Learning Objective	Practice Problems	Reading / Lecture
Running JS Locally		
Match the commands ls, cd, pwd to their descriptions		
Given a folder structure diagram, a list of 'cd (path)' commands and target files, match the paths to the target files.	Find the Path Quiz	
Use VSCode to create a folder. Within the folder create a .js file containing console.log('hello new world'); and save it.		https://open.appacademy.io/learn/js-py---aug-2020-online/week-2-aug-2020-online/local-javascript-project
Use node to execute a JavaScript file in the terminal		
POJOs		
Label variables as either Primitive vs. Reference	Primitive vs. Reference Quiz	
Identify when to use . vs [] when accessing values of an object		
Write an object literal with a variable key using interpolation		
Use the obj[key] !== undefined pattern to check if a given variable that contains a key exists in an object	POJO Project (See below)	
Utilize Object.keys and Object.values in a function	POJO Project (See below)	
Iterate through an object using a for in loop	Iterate Through Object	https://open.appacademy.io/learn/js-py---jun-2020-online/week-2-jun-2020-online/iterating-through-objects
Define a function that utilizes ...rest syntax to accept an arbitrary number of arguments	POJO Project (See below)	
Use ...spread syntax for Object literals and Array literals	POJO Project (See below)	https://open.appacademy.io/learn/js-py---jun-2020-online/week-2-jun-2020-online/rest-and-spread
Destructure an array to reference specific elements	Destructuring Quiz	
Destructure an object to reference specific values	Destructuring Quiz	
Write a function that accepts a array as an argument and returns an object representing the count of each character in the array	POJO Project (See below)	
Callbacks		
Given multiple plausible reasons, identify why functions are called “First Class Objects” in JavaScript.	Callbacks Quiz	
Given a code snippet containing an anonymous callback, a named callback, and multiple console.logs, predict what will be printed	Callbacks Quiz	
Write a function that takes in a value and two callbacks. The function should return the result of the callback that is greater.	Callback Project (See below)	
Write a function, myMap, that takes in an array and a callback as arguments. The function should mimic the behavior of Array#map.	Callback Project (See below)	
Write a function, myFilter, that takes in an array and a callback as arguments. The function should mimic the behavior of Array#filter.	Callback Project (See below)	
Write a function, myEvery, that takes in an array and a callback as arguments. The function should mimic the behavior of Array#every.	Callback Project (See below)	
Scope		
Identify the difference between const, let, and var declarations	Variables Quiz	
Explain the difference between const, let, and var declarations		https://open.appacademy.io/learn/js-py---jun-2020-online/week-2-jun-2020-online/different-variables-in-javascript
Predict the evaluation of code that utilizes function scope, block scope, lexical scope, and scope chaining	Scope Quiz	
Define an arrow function		https://open.appacademy.io/learn/js-py---jun-2020-online/week-2-jun-2020-online/arrow-functions
Given an arrow function, deduce the value of this without executing the code	Arrow Functions Quiz	
Implement a closure and explain how the closure effects scope	Scope Project (See below)	
Define a method that references this on an object literal	Scope Project (See below)	
Utilize the built in Function#bind on a callback to maintain the context of this	Scope Project (See below)	
Given a code snippet, identify what this refers to	Context Quiz	
POJO Project Focus Problems		
You should feel comfortable with all the problems in the POJO project (there's only 14 and some are pretty small)		
Callbacks Project Focus Problems		
01-my-for-each.js		
02-my-map.js		
04-my-filter.js		
05-selective-map.js		
06-reject.js		
10-my-every.js		
12-exactly.js		
18-one.js		
19-greater-callback-value.js		
20-none.js		
23-alternating-map.js		
Scope Project Focus Problems		
00-arrow-addfive.js		
02-arrow-my-map.js		
03-arrow-rest-sum.js		
04-arrow-average-value.js		
05-car-drive.js		
10-change-context.js		
11-bound-by-arg.js		
13-hidden-counter.js		
17-all-the-args.js		
19-lazy-adder.js		
20-call-on-target.js		
22-dynamic-divide.js		
23-smoothie-machine.js		
25-party-planner.js		
26-coupon.js		