

Web Coding Camp — Day 4

JavaScript: Making It Interactive

David Pouliot

Southern Oregon University

Today

HTML is content. CSS is style. **JavaScript** is *behavior* — it makes things happen.

- The console — your JavaScript playground
- Variables and functions
- Grabbing an element with `querySelector`
- Reacting to a click
- Changing text and style on the fly

Goal: add *one* interactive feature to your site — a click counter, a dark-mode toggle, a random-joke button.

The Console

Add JavaScript with `<script>`

A `<script>` tag holds your JavaScript — put it right before `</body>`.

```
1 <script>
2   console.log("Hello from JavaScript!");
3 </script>
4 </body>
```

Press **F12** in your browser and click **Console**. That's where `console.log()` messages — and any errors — show up. Keep it open.

Give your JavaScript its own file

Just like `style.css` holds all your CSS, put all your JavaScript in one file — `script.js` — and link it right before `</body>`.

```
1 <script src="script.js"></script>  
2 </body>
```

Now make `script.js` next to your HTML. Everything today goes in there — **no** `<script>` tags inside the file, just the code.

Link it *after* your elements (at the bottom) so the page is ready when the code runs.

Variables & Functions

Variables hold values

```
1 let score = 0;           // let = a value that can change
2 const name = "Alex";     // const = a value that won't change
3 let colors = ["red", "blue", "green"]; // an array (a list)
```

- let when the value will change (a score, a counter).
- const when it won't (a name, a fixed list).
- Every line ends with a ;.

Functions bundle up steps

Write the steps once, run them whenever you want.

```
1 function sayHi() {  
2   console.log("Hi there!");  
3 }  
4  
5 sayHi();    // runs it
```

- Define with `function name() { ... }`.
- Nothing happens until you **call** it: `sayHi()`.

Talking to the Page

Grab an element, then change it

querySelector finds an element by its #id (same # as CSS). Put this in a <script> *below* the element, so it exists when the code runs:

```
1 <h1 id="title">Hello</h1>
2 <script>
3   const title = document.querySelector("#title");
4   title.textContent = "New text!";    // change the words
5   title.style.color = "red";          // change the style
6 </script>
```

React to a click

`addEventListener` runs your code when something happens.

```
1 const btn = document.querySelector("#btn");  
2  
3 btn.addEventListener("click", () => {  
4   console.log("The button was clicked!");  
5 });
```

- "click" — the event to wait for.
- `() => { ... }` — the code to run when it happens.

Project: Button That Changes the Page

A working click counter

Every piece you just met, put together. Type it and click the button.

```
1 let count = 0;
2 const btn = document.querySelector("#btn");
3 const output = document.querySelector("#output");
4
5 btn.addEventListener("click", () => {
6   count = count + 1;
7   output.textContent = "You clicked " + count + " times";
8 });
```

With `<button id="btn">` and `<p id="output">` in your HTML.

Project: Add a Feature

Random joke button — your turn

Goal: click a button, show a random joke from a list.

```
1  const jokes = ["Too many bugs.", "It left its Windows open."];
2  const btn = document.querySelector("#jokeBtn");
3  const output = document.querySelector("#joke");
4
5  btn.addEventListener("click", () => {
6    // TODO: pick a random number, then show that joke
7    const i = Math.floor(Math.random() * jokes.length);
8    output.textContent = jokes[i];
9  });
```

Pick one feature to add

Starter

- Click counter
- Random joke / quote button

Standard

- Dark-mode toggle
- Colour changer
- Name greeting form

Boss challenge: a tiny quiz that scores answers, or pull a live joke from an API with `fetch()`.

Full code for every option is on the camp website. Add just *one* today.

Idea starter: dark-mode toggle

Add or remove a CSS class on <body>; style that class in CSS.

```
1 const btn = document.querySelector("#themeBtn");  
2 btn.addEventListener("click", () => {  
3   document.body.classList.toggle("dark");  
4 });
```

```
1 body.dark { background: #111; color: #eee; }
```