

# **CS50 Section 8**

**Slides and Exercises at [github.com/cjleggett/2022-section](https://github.com/cjleggett/2022-section)**

**Think.**

**Pair.**

**Share.**

# Think. Pair. Share.

- What is HTTP?
- Why do we use HTML/CSS/JavaScript?

# Outline

- Upcoming Schedule (3:05)
- HTTP (3:10)
- HTML (3:15)
- CSS (3:25)
- JavaScript (3:35)
- Test Prep (3:55)

# Upcoming Schedule

# 11/7 - 11/13: Finance + Proposal

- Normal pset, lab, section
- Maybe the most time-consuming pset
- But also one of the most fun!
- **Also time to get into groups and start thinking about final project!**

# 11/14 - 11/20: Test

- Test released 11/14
- Test Due 11/20
- No Section / Lecture / Lab
- No problem set
- Harvard-Yale is 11/19!



**Saturday, November 19 • Harvard Stadium**

# 11/21 - 11/27: Break

- Monday Lecture
- Thanksgiving Break!
- No problem set / section / lab
- You should start working on your final project!



# 11/28 - 12/4: Final Project Sprint

- No Lecture / Section / Pset / Lab
- 12/1 - 12/2: CS50 Hackathon!



# 12/5 - 12/7: Finishing Up

- No Lecture / Section / Pset / Lab
- 12/6: Final Projects Due
- 12/7: CS50 Fair

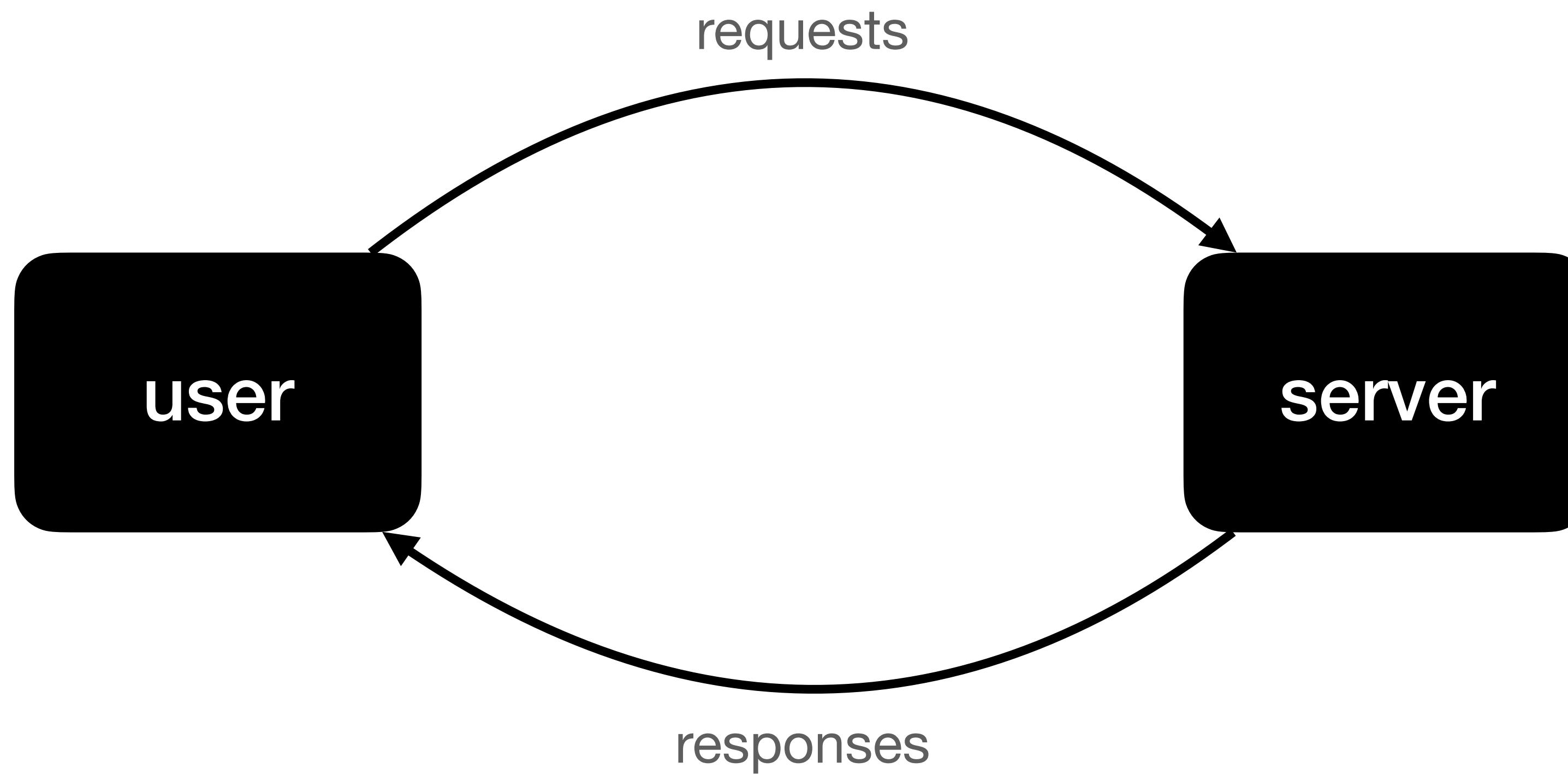


**HTTP**

# Hypertext Transfer Protocol

- A **Protocol** for how different users and websites interact
- Made up of **requests** and **responses**

# Hypertext Transfer Protocol



# Requests

- Type
  - GET: give me information
  - POST: i want to add/edit information
- Host (website we want to go to)
- Data
  - Form Data
  - Params
  - ...

# Responses

- Response Codes
  - 200 = OK
  - 404 = Not Found
  - 500 = Internal Server Error
- Content Type
- Content

# **HTML**

# What is HTML?

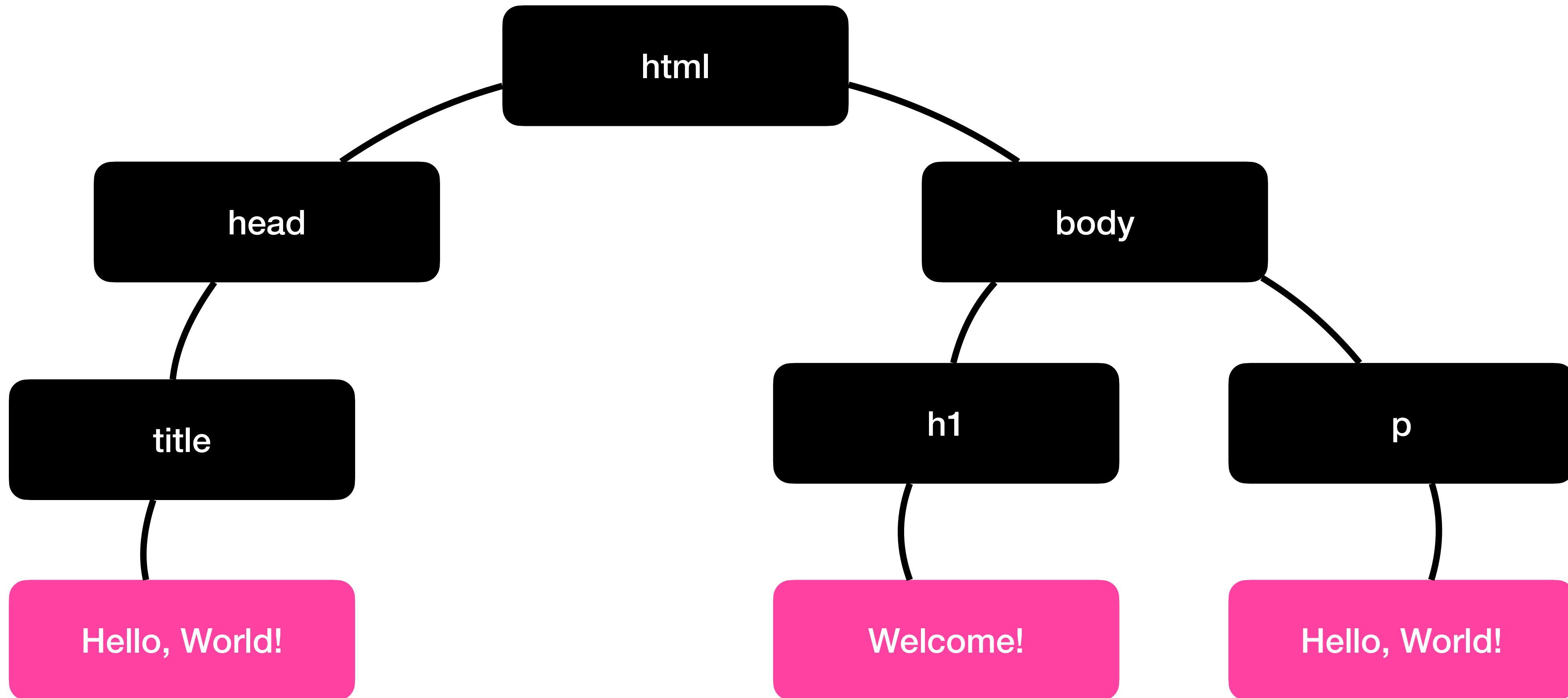
- Hypertext Markup Language
- A standard way of formatting text, images, etc.
- Your browser knows how to translate HTML -> Visuals!

# Example HTML Page

```
<!DOCTYPE html>

<html lang="en">
  <head>
    <title>Hello, World!</title>
  </head>
  <body>
    <h1>Welcome to my Website!</h1>
    <p>Hello, World!</p>
  </body>
</html>
```

# Document Object Model



# Lots of HTML Element Types

- Buttons
- Links
- Images
- Tables
- ...

# HTML Element Attributes

- id
- class
- style
- href
- ...

# **CSS**

# What is CSS?

- Cascading Style Sheets
- A way to specify how you want your webpage to look
- Three Methods for including CSS:
  - style attribute in an HTML tag
  - style element in head
  - separate CSS file

# What is CSS?

- Cascading Style Sheets
- A way to specify how you want your webpage to look
- Three Methods for including CSS:
  - style attribute in an HTML tag
  - style element in head
  - **separate CSS file**

# CSS

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

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

# CSS

CSS Selector

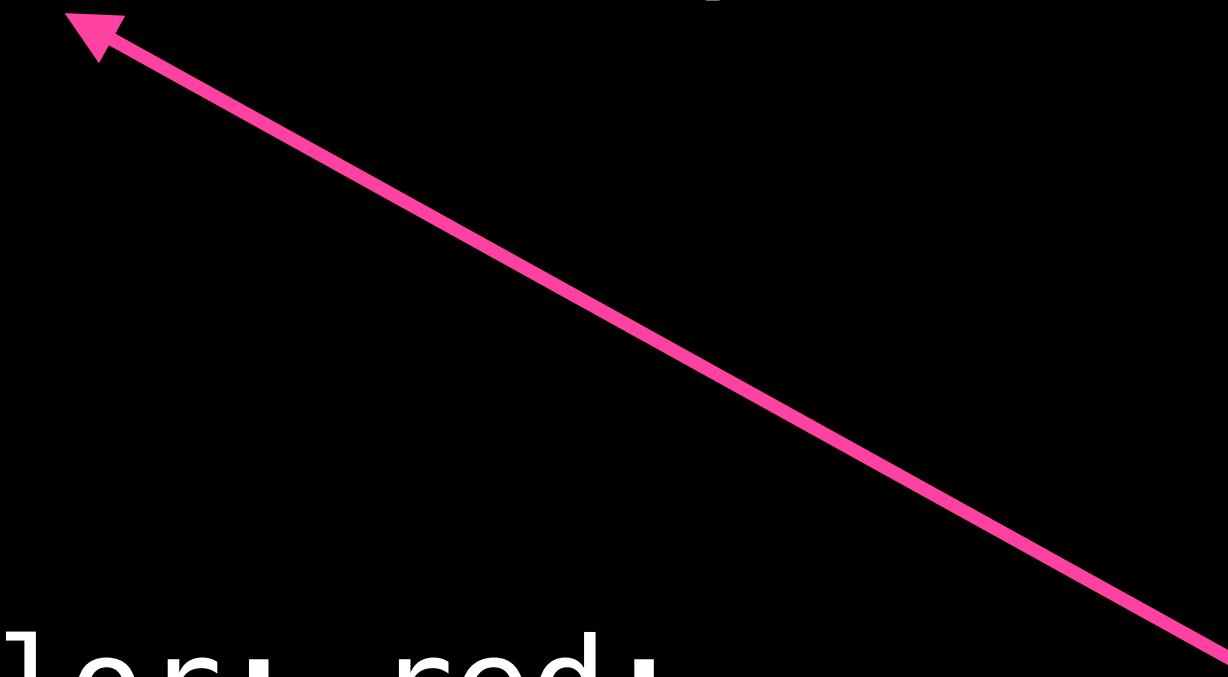
```
p  
{  
    color: blue;  
}
```

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



# CSS

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



Attribute

# CSS

```
p  
{  
    color: blue; ←  
}  
  
h1  
{  
    color: red;  
}
```



# CSS: Select By Element Type

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

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

# CSS: Select By Class Name

```
.blue-text  
{  
    color: blue;  
}  
  
h1  
{  
    color: red;  
}
```

# CSS: Select By id

```
#special-text
{
    color: blue;
}

h1
{
    color: red;
}
```

a, b	Multiple Element Selector
a b	Descendant Selector
a > b	Child Selector
a + b	Adjacent Sibling Selector
[a=b]	Attribute Selector

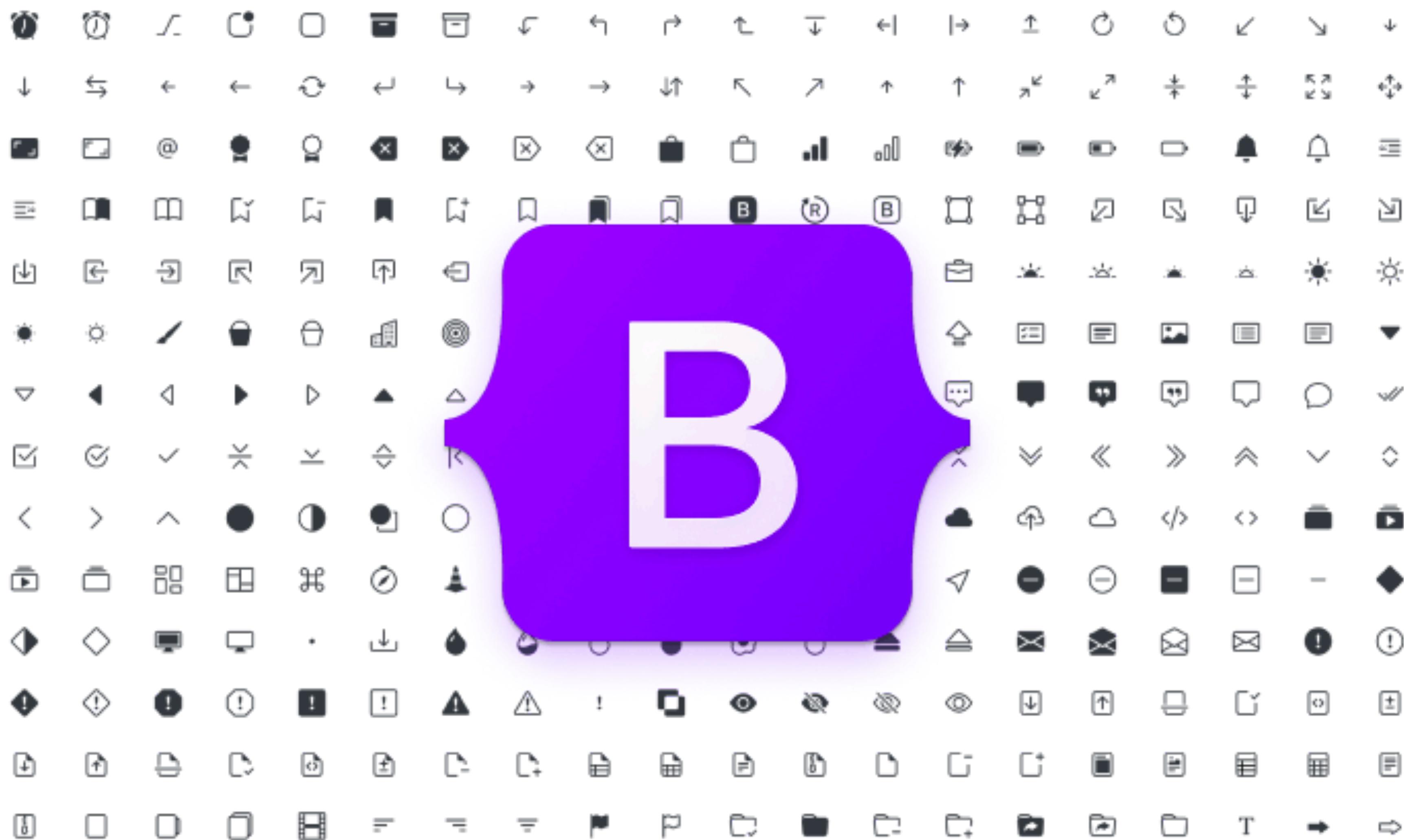
# CSS: Advanced Selectors

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

h1
{
    color: red;
}
```

# Bootstrap

# Bootstrap



# What is Bootstrap?

- Essentially, a CSS file someone's written for you
- An quicker way to get to nice-looking website
- Get started [here](#)
- Check out elements [here](#)

# JavaScript

# JavaScript

- Another programming language?!?!
- Somewhere between C and Python
- Specifically built to interact with HTML
- It's how we make websites interactive!

# Syntax

- Structurally Similar to C
- No need to declare variable types
- Functions look a bit different
- I'm not going to spend much time on syntax

# Key JavaScript Functions

- `document.querySelector(query)`
- `element.addEventListener(event, function)`

# General JavaScript Pattern

1. Find an element
2. Assign an event listener to that element
3. When the event is triggered:
  - A. Find the element you want to change (could be the same or different)
  - B. Set an attribute of that element to a new value

# Important!

- We want to make sure to only run our JavaScript code after the page has loaded
- To do this, we'll add an event listener for the page loading

```
document.addEventListener('DOMContentLoaded', function() {  
    // Your code here  
})
```

# Test Prep

# From 2020 test:

Recall that it's possible to change the HTML, and even CSS, of a web page via Chrome's **Developer Tools**. But, in no more than three sentences, why is that not considered "hacking"? Put another way, why is that not considered a threat to the website?

# From 2020 Test

**<https://cs50.harvard.edu/college/2020/fall/test/cats/>**

# From 2020 Test (stuff we don't learn!)

<https://cs50.harvard.edu/college/2020/fall/test/logical/>

# From 2020 test

**<https://cs50.harvard.edu/college/2020/fall/test/finsta/>**

# From 2018 test

**<https://cdn.cs50.net/2018/fall/quiz/alarm.html>**