

# CSS Organization

We talked about approaches to class names:

- Semantic
- Utility First/Atomic

But what about the rules/declarations in the CSS files?

How do we make it easy to:

- Find if element is styled?
- Find if property is altered?
- Add properties in that system?

# Organization is needed

You quickly start noticing the problem

- Have I styled this element already?
- Where?
- Have I declared this property already?
- Where?

# CSS Organization

Some form of organization is needed

- How to order the selectors
- What files to put the selectors in
- How to order the properties in declarations

# Multiple Options

We simply do not know what is best

We only know what is worst:

- No organization

Follow the style of your team, if you have one Otherwise, create a pattern

# Files

- Can you break up rules into multiple files?
  - Without tools, this should be limited
  - B/C multiple files means more downloads for client
- Should only consider rules that are contained

# Selectors

How are you organizing the different rules (by selector)

- Expected page order?
- Generic to specific?
- Sections (globals, header, menu, etc)

Comments for boundaries help `/* comment */`

# Properties Within A CSS Rule

How are you ordering the list of declarations?

- Alphabetical?
  - Low-value
  - Easy to automate in a large team
- By purpose (My recommendation)
  - Display/position/box model/colors/text/etc
    - You + your team decide the breakdown
  - Separate by blank lines
    - Programming is Communication

# My Preferred Style: Selector order

- "sweeping" CSS first
  - `*` selectors, `body`, `html`
  - "resets"
  - Any common styling (Ex: `p`, etc)
- General Page order for next selectors
  - Often BEM style
  - Try to select based on class



# My Preferred Style: Properties

Within each CSS Rule:

- "Paragraphs" for different categories
- Position, display, box model, colors, typography

```
.card-link {  
  margin-top: auto;  
  padding: 0.5rem;  
  border: 1px solid black;  
  border-radius: 10px;  
  
  color: black;  
  background-color: lime;  
  
  font: inherit;  
  text-decoration: none;  
}
```

# **My preference is just a preference**

- Different teams and projects make choices
  - System adds value through:
    - Consistency
    - Effectiveness of communication/finding
- No agreement on a "best"
- Course requires you have a consistent system
  - You pick what that system is

# Enforcement

How do we make sure these conventions are followed?

- Automated tooling
- Code Reviews

# Code Review

Most workplaces:

- Everyone has their code changes reviewed
- PR just like assignments

Unlike assignments:

- Less "is this correct?"
- More "Can we maintain this over time?"
  - Conventions checked here
- Some "Are there subtle bugs"

Warning: Expected you confirmed basic working!