Make that button do something!

<button>Click me</button>

– An **EVENT** is something that happens to an HTML **ELEMENT** (something on the page)

– A **BUTTON** is an **ELEMENT**
  – What are other **ELEMENTS**?
  – What happens to buttons? They are “clicked”

– OUR GOAL: tell the program what to do when the button is clicked... to **HANDLE** that event

– Buttons are clicked, so the name of this handler is **onclick**

```html
<button onclick='JAVASCRIPT INSTRUCTIONS'>
  Click Me
</button>
```

Pedagogy sidebar: use terminology consistently to help students learn it. This might be hard for some.
What should the button do?

JavaScript can modify other HTML elements

– Change content
– Change style
– Hide or display

To modify an element, JavaScript has to be able to find it.

What did we do in CSS to apply a style to only one element?

We will use the same technique so JavaScript can find the desired element.
Update your page

1. Below the button, add a new paragraph element with the id “demo”

```html
<p id="demo">Hello :-)</p>
```

2. We want to tell JavaScript to:

   - Find that element
     - `document` is this web page
     - `getElementById` looks for the “id”
     - Case sensitive!! ByID won’t work
     - Look at the example below. How are we using single and double quotes?

   - Change the content – the `innerHTML`

```javascript
<button onclick='document.getElementById("demo").innerHTML = "Goodbye :-(";'>

Click me</button>
```

https://usabilla.com/blog/how-to-design-for-color-blindness/
Pedagogy Discussion

– Is it an effective strategy for students to type along as you lecture?

– What are some alternatives?
Another example

```html
<button onclick='document.getElementById("demo").
  style.fontSize="35px";'>
  Click me</button>

- Same way to find the element
- **style** indicates a change to CSS
- JavaScript names are similar to, but NOT exactly the same as, CSS
  - CSS: font-size
  - JavaScript: fontSize
- Don’t forget =
- Best to put value in “value” (“35px”)
- Good practice to put ; after value (required in some cases)

Pedagogy sidebar:
  - how interested would your students be in changing HTML styles?
  - Is it realistic for you to know all commands?
  - Consider a “cheat sheet” (maybe online) for the commands students need. Keep it short!
Final example

<button onclick='alert("Button Clicked!");'>
  Click me too!
</button>

- Alert shows a pop-up box
- Very useful to see if button is being clicked
Document Object Model

JavaScript can:
- Change all the HTML elements on the page
- Remove existing HTML elements
- Add new HTML elements
- and more...

https://www.w3schools.com/js/js_htmldom.asp
Important takeaways

- JavaScript and HTML know how to work together
- We’ve learned how to assign an id to an `element` so that JavaScript can locate it
- Advanced/related topic: Document Object Model (DOM)
  - Professional programmers need to understand the DOM... but this workshop will focus more on basic programming constructs
- JavaScript is **CASE SENSITIVE**
- JavaScript responds to **EVENTS**
- **EVENTS** are attached to **HTML ELEMENTS**
Pedagogy Discussion

– Would you expect any gender differences related to any of today’s pedagogy sidebars?

– What would lesson plans look like?

– How would you adapt the lesson plan for students with different abilities?

– What fun things can you do with just what we’ve learned so far?
  – Memes?
  – Greeting cards?

NOTE: We have more material in Day 2 and Day 3, so less time for this type of discussion. But Christine Liebe will give a 30-minute presentation on Broadening Participation in Computing from 4-4:30 tomorrow.
End of Day 1

Time to relax!