1. [Value vs. Reference](https://www.educative.io/collection/page/5679346740101120/5707702298738688/5685265389584384/" \t "_blank) — Understand how objects, arrays, and functions are copied and passed into functions. Know that the reference is what’s being copied. Understand that primitives are copied and passed by copying the value.
2. [Scope](https://scotch.io/tutorials/understanding-scope-in-javascript#toc-scope-in-javascript) — Understand the difference between global scope, function scope, and block scope. Understand which variables are available where. Know how the JavaScript engine performs variable lookup.
3. [Hoisting](http://javascriptissexy.com/javascript-variable-scope-and-hoisting-explained/) — Understand that variable and function declarations are hoisted to the top of their available scope. Understand that function expressions are not hoisted.
4. [Closures](http://javascriptissexy.com/understand-javascript-closures-with-ease/) — Know that a function retains access to the scope that it was created in. Know what this enables us to do with our code. Understand that closures allow data hiding, memoization, and dynamic function generation.
5. [this](https://www.educative.io/collection/page/5679346740101120/5707702298738688/5676830073815040) — Know the rules of this binding. Know how it works, know how to figure out what it will be equal to in any piece of code, and know why it’s useful.
6. [new](https://codeburst.io/javascripts-new-keyword-explained-as-simply-as-possible-fec0d87b2741) — Know how it relates to object oriented programming. Know what happens to a function called with new and why such a function is known as a constructor. Understand that an object generated by using newinherits from the function’s prototype property.
7. [apply, call, bind](https://codeplanet.io/javascript-apply-vs-call-vs-bind/) — Know how each of these functions work. Know how to use them. Know what they do to this.
8. [Prototypes & Inheritance](https://codeburst.io/master-javascript-prototypes-inheritance-d0a9a5a75c4e) — Understand that inheritance in JavaScript works through the [[Prototype]] chain. Understand how to set up inheritance through functions and objects and how new helps us implement it. Know what the \_\_proto\_\_ and prototype properties are and what they do.
9. [Asynchronous JS](https://www.youtube.com/watch?v=8aGhZQkoFbQ&t=948s) — Understand the event loop. Understand how the browser deals with user input, web requests, and events in general. Know how to recognize and write asynchronous code. Understand how JavaScript is both asynchronous and single-threaded.
10. [Higher Order Functions](https://www.sitepoint.com/higher-order-functions-javascript/) — Understand that functions are first-class objects in JavaScript and what that means. Know that returning a function from another function is perfectly legal. Understand the techniques that closures and higher order functions allow us to use.