

Lesson

Weekend

Intermediate JavaScript (/intermediate-javascript)

/ Object-Oriented JavaScript

(/intermediate-javascript/object-oriented-javascript)

/ Object-Oriented JavaScript Objectives

Text

We've learned to work with just about all the fundamental tools that JavaScript has to offer: primitives, functions, arrays, looping, branching, and using built-in object methods and properties. We've also reinforced our know-how of using built-in object properties and methods by implementing Web APIs into our applications to make them interactive.

However, we've never actually created our own JavaScript objects, and in this section, that's just what we'll do! We'll learn:

- That JavaScript objects collect data AND functions into a single unit.
- How to create object literals.
- How to create object constructor functions and prototype methods.
- How JavaScript objects share functionality through prototypal inheritance.

We'll also learn more about events and Web APIs. We'll cover:

- Event bubbling

- Event delegation
- The Event object
- The Node object
- The CSS Object Model (CSSOM)

There will also be lessons on new topics, and opportunities for further exploration:

- Switch cases
- Using your browser's local storage
- Using aliases

At the very end of the section, we'll learn about whiteboarding, which is the process of solving a coding problem on a dry erase whiteboard (or in a text editor in remote environments) to demonstrate and communicate about your problem-solving process. The goal of introducing whiteboarding now is to expose you to the concept and process, and provide optional prompts to complete. (The exact schedule of whiteboarding will be up to your instructor.) Then, starting in the backend language course (C# or Ruby), you'll whiteboard at the end of most class sessions.

learn about JavaScript **objects** which collect data AND functions into a single *object* unit. We will see how business logic and user interface logic work together in the creation and updating of objects.

At the end of the section, you should be able to:

- Build an application that structures its data with JavaScript objects created with constructor functions.
- Add functionality to JavaScript objects by adding methods to the object's prototype.
- Use event delegation to attach an event listener to a parent element that fires for a child element.
- Implement 'separation of concerns' to create JavaScript functions that focus on doing just one thing.

Important note: If you find video content helpful to your learning style, we recommend supplementing with external resources that include videos. Students have found helpful videos at the following online resources:

- Treehouse (<https://teamtreehouse.com/>)
- Lynda (<https://www.lynda.com/>)

Of course, there are also many tutorials and videos available on YouTube (<https://www.youtube.com/>) as well.

Independent Project Objectives

At the end of this section, you will complete an independent project. Your code will be reviewed for the following objectives:

- Constructors and prototypes are used successfully.
- Application works as expected.
- Tests are included for each business logic behavior and code is committed after each test passes.
- Project is in a polished, portfolio-quality state.
- The prompt's required functionality and baseline project requirements are in place by the deadline.

Next (</intermediate-javascript/object-oriented-javascript/installing-and-using-node-version-manager>)

Lesson 1 of 33

Last updated March 23, 2023

disable dark mode



© 2023 Epicodus (<http://www.epicodus.com/>), Inc.