

---

## *While you're waiting:*

1. Hop on Slack
2. Vote on the two polls about last week's session
3. Add the priority guide you made last week to the shared Google Drive folder  
(include your first initial and last name in the filename, pls)

---

# *Today's agenda*

## *1. Slack checkin*

---

# *Today's agenda*

## *1. Slack checkin*

## *2. CSS*

- The cascade
- Defaults to be aware of
- Best practices
- Structuring your CSS
- Naming structures
- Some gotchas
- Sass: CSS superpowers

---

# *Deep dive into CSS*

---

## *Talking about CSS*

```
body {
```

```
  color: red;
```

```
}
```

} Ruleset

---

## *Talking about CSS*

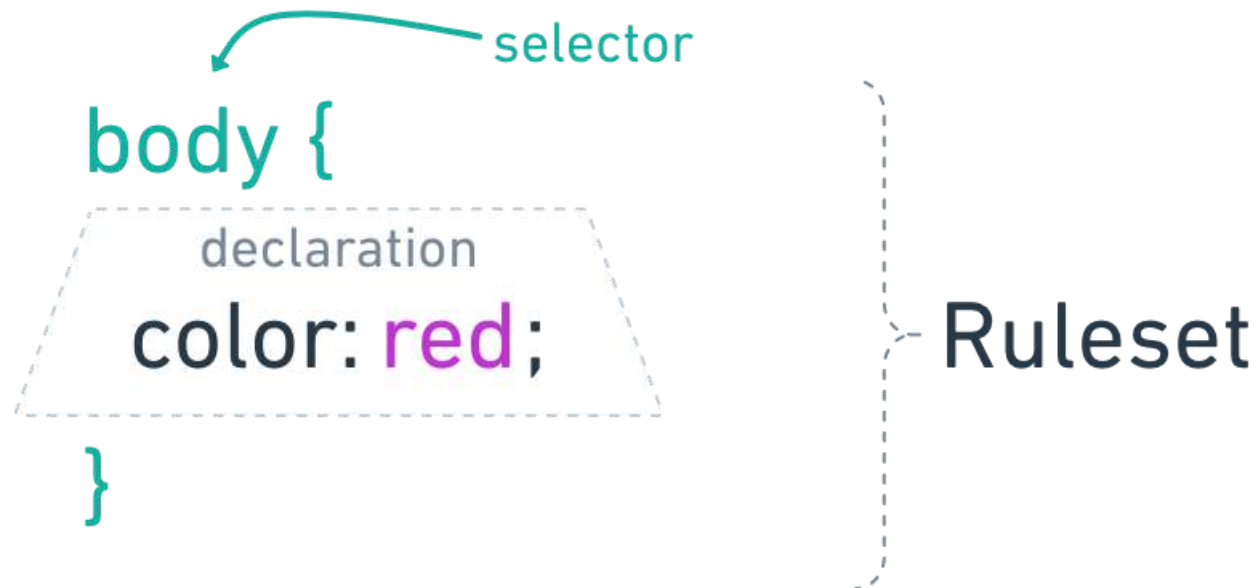
selector

```
body {  
  color: red;  
}
```

Ruleset

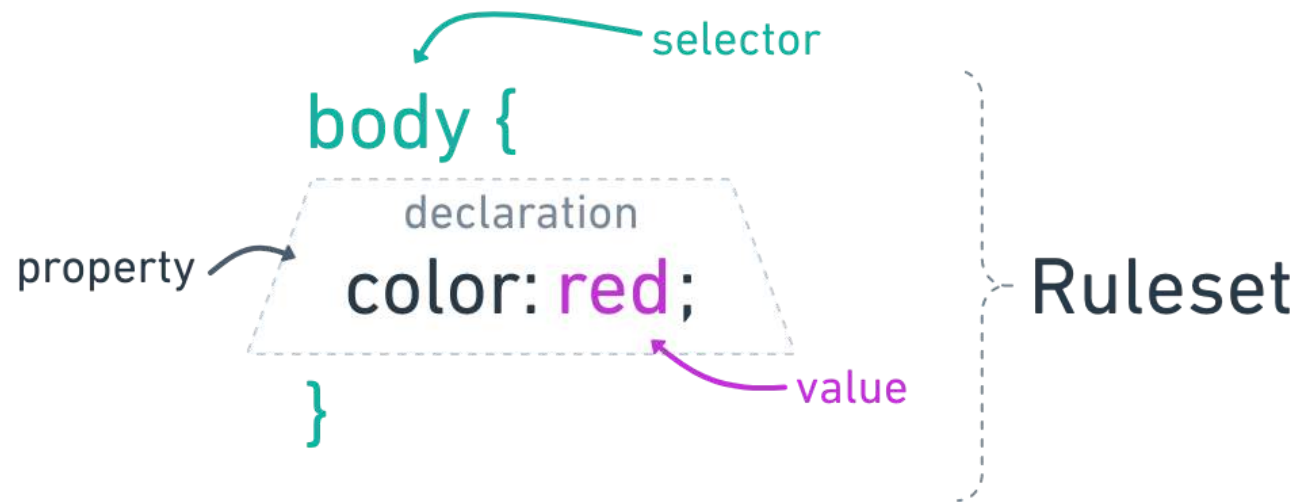
---

## *Talking about CSS*



---

## *Talking about CSS*





---

## *The cascade*

---

---

# *CSS defaults*

*Inline elements*

*Block-level elements*

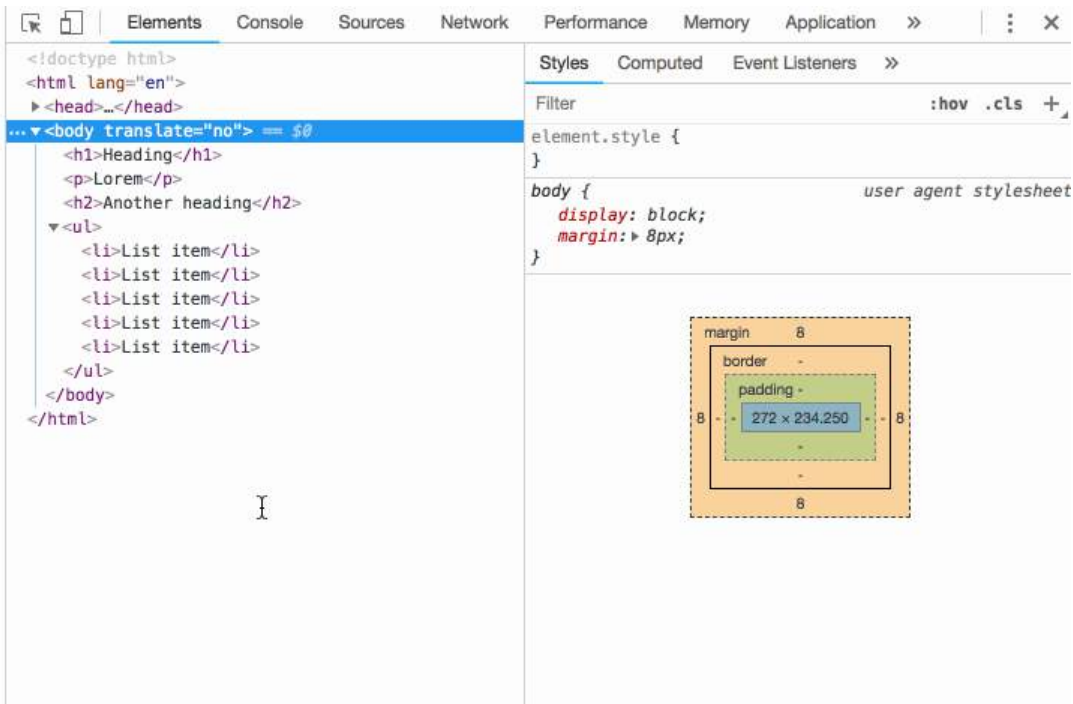
# Browser defaults

## Heading

Lorem

## Another heading

- List item
- List item
- List item
- List item
- List item



---

## *Best practices*

Use classnames over bare elements or IDs

---

## *Best practices*

Use classnames over bare elements or IDs

```
<h1 class="page-title cursive"></h1>
```

---

## *Best practices*

Use classnames over bare elements or IDs

```
<h1 class="page-title cursive"></h1>
```

Write your CSS lowercase and kebab case

---

## *Best practices*

Use classnames over bare elements or IDs

```
<h1 class="page-title cursive"></h1>
```

Write your CSS lowercase and kebab case

```
.page-title {  
  text-transform: uppercase;  
}
```

---

## *Best practices*

Use classnames over bare elements or IDs

```
<h1 class="page-title cursive"></h1>
```

Write your CSS lowercase and kebab case

```
.page-title {  
  text-transform: uppercase;  
}
```

Transform case using CSS



---

## *Best practices*

Use classnames over bare elements or IDs

```
<h1 class="page-title cursive"></h1>
```

Write your CSS lowercase and kebab case

```
.page-title {  
  text-transform: uppercase;  
}
```

Transform case using CSS

Remember your semicolon

---

## *Best practices*

Learn about methods of organizing CSS/Sass and use a system

---

## *Best practices*

Learn about methods of organizing CSS/Sass and use a system

```
div {  
  background-color:  
whitesmoke;  
  border: 3px dotted #f9c7dd;  
  color: #444;  
  display: inline-block;  
  font-family: "Montserrat",  
sans-serif;  
  font-size: 100%;  
  padding: 1em;  
}
```

---

## *Best practices*

Learn about methods of organizing CSS/Sass and use a system

```
div {  
  background-color:  
whitesmoke;  
  border: 3px dotted #f9c7dd;  
  color: #444;  
  display: inline-block;  
  font-family: "Montserrat",  
sans-serif;  
  font-size: 100%;  
  padding: 1em;  
}
```

```
div {  
  /* -- Box -- */  
  display: inline-block;  
  padding: 1em;  
  
  /* -- Border -- */  
  border: 3px dotted #f9c7dd;  
  
  /* -- Background -- */  
  background-color:  
whitesmoke;  
  
  /* -- Text -- */  
  color: #444;  
  font-family: "Montserrat",  
sans-serif;  
  font-size: 100%;  
}
```

---

# *Best practices*

Use CSS comments

---

## *Best practices*

Use CSS comments

```
/*  
* === SITE HEADER ===  
*/
```

```
/*  
* --- Header nav ---  
*/
```

---

## *Best practices*

Avoid !important

---

## *Best practices*

Avoid using empty divs to achieve your styles

```
<div class="something-fancy"></div>
```



---

## *Best practices*

Measurements of zero don't need units

---

## *Best practices*

Measurements of zero don't need units

### *Avoid*

```
div {  
  margin: 0px;  
  padding: 0px;  
  width: 0px;  
}
```

### *Do this*

```
div {  
  margin: 0;  
  padding: 0;  
  width: 0;  
}
```

---

## *Best practices*

Avoid overdoing your selectors and specificity

---

## *Best practices*

Avoid overdoing your selectors and specificity

```
<header class="site-header">
  <nav class="nav-bar">
    <ul class="nav-menu">
      <li class="nav-
menu__item"></li>
      <li class="nav-
menu__item"></li>
      <li class="nav-
menu__item"></li>
    </ul>
  </nav>
</header>
```

---

## *Best practices*

Avoid overdoing your selectors and specificity

```
<header class="site-header">
  <nav class="nav-bar">
    <ul class="nav-menu">
      <li class="nav-
menu__item"></li>
      <li class="nav-
menu__item"></li>
      <li class="nav-
menu__item"></li>
    </ul>
  </nav>
</header>
```

## *Avoid*

```
header.site-header nav.nav-
bar ul.nav-menu li.nav-
menu__item {}
```

---

## Best practices

Avoid overdoing your selectors and specificity

```
<header class="site-header">
  <nav class="nav-bar">
    <ul class="nav-menu">
      <li class="nav-
menu__item"></li>
      <li class="nav-
menu__item"></li>
      <li class="nav-
menu__item"></li>
    </ul>
  </nav>
</header>
```

### Avoid

```
header.site-header nav.nav-
bar ul.nav-menu li.nav-
menu__item {}
```

### Do this

```
.nav-menu__item {}
```

OR

```
.site-header .nav-menu__item
{}
```

---

## *Best practices*

Avoid making unnecessary assumptions about the markup

### *Avoid*

```
<span class="highlight"><a  
href='#>Highlighted link</a></span>
```

```
.highlight a { background: yellow; }
```

---

## *Best practices*

Avoid making unnecessary assumptions about the markup

### *Avoid*

```
<span class="highlight"><a  
href='#>Highlighted link</a></span>
```

```
.highlight a { background: yellow; }
```

### *Do this*

```
<a href='#' class="highlight">Highlighted  
link</a>
```

```
.highlight { background: yellow; }
```



---

## *Best practices*

Avoid writing CSS you don't need

---

## *Best practices*

Avoid writing CSS you don't need

```
div { display: block; }
```

---

## *Best practices*

Avoid writing CSS you don't need

```
div { display: block; }
```

```
div { margin: 0 auto; }  
/* This is the margin shorthand. It's the  
equivalent of: */  
/* margin-top: 0; */  
/* margin-right: auto; */  
/* margin-bottom: 0; */  
/* margin-left: auto; */
```

---

## *Best practices*

Avoid writing CSS you're immediately going to override

---

## *Best practices*

Avoid writing CSS you're immediately going to override

### *Avoid*

```
li {  
  display: inline-block;  
  margin-left: 14px;  
}  
li:first-of-type { margin-left: 0; }
```

---

## *Best practices*

Avoid writing CSS you're immediately going to override

### *Avoid*

```
li {  
  display: inline-block;  
  margin-left: 14px;  
}  
li:first-of-type { margin-left: 0; }
```

### *Do this*

```
li { display: inline-block; }  
li + li { margin-left: 14px; }
```

---

## *Best practices*

Avoid making an HTTP request for a resource when you can easily replicate the element with CSS

### *Avoid*

```
li::before {  
  content: url(green-  
circle.svg);  
}
```

---

## *Best practices*

Avoid making an HTTP request for a resource when you can easily replicate the element with CSS

### *Avoid*

```
li::before {  
  content: url(green-  
circle.svg);  
}
```

### *Do this*

```
li::before {  
  background: green;  
  border-radius: 50%;  
  content: "";  
  display: block;  
  height: 20px;  
  width: 20px;  
}
```

View on CodePen: <https://codepen.io/angeliquejw/pen/jvYvpy>



---

## *Best practices Q&A*

---

# *Structuring your CSS*

## 1. Reset

---

# *Structuring your CSS*

1. Reset
2. Utilities/helpers

---

# *Structuring your CSS*

1. Reset
2. Utilities/helpers
3. Base

---

## *Structuring your CSS*

1. Reset
2. Utilities/helpers
3. Base
4. Custom styles

---

## *Learn about resets*

### *Reset.css (group 1)*

- Read about it: <https://meyerweb.com/eric/tools/css/reset/>
- View the code: <https://codepen.io/angeliquejw/pen/BOmXNa?editors=0100>

### *Normalize.css (group 2)*

- Read about it: <http://nicolasgallagher.com/about-normalize-css/>
- View the code: <https://codepen.io/angeliquejw/pen/rZYXOz?editors=0100>

Extra time? <https://css-tricks.com/reboot-resets-reasoning/>

---

## *Learn about resets*

1. Join your group

---

## *Learn about resets*

1. Join your group
2. Establish roles:
  - Timekeeper
  - Presenter



---

## *Learn about resets*

1. Join your group
2. Establish roles:
  - Timekeeper
  - Presenter
3. As a group, spend 10 mins discussing:
  - 3-5 main things to know
  - 2 pros and 2 cons

---

# *Reset*

---

<https://s.codepen.io/angeliquejw/debug/ZMvqBj>

---

## *Utility classes*

---

<https://codepen.io/angeliquejw/pen/wEpvqN?editors=0100>

---

## *Base styles*

---

## *Base styles*

```
body {  
  background: whitesmoke;  
  color: #1b203a;  
  font-family: "Fira Sans", Sans-Serif;  
  margin: 0;  
}  
  
h1, h2, h3, h4, h5, h6, p, li {  
  line-height: 1.5625em;  
  margin: 0;  
}
```

---

## *Custom CSS*

---

## *Structuring your CSS Q&A*