Client-side validation

REMEMBER: Client-side JS is NOT secure.

- Fully visible to the user
- Fully alterable by the user

Client-side JS provides **convenience**, not **security**

"Validation" is one such convenience.

What is validation?

- Prevent user from submitting invalid info
- Inform user of needed changes

There are MANY approaches

Does **not replace** server-side validation

But may be the friendly version

Standards-based validation

Some HTML standards to automatically validate

- These standards are pretty minimal

Much validation is JS-based (AND server-based)

Simple example

Our chat application allows empty messages

We can disable the submit button until they have text

No messaging needed -- **discoverability**

Create some Client-side JS

Add to our HTML

```
  <script src="/chat.js"></script>
```

Create a chat.js file **in public/** (static asset)

```
console.log("Hello world");
```

REMEMBER client-side JS is just "text" to the server Client-side JS runs on the browser, not the server

Attempt a small change

```
const sendButton = document.querySelector(".send button");
const toSend = document.querySelector(".to-send");
sendButton.disabled = true;
```

After body

You can set all your JS to run after the page an event says the page is done loading.

But that still pauses the page to get the JS.

Better to not even load it until you have a page.

Place the `<script>` tag just before the closing `</body>` tag

Yay! Except...

You are polluting the global scope

Put your code in an IIFE:

```
(function IIFE() {
  // Your code here
})();
```

Add some complexity

```
( function IIFE() {
  const sendButton = document.querySelector(".send button");
  const toSend = document.querySelector(".to-send");

  sendButton.disabled = !toSend.value;
  toSend.addEventListener('input', (e) => {
    sendButton.disabled = !e.target.value;
  });

})();
```

Are you requiring JS?

Always consider if you're **requiring** client-side JS

JS may or may not be a reasonable requirement

You should consider the cost/benefits

Progressive Enhancement

- it works without JS
- nicer if you have JS