

Lesson

Thursday

# Introduction to Programming

## (/introduction-to-programming)

### / JavaScript and Web Browsers

## (/introduction-to-programming/javascript-and-web-browsers)

### / Debugging in JavaScript: Using debugger and Breakpoints

Text

Cheat sheet

## Using debugger

- The debugger tool freezes execution of our code wherever we add a `debugger;` statement or a breakpoint.
- The best way to add a breakpoint is to go to the DevTools **Sources** tab, click on the file you want to add a breakpoint to (in the left pane of DevTools), and then click on the line number in the source code to add a blue arrow (a breakpoint).
- The breakpoint will pause execution of the code as soon as it's reached, so it should be added to the line *after* the one you want to evaluate and debug.
- We can also make changes directly to the code in the **Sources** tab. This allows us to experiment without needing to change our actual code in Visual Studio Code.

- We can click "Resume script execution" (the blue arrow) to go to the next breakpoint. If there is no other breakpoint, the script will simply continue running.
- We can click "Step over next function call" (arrow rounded over a dot) to go to the next line of code that's supposed to execute. We can have our code run line by line until we have run through our entire scripts.
- When our code is paused, we can go to the DevTools console and access any variables in our current scope. We can even write and run statements that use those variables.
- Make sure to familiarize yourself with this tool. It's one of the most important debugging tools at our disposal.

Previous (/introduction-to-programming/javascript-and-web-browsers/debugging-in-javascript-pausing-on-exceptions)

Next (/introduction-to-programming/javascript-and-web-browsers/practice-triangle-tracker)

Lesson 73 of 75

Last updated March 24, 2023

disable dark mode



Epicodus (<http://www.epicodus.com>)

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