Gmail

Anibal Santos <anibalsantosgo@gmail.com>

[JSR] – Mastering the DOM (The Child Phase)

1 message

Zell Liew <zell@zellwk.com> Thu, Mar 7, 2019 at 1:20 PM
To: Anibal <anibalsantosgo@gmail.com>

Hey Anibal,

Were you able to answer the questions listed in the baby phase? If you could, you’re ready to move into the child phase. That’s what this lesson is for.

Now, if you were unable to answer the questions, take some time to figure them out before moving on. It’s important to get the fundamentals in place, or the child phase would be confusing and overwhelming for you.

If you’re way past the child phase, you might still want to review this lesson and see if you missed anything fundamentally important. (Hopefully not).

Moving on, here’s what you’ll learn in today’s lesson:

1. What should you focus on for the child phase?
2. How to overcome challenges that may surface for you
3. What do you need to know before moving into the teenage phase.

Now, without further ado, let’s dive into what to focus on for the child phase.

What is the child phase again?

Just a quick reminder, you’re in the child phase if you’re comfortable with the basic JavaScript syntax.

Maybe you can copy-paste and modify other people’s code after hours of googling, but you’re not confident about building things from scratch yet.

What to focus on in the child phase

Your focus for this phase is to get yourself familiarized with the DOM, so you can change it and make anything you want. You want to focus on learning to build things like accordions and carousels.

(See the questions section if you have no idea what the DOM is)

To understand how JavaScript works with the DOM, you first need to brush up a little more JavaScript fundamentals. To be specific, you need to learn these three things:

1. What are scopes and closures
2. What is asynchronous JavaScript
3. How to write asynchronous JavaScript with callbacks.

Once you have a firm understanding of these three things in JavaScript, begin to focus on learning methods you can use with the DOM, things like:

1. How to select elements
2. How to add or remove classes
3. How to add or remove attributes
4. How to add or remove elements
5. How to use event listeners to add interactivity
6. What events are there and how to use them

Here are some resources that’ll help you get through the child phase. Go through ALL of them (in order).

Free resources:

- You don’t know JS – scopes and closures
- What is a callback

- Altering the DOM with JavaScript
- JavaScript 30 by Wes Bos (You may get confused while going through this one if your JS skills aren't stable enough yet. Work through it slowly).

Useful references:

- MDN Event Reference
- MDN Element API Reference

Once you've completed these free resources, think about what components you want to build. Work on easier ones like accordions first, then move towards tougher ones like carousels.

The key is to keep building things. It's okay to use other people's code and ideas at first. Once you complete a component, build it again. This time, don't refer other people's code. Make sure you write every line of code yourself so you internalize what needs to happen.

At this point, don't worry about best practices or code quality. You can get to them later. If you try to incorporate best practices now, you're trying to learn two things at once. It's a surefire way to get confused and overwhelmed.

(If you don't know how to think like a developer, I got you covered, see below).

You will feel like giving up

This phase is hard. Most half-hearted learners give up here, thinking JavaScript is too hard for them. Getting from theory and syntax no building stuff that work is hard because you have no prior knowledge of how stuff should work.

So, expect to google a lot. Watch videos and pay attention to how other people build things. Slowly, you'll begin to learn and understand the mechanics to it all.

To help you out, I'm building a JavaScript course that'll teach you everything you need to know from the baby phase to the teenage phase. Click on this link if you're interested to hear more.

Common questions and challenges

1. How long should I spend in the child phase?

Depends on how much time you're willing to invest in learning JavaScript daily. If you learn for 2 hours a day, you might spend anywhere between a week to a month. Maybe longer.

Take the time you need to learn this phase properly. If you don't you'll just face more overwhelm and confusion later.

2. When can I move on to the next phase?

In this phase, you ideally want to build at least 6 components before you move on. If you build more things, you have more confidence in your ability to solve problems, which would help you in future.

Once you built your 6 components, head over the end of this email and see if you can answer every question there. If you can, 👍. You're ready to move on.

3. What is the DOM?

The DOM stands for Document Object Model. It is a representation of your HTML. It's your "HTML" when you open up your developer's tools. It is here where you can modify your HTML with JavaScript methods. More info in this article

4. There are so many DOM methods. Which should I learn?

Focus on the ones listed in the question section below. They are the ones you need to master. The rest are additional things that are good to know.

5. How can I remember so many methods?

We remember things by using them a lot of times. The easiest way to remember the methods is to use them. So, as long as you keep building things, these methods will stick to you for life.

It also helps to download a JavaScript snippet pack like JavaScript completions for your text editor. They reduce the amount of code you type, which helps in remembering things.

6. How can I think like a programmer/developer?

If I'm not mistaken, you're asking this question because you don't know how to start.

The best answer I can give you is to Google. Search online for how you'd build an accordion to begin with. Notice how someone else builds the component. How does he think? How does he break down the problem?

As you build more components, you'll internalize this process of breaking down big problems into smaller ones, then solving the smaller problems to fix the big problem.

This is a process that you'll need later in your development career when you can't find any of your answers on Google. It's just you, your code and your brain. It will come, so take the time to practice now.

Here's an article that goes through the steps in detail. Read it. It'll help.

(Just a reminder, don't fall into the victim trap).

7. Writing HTML in JavaScript is difficult...

Yes it is. Template strings make them easier. Check this article out.

A list of questions to check your understanding

As before, the final part of this lesson is a list of questions you can use to check your understanding.

Before looking at the list, you'd want to make sure you've built 6 components from scratch. If you did so, you'll already have the answers to most of the questions in the list.

For questions that you don't know the answer to, feel free to build 2-3 more things that involves that concept so you get some practice.

Note: This list is HUGE. Don't feel overwhelmed. The questions are just broken down into small parts so its easier for you to understand and check things off. When you can answer everything, you're 👍 to move on.

Here's the questions:

1. What is the JavaScript scope?
    1. Why should you keep global variables to a minimum?
    2. What is a closure?
    3. Why do you use closures?
2. What is a callbacks
    1. How do you use a callback?
    2. How do you write a callback?
    3. How do you write a function that accepts a callback?
    4. Is `setTimeout` a callback-accepting function?
3. Asynchronous JavaScript
    1. What does asynchronous and synchronous JavaScript mean?
    2. How do you write asynchronous JavaScript?
    3. What is an event loop?
    4. How does the event loop work?
4. DOM methods and questions
    1. What is an Element?
    2. What is a Node?
    3. How do you select an Element?
    4. How do you select multiple Element?
    5. How do you loop through multiple Element for all browsers?
    6. How do you select the parent Element?
    7. How do you select sibling Elements?
    8. How do you select children Elements?
    9. How do you add a class to a Element?
    10. How do you remove a class from a Element?
    11. How do you check if a class is present on a Element?
    12. When should you add a class to a Element?
    13. How do you add an attribute to a Element?
    14. How do you remove an attribute from a Element?
    15. How do you check if an attribute is present on a Element?
    16. When should you add or remove an attribute?
    17. How do you create a HTML Element?

    18. How do you add your Element before another Element?
    19. How do you add your Element after another Element?
    20. How do you change the style of your Element?
    21. Should you change the style with JavaScript? Why or why not?
    22. How do you get the contents of a Element?

5. Events
    1. How do you add an event listener?
    2. Why do you add event listeners?
    3. How do you remove an event listener?
    4. When should you remove event listers? Why?
    5. What are the common mouse events?
    6. What are the common keyboard events?
    7. What are the common form events?
    8. How do you get the value of a event target?
