

Common Mistakes & Debugging

- 00 Introducing the Developer Tools
- 01 How to Scale a Paywall
- 02 Common HTML Mistakes
- 03 Common CSS Mistakes

00

Introducing the Developer Tools

The Developer Tools, including the Console, Inspector, and Network tab, are a suite of handy tools that help you look under the hood of any website—and they come free with every browser.

Inspector

Hone in on a single element, its CSS styles, its current state, and more—helps you figure out what’s wrong with a single spot on the page or diagnose a problem.

Console

Run any javascript right in the browser, right on the page that you have loaded. Great for testing your code or snippets.

Network tab

Look at any and every resource downloaded by the page, as well as how long they took. Good for diagnosing slow loads or finding assets.

01

How to Scale a Paywall

The profit model of the web is advertising, or subscription models—but sometimes, we just want to take a peek.

What is a paywall?

HTML

The physical wall is often made up of an HTML element, which blocks the content behind it from being seen.

CSS

CSS can control what can and can't be scrolled on the page. Make the `<main>` content of the page unscrollable, and it can't be seen.

Javascript

Javascript is often used to check if the viewer has the correct permission to view the page, after the page loads—a somewhat sloppy solution because you can pause javascript execution.

02

Common HTML Mistakes

It's easy to forget a closing tag here or there, forget to close the tag itself, or accidentally swap a class with an id

Mistake 0

<p>
<a>
</p>

The unclosed tag

Forgetting a closing tag makes the element continue, and “swallow” up the rest of your HTML content.

Mistake 1

```
<a  
href="google.com"  
Google.com</a>
```

The unfinished tag

Forgetting to close your tags with > will make your tag appear in plaintext on the page.

03

Common CSS Mistakes

It's easy to forget a closing tag here or there, forget to close the tag itself, or accidentally swap a class with an id

Mistake 0

```
a {  
  color: red;  
}
```

Missing end brace

The CSS processor will keep reading and “swallow” up more CSS until the next end brace.

```
p {  
  color: blue;  
}
```

Mistake 1

```
<a id="homelink">  
Google.com</a>
```

Swapping class & ids

A class is for many
elements; an id is for one.

```
.homelink {  
    color: red;  
}
```