CSS/CSS3
for Front End Developers

phuong.vu@htklabs.com

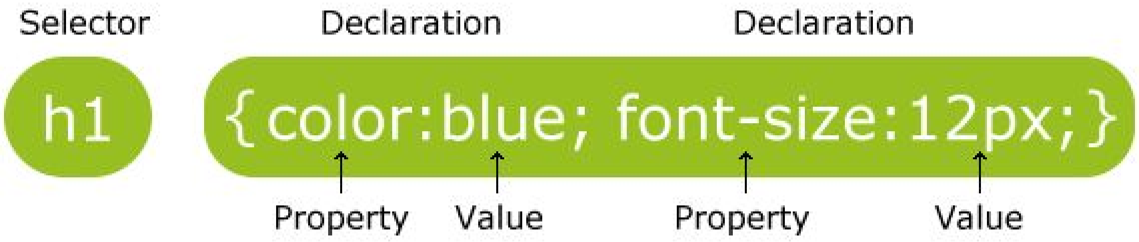
Agenda

- About CSS
- CSS Syntax
- CSS Placement
- Cascading Order
- CSS Selectors
- CSS Normalize
- CSS Lint
- CSS Preprocessors

About CSS

- A stylesheet language that describes the presentation of an HTML (or XML) document.
- Describes how elements must be rendered on screen, on paper, or in other media
- Saves a lot of work. It can control the layout of multiple web pages all at once.
- Is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

CSS Syntax



Selector

Declaration

Declaration

```
h1 { color:blue; font-size:12px;}
```

Property

Value

Property

Value

CSS Placement

- External style sheet:
    - Using <link> element inside the <head> section
    - Ex: `<link rel="stylesheet" type="text/css" href="mystyle.css">`
- Internal style sheet:
    - Using <style> element inside the <head> section
    - Ex: `<style>.....</style>`
- Inline style: add the style attribute to the relevant element

Cascading Order Priority

1. Inline style
2. External and internal style sheets.
   - If some properties have been defined for the same selector (element) in different style sheets, the value from the last read style sheet will be used.
3. Browser default

- **Number one has the highest priority**

CSS Selectors

- Element Selector
- ID Selector
- Class Selector
- Group Selector
- Attribute Selector
- Combinators Selector
- Pseudo-classes Selector
- Pseudo-elements Selector

CSS Selectors (con’t)

- **Element Selector:** selects elements based on the element name.
  - Syntax: *element*
  - *Ex:* div
- **ID Selector:** select a specific element with given id
  - Syntax: *#id*
  - *Ex:* #para
- **Class Selector:** select elements with given class
  - Syntax: *.class*
  - *Ex:* .bar
- **Group Selector:** Many selectors with the same style definitions.
  - Syntax: *selector1, selector2*
  - *Ex:* div, p

CSS Selectors (con’t) - Attribute Selector

- *[attribute]*: selects elements with a specified attribute
  - *Ex:* [target]
- *[attribute=”value”]*: selects elements with a specified attribute and value
  - *Ex:* [target=”_blank”]
- *[attribute~=”value”]*: select elements with an attribute value containing a specified word
  - *Ex:* [title~=”training”]
- *[attribute|=”value”]*: select elements with the specified attribute starting with the specified value
  - Ex: [class|=”top”]
- *[attribute^=”value”]*: select elements whose attribute value begins with a specified value
  - *Ex:* [class^=”top”]
- *[attribute$=”value”]*: select elements whose attribute value ends with a specified value
  - Ex: [class$=”test”]
- *[attribute*=”value”]*: select elements whose attribute value contains a specified value
  - Ex: [class*=”hi”]

CSS Selectors (con’t) - Combinators Selector

- **Descendant Selector:** matches all elements that are descendants of a specified element
  - Syntax: *element element* | *Ex:* div p
- **Child Selector:** selects all elements that are the immediate children of a specified element.
  - Syntax: *element* > *element* | *Ex:* div > p
- **Adjacent Sibling Selector:** selects all elements that are the adjacent siblings of a specified element
  - Syntax: *element* + *element* | *Ex*: div + p
- **General Sibling Selector:** selects all elements that are siblings of a specified element
  - Syntax: *element* ~ *element* | *Ex*: div ~ p

CSS Selectors (con’t) - Pseudo-class Selector

- **Pseudo-class Selector:** Used to define a special state of an element.
  - Syntax: *selector:pseudo-class*
  - *Ex:* div:hover
- **Pseudo-element Selector:** Used to style specified parts of an element
  - Syntax: *selector::pseudo-element*
  - *Ex:* p::first-line

CSS Coding Convention

- **Class naming conventions:**
  - Never reference **js-** prefixed class names from CSS files. **js-** are used exclusively from JS files.
  - Use the **is-** prefix for state rules that are shared between CSS and JS.
  - The class names *disabled, mousedown, danger, hover, selected,* and *active* should **always be** namespaced by a class.
- **Classes vs. ids:** Elements that occur **exactly once** inside a page should use IDs, otherwise, use classes.

CSS Coding Convention (con’t)

- **Spacing:**
  - Use soft-tabs with a two space indent.
  - Put spaces after **:** in property declarations.
  - Put spaces before **{** in rule declarations.
  - Put line breaks between rulesets.
  - When grouping selectors, keep individual selectors to a single line.
  - Place closing braces **}** of declaration blocks on a new line.
  - Each declaration should appear on its own line for more accurate error reporting.

CSS Coding Convention (con’t)

- **Formatting:**
  - Use hex color codes #000 unless using rgba() in raw CSS.
  - Use /* */ for comment blocks.
  - Avoid specifying units for zero values.

CSS Normalize

- Is a small CSS file that provides better cross-browser consistency in the default styling of HTML elements.
- It’s a modern, HTML5-ready, alternative to the traditional CSS reset.
- Site: http://necolas.github.io/normalize.css/
- Source: https://github.com/necolas/normalize.css

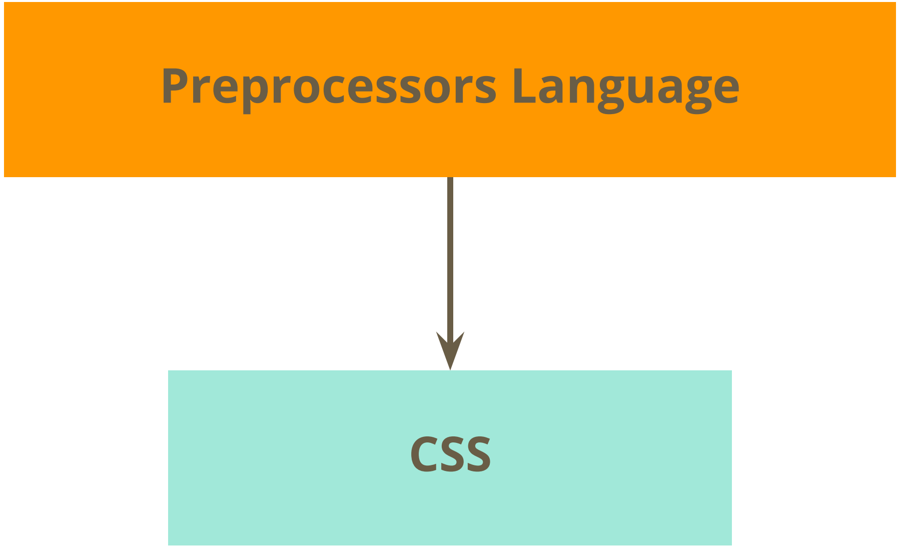
CSS Normalize - The aims

- Preserve useful browser defaults rather than erasing them.
- Normalize styles for a wide range of HTML elements.
- Correct bugs and common browser inconsistencies.
- Improve usability with subtle improvements.
- Explain the code using comments and detailed documentation.

CSS Lint

- Is an open source CSS code quality tool.
- points out problems with your CSS code.
- CLI: https://github.com/CSSLint/csslint/wiki/Command-line-interface

CSS Preprocessors



Preprocessors Language

CSS

CSS Preprocessor Feature

Variables

Math

Mixins

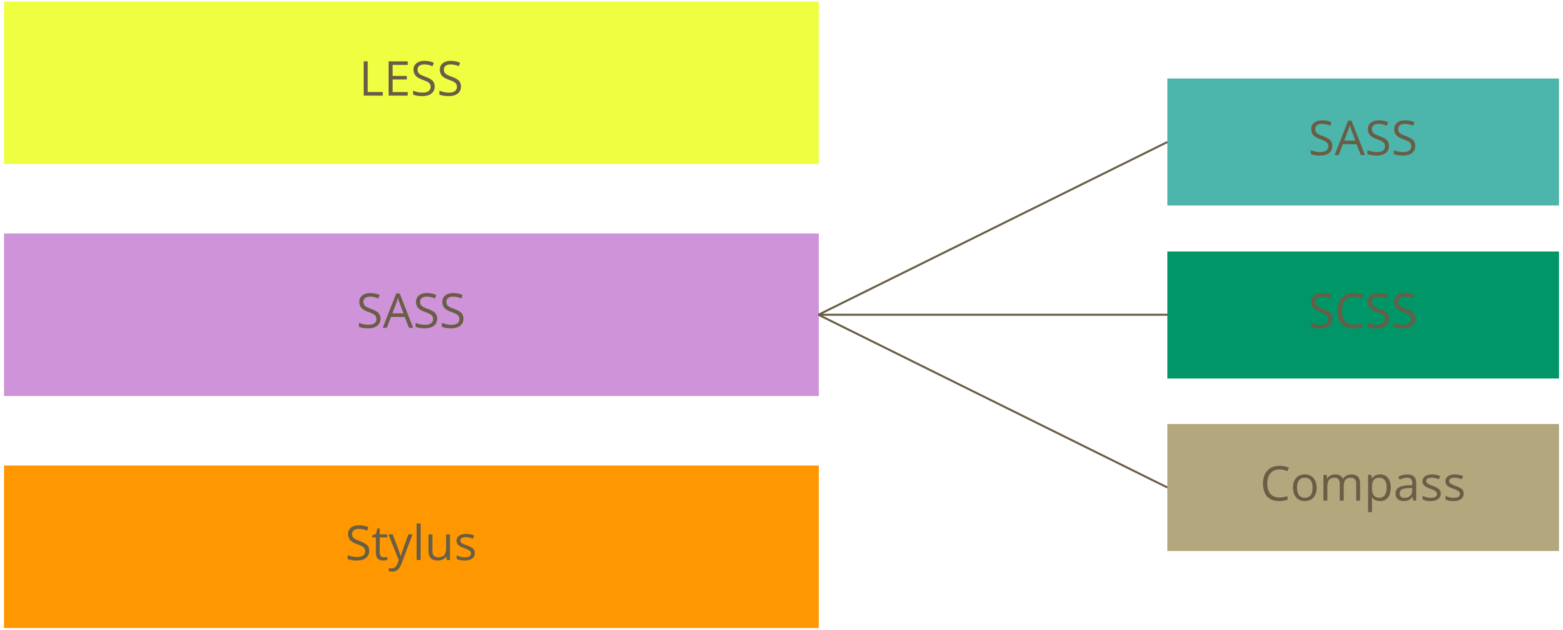
Nested Rules

Logic

CSS Preprocessor Benefit

- Cleaner code with reusable pieces.
- More flexibility to do thing on the fly.
- Shareable snippets and libraries.
- Easily product CSS that works across browser.

CSS Preprocessor List



LESS

SASS

Stylus

SASS

SCSS

Compass

Assignment

1. Apply the CSS normalize in the HTML project in the previous assignment.
2. Apply your CSS in that project. Your CSS have enough 8 types of CSS Selectors.
3. Investigate and make the present about CSS Preprocess Stylus.

Q&A

Thanks for your listening!