

# Client-Side Web Programming: CSS

Copyright © 2026 by  
Robert M. Dondero, Ph.D.  
Princeton University

# Objectives

- We will cover:
  - How to make good looking **responsive** web apps

# Objectives

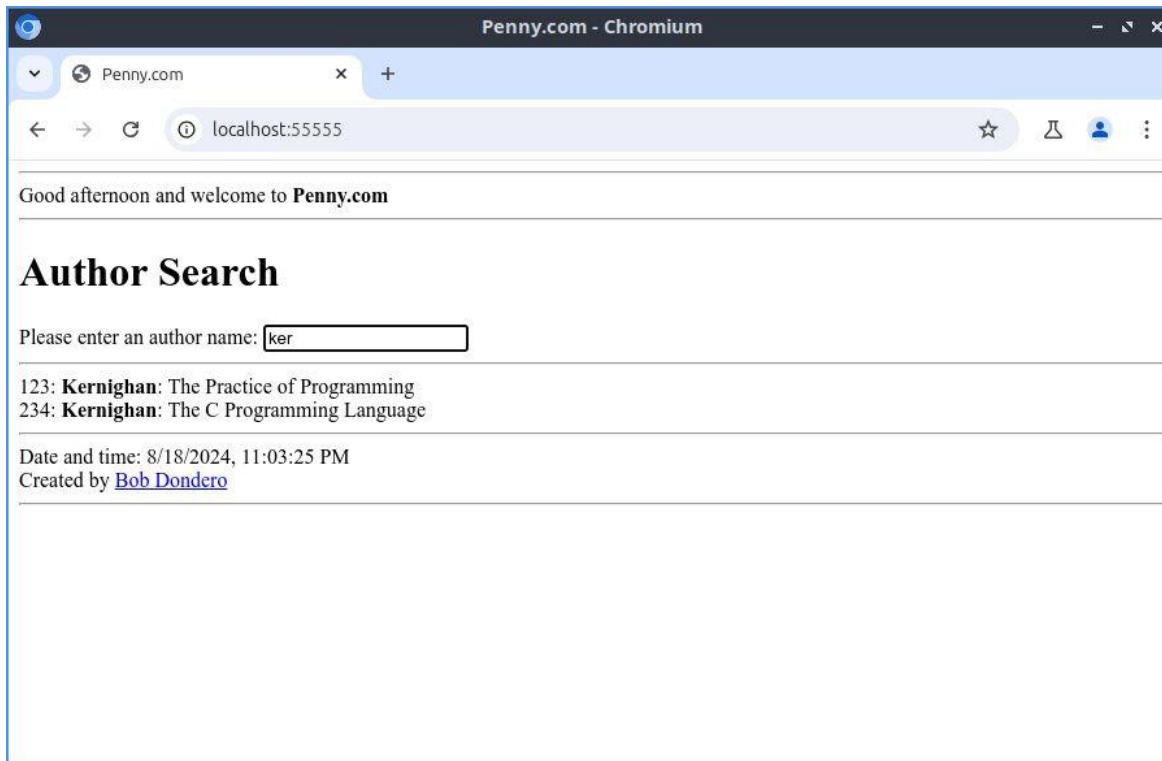
- We will cover:
  - The smartphone viewport
  - Cascading style sheets (CSS)
  - Responsive web apps
  - CSS libraries

# Agenda

- **The smartphone viewport**
- CSS
- Responsive web apps
- CSS libraries

# Smartphone Viewport

- Recall PennyAjax4 app
  - Laptop computer (large browser window)



OK

# Smartphone Viewport

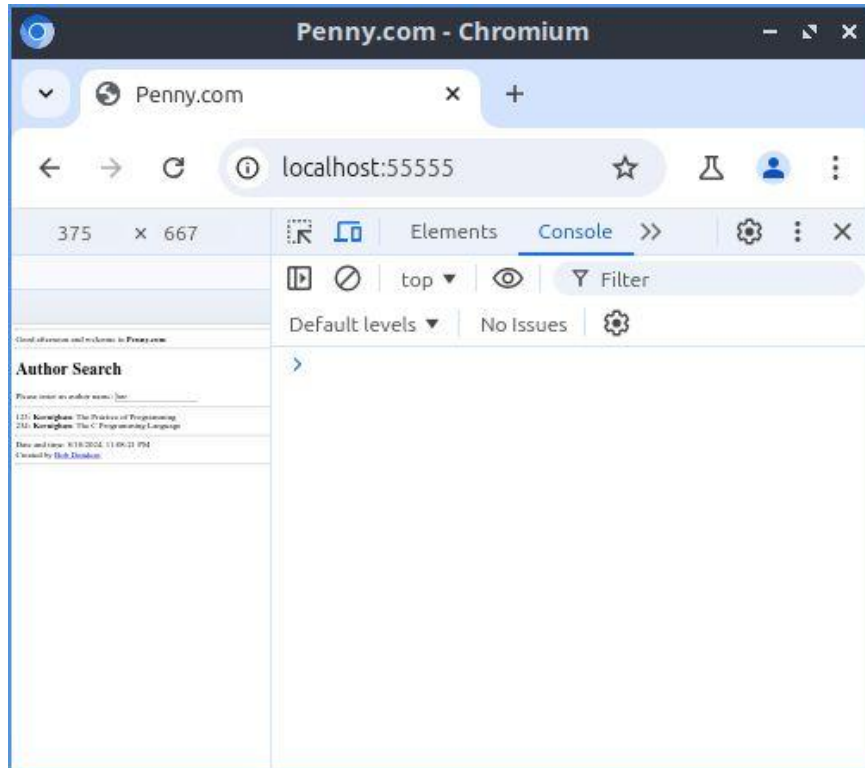
- Recall PennyAjax4 app (cont.)
  - Laptop computer (small browser window)



OK

# Smartphone Viewport

- Recall PennyAjax4 app (cont.)
  - Smartphone (very small browser window)



**Chrome:** More tools -> Developer Tools

**Unacceptable**

# Smartphone Viewport

**Original problem:** Truncation

Laptop Window

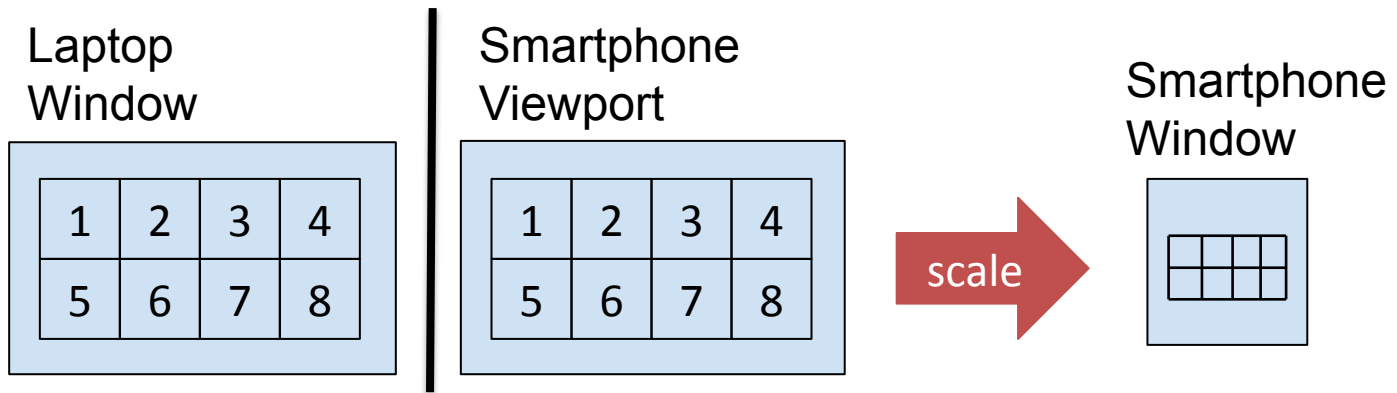
1	2	3	4
5	6	7	8

Smartphone Window

1	2	3	4
5	6	7	8

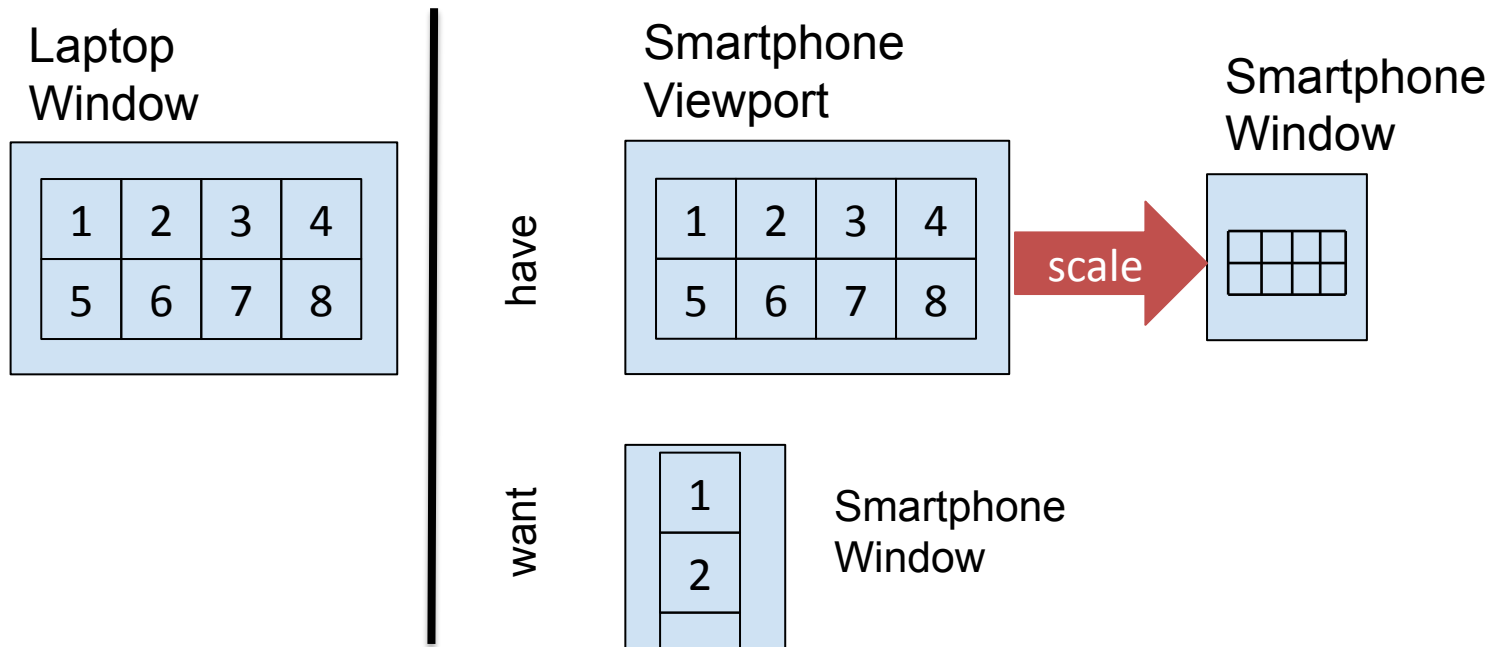
# Smartphone Viewport

Original solution: Smartphone *viewport*



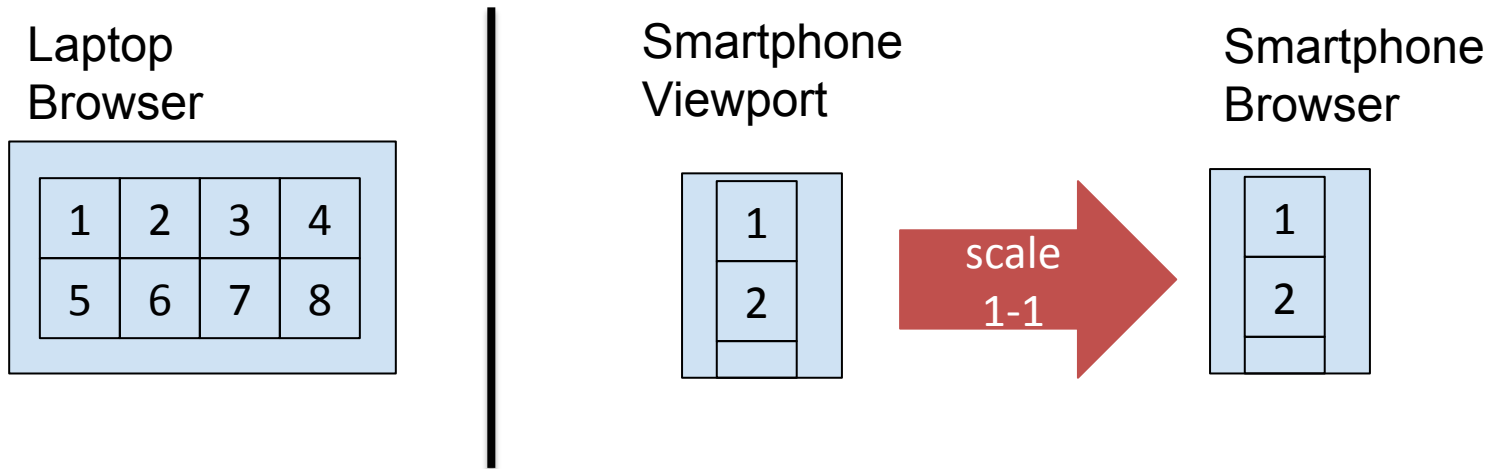
# Smartphone Viewport

**Current problem:** Viewport inappropriate for responsive apps



# Smartphone Viewport

**Current solution:** Bypass viewport concept

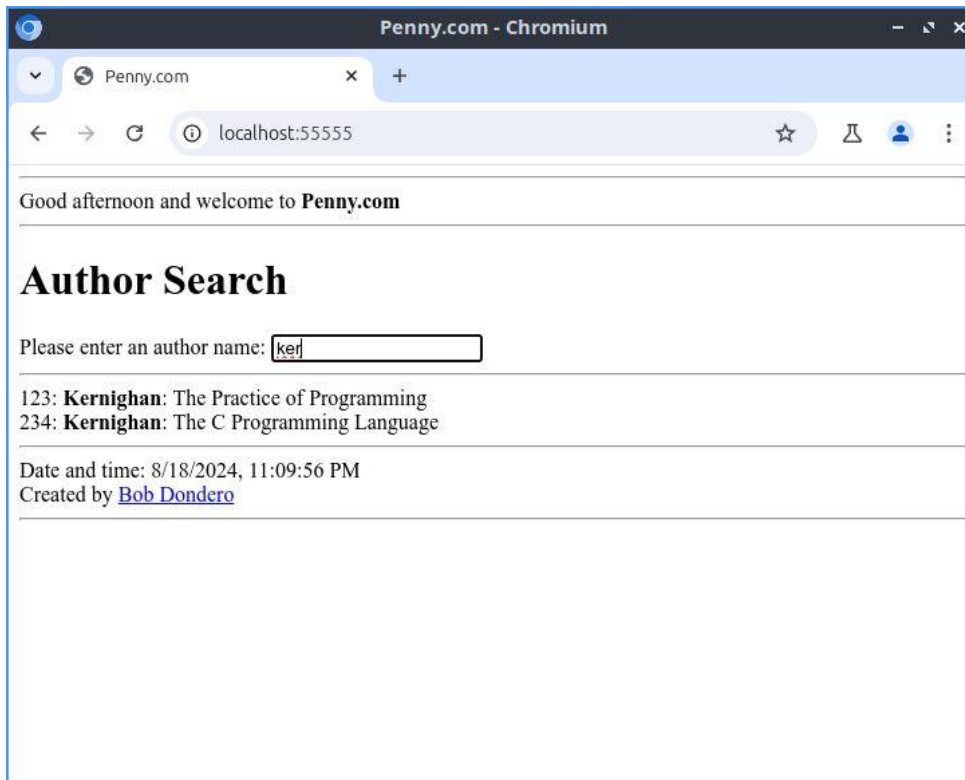


# Smartphone Viewport

- See **PennyViewport** app
  - runserver.py
  - penny.sql, penny.sqlite
  - database.py
  - **penny.py** (same as previous)
  - **index.html**

# Smartphone Viewport

- See **PennyViewport** app (cont.)
  - Laptop computer (large browser window)



OK

# Smartphone Viewport

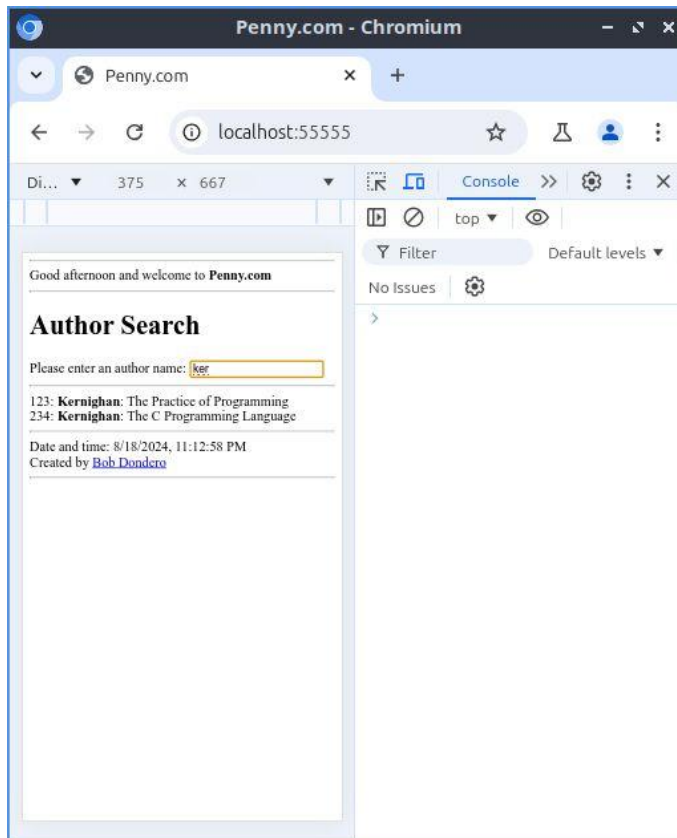
- See **PennyViewport** app (cont.)
  - Laptop computer (small browser window)



OK

# Smartphone Viewport

- See **PennyViewport** app (cont.)
  - Smartphone (very small browser window)



Maybe OK?

# Agenda

- The smartphone viewport
- **CSS**
- Responsive web apps
- CSS libraries

# CSS

- **Problem:** (Prior to CSS)
  - Specification of style info was repetitive & difficult to maintain

somefile.html

```
<html>
  ...
  <body>
    <font color="red"><h1>First Heading</h1></font>
    ...
    <font color="red"><h1>Second Heading</h1></font>
    ...
  </body>
</html>
```

# CSS

- **Solution:** Cascading Style Sheets (CSS)
  - W3C standard
  - Defines rules that browser uses to render HTML elements

# CSS

- ***External rules***: rules in an external style sheet

mystyle.css

```
h1 {color:red;}
```

somefile.html

```
<html>
  <head>
    <link rel="stylesheet"
          href="mystyle.css">
    ...
  </head>
  <body>
    <h1>First Heading</h1>
    ...
    <h1>Second Heading</h1>
    ...
  </body>
</html>
```

# CSS

- ***Embedded rules***: rules in an embedded style sheet

somefile.html

```
<html>
  <head>
    <style>
      h1 {color:red;}
    </style>
    ...
  </head>
  <body>
    <h1>First Heading</h1>
    ...
    <h1>Second Heading</h1>
    ...
  </body>
</html>
```

# CSS

- *Inline rules*

somefile.html

```
<html>
  <body>
    <h1 style="color:red;">
      First Heading
    </h1>
    ...
    <h1 style="color:red;">
      Second Heading
    </h1>
    ...
  </body>
</html>
```

# CSS

- The browser “cascade”
  - To style a HTML document, browser applies rules in this order...

# CSS

Rules created by the browser authors  
Rules created by the browser user  
Rules created by the document authors  
    External rules  
    Embedded rules  
    Inline rules

Rules created by the document author  
    Rules with element selectors  
    Rules with class selectors  
    Rules with ID or attribute selectors

# CSS

- CSS rule syntax:

```
selector {  
    property: value;  
    property: value;  
    ...  
}
```

# CSS

```
selector {  
    property:value;  
    property:value;  
    ...  
}
```

# CSS

From [https://www.w3schools.com/cssref/css\\_selectors.asp](https://www.w3schools.com/cssref/css_selectors.asp)

Selector	Example	Example Description
<i>.class</i>	.xxx	Select all elements with a <code>class</code> attribute whose value is <code>xxx</code>
<i>#id</i>	#xxx	Select the element with an <code>id</code> attribute whose value is <code>xxx</code>
*	*	Select all elements
<i>element</i>	p	Select all <code>&lt;p&gt;</code> elements
<i>element, element</i>	div, p	Select all <code>&lt;div&gt;</code> elements and all <code>&lt;p&gt;</code> elements
<i>element element</i>	div p	Select all <code>&lt;p&gt;</code> elements inside <code>&lt;div&gt;</code> elements
<i>element&gt;element</i>	div > p	Select all <code>&lt;p&gt;</code> elements where the parent is a <code>&lt;div&gt;</code> element

# CSS

From [https://www.w3schools.com/cssref/css\\_selectors.asp](https://www.w3schools.com/cssref/css_selectors.asp)

Selector	Example	Example Description
<code>[attribute]</code>	<code>[type]</code>	Select all elements with a <code>type</code> attribute
<code>[attribute=value]</code>	<code>[type=text]</code>	Select all elements with a <code>type</code> attribute whose value is <code>text</code>

And **many** more (58 of them)

# CSS

```
selector {  
    property: value;  
    property: value;  
    ...  
}
```

# CSS

From <https://www.w3schools.com/cssref/default.asp>

Property	Example	Example Description
<code>font-family</code>	<code>font-family:Arial</code>	Set the font family of text within the selected element(s) to Arial
<code>font-size</code>	<code>font-size:20px</code>	Set the font size of text within the selected element(s) to 20 pixels
<code>color</code>	<code>color:white</code>	Set the color of text within the selected element(s) to white
<code>background-color</code>	<code>background-color:#295078</code>	Set the background color of the selected element(s) to r:29, g:50, b:78
<code>text-align</code>	<code>text-align:center</code>	Set the horizontal alignment of text within the selected element(s) to center
<code>border-style</code>	<code>border-style:solid</code>	Set the style of the four borders of the selected element(s) to solid
<code>border-color</code>	<code>border-color:gray</code>	Set the color of the four borders of the selected element(s) to gray
<code>border-width</code>	<code>border-width:1px</code>	Set the width of the four borders of the selected element(s) to 1 pixel

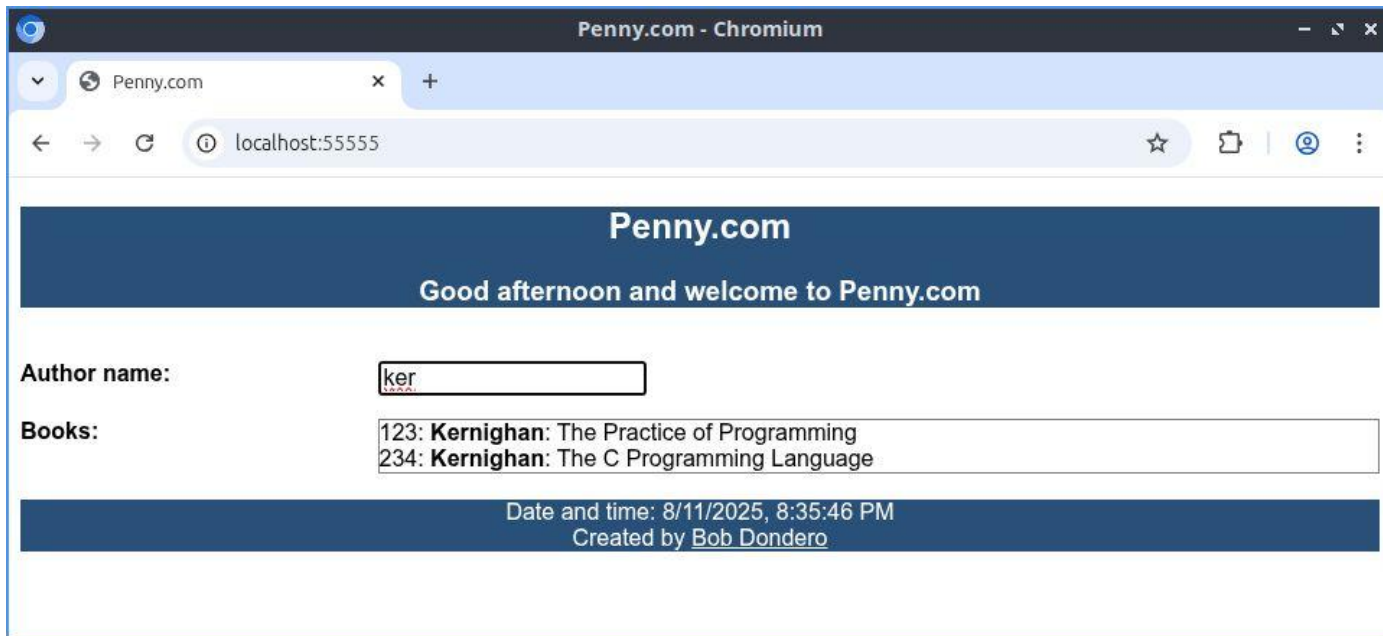
And **many** more (over 200 of them)

# CSS

- See **PennyCss** app
  - runserver.py
  - penny.sql, penny.sqlite
  - database.py
  - penny.py
  - **static/penny.css**
  - **index.html**

# CSS

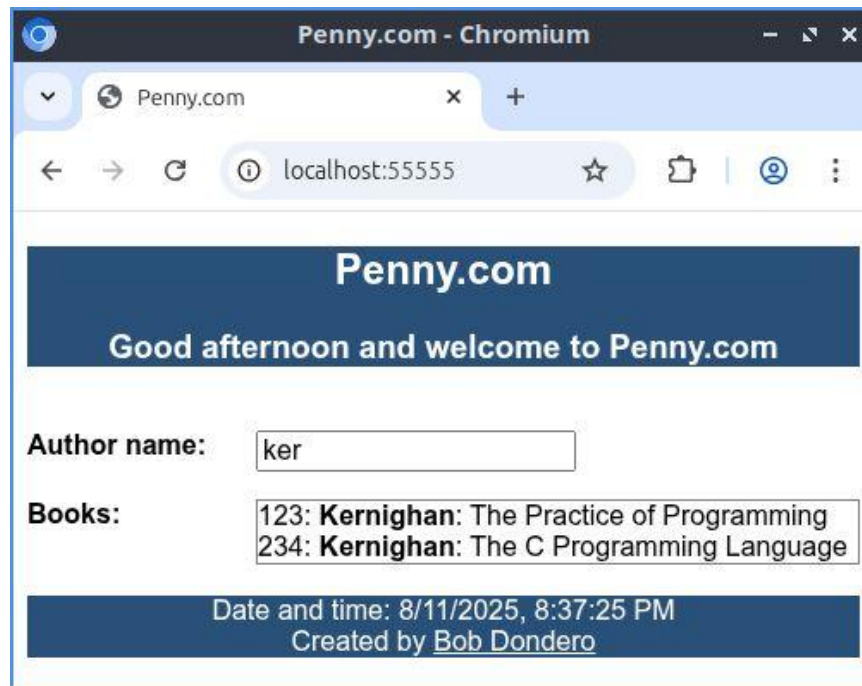
- See **PennyCss** app (cont.)
  - Laptop computer (large browser window)



Good

# CSS

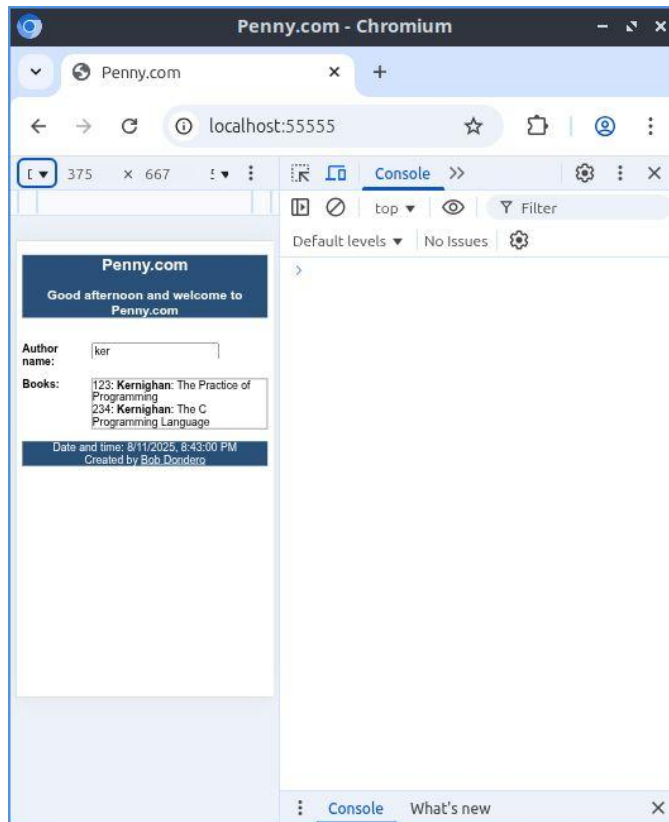
- See **PennyCss** app (cont.)
  - Laptop computer (small browser window)



OK

# CSS

- See **PennyCss** app (cont.)
  - Smartphone (very small browser window)



Maybe  
OK

# Agenda

- The smartphone viewport
- CSS
- **Responsive web apps**
- CSS libraries

# Responsive Apps

- **Problem:**
  - Web apps should adjust themselves based upon browser characteristics
- **Solution:**
  - Responsive web apps

# Responsive Apps

- Responsive web app
  - Rearranges tables
    - Large window => multiple columns
    - Small window => fewer columns
  - Collapses menus
    - Large window => ordinary menu
    - Small window => drop-down menu
  - Hides some content
    - Large window => display all content
    - Small window => hide some content
  - Etc.

# Responsive Apps

- See **PennyResponsive** app (cont.)
  - runserver.py
  - penny.sql, penny.sqlite
  - database.py
  - penny.py
  - **static/penny.css**
  - index.html (same as previous)

# Responsive Apps

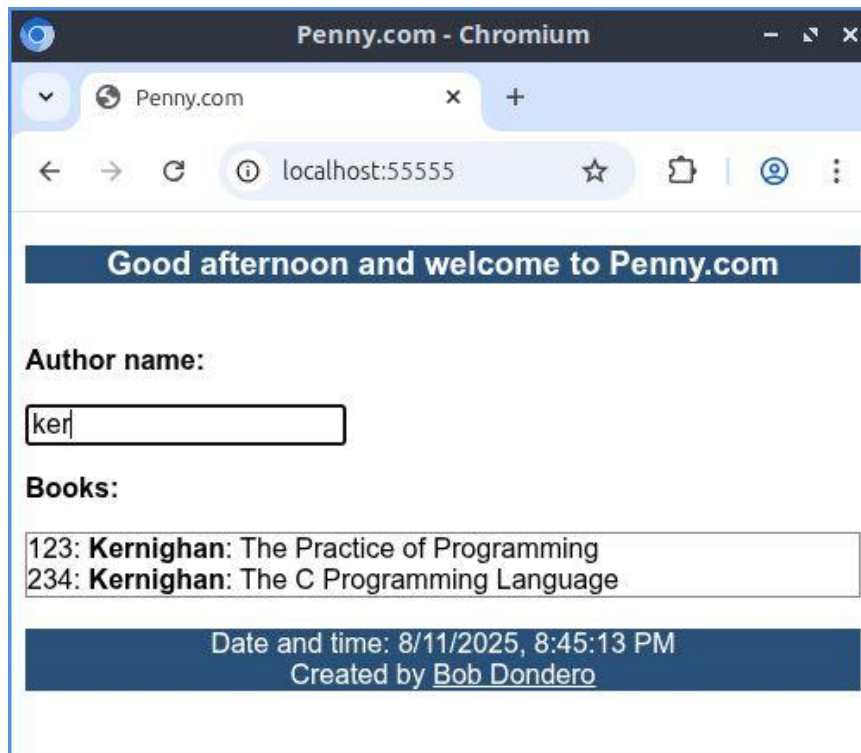
- See **PennyResponsive** app (cont.)
  - Laptop computer (large browser window)



Good

# Responsive Apps

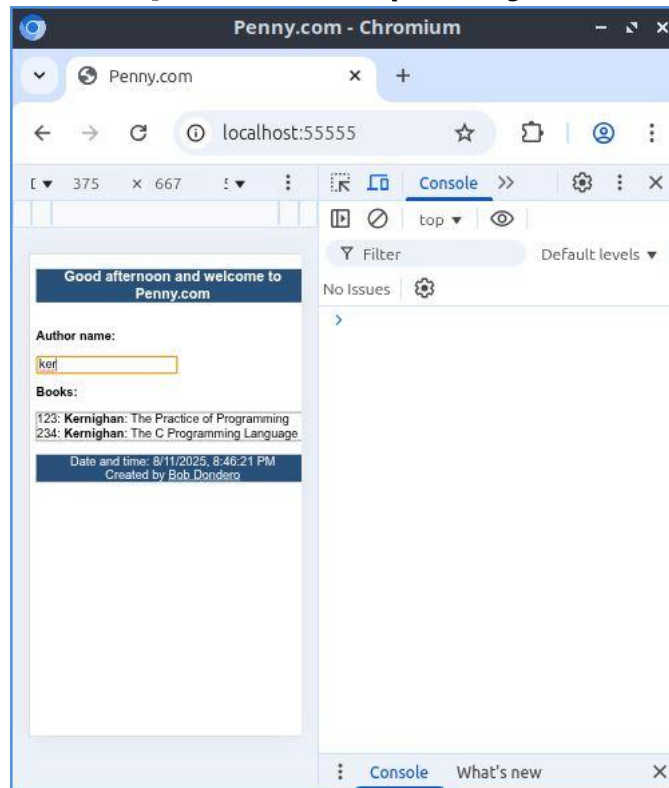
- See **PennyResponsive** app (cont.)
  - Laptop computer (small browser window)



Good!

# Responsive Apps

- See **PennyResponsive** app (cont.)
  - Smartphone (very small browser window)



Good!!!

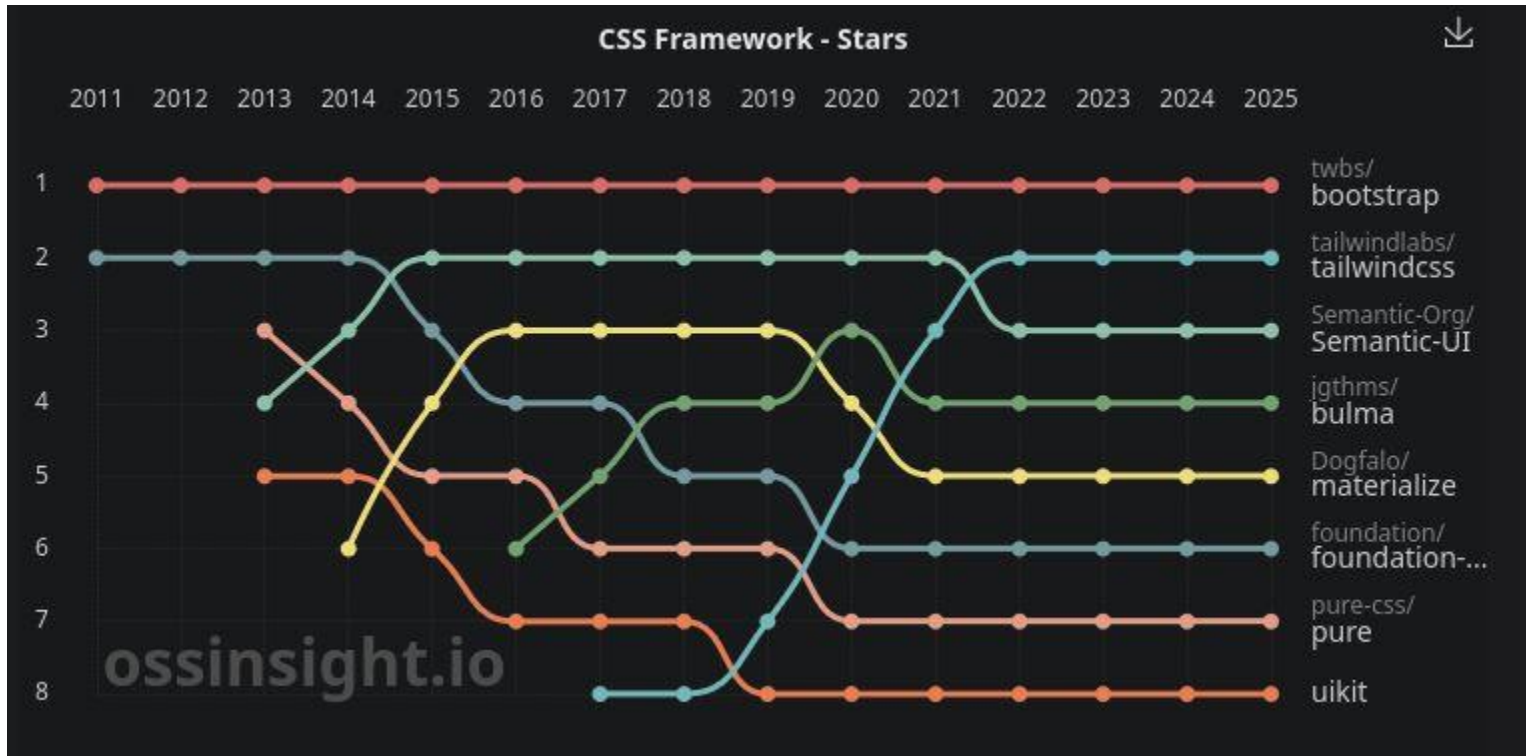
# Agenda

- The smartphone viewport
- CSS
- Responsive web apps
- **CSS libraries**

# CSS Libraries

- **Problem:**
  - Composing CSS rules is tedious
    - Redundancy across apps
- **Solution:** *CSS libraries*
  - Define CSS rules

# CSS Libraries



Fetches 10/25 from

<https://ossinsight.io/collections/css-framework/>

# CSS Libraries

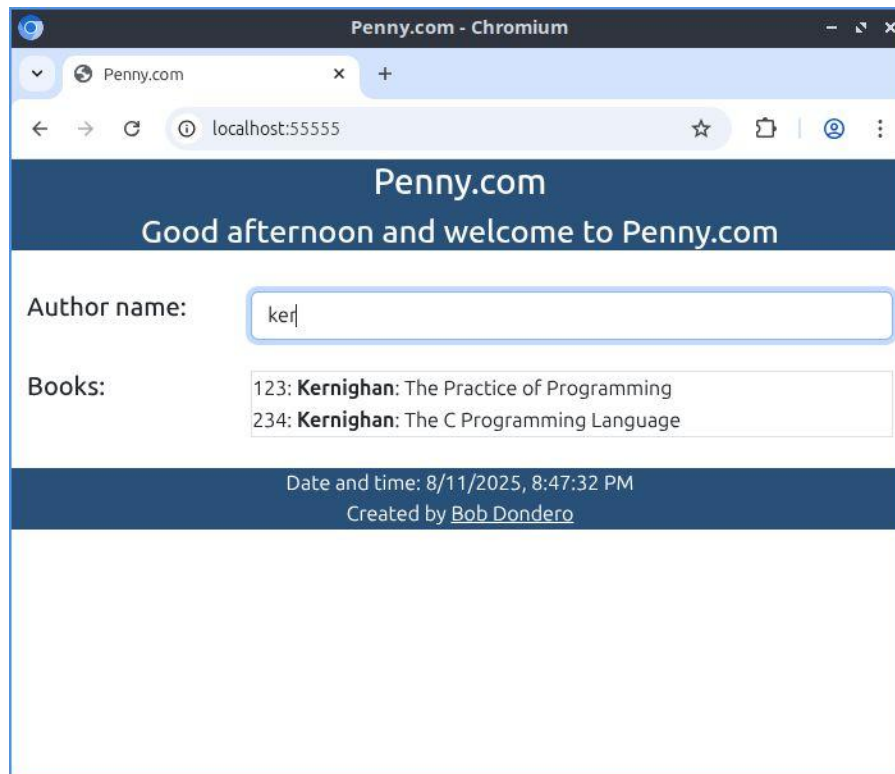
- ***Bootstrap***
  - From X (Twitter)
  - The most popular CSS framework
- ***W3.CSS***
  - From W3Schools
  - Not popular, but simple & pleasant

# CSS Libraries: Example

- See **PennyBootstrap** app
  - runserver.py
  - penny.sql, penny.sqlite
  - database.py
  - penny.py
  - ~~- static/penny.css~~
  - **index.html**

# CSS Libraries: Example

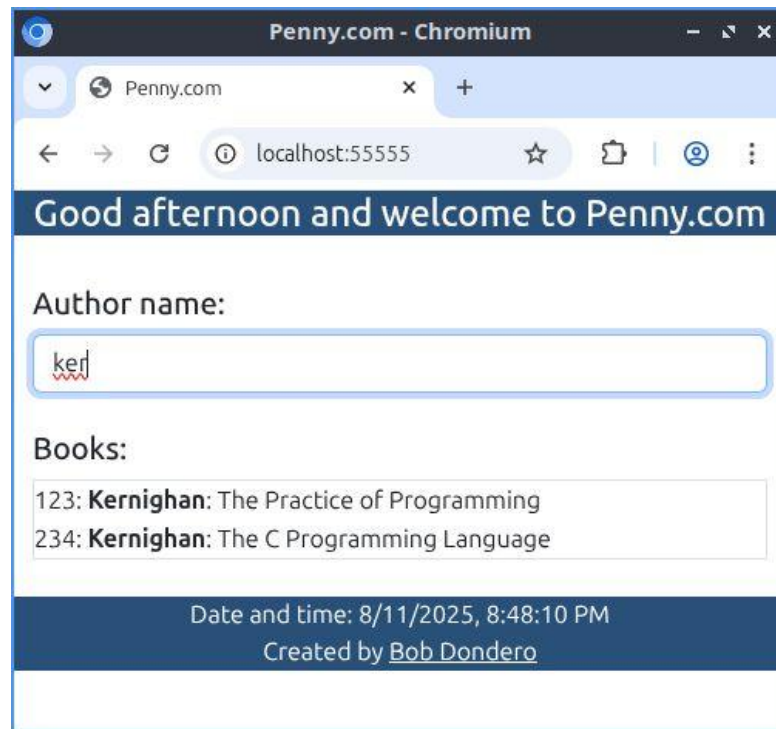
- See **PennyBootstrap** app (cont.)
  - Laptop computer (large browser window)



Good

# CSS Libraries: Example

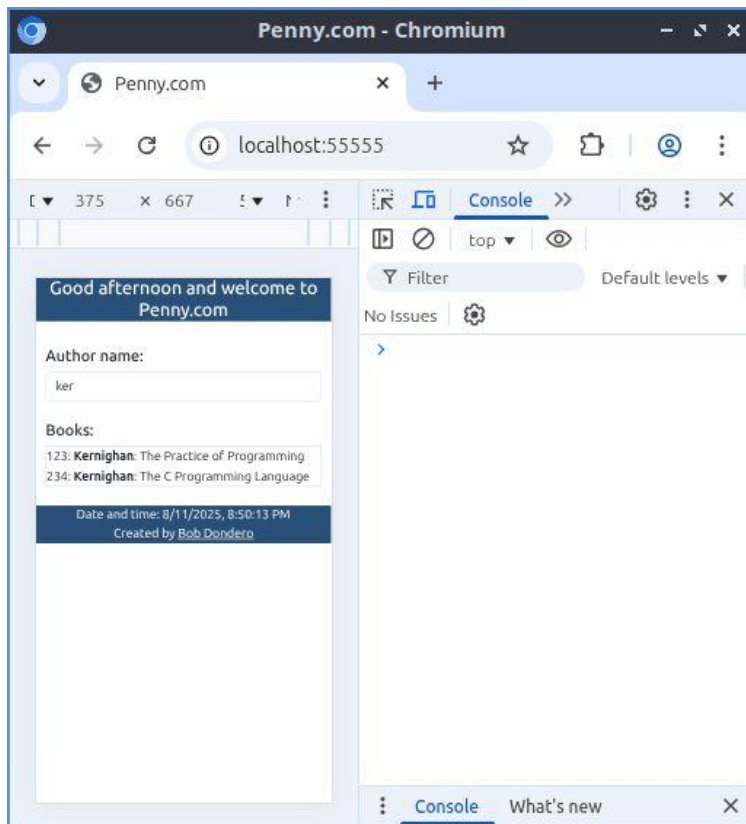
- See **PennyBootstrap** app (cont.)
  - Laptop computer (small browser window)



Good!

# CSS Libraries: Example

- See **PennyBootstrap** app (cont.)
  - Smartphone (very small browser window)



Good!!!

# Aside: Bootstrap Modals

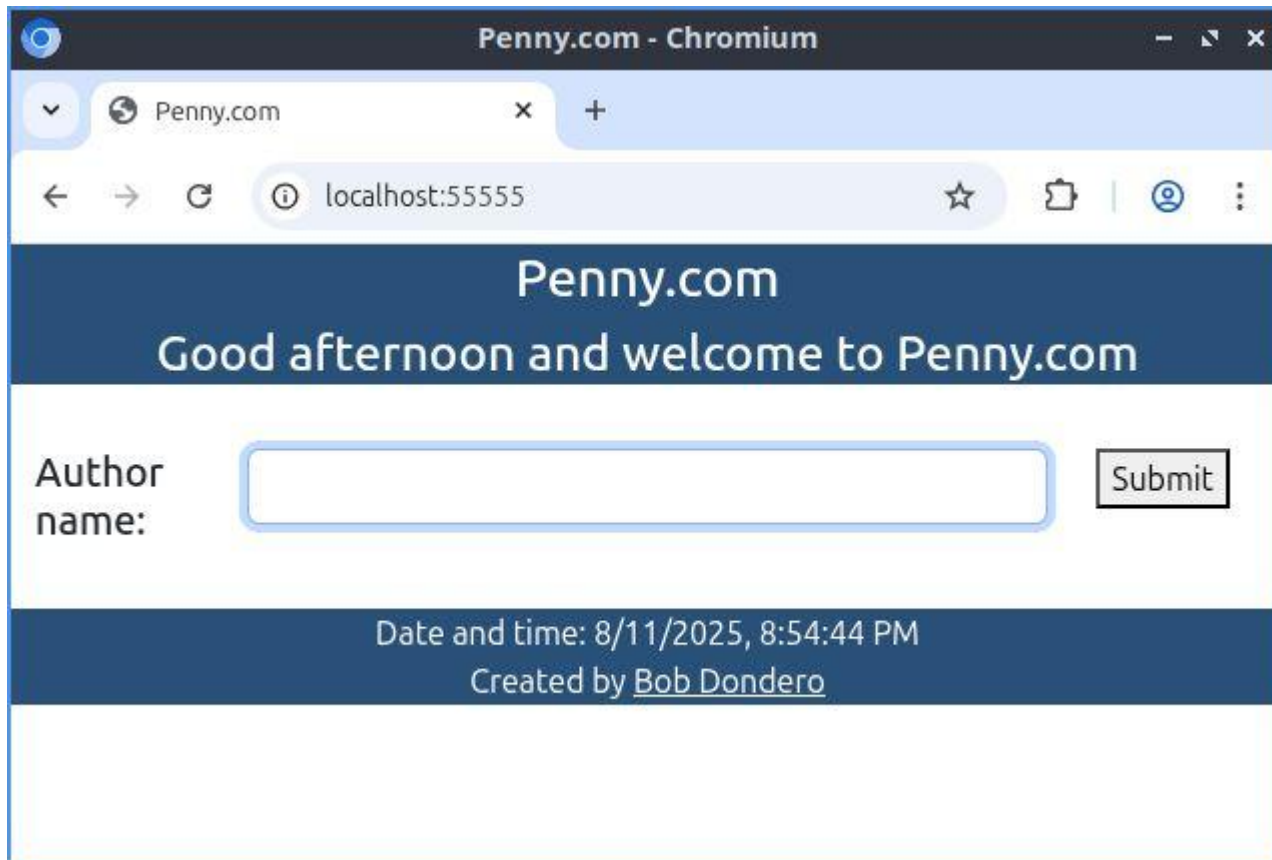
- **Aside:** *modal*
  - Alias *modal window*
  - Alias *modal dialog*
  - Alias *modal dialog box*
  - A window that:
    - Is subordinate to the main window
    - Typically occludes (part of) the main window
    - Disables user interaction with the main window until dismissed

# Aside: Bootstrap Modals

- **Aside:** modals in Bootstrap
  - An inferior but potentially useful version of Penny...

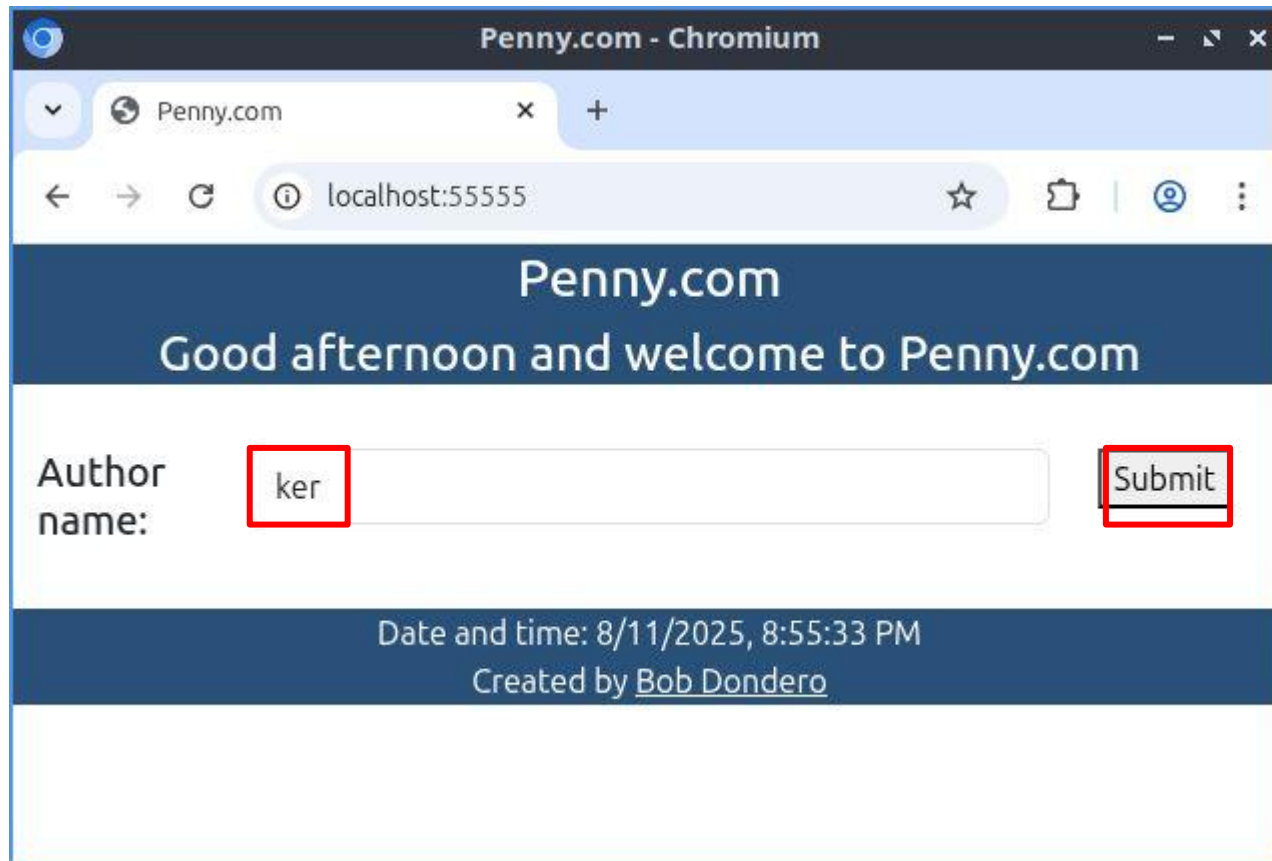
# Aside: Bootstrap Modals

- See **PennyModal** app



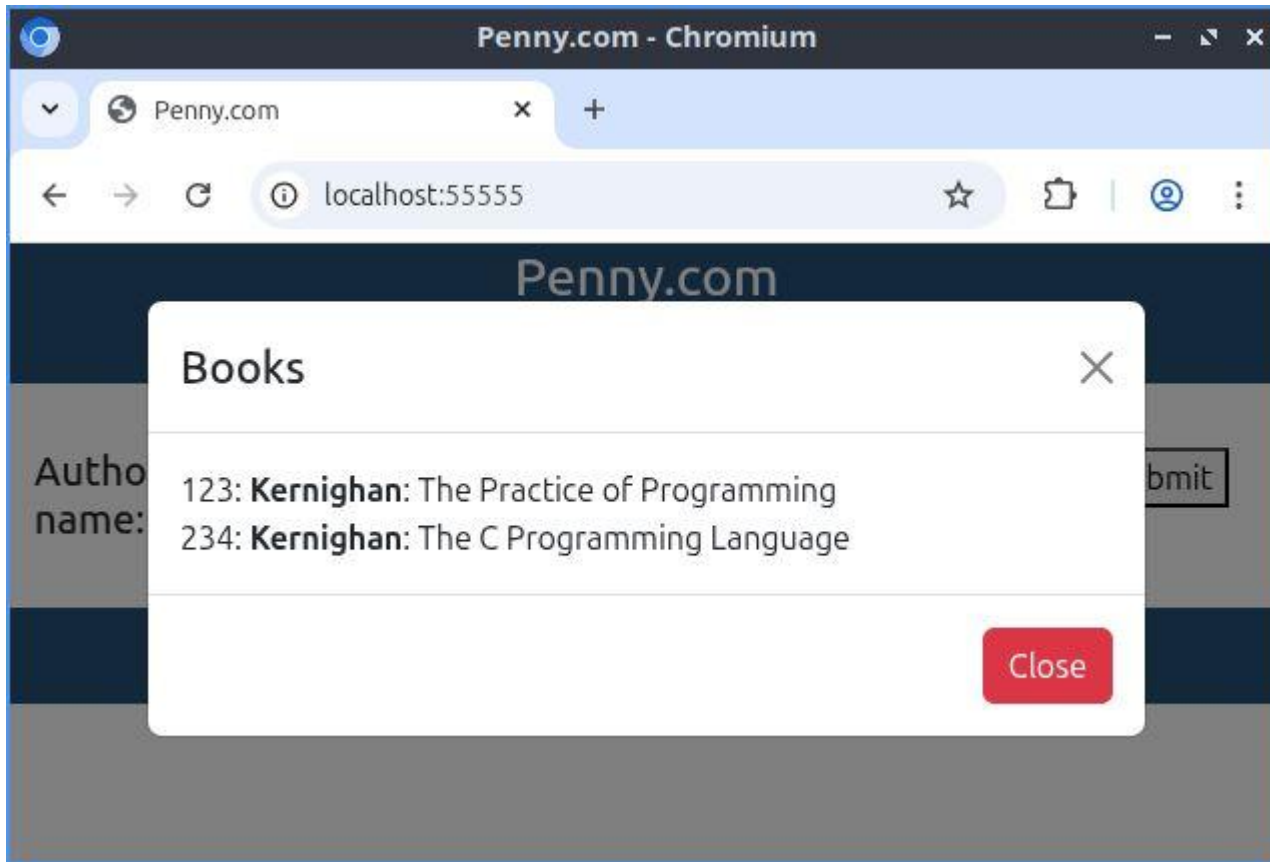
# Aside: Bootstrap Modals

- See PennyModal app (cont.)



# Aside: Bootstrap Modals

- See **PennyModal** app (cont.)



# Aside: Bootstrap Modals

- See **PennyModal** app (cont.)
  - runserver.py
  - penny.sql, penny.sqlite
  - database.py
  - penny.py
  - **index.html**
    - Choose indexRaw.html, indexjQuery.html, or indexReact.html

# Aside: Bootstrap Modals

- See **PennyModal** app (cont.)
  - **indexRaw.html**
    - Uses Bootstrap CSS + Bootstrap JavaScript
  - **indexjQuery.html**
    - Uses Bootstrap CSS + Bootstrap JavaScript + jQuery
  - **indexReact.html**
    - Uses Bootstrap CSS + *React Bootstrap JavaScript* + React

# Lecture Summary

- In this lecture we covered:
  - The smartphone viewport
  - Cascading style sheets (CSS)
  - Responsive web apps
  - CSS libraries
    - Bootstrap

# More Information

- The COS 333 *Lectures* web page provides references to supplementary information