

# The Anatomy of a Web Map App: Part 2

Patrick McKinney  
Cumberland County

[pmckinney@ccpa.net](mailto:pmckinney@ccpa.net)

<https://pnmcartodesign.com/>



pmacMaps

# Cascading Style Sheets (CSS)

- Describes the presentation of a document written in HTML or XML
- [Mozilla Developer Network Reference](#)
- CSS allows you to:
  - Set color, size, and font for text
  - Set layout for sections of a website/app
  - Set rules based upon width of device/screen

# Where To Set CSS Rules

1. Link to an external stylesheet

❖ `<link rel="stylesheet" href="css/main.css">`

2. Use `<style>` tag within the HTML file

3. Use inline-styles on an element

❖ `<h1 style="color:#333; font-size: 2em;">text</h1>`

# Categories of Selectors

- Element
  - `h1`, `p`, `a`, `img`
- ID
  - `#map`, `#title`
- Class
  - `.nav-link`
- Attribute
  - `tr[aria-label="Filter"]`

# Sample CSS Rules

```
html {  
    font-family: Verdana,Geneva,sans-serif;  
}
```

```
#intro-section {  
    margin: 0 auto;  
    padding: 15px;  
    color: #333;  
}
```

```
.link-external {  
    color: #3dbe98;  
    text-decoration: underline;  
}
```

# Specificity Wars

- Hierarchy for how rules are applied
  - Smashing Magazine [article](#)
- When selectors have an equal specificity value, the latest rule is the one that counts
- **important!** Declaration beats everything
  - `.site-title { font-size: 2em !important; }`

# DRY Principle

- Don't Repeat Yourself
- Group elements together that have same rules
- Consider using **Block Element Modifier (BEM)**
  - `.button`
  - `.button-state-success`
  - `.button-state-danger`

# CSS Frameworks

- Preset rules that make standing up a site/app easier
- [PaMAGIC Membership Map](#)
- Examples
  - [Bootstrap](#)
  - [Foundation](#)
  - [Normalize](#)



# CSS Evolves

- New features are being added
- But not all browsers support these!
- [Can I Use](#) shows which browsers support features
- [Should I Prefix](#) shows which browsers need prefixes to rules to work

An aerial, high-angle view of a dense urban skyline, likely New York City. The image shows a vast number of skyscrapers and high-rise buildings packed closely together. The Empire State Building is prominent in the center-left. The sky is a pale, hazy blue, suggesting a clear day. The overall tone is bright and airy, with a soft light filtering over the city.

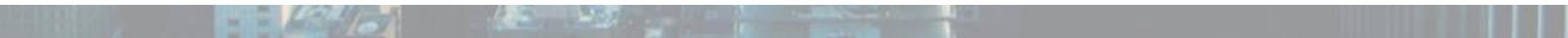
# Let's Look at Some CSS!

# JavaScript

- lightweight interpreted programming language used in web browsers
- Adds interactivity to websites/apps
- ECMAScript 2015 (ES6) added great features
  - But not fully supported yet ☹️
- [Mozilla Developer Network Reference](#)

# JavaScript Enables



- Building interactive map apps
  - Manipulating elements on a page
  - Executing code based upon user interactions
  - Pull in data from other sources without refreshing page
- 

# JavaScript Types

- Number
- String
- Boolean
- Object
- Symbol
- Null
- Undefined

# Variables

- Data item that holds a value
- **Let**: block-level scope; value can change
  - `Let legendContent = "<ul>";`
- **Const**: value does not change
  - `const someMapServiceLegendUrl =  
 'https://[some_domain]/[...]/legend';`
- **Var**: pre-ES6 variable declaration
  - `var customLegendBtn =  
 $('#customLegendBtn');`

# Conditional Statements

- Run blocks of code based upon a condition being true or false

```
if (variable === "value") {  
    // execute code  
} else if (variable === "other value") {  
    // execute other code  
} else {  
    // execute different code  
}
```

# Loops

- **while/do-while**
  - Run code while a variable is true or false
- **for/for...of/for...in**
  - Iterate over an array or object
  - Run code using current value of array or object



# Loops

```
For (let item of layers) {  
    // iterate over layers array  
    // run code  
    // value of item changes each time  
}
```

# Functions

- Take zero or more named parameters
- Executes code using those parameters
- Can return a variable

```
// click event of an element
customLegendBtn.click(function() {
    // execute toggle function on an
    element
    $('#customLegend').toggle();
});
```

An aerial photograph of a dense city skyline, likely New York City, featuring numerous skyscrapers and buildings. The image is slightly hazy, giving it a soft, atmospheric feel. Overlaid on the top half of the image is the text "Let's Look at Some JavaScript!" in a large, bold, black serif font. The text is centered and spans across the width of the image.

**Let's Look at Some  
JavaScript!**