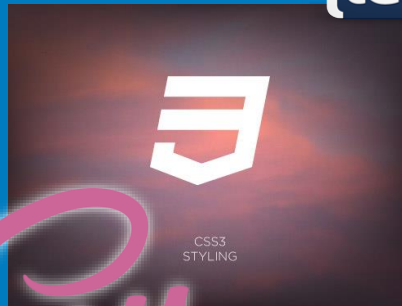


Building layouts with frameworks and preprocessors

{less}



Sass

Agenda

■ Building layouts

- Fixed layouts
- Fluid layouts
- Responsive layouts

■ {less}

- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

■ LESS vs Sass

■ CSS frameworks

- 960 grid system
- Bootstrap, from Twitter

■ Sass

- Getting started
- Mixins and extending
- Nested rules
- Operations
- Control directives
- More Sass

Layouts: The fixed (1/2)

- Content area has a fixed width, e.g. 760 pixels

Pros

- Tight control over the readable content area
- Often easy to style

Cons

- The content area may appear narrow on wide screens
- Limits the reasonable amount of displayable content



3

Layouts: The fixed (2/2)

- A fixed width, often positioned in the middle of the page

CSS

```
#container {
  width: 700px;
  margin-left: auto;
  margin-right: auto;
}
```

CSS

```
#container {
  width: 700px;
  position: absolute;
  left: 50%;
  margin-left: -350px;
}
```

4

Agenda

■ Building layouts

- Fixed layouts
- Fluid layouts
- Responsive layouts



- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

■ LESS vs Sass

■ CSS frameworks

- 960 grid system
- Bootstrap, from Twitter



- Getting started
- Mixins and extending
- Nested rules
- Operations
- Control directives
- More Sass

5

Layouts: The fluid

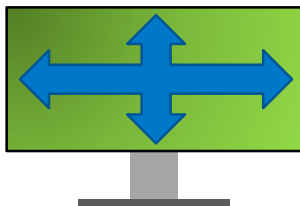
■ Content area has a relative width, e.g. 76%

Pros

- Makes good use of the available screen estate

Cons

- Readability suffers if text spans the entire screen
- More work to implement successfully
- Modern browsers support page zoom instead of text zoom



6

Agenda

- Building layouts
 - Fixed layouts
 - Fluid layouts
 - Responsive layouts
- **less**
 - Getting started
 - Mixins
 - Nested rules
 - Operations
 - More LESS
- LESS vs Sass
- CSS frameworks
 - 960 grid system
 - Bootstrap, from Twitter
- **Sass**
 - Getting started
 - Mixins and extending
 - Nested rules
 - Operations
 - Control directives
 - More Sass

7

Layouts: The responsive (1/2)

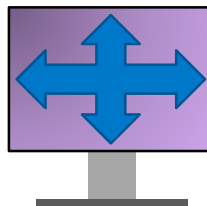
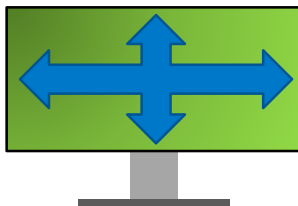
- Respond with the layout that's best suited

Pros

- Offers an optimized user experience based on screen resolution

Cons

- More data to retrieve
- More work to implement



8

Layouts: The responsive (2/2)

- Respond to device capabilities with CSS3 **media queries**
- Use media queries to link external stylesheets

```
<link href="mobile.css" rel="stylesheet"
      media="screen and (max-width: 400px)" />
```

HTML

- Use media queries inside a stylesheet

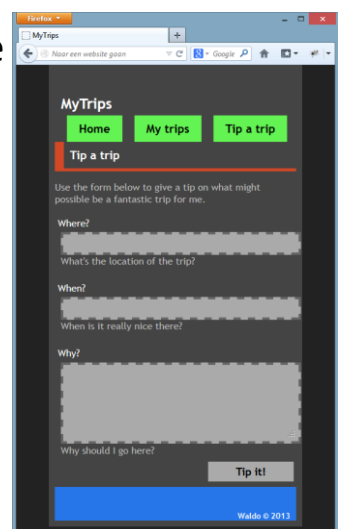
```
@media screen and (orientation: portrait) {
  body {
    font-family: Verdana;
  }
}
```

CSS

9

Lab: Building layouts

- Exercise 1: Fully set up a layout
- Exercise 2: Make it responsive with a mobile layout
- Exercise 3: Build a form



10

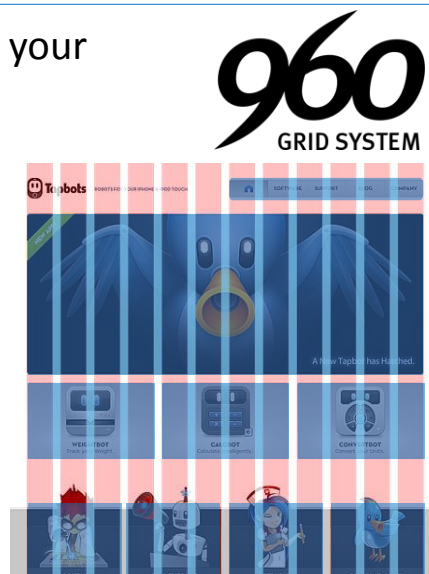
Agenda

- Building layouts
 - Fixed layouts
 - Fluid layouts
 - Responsive layouts
- **less**
 - Getting started
 - Mixins
 - Nested rules
 - Operations
 - More LESS
- LESS vs Sass
- CSS frameworks
 - 960 grid system
 - Bootstrap, from Twitter
- **Sass**
 - Getting started
 - Mixins and extending
 - Nested rules
 - Operations
 - Control directives
 - More Sass

11

960 grid system

- A CSS template to build your layout upon
- Divide 960 pixels over 12, 16 or 24 columns



12

960 grid system: Usage (1/3)

■ Declare the grid container

```
<div class="container_12">
  ...
</div>
```

Use 12 columns for the grid

HTML

■ Use the grid

```
<div class="container_12">
  <div class="grid_3">
    ...
  </div>
</div>
```

Let this DIV span across three columns

HTML

13

960 grid system: Usage (2/3)

■ Indent with prefixes and suffixes

```
<div class="container_12">
  <div class="grid_3 prefix_2 suffix_4">
    ...
  </div>
</div>
```

Provides two columns of indenting **before** the DIV

Provides four columns of indenting **after** the DIV

HTML

– Uses padding-left and padding-right

14

960 grid system: Usage (3/3)

■ Push and pull space

- Rearrange elements independent of the order in the markup

```
<div class="container_12">
  <div class="grid_3 pull_2">
    ...
  </div>
</div>
```

HTML

“Pulls” the element two columns to the left

- Uses relative positioning along with left and right

15

Agenda

■ Building layouts

- Fixed layouts
- Fluid layouts
- Responsive layouts

■ CSS frameworks

- 960 grid system
- Bootstrap, from Twitter



- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

■ LESS vs Sass

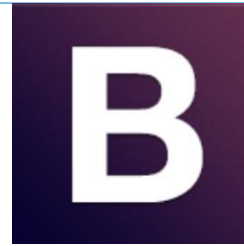


- Getting started
- Mixins and extending
- Nested rules
- Operations
- Control directives
- More Sass

16

Bootstrap, from Twitter

- A framework to speed up front-end development
- Features:
 - **Positioning** with a 12-column responsive grid
 - **Styling** of tables, fonts, buttons and forms
 - **UI components**, including dropdown menus, progress indicators, alerts and pagination
 - **jQuery plugins**, including modal dialogs, tabs, tooltips, button states and scrollspy



17

Bootstrap: Usage

- Include Bootstrap CSS for styling

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

HTML

- Bootstrap uses CSS3, so make sure the HTML5 doctype is set

- Include Bootstrap JavaScript to make use of components

- Include jQuery as well!

```
<script src="scripts/jquery.js"></script>
<script src="bootstrap/js/bootstrap.js"
      type="text/javascript"></script>
```

HTML

18

Bootstrap, from Twitter

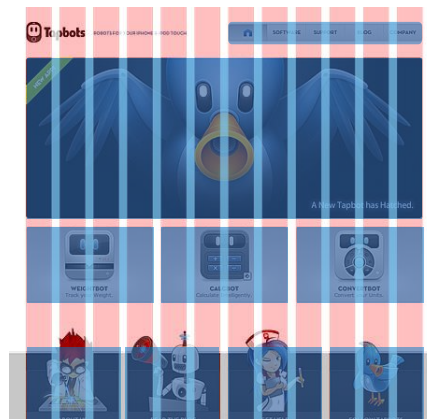
- A framework to speed up front-end development
- Features:
 - **Positioning** with a 12-column responsive grid
 - **Styling** of tables, fonts, buttons and forms
 - **UI components**, including dropdown menus, progress indicators, alerts and pagination
 - **jQuery plugins**, including modal dialogs, tabs, tooltips, button states and scrollspy



19

B Positioning

- A grid that:
 - Stretches over 940 pixels
 - Has 12 columns
 - Has capabilities for fixed and fluid layouts
- Support for building responsive layouts



20

B Positioning: Grid (1/3)

■ Provide a fixed-width layout

```
<div class="container">
  ...
</div>
```

HTML

■ And use the grid

```
<div class="row">
  <div class="span4">
    ...
  </div>
</div>
```

Let this DIV span across four columns

HTML

21

B Positioning: Grid (2/3)

■ Provide a fluid layout

```
<div class="container-fluid">
  ...
</div>
```

HTML

■ And use the grid

```
<div class="row-fluid">
  <div class="span4">
    ...
  </div>
</div>
```

Let this DIV span across four columns

HTML

22

B Positioning: Grid (3/3)

■ Indent with offsets

```
<div class="row">
  <div class="span4 offset2">
    ...
  </div>
</div>
```

Provides two columns of indenting **before** the DIV

HTML

– Uses margin-left

23

B Positioning: Responsive support

■ Bootstrap offers **utility classes** for showing/hiding content **by device width**

.visible-phone

.visible-tablet

.visible-desktop

.hidden-phone

.hidden-tablet

.hidden-desktop

From 0 to 767px

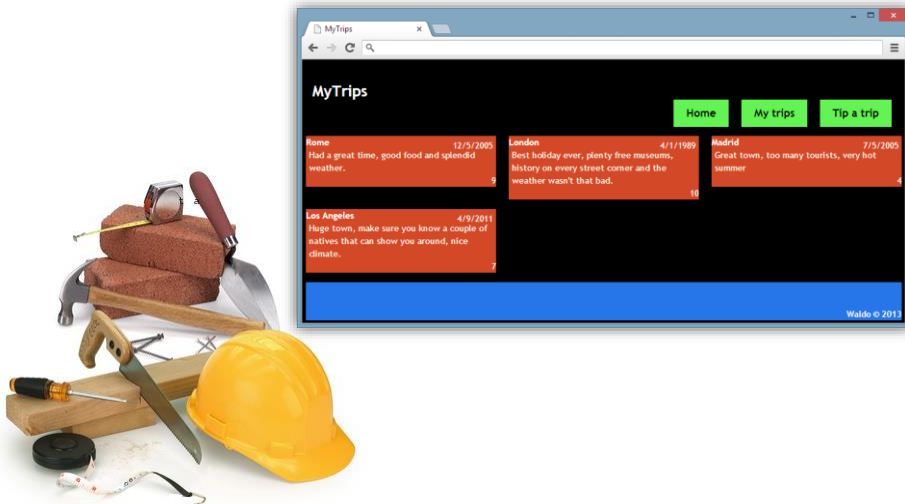
From 768px to 979px

980px and higher

24

Lab: Bootstrap

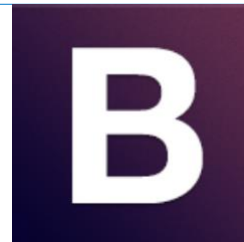
■ Exercise 4: Build a layout with Bootstrap



25

Bootstrap, from Twitter

- A framework to speed up front-end development



■ Features:

- **Positioning** with a 12-column responsive grid
- **Styling** of tables, fonts, buttons and forms
- **UI components**, including dropdown menus, progress indicators, alerts and pagination
- **jQuery plugins**, including modal dialogs, tabs, tooltips, button states and scrollspy

26

B Styling

■ Common elements

- ``, `<i>`, `<abbr>`, `<blockquote>`, ``, ``, `<dl>`, `<code>` and `<pre>`

■ Typical usage scenarios

- Zebra-striped tables
- Displaying form input fields
- Buttons and images
- Emphasis classes

27

B Styling: Tables (1/6)

■ Use table styling

```
<table class="table">
...
</table>
```

HTML

#	First name	Last name	Date of birth
1	Frank	Young	July 5th, 1981
2	Susan	Chi	November 22th, 1966
3	Paul	Wineless	March 1st, 1991

28

B Styling: Tables (2/6)

■ Add more explicit borders

```
<table class="table table-bordered">
...
</table>
```

HTML

#	First name	Last name	Date of birth
1	Frank	Young	July 5th, 1981
2	Susan	Chi	November 22th, 1966
3	Paul	Wineless	March 1st, 1991

29

B Styling: Tables (3/6)

■ Add zebra stripes to a table

```
<table class="table table-striped">
...
</table>
```

HTML

#	First name	Last name	Date of birth
1	Frank	Young	July 5th, 1981
2	Susan	Chi	November 22th, 1966
3	Paul	Wineless	March 1st, 1991

– Uses CSS3's `nth-child()` and does not work in older browsers

30

B Styling: Tables (4/6)

■ Make tables more compact

```
<table class="table table-condensed">
...
</table>
```

[HTML](#)

#	First name	Last name	Date of birth
1	Frank	Young	July 5th, 1981
2	Susan	Chi	November 22th, 1966
3	Paul	Wineless	March 1st, 1991

31

B Styling: Tables (5/6)

■ Enable a hover state

```
<table class="table table-hover">
...
</table>
```

[HTML](#)

#	First name	Last name	Date of birth
1	Frank	Young	July 5th, 1981
2	Susan	Chi	November 22th, 1966
3	Paul	Wineless	March 1st, 1991

32

B Styling: Tables (6/6)

■ Apply contextual classes

.success

.error

.warning

.info

HTML

```
<table class="table">
  <tr class="success">
    ...
  </tr>
</table>
```

#	First name	Last name	Date of birth
1	Frank	Young	July 5th, 1981
2	Susan	Chi	November 22th, 1966
3	Paul	Wineless	March 1st, 1991

33

B Styling: Buttons (1/3)

■ Bootstraps offers buttons that aid the user

Button	CSS class(es)	Meaning
Default	btn	Default button
Primary	btn btn-primary	Identifies primary action in a set of buttons
Info	btn btn-info	Alternative to the default styles
Success	btn btn-success	Indicates successful or positive action
Warning	btn btn-warning	Indicates caution should be taken
Danger	btn btn-danger	Indicates a dangerous or negative action
Inverse	btn-inverse	Alternate dark gray button, no specific meaning
Link	btn btn-link	Deemphasize a button

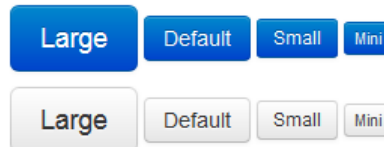
34

B Styling: Buttons (2/3)

- Use bigger or smaller buttons to attract more or less attention

```
<button class="btn btn-large">Large</button>
<button class="btn">Default</button>
<button class="btn btn-small">Small</button>
<button class="btn btn-mini">Mini</button>
```

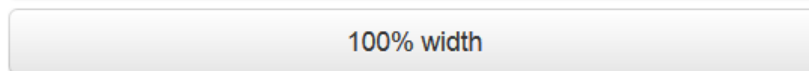
HTML



- Block-level buttons

```
<button class="btn btn-large btn-block">100% width</button>
```

HTML



35

B Styling: Buttons (3/3)

- Apply the button look to the element you wish

```
<a class="btn" href="#">Link</a>
<button class="btn" type="submit">Button</button>
<input class="btn" type="button" value="Input">
<input class="btn" type="submit" value="Submit">
```

HTML



36

B Styling: Forms (1/8)

- Bootstrap styles form elements
 - Input fields get rounded corners, padding and a glow on focus
 - Contextual classes for validation
 - Building the form gets easier with helpful additional classes

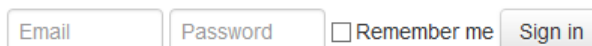
37

B Styling: Forms (2/8)

- Display a form inline

```
<form class="form-inline">
  <input type="text" class="input-small"
    placeholder="Email">
  <input type="password" class="input-small"
    placeholder="Password">
  <label class="checkbox">
    <input type="checkbox">
    Remember me
  </label>
  <button type="submit" class="btn">Sign in</button>
</form>
```

HTML



38

B Styling: Forms (3/8)

■ Display elements on the same line

```
<form class="form-horizontal">
  <div class="control-group">
    <label class="control-label"
      for="inputEmail">E-mail</label>
    <div class="controls">
      <input type="text" id="inputEmail"
        placeholder="E-mail">
    </div>
  </div>
  [...more control groups with labels and controls...]
</form>
```

HTML

E-mail

Password

39

B Styling: Forms (4/8)

■ Use contextual classes for validation

.success

.error

.warning

.info

```
<form class="form-horizontal">
  <div class="control-group error">
    [...labels and controls...]
  </div>
  [...more control groups with labels and controls...]
</form>
```

HTML

E-mail

Password

40

B Styling: Forms (5/8)

■ Default support for search forms

```
<form class="form-search">
  <input type="text" class="input-medium search-query">
  <button type="submit" class="btn">Search</button>
</form>
```

HTML

41

B Styling: Forms (6/8)

■ Prepend an input field

```
<div class="input-prepend">
  <span class="add-on">@</span>
  <input type="text" placeholder="Twitter name">
</div>
```

HTML

■ Append an input field

```
<div class="input-prepend input-append">
  <span class="add-on">&euro;</span>
  <input class="span1" type="text">
  <span class="add-on">,<sup>00</sup></span>
  <button class="btn">Submit</button>
</div>
```

HTML

Buttons are
automatically appended

42

B Styling: Forms (7/8)

■ Size input fields

```
<input class="input-mini" type="text"
placeholder=".input-mini">
<input class="input-medium" type="text"
placeholder=".input-medium">
<input class="input-xxlarge" type="text"
placeholder=".input-xxlarge">
<input class="span3" type="text" placeholder=".span3">
```

HTML

.input-mini

.input-medium

.input-xxlarge

.span3

Size is set to three
grid columns

43

B Styling: Forms (8/8)

■ Emphasize form actions

```
<div class="form-actions">
  <button type="submit"
class="btn btn-primary">Save changes</button>
  <button class="btn">Cancel</button>
</div>
```

HTML

- Buttons are automatically lined up with the form controls

E-mail address

E-mail

Password

Password

Save changes

Cancel

44

Lab: Bootstrap

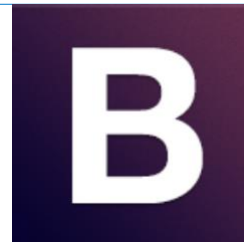
■ Exercise 5: Set up a form with Bootstrap



45

Bootstrap, from Twitter

- A framework to speed up front-end development
- Features:
 - **Positioning** with a 12-column responsive grid
 - **Styling** of tables, fonts, buttons and forms
 - **UI components**, including dropdown menus, progress indicators, alerts and pagination
 - **jQuery plugins**, including modal dialogs, tabs, tooltips, button states and scrollspy



46

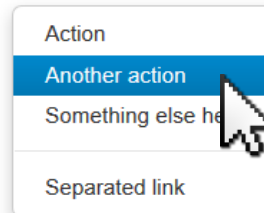
B UI components: Dropdowns

■ Support for dropdown menus

```
<ul class="dropdown-menu">
  <li><a href="#">Action</a></li>
  <li><a href="#">Another action</a></li>
  <li><a href="#">Something else here</a></li>
  <li class="divider"></li>
  <li><a href="#">Separated link</a></li>
</ul>
```

HTML

– Highly useful in combination with other UI components, e.g. buttons



47

B UI components: Alerts

■ Display notifications

```
<div class="alert">
  <button type="button" class="close"
    data-dismiss="alert">&times;</button>
  <strong>Warning!</strong> Something went wrong.
</div>
```

HTML

Warning! Something went horribly, horribly wrong. ✕

■ Support for context

```
<div class="alert alert-info">
  ...
</div>
```

Warning! Something went horribly, horribly wrong. ✕

.alert-success

.alert-error

.alert-info

48

B UI components: Navbar (1/2)

■ Display a navigation bar

```
<div class="navbar">
  <div class="navbar-inner">
    <a class="brand" href="#">Title</a>
    <ul class="nav">
      <li class="active"><a href="#">Home</a></li>
      <li><a href="#">Link</a></li>
      <li><a href="#">Link</a></li>
    </ul>
  </div>
</div>
```

HTML

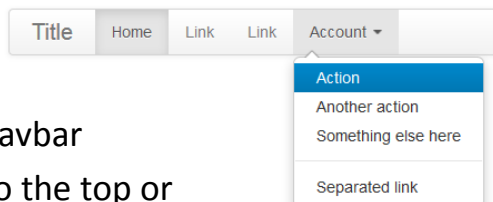


49

B UI components: Navbar (2/2)

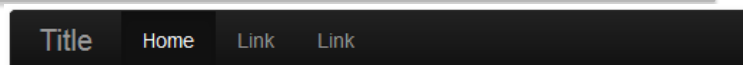
■ The navbar has built-in support for:

- Dropdowns for menu items
- Forms within the navbar
- Fixing the navbar to the top or bottom of the page
- An alternative “inverted” look



```
<div class="navbar navbar-inverse">
  ...
</div>
```

HTML



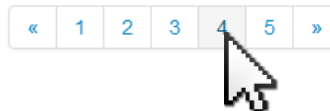
50

B UI components: Pagination

■ Display a pagination bar

```
<div class="pagination">
  <ul>
    <li><a href="#">&laquo;</a></li>
    <li><a href="#">1</a></li>
    <li><a href="#">2</a></li>
    <li><a href="#">3</a></li>
    <li><a href="#">4</a></li>
    <li><a href="#">5</a></li>
    <li><a href="#">&raquo;</a></li>
  </ul>
</div>
```

HTML



51

B UI components: Progress (1/3)

■ Show a progress indicator

```
<div class="progress">
  <div class="bar" style="width: 60%;"></div>
</div>
```

HTML

CSS



■ Indicate status along with progress

.progress-success

.progress-warning

.progress-danger

.progress-info

```
<div class="progress progress-danger">
  <div class="bar" style="width: 60%;"></div>
</div>
```

HTML

CSS



52

B UI components: Progress (2/3)

■ Stack progress bars

```
<div class="progress">
  <div class="bar bar-success"
    style="width: 35%;"></div>
  <div class="bar bar-warning"
    style="width: 20%;"></div>
  <div class="bar bar-danger"
    style="width: 10%;"></div>
</div>
```

HTML

CSS



53

B UI components: Progress (3/3)

■ Additional possibilities

– Striped progress bar

```
<div class="progress progress-striped">
  <div class="bar" style="width: 60%;"></div>
</div>
```

HTML

CSS



– Add animation to the striped progress bar

```
<div class="progress progress-striped active">
  <div class="bar" style="width: 60%;"></div>
</div>
```

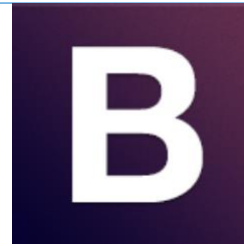
HTML

CSS

54

Bootstrap, from Twitter

- A framework to speed up front-end development
- Features:
 - **Positioning** with a 12-column responsive grid
 - **Styling** of tables, fonts, buttons and forms
 - **UI components**, including dropdown menus, progress indicators, alerts and pagination
 - **jQuery plugins**, including modal dialogs, tabs, tooltips, button states and scrollspy



55

B jQuery plugins: Modal dialog

- Display a modal dialog

```
<a href="#myModal" data-toggle="modal">Launch modal</a>
```

HTML

```
<div id="myModal" class="modal hide fade">
  <div class="modal-header">
    <button data-dismiss="modal"
      class="close">&times;</button>
    <h3>Some heading here</h3>
  </div>
  <div class="modal-body">
    <p>Lorem ipsum dolor sit amet, ...</p>
  </div>
  <div class="modal-footer">
    <button class="btn" data-dismiss="modal"
      aria-hidden="true">Close</button>
    <button class="btn btn-primary">Save</button>
  </div>
</div>
```

HTML

57

B jQuery plugins: Tabs

■ Divide content over tabs

```
<ul class="nav nav-tabs">
  <li><a href="#home" data-toggle="tab">General</a></li>
  <li><a href="#videos" data-toggle="tab">Videos</a></li>
  <li><a href="#extras" data-toggle="tab">Extras</a></li>
</ul>
<div class="tab-content">
  <div class="tab-pane active" id="home">General</div>
  <div class="tab-pane" id="videos">Some videos</div>
  <div class="tab-pane" id="extras">Cool extra stuff</div>
</div>
```

HTML



58

B jQuery plugins: Tooltips

■ Show a tooltip with helpful information

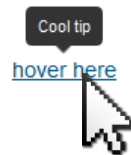
```
<a href="#" rel="tooltip" title="Cool tip">hover here</a>
```

HTML

- Due to performance reasons, tooltips must be explicitly activated

```
$("a[rel=tooltip]").tooltip();
```

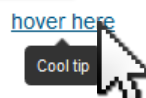
JS



■ Control placement

```
<a href="#" rel="tooltip" title="Cool tip"
  data-placement="bottom">hover here</a>
```

HTML



59

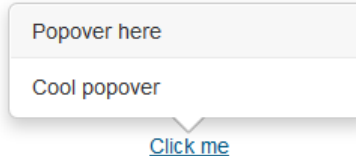
B jQuery plugins: Popovers

■ Display an extended tooltip

```
<a href="#" rel="popover" data-content="Cool popover"
  title="Popover here">Click me</a>
```

HTML

- Due to performance reasons, popovers must be explicitly activated



Click me

```
$( "a[rel=popover]" ).popover();
```

JS

■ Control placement

```
<a href="#" rel="popover" data-placement="right" data-
  content="Cool popover" title="Popover here!">Click me</a>
```

HTML

60

B jQuery plugins: Buttons (1/2)

■ Do more with buttons by supplying feedback

- Button gets disabled and shows a loading text

```
<button id="btnDoWork"
  data-loading-text="Loading...">Do work</button>
```

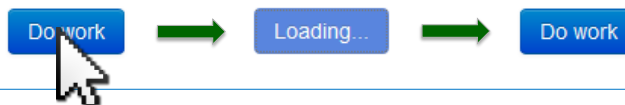
HTML

```
$('#btnDoWork').click(function () {
  $(this).button('loading');

  // Do amazing work

  $(this).button('reset');
});
```

JS



61

B jQuery plugins: Buttons (2/2)

■ Make buttons behave like radiobuttons

```
<div data-toggle="buttons-radio">
  <button class="btn btn-info">Left</button>
  <button class="btn btn-info">Middle</button>
  <button class="btn btn-info">Right</button>
</div>
```

HTML

Left Middle Right

■ Or like checkboxes

```
<div class="btn-group" data-toggle="buttons-checkboxes">
  <button class="btn btn-danger">Left</button>
  <button class="btn btn-danger">Middle</button>
  <button class="btn btn-danger">Right</button>
</div>
```

HTML

Left Middle Right

62

B jQuery plugins: Typeahead

■ Support for autocompletion

```
<input type="text" data-provide="typeahead" data-items="5"
data-source='["Alabama", "Alaska", "Arizona", [...],
"Washington", "West Virginia", "Wisconsin", "Wyoming"]'>
```

HTML

■ Autocompletion also possible with JavaScript

```
$('.typeahead').typeahead({
  source: function (typeahead, qry) {
    return $.get('/ta', { query: qry }, function (data) {
      return typeahead.process(data);
    });
  }
});
```

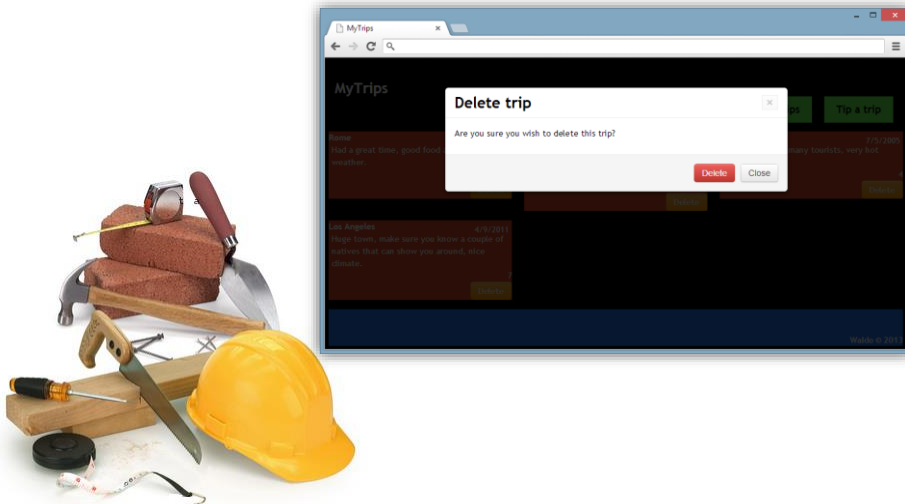
JS

- Arizona
- Arkansas
- Delaware
- Maryland
- North Carolina

63

Lab: Bootstrap

■ Exercise 6: Use modal dialogs for deleting trips







65

CSS frameworks wrap-up

- CSS frameworks speed up front-end development
 - Support for positioning, typography and common UI components
- There are many
 - 960 and Bootstrap, but also Blueprint, Skeleton, jQuery UI, Formalize, 1140 CSS Grid, Boilerplate and many more

66

Agenda

- Building layouts
 - Fixed layouts
 - Fluid layouts
 - Responsive layouts
-  
 - Getting started
 - Mixins
 - Nested rules
 - Operations
 - More LESS
- LESS vs Sass
- CSS frameworks
 - 960 grid system
 - Bootstrap, from Twitter
-  
 - Getting started
 - Mixins and extending
 - Nested rules
 - Operations
 - Control directives
 - More Sass

67



- A **dynamic stylesheet language** that translates to CSS
- Backwards compatible with CSS
- Helps in **maintaining CSS** by offering
 - Calculations through formulas
 - Let <h1> be 8 pixels bigger than the default font size
 - Variables
 - Define a color once and use it multiple times
 - Nested styling

68

Usage

- LESS files can be translated on both the **client-side** and **server-side**
- Client-side translation
 - JavaScript as the translator - www.lesscss.org
- Server-side
 - ASP.NET: .Less - www.dotlesscss.org
(also available as a NuGet package: dotless)
 - Java: Asual - www.asual.com/lesscss/

69

Client-side translation

- Link your LESS stylesheets

```
<link href="default.less"
      rel="stylesheet/less" type="text/css" />
```

HTML

- Parse the LESS stylesheets with JavaScript

```
<script src="less.js" type="text/javascript"></script>
```

HTML

Downloadable from
www.lesscss.org

- Drawbacks
 - The page renders a bit slower
 - JavaScript is required

70

Variables

■ Define variables in your stylesheet

```
@main-color: #5c87b2;
body {
  background-color: @main-color;
}
```

– Note: They're really constants

■ Examples of variable definitions

```
@var1: blue; // Color
```

```
@var2: 12px; // Unit
```

```
@var3: 1.1em; // Unit
```

```
@var4: @var2 + 8px; // Formula
```

71

Agenda

■ Building layouts

- Fixed layouts
- Fluid layouts
- Responsive layouts

■

- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

■ LESS vs Sass

■ CSS frameworks

- 960 grid system
- Bootstrap, from Twitter

■

- Getting started
- Mixins and extending
- Nested rules
- Operations
- Control directives
- More Sass

72

less Mixins (1/8)

■ Define a mixin

```
.border {
  border-top: 3px dotted #666666;
  border-bottom: 4px solid black;
}
```

Note: This class
also appears in
your compiled CSS

■ And reuse the defined CSS properties

LESS

```
.message {
  .border;
}

#nav li {
  .border;
}
```



CSS

```
.message {
  border-top: 3px dotted #666666;
  border-bottom: 4px solid black;
}

#nav li {
  border-top: 3px dotted #666666;
  border-bottom: 4px solid black;
}
```

73

less Mixins (2/8)

■ Supply parameters to a mixin

```
.transition(@duration) {
  -webkit-transition: all @duration ease;
  -o-transition: all @duration ease;
  -ms-transition: all @duration ease;
  -moz-transition: all @duration ease;
  transition: all @duration ease;
}
```

■ And use the mixin

LESS

```
#nav a {
  .transition(0.4s);
}
```

CSS

```
#nav a {
  -webkit-transition: all 0.4s ease;
  -o-transition: all 0.4s ease;
  [...]
}
```

74

less Mixins (3/8)

■ Support for overloading

```
.transition(@duration) {
  transition: all @duration ease;
}
```

```
.transition(@property, @duration) {
  transition: @property @duration ease;
}
```

■ Usage

```
p {
  .transition(5s);
}
```

```
p {
  .transition(background-color, 3s);
}
```

75

less Mixins (4/8)

■ Support for default values

```
.border(@thickness: 5px) {
  border: @thickness solid black;
}
```

■ Usage

```
p {
  .border;
}
```

```
p {
  .border();
}
```

```
p {
  .border(10px);
}
```

76

Mixins (5/8)

■ Parametric mixins without parameters

```
.border() {
  border-top: 3px dotted #666666;
  border-bottom: 4px solid black;
}
```

– Advantage: `.border` does not appear in the compiled CSS anymore

■ Usage is unchanged

```
.message {
  .border;
}
#nav li {
  .border;
}
```

```
.message {
  .border();
}
#nav li {
  .border();
}
```

77

Mixins (6/8)

■ Match patterns by supplying static values

```
.mixin (dark, @color) {
  background-color: darken(@color, 10%);
}
.mixin (light, @color) {
  background-color: lighten(@color, 10%);
}
.mixin (@_, @color) {
  display: block;
}
```

```
@switch: dark;
#div1 {
  .mixin(@switch, #888);
}
```

78

less Mixins (7/8)

■ Match on expressions with **guards**

```
.mixin (@a) when (lightness(@a) >= 50%) {
  background-color: black;
}
.mixin (@a) when (lightness(@a) < 50%) {
  background-color: white;
}
```

LESS

```
#div1 { .mixin(#ddd); }
#div2 { .mixin(#555); }
```

Note: Not supported
by all implementations

CSS

```
#div1 {
  background-color: black;
}
#div2 {
  background-color: white;
}
```

79

less Mixins (8/8)

■ Helpful functions for using with guards

```
iscolor(@color);
```

```
isstring(@string);
```

```
isnumber(@number);
```

```
iskeyword(@keyword);
```

```
ispercentage(@percentage);
```

```
isurl(@url);
```

```
isem(@number);
```

```
ispixel(@number);
```



```
.mixin (@a, @b: black) when (iscolor(@b)) { ... }
```

```
.mixin (@a) when (isnumber(@a)) and (@a > 0) { ... }
```

```
.mixin (@a) when (isnumber(@a)) and not (@a > 0) { ... }
```

80

Agenda

- Building layouts
 - Fixed layouts
 - Fluid layouts
 - Responsive layouts
- 
 - Getting started
 - Mixins
 - Nested rules
 - Operations
 - More LESS
- LESS vs Sass
- CSS frameworks
 - 960 grid system
 - Bootstrap, from Twitter
- 
 - Getting started
 - Mixins and extending
 - Nested rules
 - Operations
 - Control directives
 - More Sass

81

Nested rules

■ Without nested rules...

```
#nav {
  background-color: #660066;
}
#nav a {
  display: block;
  float: left;
  padding: 10px;
  text-transform: uppercase;
  font-weight: bold;
  text-decoration: none;
  color: White;
}
#nav a:hover, nav a.current-page {
  background-color: #bb00bb;
}
```

CSS

Nested rules

■ With nested rules...

```
#nav {
  background-color: #660066;
  a {
    display: block;
    float: left;
    padding: 10px;
    text-transform: uppercase;
    font-weight: bold;
    text-decoration: none;
    color: White;
    &:hover, &.current-page {
      background-color: #bb00bb;
    }
  }
}
```

↗ & to concatenate to its parent
instead of acting as a descendant

Agenda

■ Building layouts

- Fixed layouts
- Fluid layouts
- Responsive layouts

■

- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

■ LESS vs Sass

■ CSS frameworks

- 960 grid system
- Bootstrap, from Twitter

■

- Getting started
- Mixins and extending
- Nested rules
- Operations
- Control directives
- More Sass

84

Operations (1/2)

■ Perform operations

```
h1 {
  font-size: 14px + 4px; // 18px
}
```

```
@base-color: #888 / 4;
h1 {
  font-size: @base-color + #111; // #333333
}
```

```
@container-width: 960px;
body {
  width: (@container-width / 3) + 10; // 330px
}
```

85

Operations (2/2)

■ Math operations

```
round(@number);
```

```
floor(@number);
```

```
ceil(1.7);
```



```
percentage(0.5);
```

■ Useful during calculations

```
@container-width: 1000px;
#container {
  width: ceil(@container-width / 3); // 334px
}
```

86

Agenda

- Building layouts
 - Fixed layouts
 - Fluid layouts
 - Responsive layouts
- 
 - Getting started
 - Mixins
 - Nested rules
 - Operations
 - More LESS
- LESS vs Sass
- CSS frameworks
 - 960 grid system
 - Bootstrap, from Twitter
- 
 - Getting started
 - Mixins and extending
 - Nested rules
 - Operations
 - Control directives
 - More Sass

87

Colors (1/2)

- Extensive support for dealing with colors
 - Retrieving color information
 - Transforming colors

■ Retrieve color information

```
hue(@color);
```

```
saturation(@color);
```

```
lightness(@color);
```

```
alpha(@color);
```

■ Useful when creating new colors

```
#menu {
  background-color: hsl(hue(@base-color), 45%, 72%);
}
```

88

Colors (2/2)

- When transforming, colors are first converted to **HSL**

```
lighten(@base-color, 10%);
```

```
fadein(@base-color, 10%);
```

```
darken(@base-color, 10%);
```

```
fadeout(@base-color, 10%);
```

```
saturate(@base-color, 10%);
```

```
fade(@base-color, 10%);
```

```
desaturate(@base-color, 10%);
```

```
spin(@base-color, 10);
```

```
mix(@base-color, @new-color);
```

```
spin(@base-color, -10);
```

```
h1 {
  color: darken(@base-color, 25%);
}
```

89

Import

- Import LESS stylesheets
 - Reuse the defined variables and mixins

```
@import 'general.less';
```

```
@import 'general';
```

- Import CSS stylesheets

```
@import 'general.css';
```

CSS

- This statement simply ends up in the CSS output

90

String interpolation

- Parse variables within strings

```
@base-url: "http://www.myserverhere.com";  
body  
{  
  background-image: url("@{base-url}/images/bg.png");  
}
```

91

Comments

- Support for single-line comments

```
@var1: blue; // Color
```

– These comments **do not appear** in the CSS output

- Multi-line comments do remain visible in the CSS output

```
/* Navigation */  
#nav {  
  background-color: #660066;  
}  
[...]
```

CSS

92

Lab: LESS is more

■ Exercise 7: Apply LESS to your stylesheets



93

Agenda

- Building layouts
 - Fixed layouts
 - Fluid layouts
 - Responsive layouts



- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

■ LESS vs Sass

- CSS frameworks
 - 960 grid system
 - Bootstrap, from Twitter



- Getting started
- Mixins and extending
- Nested rules
- Operations
- Control directives
- More Sass

94

Syntactically Awesome Stylesheets

- A **dynamic stylesheet language** that translates to CSS
- Helps in **maintaining CSS** by offering
 - Calculations through formulas
 - Let <h1> be 8 pixels bigger than the default font size
 - Variables
 - Define a color once and use it in various places
 - Support for reusing styles and nesting styles
 - All that and more

95

Usage

- Server-side
 - Originated in Ruby
 - ASP.NET: SassAndCoffee
<http://nuget.org/packages/SassAndCoffee>
- Client-side translation is ongoing
 - JavaScript as the translator

96

Usage

■ Sass actually comes in two syntaxes

■ SCSS

- A superset of CSS3

```
$blue: #3bbfce;
$margin: 16px;

.content-navigation {
  border-color: $blue;
}

.border {
  padding: $margin / 2;
  border-color: $blue;
}
```

■ Sass

- The “indented syntax”

```
$blue: #3bbfce
$margin: 16px

.content-navigation
  border-color: $blue

.border
  padding: $margin / 2
  border-color: $blue
```

97

Variables

■ Define variables in your stylesheet

```
$main-color: #5c87b2;
body {
  background-color: @main-color;
}
```

■ Examples of variable definitions

```
$var1: blue; // Color
```



```
$var2: 12px; // Unit
```

```
$var3: 1.1em; // Unit
```

```
$var4: $var2 + 8px; // Formula
```

98

Agenda

- Building layouts
 - Fixed layouts
 - Fluid layouts
 - Responsive layouts
- 
 - Getting started
 - Mixins
 - Nested rules
 - Operations
 - More LESS
- LESS vs Sass
- CSS frameworks
 - 960 grid system
 - Bootstrap, from Twitter
- 
 - Getting started
 - Mixins and extending
 - Nested rules
 - Operations
 - Control directives
 - More Sass

99

Mixins (1/3)

■ Define a mixin

```
@mixin border {
  border-top: 3px solid #666666;
  border-bottom: 4px solid black;
}
```

■ And reuse the defined CSS properties

Sass

```
.message {
  @include border;
}
#nav li {
  @include border;
}
```



CSS

```
.message {
  border-top: 3px dotted #666666;
  border-bottom: 4px solid black;
}
#nav li {
  border-top: 3px dotted #666666;
  border-bottom: 4px solid black;
}
```

100

Mixins (2/3)

■ Supply parameters to a mixin

```
@mixin transition($duration) {
  -webkit-transition: all $duration ease;
  -o-transition: all $duration ease;
  -ms-transition: all $duration ease;
  -moz-transition: all $duration ease;
  transition: all $duration ease;
}
```

■ And use the mixin

Sass

```
#nav a {
  @include transition(0.4s);
}
```

CSS

```
#nav a {
  [...]
  transition: all 0.4s ease;
}
```

101

Mixins (3/3)

■ Support for default values

```
@mixin border($thickness: 5px) {
  border: $thickness solid black;
}
```

■ Usage

```
p {
  @include border;
}
```

```
p {
  @include border();
}
```

```
p {
  @include border(3px);
}
```

102

Extending

■ Inheritance in CSS

Sass

```
.error {
  border: 1px solid red;
  background: pink;
}
.error.intrusion {
  background: darkred;
}
.serious-error {
  @extend .error;
  border-width: 3px;
}
```

CSS

```
.error, .serious-error {
  border: 1px solid red;
  background: pink;
}
.error.intrusion,
.intrusion.serious-error {
  background: darkred;
}
.serious-error {
  border-width: 3px;
}
```

– Multiple extensions are supported

103

Agenda

■ Building layouts

- Fixed layouts
- Fluid layouts
- Responsive layouts



- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

■ LESS vs Sass

■ CSS frameworks

- 960 grid system
- Bootstrap, from Twitter



- Getting started
- Mixins and extending
- Nested rules
- Operations
- Control directives
- More Sass

104

Sass Nested rules

■ Without nested rules...

```
#nav {
  background-color: #660066;
}
#nav a {
  display: block;
  float: left;
  padding: 10px;
  text-transform: uppercase;
  font-weight: bold;
  text-decoration: none;
  color: White;
}
#nav a:hover, nav a.current-page {
  background-color: #bb00bb;
}
```

CSS

Sass Nested rules

■ With nested rules...

```
#nav {
  background-color: #660066;
  a {
    display: block;
    float: left;
    padding: 10px;
    text-transform: uppercase;
    font-weight: bold;
    text-decoration: none;
    color: White;
    &:hover, &.current-page {
      background-color: #bb00bb;
    }
  }
}
```

↑ & to concatenate to its parent instead of acting as a descendant

Nested rules

■ With nested rules...

```
#nav {
  background-color: #660066;
  a {
    display: block;
    float: left;
    padding: 10px;
    text-transform: uppercase;
    font-weight: bold;
    text-decoration: none;
    border: {
      left: 3px solid #660000;
      right: 10px solid #000080;
    }
  }
}
```

↖ & to concatenate to its parent
instead of acting as a descendant

Agenda

■ Building layouts

- Fixed layouts
- Fluid layouts
- Responsive layouts



- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

■ LESS vs Sass

■ CSS frameworks

- 960 grid system
- Bootstrap, from Twitter



- Getting started
- Mixins and extending
- Nested rules
- Operations
- Control directives
- More Sass

108

Sass Operations (1/2)

■ Perform operations

```
h1 {
  font-size: 14px + 4px; // 18px
}
```

```
$base-color: #888 / 4;
h1 {
  font-size: $base-color + #111; // #333333
}
```

```
@container-width: 960px;
body {
  width: ($container-width / 3) + 10; // 330px
}
```

109

Sass Operations (2/2)

■ Math operations

```
round($number);
```

```
floor($number);
```

```
ceil(1.7);
```

```
percentage(0.5);
```

```
min(8px, 1px, 4px);
```



```
max(8px, 1px, 4px);
```

■ Useful during calculations

```
$container-width: 1000px;
#container {
  width: ceil($container-width / 3); // 334px
}
```

110

Agenda

- Building layouts
 - Fixed layouts
 - Fluid layouts
 - Responsive layouts
- 
 - Getting started
 - Mixins
 - Nested rules
 - Operations
 - More LESS
- LESS vs Sass
- CSS frameworks
 - 960 grid system
 - Bootstrap, from Twitter
- 
 - Getting started
 - Mixins and extending
 - Nested rules
 - Operations
 - Control directives
 - More Sass

111

Control Directives (1/4)

- Support for loops with @for

Sass

```
@for $i from 1 through 3 {
  .item-#{ $i } { width: 2em * $i; }
}
```

```
.item-1 { width: 2em; }
.item-2 { width: 4em; }
.item-3 { width: 6em; }
```

CSS

112

Sass Control Directives (2/4)

■ Support for loops with @each

Sass

```
@each $animal in puma, sea-slug, egret, salamander {
  .#{$animal}-icon {
    background-image: url('/images/#{$animal}.png');
  }
}
```

```
.puma-icon {
  background-image: url("/images/puma.png");
}
.sea-slug-icon {
  background-image: url("/images/sea-slug.png");
}
[...]
```

CSS

113

Sass Control Directives (3/4)

■ Support for loops with @while

Sass

```
$i: 6;
@while $i > 0 {
  .item-#{$i} { width: 2em * $i; }
  $i: $i - 2;
}
```

```
.item-6 { width: 12em; }
.item-4 { width: 8em; }
.item-2 { width: 4em; }
```

CSS

114

Control Directives (4/4)

■ Support for loops with @if

Sass

```
$type: monster;
p {
  @if $type == ocean { color: blue; }
  @else if $type == monster { color: green; }
  @else { color: black; }
}
```

CSS

```
p {
  color: green;
}
```

115

Agenda

■ Building layouts

- Fixed layouts
- Fluid layouts
- Responsive layouts



- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

■ LESS vs Sass

■ CSS frameworks

- 960 grid system
- Bootstrap, from Twitter



- Getting started
- Mixins and extending
- Nested rules
- Operations

– Control directives

– More Sass

116

Sass Colors (1/2)

■ Extensive support for dealing with colors

- Retrieve color information
- Transform colors

■ Retrieve color information

```
hue(@color);
```

```
saturation(@color);
```

```
lightness(@color);
```

```
alpha(@color);
```

■ Useful when creating new colors

```
#menu {
  background-color: hsl(hue(@base-color), 45%, 72%);
}
```

117

Sass Colors (2/2)

■ Support for transforming colors

```
lighten($base-color, 10%);
```

```
invert($base-color);
```

```
darken($base-color, 10%);
```

```
grayscale($base-color);
```

```
saturate($base-color, 10%);
```

```
fade-in($base-color, 0.1);
```

```
desaturate($base-color, 10%);
```

```
fade-out($base-color, 0.1);
```

```
mix($base-color, $new-color);
```

```
rgba($acolor, $alpha);
```

```
h1 {
  color: darken($base-color, 25%);
}
```

118

Import

■ Import Sass stylesheets

- Reuse the defined variables and mixins

```
@import 'general.scss';
```

```
@import 'general';
```

■ Import CSS stylesheets

```
@import 'general.css';
```

CSS

- This statement simply ends up in the CSS output

119

String interpolation

■ Parse variables within strings

```
$base-url: "http://www.myserverhere.com";  
body  
{  
  background-image: url("#{ $base-url }/images/bg.png");  
}
```

120

Comments

- Support for single-line comments

```
$var1: blue; // Color
```

– These comments **do not appear** in the CSS output

- Multi-line comments do remain visible in the CSS output

```
/* Navigation */
#nav {
  background-color: #660066;
}
[...]
```

CSS

121

Agenda

- Building layouts

- Fixed layouts
- Fluid layouts
- Responsive layouts



- Getting started
- Mixins
- Nested rules
- Operations
- More LESS

- LESS vs Sass

- CSS frameworks

- 960 grid system
- Bootstrap, from Twitter



- Getting started
- Mixins and extending
- Nested rules
- Operations
- Control directives

- More Sass

122

less vs *Sass*



- Basic reuse and readability improvements
- Client-side parser
- Website niceitude

Winner:



- Basic reuse and readability improvements
- More explicit with @extend, @include and @mixin
- Respects CSS more with \$
- More robust
 - Control directives
 - Compass
- Community presence

123

Questions



124