CSS Foundations

Alexander Thalhammer | @LX_T

Foundations

– About CSS

– Using CSS

– Properties & Values

– Selectors

– Specificity

– Combinators

Agenda

Cascading Style Sheets

What are CSS?

– they are optional (HTML is required)

– allow you to style HTML

– make your HTML look good

– "cascading" → multiple rules apply to same element

History

- CSS 1 (1996)

- CSS 2 (1998)

- CSS 3 (current)
  - in development – forever ☺

Using CSS

- Inline styling
  - Rules via style[attribute] <element style="" />

- Via selector
  - Put rules into <style> tag or
  - External stylesheet file(s) .css

- What's the best way & why?

Properties & Values

– property: value

– property-a: value-a; property-b: value-b

– a lot of properties!
  – https://www.w3schools.com/cssref/
  – https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Properties_Reference

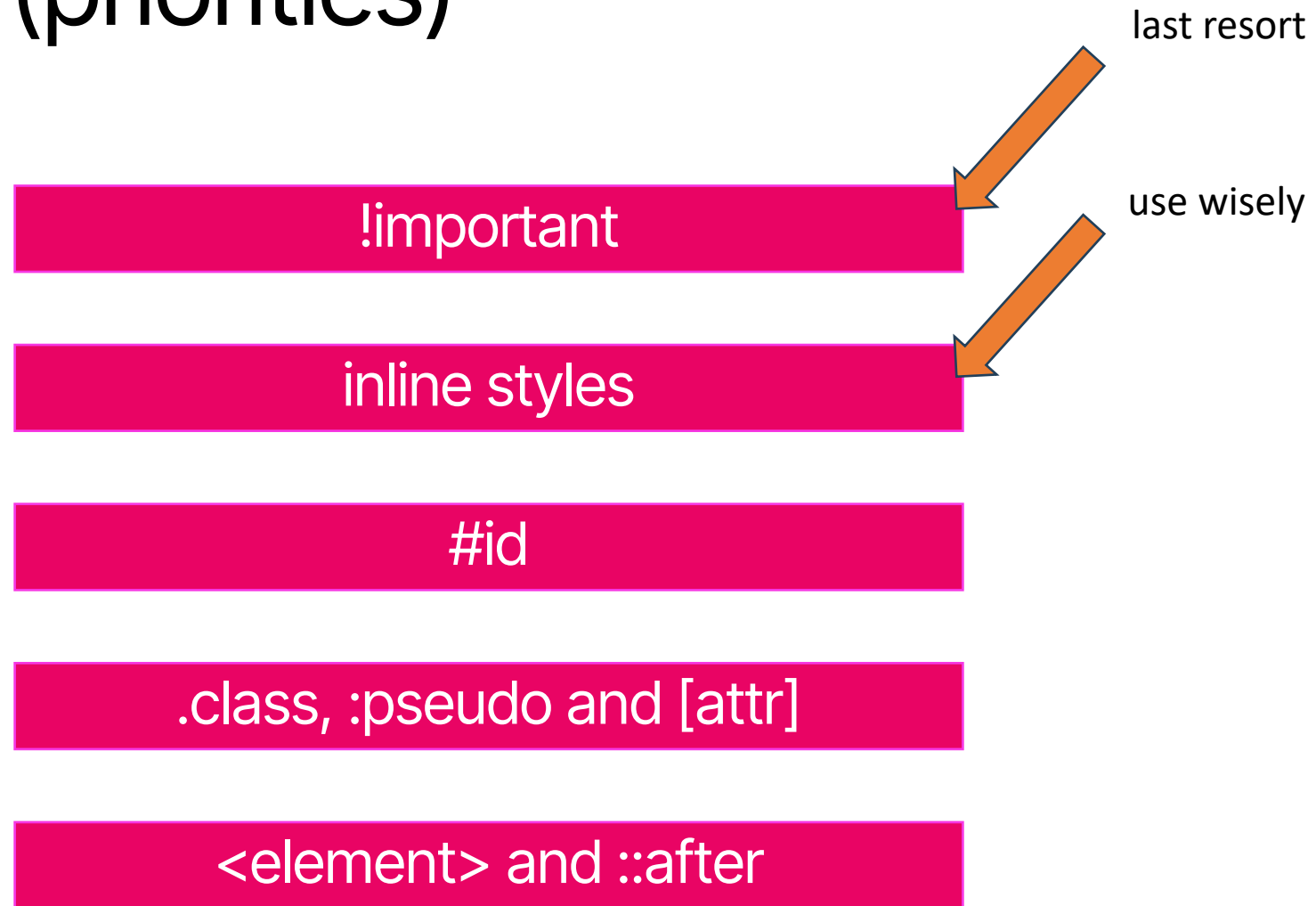– we don't have to know them all ☺

Values

- pre-defined
  - display: block;
  - margin: auto;

- colors
  - black & white
  - #000000 & #fff

- numbers
  - margin: 16px / 1em / 1rem
  - width: 100%

- functions
  - scale(50%)

Selectors (in JS → Query Selectors)

— *

— element

— .class

— #id (only applicable once)

— [attribute=value] or just [disabled]

— :hover, :focus, :nth-child(2n + 1), (pseudo-class)

   — :not(...)

— ::after, ::before (pseudo-element)

Specificity (priorities)

last resort

use wisely

!important

inline styles

#id

.class, :pseudo and [attr]

<element> and ::after

https://developer.mozilla.org/en-US/docs/Web/CSS/Specificity

ANGULAR
ARCHITECTS

Inheritance

– Styling the <body> tag

– property: inherit
  – explicitly override current element's styling with parent's styling

Combinators (increase specifity)

– .class-a.class-b (more specific)

– " " Descendant like div p {}
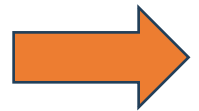
⟹ – > Direct child div > p {}

– ~ General sibling div ~ p {} ⟵ following

– + Adjacent sibling div + p {} ⟵ immediately following

https://developer.mozilla.org/…/Combinators_and_multiple_selectors

Demo

Foundations

ANGULAR
ARCHITECTS

Best Practices

– Use .class (not #id) to make styles reusable

– Combine maximal 3 selectors

– Use zero values without units

– Try to avoid !important

Tips

Foundations

– Rules

– Properties & Values

– Selectors

– Specificity

– Combinators

Summary

Questions?

ANGULAR
ARCHITECTS