
[JSR] – Learning Javascript from Scratch (The Baby Phase)

1 message

Zell Liew <zell@zellwk.com>

Wed, Mar 6, 2019 at 1:22 PM

To: Anibal <anibalsantosgo@gmail.com>

Hey Anibal,

Two days ago, you learned about the four phases of growing up in JavaScript-land. Do you know which phase you're in right now?

If you're in the baby phase, this lesson is for you. Here, you'll learn:

1. What to focus on for the baby phase.
2. How to overcome challenges that may surface for you.
3. What you need to know before moving on to the next phase.

If you're way past the baby phase, you might still want to review this lesson and see if you missed anything fundamentally important. (I hope you didn't!).

If you don't know which phase you're in (or if you're in a more advanced phase), it might be worth it to read through this lesson to see if you're already past the baby phase. If you are, you can look forward to the next lesson.

Without further ado, let's dive into what to focus on for the baby phase.

What to focus on in the baby phase

The focus of the baby phase is to get you familiarized with the JavaScript syntax. In this phase, you'll:

1. Learn how to declare variables.
2. Learn what are Strings, Numbers, Booleans, Null and Undefined.
3. Learn what are Arrays, Objects and Functions.
4. Learn to use `if` and `else` statements.
5. Learn how to compare variables.
6. Learn to use the `for` loop.

Here are some resources that'll help you through the baby phase. Feel free to go through any one of them. If you want to, you can go through all of them to get a solid understanding before moving on.

Free resources:

- [Udacity – Intro to JavaScript](#)
- [You don't know JS – Up and going](#)
- [Code Academy – Learn JavaScript](#)

Paid resources:

- [Code School – JavaScript Road Trip Parts 1, 2 and 3](#)
- [Frontend Masters – Introduction to JavaScript Programming](#)
- [Frontend Masters – From Fundamentals to Functional JS](#)

As you go through these resources, focus on learning how to write JavaScript. Follow through with the examples. Make sure you type your code word for word in your text editor so you get used to writing JavaScript.

Don't worry if you make a mistake. You probably had a typo error you couldn't spot. Learn how to use `console.log` to debug your errors so you know what went wrong.

Also, don't worry if you're not using best practices at this point. They key here is just to get familiar with JavaScript and be confident in using the basic things listed above.

At the end of this email, you'll find a list of questions you can use to check your understanding before moving on to the next phase.

But before that, let's tackle some common questions or challenges that'll surface at this point:

Common questions and challenges

1. How long would the baby phase take?

It depends on how much time you're willing to commit to learning. Generally, emerging from the baby phase with firm fundamentals would take you anywhere between one day to one week.

2. How do I use these syntax stuff that I just learned?

The easiest way to start using what you've learned is to link your JavaScript file to your HTML file with the `<script>` tag.

Once linked, open up your developer tools and navigate to the "console" section. (In Chrome, its view > developer > developer tools). You'll be able to use `console.log` to output your statements into the console.

Don't worry about using them to manipulate HTML and other stuff right now. You'll cover them in the next phase. Trust the system.

3. What is this ES6 thing that people keep talking about?

JavaScript is also named ECMAScript. The number 6 tagged at the end is a version of JavaScript. Traditionally, the most popular ones are version 3 (super old now), 5 and 6.

ES6 is one of the recent advancements in JavaScript. Now, almost all ES6 features have been integrated into JavaScript, so you can effectively use anything ES6 related.

At this point, diving too much into ES6 would be confusing for you. If you're interested though, read [this article](#) to find out more. Focus on the "const and let" part and the "arrow functions".

4. Why learn JavaScript, not jQuery?

Great question. jQuery is a library that's built on top of JavaScript. It used to be crucial to manipulate HTML. But, with recent advancements in JavaScript, jQuery is no longer needed.

Eventually, it's more beneficial in the long run to learn JavaScript because you can apply its concepts to any library you use in the future.

5. Why does JavaScript seem so illogical?

Every language is different. They have their pros and cons. Their idiosyncrasies. Just like how English is completely different from French, JavaScript is different from any other language you may have come across, even though there may be similarities.

Instead of feeling that JavaScript is illogical, try embracing JavaScript's weirdness. Be open to it. Accept it for what it is for and you'll learn it quickly. Maybe you'll think JavaScript is awesome once you finally get it?

A list of questions to check your understanding

The final part of this lesson is a list of questions you can use to check your understanding.

Don't be overwhelmed by how big it looks. Try to answer each of the questions within. When you can answer everything, you know your fundamentals are firm enough to move on.

Here we go:

1. What are the five primitive values in JavaScript? (There are six if you consider ES6).
2. How do you declare and assign variables in JavaScript?
3. What's the difference between `const`, `let` and `var`?
4. What do each of the following operators do?
 1. `+`
 2. `-`
 3. `*`
 4. `/`
 5. `%`
5. What do each of the following comparison operators do?
 1. `===`

2. !==
 3. >
 4. >=
 5. <
 6. <=
6. How do you use the following conditionals?
1. if
 2. if else
 3. else
7. How do you use a for loop?
8. What is an array?
1. How do you put values into arrays?
 2. How do you get values out of arrays?
 3. How do you remove a value from an array?
 4. How do you loop through every value of an array?
9. What is an object?
1. How do you put values into objects?
 2. How do you get values out from objects?
 3. How do you remove a property from an object?
 4. How do you loop through every value of an object?
 5. What is a method on an object?
 6. How do you define methods?
 7. How do you call/invoke a method?
10. What is a function?
1. How do you define functions?
 2. How do you call/invoke/execute functions?
 3. How do you pass arguments into a function?
 4. What does the return keyword do in a function?

That's it!

If you can answer these questions, you're set to move into the Child phase :)

Stay awesome,
Zell

-
- Need 1-1 help? [Let me know](#)
 - If you no longer wish to receive emails from me, you can [unsubscribe](#) anytime.
 - If you want to change your name or email address, you can use [this link](#).
 - Here's my address if you want to send something over: [Blk 427 Serangoon Central #05-310](#), Singapore, Singapore 550427