

Learning JavaScript

The language that makes web pages interactive — variables, functions, and talking to the page itself.

What JavaScript is for

HTML structures a page and CSS styles it, but neither can react to anything. JavaScript is the layer that listens for clicks, changes what's on screen, remembers information, and makes decisions while the page is open.

Variables and values

```
let name = "Leo";  
let age = 18;  
let isDeveloper = true;
```

Let creates a variable you can change later; const creates one you can't. Use const by default, and only switch to let once you know the value actually needs to change.

Functions

```
function greet(name) {  
  return `Hello, ${name}!`;  
}  
  
console.log(greet("Leo")); // Hello, Leo!
```

A function is a reusable block of instructions. Anything you find yourself doing more than once is a good candidate for one.

Conditionals and loops

```
if (age >= 18) {
  console.log("Adult");
} else {
  console.log("Minor");
}

for (let i = 0; i < 5; i++) {
  console.log(i);
}
```

if/else make decisions; loops repeat something a set number of times, or once for every item in a list.

Talking to the page (the DOM)

The DOM (Document Object Model) is JavaScript's view of your HTML — every tag on the page becomes an object you can read or change.

```
const button = document.querySelector(".play-btn");

button.addEventListener("click", () => {
  button.classList.toggle("playing");
});
```

This exact pattern — `querySelector` plus `addEventListener` — runs the music player and the mobile menu on this site. Most interactive websites are built from a handful of patterns like this one, repeated.

A small real example

```
<button id="toggle">Show more</button>
<p id="details" style="display:none">Here are the details.</p>

<script>
document.getElementById("toggle").addEventListener("click", () => {
  const details = document.getElementById("details");
  details.style.display = details.style.display === "none" ? "block" : "none";
});
</script>
```

Copy that into an HTML file and open it. That's a real, working feature, built entirely from what's above it in this guide.

Where to go from here

- Rebuild small pieces of pages you use every day — a like button, a dropdown, a counter.
- Get comfortable with the browser console (right-click → Inspect → Console). It's where you'll spend a lot of time debugging.
- Don't try to learn the whole language before building something. Build the thing, and look up what you don't know exactly when you hit it.
- Once loops and functions feel natural, the “Using AI to Learn to Code” guide covers how to speed up everything after this.

From the Notes & Guides section of leocebillo.github.io — written to explain, not just to answer.