

download as zip:

<https://github.com/tpoulos/html-talk>

Replace the information in  
start\_here/index.html  
with your own information

Be done by 6:10 



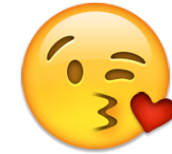
# HTML & CSS!

CrashCourse 2016

What did you learn?



# html



# CSS

- Content
- English, surrounded by `<tags></tags>`
- Pretty solid

- Styling
- selector:  
`{rule:attribute;}`
- Universally hated

```
<!DOCTYPE html>
<html>
<head>
<title>Title of the document</title>
</head>

<body>
The content of the document
</body>

</html>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<title>Title of the document</title>
```

```
The content of the document
```

```
</body>
```

```
</html>
```

- HTML (and CSS) are very forgiving
- In practice, it would be better if they weren't so passive aggressive

# HTML is a tree

VALID:

```
<p><strong>HELLO!</strong></p>
```

INVALID:

```
<p><strong>HELLO!</p></strong>
```



## HTML is a tree

### VALID:

```
<p><strong>HELLO!</strong></p>
```

### INVALID:

```
<p><strong>HELLO!</p></strong>
```

Most browsers won't tell you if the HTML you write is invalid

- Use the [W3C Validator](#)!

# HTML is composed of tags

- `<p>`
- `<strong>`
- `<header>`
- `<div>`
- `<a>`
- And many more

## Tags have attributes

- `<a href=https://developer.mozilla.org">MDN</a>`
- Look at the documentation



- Also a markup language like HTML
- Controls the styling

```
body {  
    font-family: sans-serif;  
}  
  
strong {  
    color: red;  
}  
  
h1 {  
    text-align: center;  
}
```

```
<h1> Hello! </h1>
```

```
<p> and welcome </p>
```

```
<h1 id="requirements">  
Requirements Section </h1>
```

```
<p> Bring a computer </p>
```

```
<p class="note"> Mac or PC  
is fine </p>
```

```
h1 {  
  
    text-align: center;  
  
}  
  
.note {  
  
    font-size: 10px;  
  
}  
  
#requirements {  
  
    color: blue;  
  
}
```

```
.section-title {  
    color: green;  
}
```

```
<h1 id="requirements" class="section-title">
```

Requirements Section

```
</h1>
```

```
.section-title {  
    color: red;  
}
```

```
.section-title{  
    color: blue;  
}
```

```
<h1 id="requirements" class="section-title">
```

Requirements Section

```
</h1>
```

```
#requirements {
```

```
    color: blue;
```

```
}
```

```
h1 {
```

```
    color: green;
```

```
}
```

```
.section-title {
```

```
    color: red;
```

```
}
```





# Make it Classy

- Change colors and text-sizes
- Add a google font, or two
- Add an image in the circle at the top, and on the background
- DO NOT COPY AND PASTE

Be done by 6:40 

What did you learn?

# Media Queries

- The web needs to work on all different devices
- Sometimes, we need to change the CSS based on the type of device

```
@media (min-width: 600px) {  
  .sidebar {  
    display: none;  
  }  
}
```

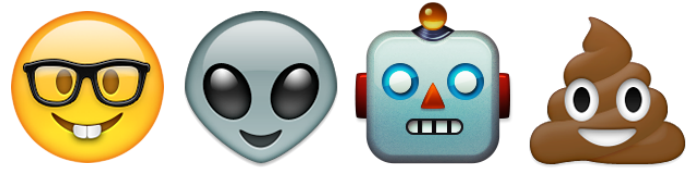


# Make it Responsive

On thin screens, remove the side margins and change the text size

Be done by 6:50   

What did you learn?



# What to learn next

- There is so much more to HTML/CSS
  - Especially positioning/moving around elements
- Check out [Shay Howe's tutorial](#) for a good introduction
- Make stuff; that's the best way to learn



# Taylor Poulos

- [tpoulos@andrew.cmu.edu](mailto:tpoulos@andrew.cmu.edu)
- [twp.io](http://twp.io)