COMPSCI 326 - Web Programming
Basic CSS and Grids

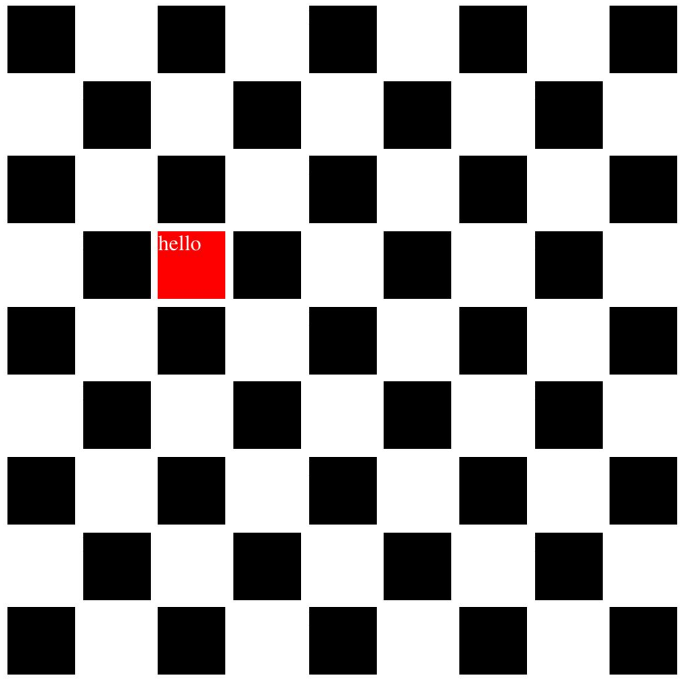
join on the Slack #q-and-a channel as well as Zoom
remember, you can ask questions of your teammates on your group Slack!
please **turn on your webcam** if you can
mute at all times when you aren't asking a question
(https://docs.google.com/document/d/1uUl8I6zmLck58o9ZPODGL9xR4graNYnRXCdrisM5Dyo/edit?usp=sharing)

Today: Basic CSS and grids

resources:

- CSS (with classes and ids) - https://developer.mozilla.org/en-US/docs/Web/CSS
- CSS pseudo-classes (e.g., hover) - https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes
- CSS grids - https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout

Exercise today: heading towards this checkerboard (demo)



CSS = "cascading style sheets"

- totally stupid name
- think of it as <STYLE> while <HTML> is *structure*
- e.g., color, height, width, background-color...

CSS has rules that work like this:

- SELECTOR { STYLES }
  - "selector" == WHAT do you want to set the style of
  - then { ... } == HOW you want them styled
    - example:
      ```
      <STYLE>
      p {
        color : green;
      ```

    }
    </STYLE>

    - you can also make your "stylesheet" a separate file, so `<link href="/path/to/style.css" rel="stylesheet">` can be added to the head element
    - In-line (not preferred):
      <p style="color:green">GREEN MONSTAH</p> *this is for just one P, but the other is for EVERY P -- we will see how you can be more precise*
- you can set many styles automatically at once

  ```
  p {
    color: green;
    background-color: yellow;
    font-size: 16pt;
  }
  ```

  sets every <p>*whatever*</p>
- **very useful:**
  <DIV> = a container, breaks a line (nothing visual, just lets you set style for stuff inside)
  <SPAN> = a container, doesn't break the line (as above)

other styles you can set:

- color -- can be a name, or rgb(n,n,n), or #FOFOFO, OR rgba(red,green,blue, ALPHA) = opacity
- background-color
- background-image
- font-size, family (like "serif", "sans-serif", or "Times New Roman"), weight (like "bold"), style (like "italic"), and other fun stuff like font-variant (small caps)
- width, height - for "content" -- **px** or inches ("em" "en")

so what's this CASCADING thing?

- basically, styles are INHERITED from their parent (the CSS declaration)
  <p>this is green and big<p style="color:red">this is red and big</p></p>

CSS can match ordinary HTML elements like P but also:

- **classes**
  <span **class**="pretty">...</span>
  - these you match with a DOT:

```
.pretty {
  color: pink;
}
```

- **ids**

  <span **id**="kitty">...</span>

  - these you match with a HASH:

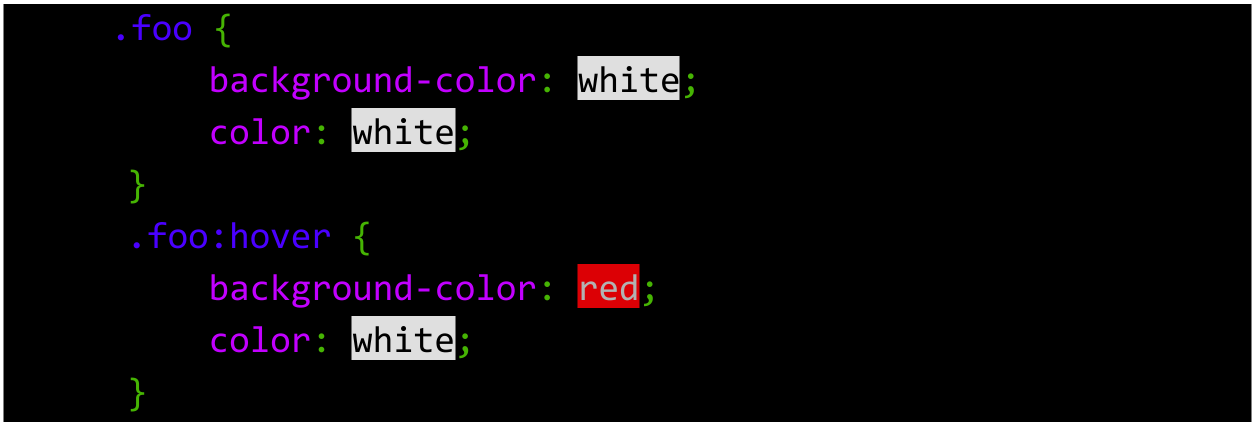
  ```
  #kitty {
   font-size: 1pt;
  }
  ```

there are also ***pseudo-classes*** which correspond to things that "happen" (kind of like events)

- hover (mouseover)

  - these you match with COLONS

  ```
  p:hover {
   ...
  }
  ```



```
.foo {
     background-color: white;
     color: white;
 }
 .foo:hover {
     background-color: red;
     color: white;
 }
```

We'll talk about more advanced CSS stuff with selectors later

- **grid**
  - new thing in CSS that makes it super easy to specify a grid (like a checkerboard, or a spreadsheet, or, you know, a Scrabble board)
    - anything in the "grid-container" class is a grid
      <div class="grid-container" id="thegrid">
      *note: divs can have **multiple** classes separated by spaces, but just **one** id*

- then, inside that div, each item has to have the class grid-item

```
    <div class="grid-container" id="thegrid">
      <div class="foo grid-item">1</div>
      <div class="foo grid-item">Hello</div>
      <div class="bar grid-item">2</div>
      <div class="bar grid-item">World</div>
...
```

Then you specify the characteristics of the grid in the STYLE section:

```
      .grid-container {
          display: grid;
          grid-template-columns: auto auto auto auto auto auto auto auto auto;
          grid-gap: 1px;
          width: 500px;
          height: 500px;
      }
      .grid-item {
          height: 50px;
          width: 50px;
      }
```

Exercise!

https://docs.google.com/document/d/1D04hChcHINVKMjb_8GghJffgs-Cu18WRPeY5bLBRUr4/edit?usp=sharing