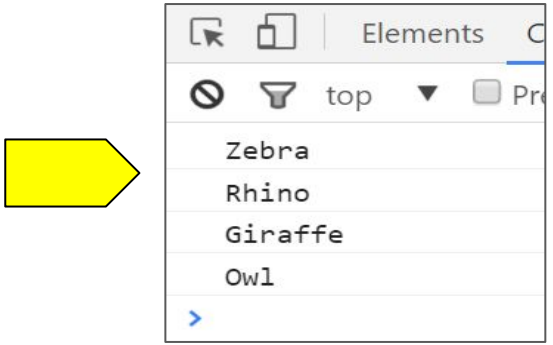
```
// Array of zoo animals.
var zooAnimals = ["Zebra", "Rhino", "Giraffe", "Owl"];

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



What's Wrong Here?

Repeated code! Let's be more efficient.

```
// 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];
```




<DRY>
DON'T REPEAT YOURSELF

`for` Loops

for Loops

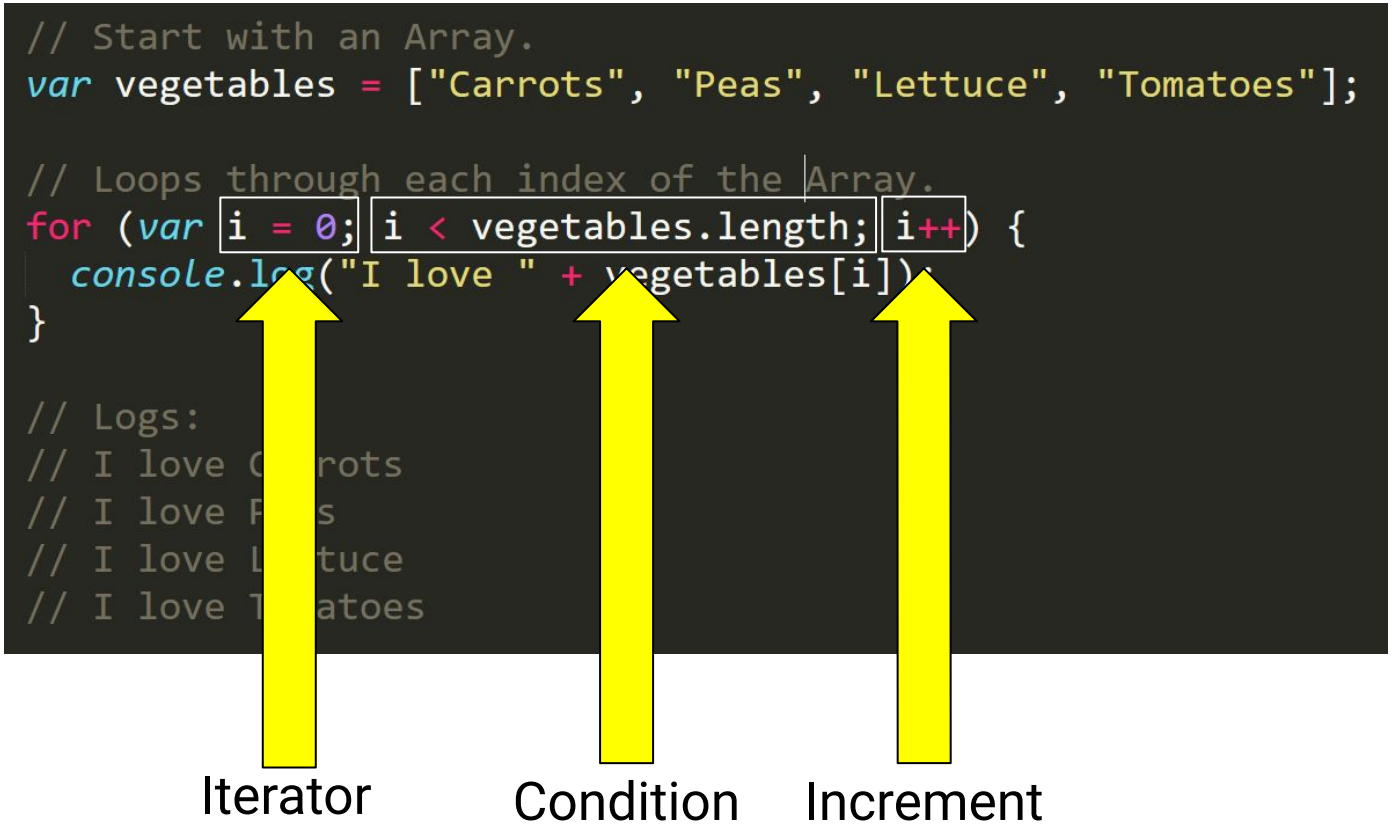
for loops are **critical** in programming. We use them to run **repeated blocks of code** over a set period.

Each for loop is composed of a:

- Variable declaration or counter (iterator)
- Loop condition
- 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]);
}
```

for Loops


// 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

for Loops

Code between the { } gets repeated each time the iterator is smaller than the condition (in this case, as long as i less than 4).

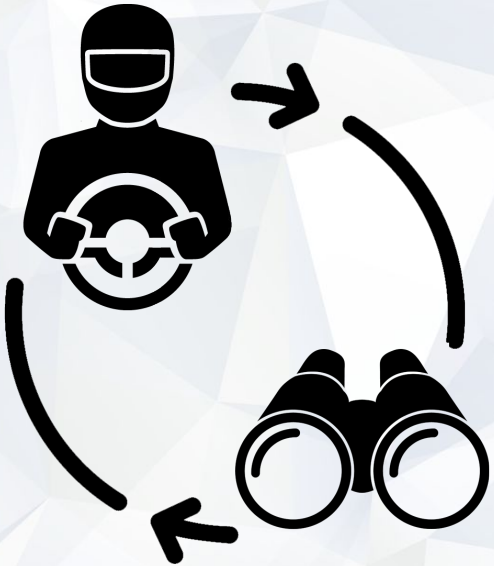
```
// Start with an Array.
var vegetables = ["Carrots", "Peas", "Lettuce", "Tomatoes"];

// 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
```

for Loops

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

```
// 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
```



Pair Programming Activity:

for Loop Dissection

In this activity, you'll...

(Instructions sent via Slack)

Suggested Time:

5 Minutes

Activity: for Loop Dissection

- With a partner, spend a few moments trying to dissect the code sent to you.
- Try to explain to one another what is happening with each line of code.
- Feel free to do research if you are stumped.

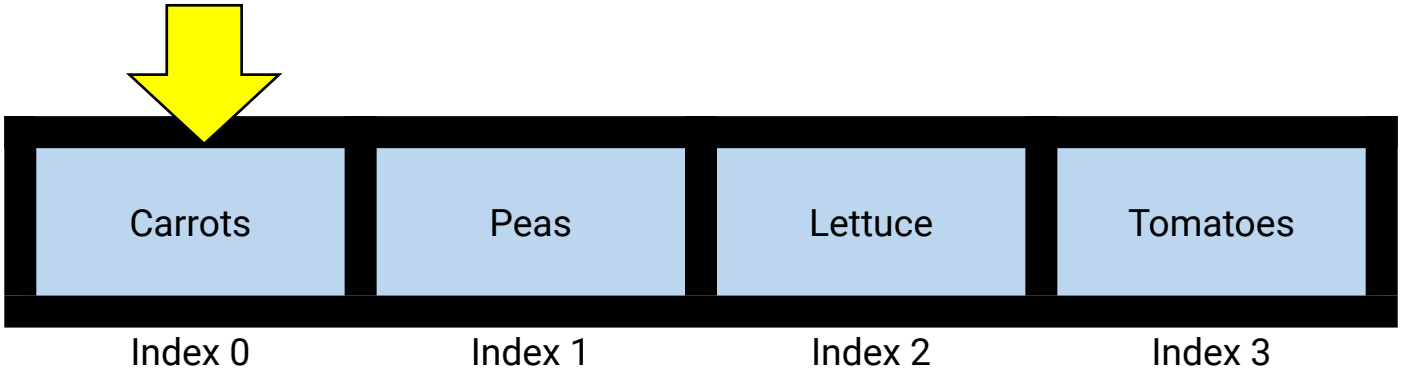
Hint: Look into the phrase “for loop.”

for Loops Continued

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]);
}
```

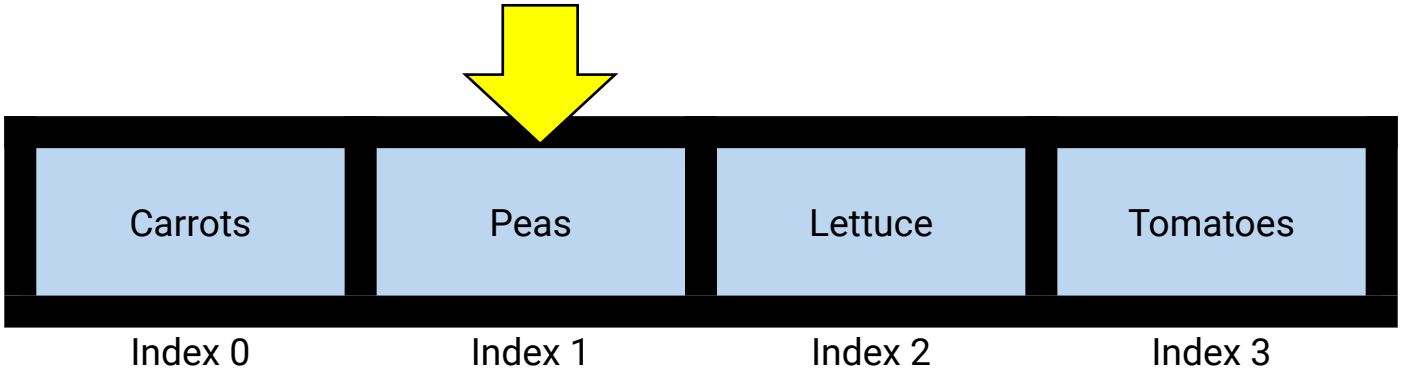
When i = 0 … `console.log`(“I love Carrots”)



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]);
}
```

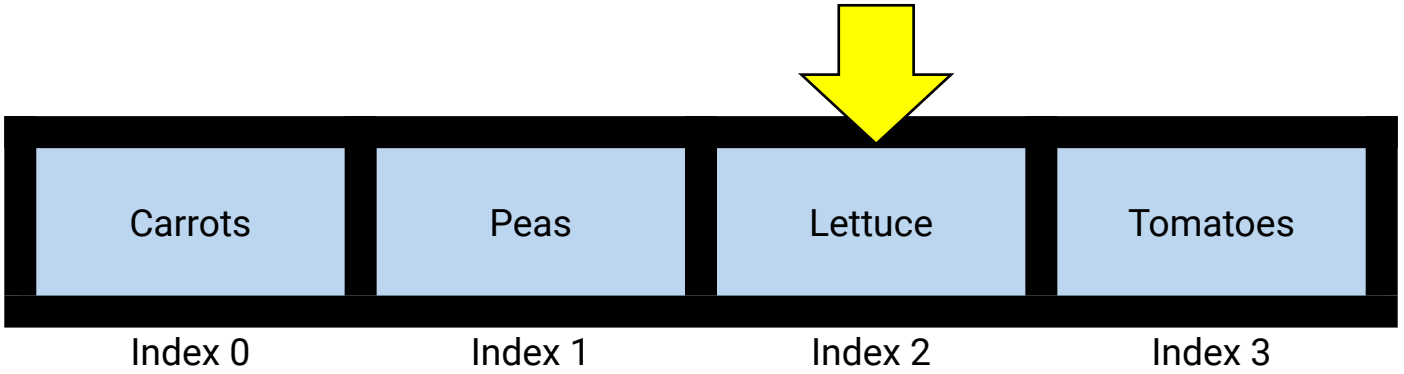
When i = 1 … `console.log`(“I love Peas”)



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]);
}
```

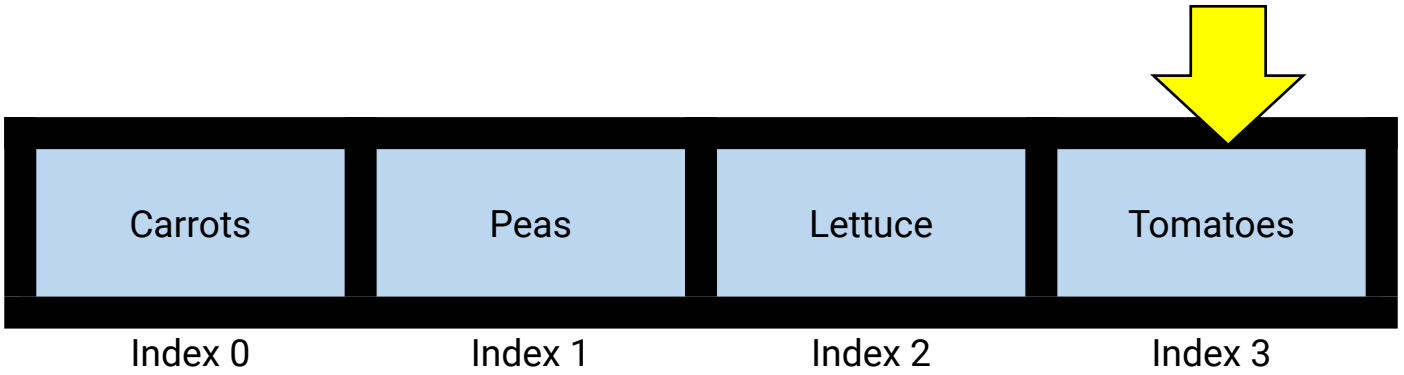
When i = 2 … `console.log`(“I love Lettuce”)



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”)



Activity: for Loop Zoo

In this activity, you'll...

(Instructions sent via Slack)

Suggested Time:

15 Minutes

Activity: for Loop Zoo

Use `for` loops to rewrite the file sent to you via Slack.

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

Once you think your code is functioning properly, share on slack and ask for feedback.

```
// 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];
```

Time's Up! Let's Review.

Questions?


Break

Activity: Another Loop

In this activity, you'll...

(Instructions sent via Slack)

Suggested Time:

Optional

Activity: Another Loop

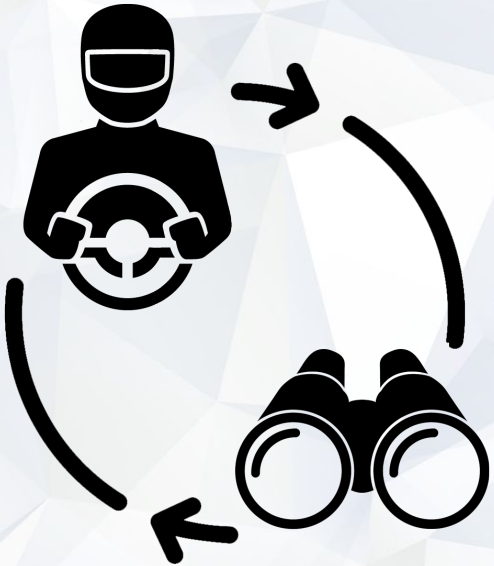
Starting from scratch, create a `for` loop that prints the following lines:

```
I am 0
I am 1
I am 2
I am 3
I am 4
```

Don’t use an array!

Time's Up! Let's Review.

Questions?



Pair Programming Activity:

Loop With Conditions

In this activity, you'll...

(Instructions sent via Slack)

Suggested Time:

10 Minutes

Partner Activity: Loop With Conditions

Starting from scratch, write code that loops through the following array and that logs the name of each animal on the farm to the console:

```
var myFarm = ["chickens", "pigs", "cows", "horses" , "ostriches"];
```

Then check if the first letter in the animal's name begins with a “c” or “o.” If it does, create an alert: “Starts with ‘c’ or ‘o’!”

Hint: You can access the first character of a string as if it were the first element of an array.

Time's Up! Let's Review.

Questions?

Activity: Random Number Loop

In this activity, you'll...

(Instructions sent via Slack)

Suggested Time:

7 Minutes

Activity: Random Number Loop

Research how to use `Math.random()` to generate a whole number between 1 and 10.

Open `21-RandomNumbers/Unsolved` and modify the code so that is logs random *whole numbers* from 1 to 10 inclusive.

Time's Up! Let's Review.

Questions?

Instructor Demonstration

Demo RPS

Demo RPS

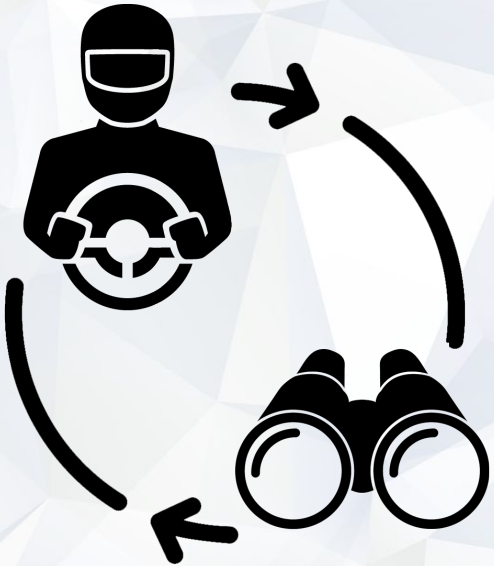
You'll need to use conditional statements and the `&&`/`||` operators to make things work.

It's okay if you have to write a long chain of `if`, `else-if`, or `else` statements. If they find themselves doing this, they're on the right track.

The computer *randomly* chooses each time.

Rock-Paper-Scissors With a Partner!

Play five rounds.



Pair Programming Activity:

Pseudocode RPS

In this activity, you'll create a pseudocoded solution that lays out the steps involved in playing rock-paper-scissors against a computer.

(Instructions sent via Slack)

Suggested Time:

3 Minutes

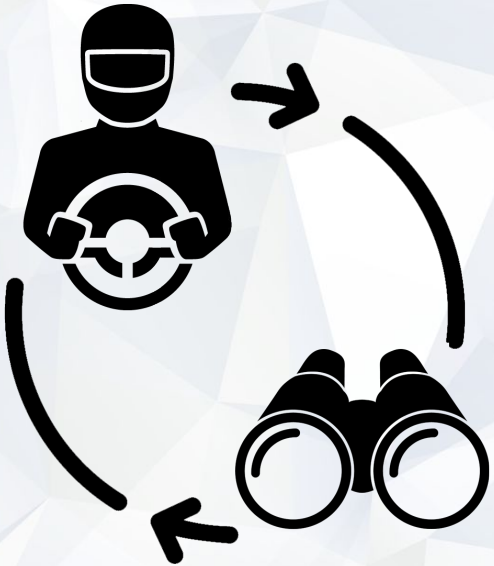
Partner Activity: Pseudocode RPS

- With a partner, spend a few moments outlining all the steps and conditions that go into a single game of rock-paper-scissors.
- Try to break it down into steps that you could “code out.”
- Think of basic elements like loops, `if-else` statements, arrays, alerts, etc.
- Be prepared to share your outlined approach.

You just **pseudocoded!**

And now, for the rest of the class you will be coding it out!


Don’t worry.
We’ll be here
to help you
along the way.



Pair Programming Activity:

Coding out RPS

In this activity, you'll begin the process of coding out the rock-paper-scissors game.

(Instructions sent via Slack)

Suggested Time:

60 Minutes

Partner Activity: Coding Out RPS

In groups of four, begin the process of coding out the rock-paper-scissors game.

Play the game 10 times, then show you total scores.

Do as much as you can on your own, but don't be afraid to ask for help if you feel your team is struggling.

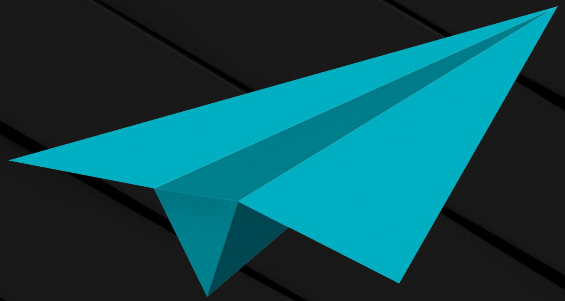
Note: Don’t worry. We know this is a very challenging assignment. We also know that you won’t know where to start. In fact, we haven’t shown you EVERYTHING you need yet, but that’s okay. Part of being a developer is figuring things out on your own through trial and error.

Time's Up! Let's Review.

Questions?


RECAP


The End



Office Hours

30 Minutes