# SECTION 1 – WELCOME, WELCOME, WELCOME!

# TABLE OF CONTENTS: THEORY LECTURES (CLICK THE TITLES)


1. Three Pillars of Writing Good HTML and CSS (Never Forget Them!) →
2. How CSS Works Behind the Scenes: An Overview →
3. How CSS is Parsed, Part 1: The Cascade and Specificity →
4. How CSS is Parsed, Part 2: Value Processing →
5. How CSS is Parsed, Part 3: Inheritance →
6. How CSS Renders a Website: The Visual Formatting Model →
7. CSS Architecture, Components and BEM →
8. What is Sass? →
9. Review: Basic Principles of Responsive Design and Layout Types →
10. Mobile-first vs Desktop-first and Breakpoints →
11. An Overview of Responsive Images →
12. Why Flexbox: An Overview of the Philosophy Behind Flexbox →
13. Why CSS Grid: A Whole New Mindset →

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

WELCOME, WELCOME, WELCOME!

## LECTURE

WELCOME TO THE MOST ADVANCED CSS COURSE EVER!

02 Natours Project — Setup and First Steps (Part 1)

03 How CSS Works: A Look Behind the Scenes

04 Introduction to Sass and NPM

05 Natours Project — Using Advanced CSS and Sass (Part 2)

06 Natours Project — Advanced Responsive Design (Part 3)

07 Trillo Project — Master Flexbox!

08 A Quick Introduction to CSS Grid Layouts

09 Nexter Project — Master CSS Grid Layouts!

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

WELCOME, WELCOME, WELCOME!

## LECTURE

SETTING UP OUR TOOLS

# SECTION 2 – NATOURS PROJECT PART 1

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - SETUP AND FIRST STEPS (PART 1)

## LECTURE

SECTION INTRO

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - SETUP AND FIRST STEPS (PART 1)

## LECTURE

PROJECT OVERVIEW

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - SETUP AND FIRST STEPS (PART 1)

## LECTURE

BUILDING THE HEADER - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- The best way to perform a basic reset using the universal selector;
- How to set project-wide font definitions.
- How to clip parts of elements using `clip-path`.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - SETUP AND FIRST STEPS (PART 1)

## LECTURE

BUILDING THE HEADER - PART 2

# WHAT YOU WILL LEARN IN THIS LECTURE

- The easiest way to center anything with the `transform`, `top` and `left` properties.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – SETUP AND FIRST STEPS (PART 1)

## LECTURE

CREATING COOL CSS ANIMATIONS

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to create CSS animations using `@keyframes` and the `animation` property.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – SETUP AND FIRST STEPS (PART 1)

## LECTURE

BUILDING A COMPLEX ANIMATED BUTTON – PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- What pseudo-elements and pseudo-classes are;
- How and why to use the `::after` pseudo-element;
- How to create a creative hover animation effect using the `transition` property.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - SETUP AND FIRST STEPS (PART 1)

## LECTURE

BUILDING A COMPLEX ANIMATED BUTTON - PART 2

# SECTION 3 – HOW CSS WORKS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

SECTION INTRO

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

THREE PILLARS OF WRITING GOOD HTML AND CSS (NEVER FORGET THEM!)

# THREE PILLARS TO WRITE GOOD HTML AND CSS... AND BUILD GOOD WEBSITES

- Fluid layouts
- Media queries
- Responsive images
- Correct units
- Desktop-first vs mobile-first

- Clean
- Easy-to-understand
- Growth
- Reusable
- How to organize files
- How to name classes
- How to structure HTML

- Less HTTP requests
- Less code
- Compress code
- Use a CSS preprocessor
- Less images
- Compress images

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

HOW CSS WORKS BEHIND THE SCENES: AN OVERVIEW

# WHAT HAPPENS TO CSS WHEN WE LOAD UP A WEBPAGE?

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

HOW CSS IS PARSED, PART 1: THE CASCADE AND SPECIFICITY

# QUICK REVIEW: CSS TERMINOLOGY

## A CSS RULE

Selector

Declaration block

```
.my-class {
    color: blue;
    text-align: center;
    font-size: 20px;
}
```

Declaration

Property

Declared value

# THE CASCADE (THE "C" IN CSS)

Parse CSS

Resolve conflicting CSS declarations (cascade)

Process final CSS values

## CASCADE

Process of combining different stylesheets and resolving conflicts between different CSS rules and declarations, when more than one rule applies to a certain element.

- Author
- User
- Browser (user agent)

IMPORTANCE (WEIGHT) > SPECIFICITY > SOURCE ORDER

IMPORTANCE > SPECIFICITY > SOURCE ORDER

1. User `!important` declarations
2. Author `!important` declarations
3. Author declarations
4. User declarations
5. Default browser declarations

Same importance?

```
.button {
    font-size: 20px;
    color: white;
    background-color: blue !important;
}

#nav .pull-right .button {
  background-color: green;
}
```

Don't click here!

# IMPORTANCE > SPECIFICITY > SOURCE ORDER

1. User `!important` declarations
2. Author `!important` declarations
3. Author declarations
4. User declarations
5. Default browser declarations

Same importance?

1. Inline styles
2. IDs
3. Classes, pseudo-classes, attribute
4. Elements, pseudo-elements

Same specificity?

The last declaration in the code will override all other declarations and will be applied.

1
```
.button {
    font-size: 20px;
    color: white;
    background-color: blue;
}
```

2
```
nav#nav div.pull-right .button {
    background-color: green;
}
```

3
```
a {
  background-color: purple;
}
```

4
```
#nav a.button:hover {
  background-color: yellow;
}
```

| Inline | IDs | Classes | Elements |
|---|---|---|---|
| ~~(0, 0, 1, 0)~~ | | | |
| (0, 1, 2, 2) | | | |
| ~~(0, 0, 0, 1)~~ | | | |
| ~~(0, 1, 2, 1)~~ | | | |

Don't click here!

# CASCADE AND SPECIFICITY: WHAT YOU NEED TO KNOW

- CSS declarations marked with `!important` have the highest priority;
- But, only use `!important` as a last resource. It's better to use correct specificities — **more maintainable code**!
- Inline styles will always have priority over styles in external stylesheets;
- A selector that contains **1** ID is more specific than one with **1000** classes;
- A selector that contains **1** class is more specific than one with **1000** elements;
- The universal selector `*` has no specificity value (0, 0, 0, 0);
- Rely more on **specificity** than on the **order** of selectors;
- But, rely on order when using 3rd-party stylesheets — always put your author stylesheet last.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

SPECIFICITY IN PRACTICE

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

HOW CSS IS PARSED, PART 2: VALUE PROCESSING

# HOW CSS VALUES ARE PROCESSED

| | width (paragraph) | padding (paragraph) | font-size (root) | font-size (section) | font-size (paragraph) |
|---|---|---|---|---|---|
| **1. Declared value** (author declarations) | 140px<br>66% | – | – | 1.5rem | – |
| **2. Cascaded value** (after the cascade) | 66% | – | 16px (Browser default) | 1.5rem | – |
| **3. Specified value** (defaulting, if there is no cascaded value) | 66% | 0px (Initial value) | 16px | 1.5rem | 24px |
| **4. Computed value** (converting relative values to absolute) | 66% | 0px | 16px | 24px (1.5 * 16px) | 24px |
| **5. Used value** (final calculations, based on layout) | 184.8px | 0px | 16px | 24px | 24px |
| **6. Actual value** (browser and device restrictions) | 185px | 0px | 16px | 24px | 24px |

(66% of 280px)

INHERITANCE

```html
<div class="section">
  <p class="amazing">CSS is absolutely amazing</p>
</div>
```

```css
.section {
  font-size: 1.5rem;
  width: 280px;
  background-color: orangered;
}

p {
  width: 140px;
  background-color: green;
}

.amazing {
  width: 66%;
}
```

(Let's analyse the green paragraph)

# HOW UNITS ARE CONVERTED FROM RELATIVE TO ABSOLUTE (PX)

| | | Example (x) | How to convert to pixels | Result in pixels |
|---|---|---|---|---|
| | % (fonts) | 150% | x% * parent's computed font-size | 24px |
| | % (lengths) | 10% | x% * parent's computed **width** | 100px |
| Font-based | em (font) | 3em | x * **parent** computed font-size | 72px (3 * 24) |
| Font-based | em (lengths) | 2em | x * **current element** computed font-size | 48px |
| Font-based | rem | 10rem | x * root computed font-size | 160px |
| Viewport-based | vh | 90vh | x * 1% of viewport height | 90% of the current viewport height |
| Viewport-based | vw | 80vw | x * 1% of viewport width | 80% of the current viewport width |

**4.** Computed value
(converting relative values to absolute)

```
html, body {
  font-size: 16px;
  width: 80vw;
}

header {
  font-size: 150%;
  padding: 2em;
  margin-bottom: 10rem;
  height: 90vh;
  width: 1000px;
}

.header-child {
  font-size: 3em;
  padding: 10%;
}
```

# CSS VALUE PROCESSING: WHAT YOU NEED TO KNOW

- Each property has an initial value, used if nothing is declared (and if there is no inheritance — see next lecture);
- Browsers specify a **root `font-size`** for each page (usually `16px`);
- Percentages and relative values are always converted to pixels;
- Percentages are measured relative to their parent's **`font-size`**, if used to specify `font-size`;
- Percentages are measured relative to their parent's **`width`**, if used to specify lengths;
- `em` are measured relative to their **parent** `font-size`, if used to specify `font-size`;
- `em` are measured relative to the **current** `font-size`, if used to specify lengths;
- `rem` are always measured relative to the **document's root** `font-size`;
- `vh` and `vw` are simply percentage measurements of the viewport's `height` and `width`.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

HOW CSS IS PARSED, PART 3: INHERITANCE

# INHERITANCE IN CSS

Every CSS property must have a value

Is there a cascaded value?

**YES** → Specified value = Cascaded value

**NO** → Is the property inherited? (*specific to each property*)

**NO** → Specified value = Initial value (*specific to each property*)

**YES** → Specified value = Computed value of parent element → `30px`

THIS IS INHERITANCE!

## EXAMPLE

Let's analyse `line-height` on `.child`

```
.parent {
  font-size: 20px;
  line-height: 150%;
}

.child {
  font-size: 25px;
}
```

### `line-height` property specification

| | |
|---|---|
| Initial value | `normal` |
| Applies to | all elements. It also applies to `::first-letter` and `::first-line`. |
| Inherited | yes |
| Percentages | refer to the font size of the element itself |
| Media | visual |
| Computed value | for percentage and length values, the absolute length, otherwise as specified |
| Animation type | either number or length |
| Canonical order | the unique non-ambiguous order defined by the formal grammar |

**Source:** https://developer.mozilla.org/en/docs/Web/CSS/line-height

# INHERITANCE: WHAT YOU NEED TO KNOW

- Inheritance passes the values for some specific properties from parents to children — **more maintainable code**;
- Properties related to text are inherited: `font-family`, `font-size`, `color`, etc;
- The computed value of a property is what gets inherited, **not** the declared value.
- Inheritance of a property only works if no one declares a value for that property;
- The `inherit` keyword forces inheritance on a certain property;
- The `initial` keyword resets a property to its initial value.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

CONVERTING PX TO REM: AN EFFECTIVE WORKFLOW

# WHAT YOU WILL LEARN IN THIS LECTURE

- How and why to use `rem` units in our project;
- A great workflow for converting `px` to `rem`.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

HOW CSS RENDERS A WEBSITE: THE VISUAL FORMATTING MODEL

# REMEMBER...?

Load HTML
Parse HTML
Document Object Model (DOM)
Parse CSS
Load CSS
Resolve conflicting CSS declarations (cascade)
CSS Object Model (CSSOM)
Process final CSS values
Final rendered website
Website rendering: the visual formatting model
Render tree

# THE VISUAL FORMATTING MODEL

**DEFINITION**

Algorithm that calculates boxes and determines the layout of theses boxes, for each element in the render tree, in order to determine the final layout of the page.

- **Dimensions of boxes**: the box model;
- **Box type**: inline, block and inline-block;
- **Positioning scheme**: floats and positioning;
- **Stacking contexts**;
- Other elements in the render tree;
- Viewport size, dimensions of images, etc.

# 1. THE BOX MODEL

- **Content**: text, images, etc;
- **Padding**: transparent area around the content, inside of the box;
- **Border**: goes around the padding and the content;
- **Margin**: space between boxes;
- **Fill area**: area that gets filled with background color or background image.

# 1. THE BOX MODEL: HEIGHTS AND WIDTHS

**total width** = right border + right padding + specified width + left padding + left border

**total height** = top border + top padding + specified height + bottom padding + bottom border

**Example:** height = 0 + 20px + 100px + 20px + 0 = 140px

# 1. THE BOX MODEL WITH BOX-SIZING: BORDER-BOX

**total width** = ~~right border~~ + ~~right padding~~ + specified width + ~~left padding~~ + ~~left border~~

**total height** = ~~top border~~ + ~~top padding~~ + specified height + ~~bottom padding~~ + ~~bottom border~~

**Example:** height = 0 + ~~20px~~ + 100px + ~~20px~~ + 0 = 100px

# 2. BOX TYPES: INLINE, BLOCK-LEVEL AND INLINE-BLOCK

## Block-level boxes

- Elements formatted visually as blocks
- 100% of parent's width
- Vertically, one after another
- Box-model applies as showed

```
display: block
(display: flex)
(display: list-item)
(display: table)
```

## Inline-block boxes

- A mix of block and inline
- Occupies only content's space
- No line-breaks
- Box-model applies as showed

```
display: inline-block
```

## Inline boxes

- Content is distributed in lines
- Occupies only content's space
- No line-breaks
- No heights and widths
- Paddings and margins only horizontal (left and right)

```
display: inline
```

# 3. POSITIONING SCHEMES: NORMAL FLOW, ABSOLUTE POSITIONING AND FLOATS

## Normal flow

- Default positioning scheme;
- **NOT** floated;
- **NOT** absolutely positioned;
- Elements laid out according to their source order.

```
Default
position: relative
```

## Floats

- **Element is removed from the normal flow;**
- Text and inline elements will wrap around the floated element;
- The container will not adjust its height to the element.

```
float: left
float: right
```

=

≠

## Absolute positioning

- **Element is removed from the normal flow**
- No impact on surrounding content or elements;
- We use `top`, `bottom`, `left` and `right` to offset the element from its relatively positioned container.

```
position: absolute
position: fixed
```

# 4. STACKING CONTEXTS

```
z-index: 3
position: relative
```

```
z-index: 2
position: absolute
```

```
z-index: 1
position: relative
```

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

CSS ARCHITECTURE, COMPONENTS AND BEM

# THE THINK - BUILD - ARCHITECT MINDSET

- Clean
- Modular
- Reusable
- Ready for growth

## THINK

**Think** about the layout of your webpage or web app before writing code.

## BUILD

**Build** your layout in HTML and CSS with a consistent structure for naming classes.

## ARCHITECT

Create a logical **architecture** for your CSS with files and folders.

# THINKING ABOUT THE LAYOUT

THINK → BUILD → ARCHITECT

## COMPONENT-DRIVEN DESIGN

- **Modular building blocks** that make up interfaces;
- Held together by the **layout** of the page;
- **Re-usable** across a project, and between different projects;
- **Independent**, allowing us to use them anywhere on the page.

ATOMIC DESIGN

COMPONENT

[Image taken from bradfrost.com]

# BUILDING WITH MEANINGFUL CLASS NAMES

THINK → **BUILD** → ARCHITECT

## BEM

- **B**lock **E**lement **M**odifier
- **BLOCK**: standalone component that is meaningful on its own.
- **ELEMENT**: part of a block that has no standalone meaning.
- **MODIFIER**: a different version of a block or an element.

```
.block {}
.block__element {}
.block__element--modifier {}
```

Low-specificity BEM selectors

```
<figure class="recipe">
  <div class="recipe__hero">
    <img class="recipe__img" src="
  </div>
  <div class="recipe__info">
    <div class="recipe__category">
      Veggie
    </div>
    <figcaption class="recipe__details">
      <h2 class="recipe__title">Pizza Vegetale 👌</h2>
      <p class="recipe__description">
        Yummy veggie pizza with tasty olives
      </p>
    </figcaption>
    <div class="recipe__stats-box">
      <div class="recipe__stat">
        <span class="recipe__stat-value">4.9</span>
        <span class="recipe__stat-name recipe__stat-name--1">Stars</span>
      </div>
      <div class="recipe__stat">
        <span class="recipe__stat-value">45</span>
        <span class="recipe__stat-name recipe__stat-name--2">Minutes</span>
      </div>
      <div class="recipe__stat">
        <span class="recipe__stat-value">2</span>
        <span class="recipe__stat-name recipe__stat-name--3">Persons</span>
      </div>
    </div>
  </div>
  <a class="recipe__btn btn btn--round" href="#">Try</a>
</figure>
```

# ARCHITECTING WITH FILES AND FOLDERS

THINK → BUILD → ARCHITECT

## THE 7-1 PATTERN

7 different folders for partial Sass files, and
1 main Sass file to import all other files into
a compiled CSS stylesheet.

### THE 7 FOLDERS

- base/
- components/
- layout/
- pages/
- themes/
- abstracts/
- vendors/

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

HOW CSS WORKS: A LOOK BEHIND THE SCENES

## LECTURE

IMPLEMENTING BEM IN THE NATOUR PROJECT

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to use the BEM method in practice.

# SECTION 4 – INTRODUCTION TO SASS AND NPM

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

SECTION

INTRODUCTION TO SASS AND NPM

LECTURE

SECTION INTRO

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

SECTION

INTRODUCTION TO SASS AND NPM

LECTURE

WHAT IS SASS?

# WHAT IS SASS AND HOW DOES IT WORK?

## SASS

Sass is a CSS preprocessor, an extension of CSS that adds power and elegance to the basic language.

SASS SOURCE CODE → (Sass compiler) → COMPILED CSS CODE

# MAIN SASS FEATURES

- **Variables**: for reusable values such as colors, font-sizes, spacing, etc;
- **Nesting**: to nest selectors inside of one another, allowing us to write less code;
- **Operators**: for mathematical operations right inside of CSS;
- **Partials and imports**: to write CSS in different files and importing them all into one single file;
- **Mixins**: to write reusable pieces of CSS code;
- **Functions**: similar to mixins, with the difference that they produce a value that can than be used;
- **Extends**: to make different selectors inherit declarations that are common to all of them;
- **Control directives**: for writing complex code using conditionals and loops (not covered in this course).

# SASS AND SCSS: CLEARING UP THE CONFUSION

Sass syntax

```
.navigation
  list-style: none
  float: left

  & li
    display: inline-block
    margin-left: 30px
```

SCSS syntax

```
.navigation {
  list-style: none;
  float: left;

  & li {
    display: inline-block;
    margin-left: 30px;
  }
}
```

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

INTRODUCTION TO SASS AND NPM

## LECTURE

FIRST STEPS WITH SASS: VARIABLES AND NESTING

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

INTRODUCTION TO SASS AND NPM

## LECTURE

FIRST STEPS WITH SASS: MIXINS, EXTENDS AND FUNCTIONS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

INTRODUCTION TO SASS AND NPM

## LECTURE

A BRIEF INTRODUCTION TO THE COMMAND LINE

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

INTRODUCTION TO SASS AND NPM

## LECTURE

NPM PACKAGES: LET'S INSTALL SASS LOCALLY

# A BRIEF INTRODUCTION TO NPM AND THE NODE ECOSYSTEM

Allows developers to write and run JavaScript applications on the server. Developers started using node.js to also write tools to help them with **local web development**.

NPM is a simple command line interface that allows developers to **install and manage packages** on their local computers. There are all kinds of open-source **tools, libraries and frameworks** needed for modern development. Modern web development could simply not exist without a package manager.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

INTRODUCTION TO SASS AND NPM

## LECTURE

NPM SCRIPTS: LET'S WRITE AND COMPILE SASS LOCALLY

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

INTRODUCTION TO SASS AND NPM

## LECTURE

THE EASIEST WAY OF AUTOMATICALLY RELOADING A PAGE ON FILE CHANGES

# SECTION 5 – NATOURS PROJECT PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

SECTION INTRO

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

CONVERTING OUR CSS CODE TO SASS: VARIABLES AND NESTING

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

IMPLEMENTING THE 7-1 CSS ARCHITECTURE WITH SASS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

REVIEW: BASIC PRINCIPLES OF RESPONSIVE DESIGN AND LAYOUT TYPES

# REVIEW: BASIC RESPONSIVE DESIGN PRINCIPLES

1

## FLUID GRIDS AND LAYOUTS

To allow content to easily adapt to the current viewport width used to browse the website. Uses `%` rather than `px` for all layout-related lengths.

2

## FLEXIBLE/RESPONSIVE IMAGES

Images behave differently than text content, and so we need to ensure that they also adapt nicely to the current viewport.

3

## MEDIA QUERIES

To change styles on certain viewport widths (breakpoints), allowing us to create different version of our website for different widths.

# LAYOUT TYPES

THIS PROJECT

FLOAT LAYOUTS

2ND PROJECT

FLEXBOX

3RD PROJECT

CSS GRID

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING A CUSTOM GRID WITH FLOATS

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to architect and build a simple grid system;
- How the attribute selector works;
- How the `:not` pseudo-class works;
- How `calc()` works, and what's the difference between `calc()` and simple Sass operations.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE ABOUT SECTION - PART 1

@JONASSCHMEDTMAN

# WHAT YOU WILL LEARN IN THIS LECTURE

- Thinking about components;
- How and why to use utility classes;
- How to use the `background-clip` property;
- How to `transform` multiple properties simultaneously;
- How to use the `outline-offset` property together with `outline`;
- How to style elements that are NOT hovered while others are.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE ABOUT SECTION - PART 2

@JONASSCHMEDTMAN

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE ABOUT SECTION – PART 3

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE FEATURES SECTION

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to include and use an icon font;
- Another way of creating the “skewed section” design;
- How and when to use the direct child selector.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE TOURS SECTION - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to build an amazing, rotating card;
- How to use `perspective` in CSS;
- How to use the `backface-visibility` property;
- Using background blend modes;
- How and when to use `box-decoration-break`;

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE TOURS SECTION – PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE TOURS SECTION - PART 3

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE STORIES SECTION - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to make text flow around shapes with `shape-outside` and `float`;
- How to apply a `filter` to images;
- How to create a background video covering an entire section;
- How to use the `<video>` HTML element;
- How and when to use the `object-fit` property.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE STORIES SECTION - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE STORIES SECTION - PART 3

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE BOOKING SECTION - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to implement “solid-color gradients”;
- How the general and adjacent sibling selectors work and why we need them;
- How to use the `::input-placeholder` pseudo-element;
- How and when to use the `:focus`, `:invalid`, `placeholder-shown` and `:checked` pseudo-classes;
- Techniques to build custom radio buttons.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE BOOKING SECTION - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE BOOKING SECTION – PART 3

@JONASSCHMEDTMAN

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE FOOTER

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to design a simple website footer.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE NAVIGATION – PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- What the “checkbox hack” is and how it works;
- How to create custom animation timing functions using cubic bezier curves;
- How to animate “solid-color gradients”;
- How and why to use `transform-origin`;
- In general: create an amazingly creative effect!

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE NAVIGATION – PART 2

@JONASSCHMEDTMAN

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING THE NAVIGATION – PART 3

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING A PURE CSS POPUP - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to build a nice popup with only CSS;
- How to use the `:target` pseudo-class;
- How to create boxes with equal height using `display: table-cell`;
- How to create CSS text columns;
- How to automatically hyphenate words using `hyphens`.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - USING ADVANCED CSS AND SASS (PART 2)

## LECTURE

BUILDING A PURE CSS POPUP - PART 2

# SECTION 6 – NATOURS PROJECT PART 3

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

SECTION INTRO

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

MOBILE-FIRST VS DESKTOP-FIRST AND BREAKPOINTS

# RESPONSIVE DESIGN STRATEGIES

```
html { font-size: 20px; }
@media (max-width: 600px) {
  html { font-size: 16px; }
}
```

## DESKTOP-FIRST

- Start writing CSS for the desktop: large screen;
- Then, media queries shrink design to smaller screens.

## MOBILE-FIRST

- Start writing CSS for mobile devices: small screen;
- Then, media queries expand design to a large desktop screen;
- Forces us to reduce websites and apps to the absolute essentials.

```
html { font-size: 16px; }
@media (min-width: 600px) {
  html { font-size: 20px; }
}
```

# RESPONSIVE DESIGN STRATEGIES: MAX-WIDTH AND MIN-WIDTH

Media queries don't add any importance or specificity to selectors, so code order matters — **media queries at the end**

# IS MOBILE-FIRST RIGHT FOR YOU?

## PROS

- 100% optimised for the mobile experience;
- Reduces websites and apps to the absolute essentials;
- Results in smaller, faster and more efficient products;
- Prioritizes content over aesthetic design, which may be desirable.

## CONS

- The desktop version might feel overly empty and simplistic;
- More difficult and counterintuitive to develop;
- Less creative freedom, making it more difficult to create distinctive products;
- Clients are used to see a desktop version of the site as a prototype;
- Do your users even use the mobile internet? What's the purpose of your website?

**NO MATTER WHAT YOU DO, ALWAYS KEEP BOTH DESKTOP AND MOBILE IN MIND**

# SELECTING OUR BREAKPOINTS: THE OPTIONS

BAD

GOOD

PERFECT

# SELECTING OUR BREAKPOINTS: A GOOD APPROACH

[http://gs.statcounter.com/screen-resolution-stats#monthly-201608-201708-bar]

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

LET'S USE THE POWER OF SASS MIXINS TO WRITE MEDIA QUERIES

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to use a powerful Sass mixing to write all our media queries;
- How to use the `@content` and `@if` Sass directives;
- Taking advantage of Chrome DevTools for responsive design.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

WRITING MEDIA QUERIES - BASE, TYPOGRAPHY AND LAYOUT

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

WRITING MEDIA QUERIES - LAYOUT, ABOUT AND FEATURES SECTIONS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

WRITING MEDIA QUERIES - TOURS, STORIES AND BOOKING SECTIONS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

AN OVERVIEW OF RESPONSIVE IMAGES

# WHAT ARE RESPONSIVE IMAGES ANYWAY?

Responsive design

Maintainable and scalable code

Web performance

## RESPONSIVE IMAGES

The goal of responsive images is to serve the **right image** to the **right screen size** and device, in order to avoid downloading unnecessary large images on smaller screens.

# WHEN TO USE RESPONSIVE IMAGES: THE 3 USE CASES

## RESOLUTION SWITCHING

Large screen

Small screen

(Decrease image resolution on smaller screen)

## DENSITY SWITCHING

@2x screen (high-res)

@1x screen (low-res)

(Half the image resolution on @1x screen)

## ART DIRECTION

Large screen

Small screen

*Image details were preserved, but the image is different*

(Different image on smaller screen)

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

RESPONSIVE IMAGES IN HTML - ART DIRECTION AND DENSITY SWITCHING

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to use the `srcset` attribute on the `<img>` and `<source>` elements, together with density descriptors;
- How and why to use the `<picture>` element for art direction;
- How to write media queries in HTML.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT - ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

RESPONSIVE IMAGES IN HTML - DENSITY AND RESOLUTION SWITCHING

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to allow the browser to decide the best image to download, using the `srcset` attribute, width descriptors, and the `sizes` attribute of the `<img>` element.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

RESPONSIVE IMAGES IN CSS

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to implement responsive images in CSS;
- How to use resolution media queries to target high-resolution screens with 2x.
- How to combine multiple conditions in media queries.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

TESTING FOR BROWSER SUPPORT WITH @SUPPORTS

# WARNING!

MANY OF THE NEW CSS FEATURES I SHOWED YOU ARE HIGHLY EXPERIMENTAL AND ONLY WORK IN TOP MODERN BROWSERS

ALWAYS CHECK CANIUSE.COM BEFORE USING A MODERN CSS PROPERTY IN PRODUCTION

USE GRACEFUL DEGRADATION WITH @SUPPORTS

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to use `@supports` feature queries;
- Implement graceful degradation on selected properties;
- How to use `backdrop-filter`.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

SETTING UP A SIMPLE BUILD PROCESS WITH NPM SCRIPTS

# A SIMPLE BUILD PROCESS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NATOURS PROJECT – ADVANCED RESPONSIVE DESIGN (PART 3)

## LECTURE

WRAPPING UP THE NATOURS PROJECT: FINAL CONSIDERATIONS

# SECTION 7 – TRILLO PROJECT: MASTER FLEXBOX!

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

SECTION INTRO

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

WHY FLEXBOX: AN OVERVIEW OF THE PHILOSOPHY BEHIND FLEXBOX

# A TRUE REVOLUTION

## FLEXBOX

- Flexbox is a new module in CSS3 that makes it easy to align elements to one another, in different directions and orders;
- The main idea behind flexbox is to give the container the ability to expand and to shrink elements to best use all the available space;
- Flexbox replaces float layouts, using less, and more readable and logical code;
- Flexbox completely changes the way that we build one-dimensional layouts;
- A true revolution in CSS!

# MAIN FLEXBOX CONCEPTS

```
display: flex
```

(display: flex-inline)

# FLEXBOX PROPERTIES OVERVIEW

## CONTAINER

1. **flex-direction: row** | row-reverse | column | column-reverse
2. **flex-wrap: nowrap** | wrap | wrap-reverse
3. **justify-content: flex-start** | flex-end | center | space-between | space-around | space-evenly
4. **align-items: stretch** | flex-start | flex-end | center | baseline
5. **align-content: stretch** | flex-start | flex-end | center | space-between | space-around

## ITEM

1. **align-self: auto** | stretch | flex-start | flex-end | center | baseline
2. **order: 0** | <integer>
3. **flex-grow: 0** | <integer>
4. **flex-shrink: 1** | <integer>
5. **flex-basis: auto** | <length>

**flex: 0 1 auto** | <int> <int> <len>

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

A BASIC INTRO TO FLEXBOX: THE FLEX CONTAINER

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

A BASIC INTRO TO FLEXBOX: FLEX ITEMS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

A BASIC INTRO TO FLEXBOX: ADDING MORE FLEX ITEMS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

PROJECT OVERVIEW

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

DEFINING PROJECT SETTINGS AND CUSTOM PROPERTIES

# WHAT YOU WILL LEARN IN THIS LECTURE

- How and why to use CSS custom properties.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE OVERALL LAYOUT

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to think about the overall layout of an app;
- Use flexbox in a real-world project for the first time.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE HEADER - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- Why to use SVG icons vs. font icons;
- How to find, generate and use SVG sprites in HTML;
- How to change the color of an SVG icon in CSS;
- How to use more advanced flexbox alignment techniques, including `justify-content`, `align-items`, `align-self` and `flex`.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE HEADER - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE HEADER - PART 3

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE NAVIGATION - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to use `scaleY` and multiple transition properties with different settings, to create a creative hover effect;
- How and why to use the `currentColor` CSS variable;
- How to use some more advanced flexbox alignment techniques, including `flex-direction`, `justify-content` and `align-items`.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE NAVIGATION - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE HOTEL OVERVIEW - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to create an infinite animation;
- How to use `margin: auto` with flexbox, and why it's so powerful;
- Continue to use flexbox properties for easy positioning and alignment.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE HOTEL OVERVIEW - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE DESCRIPTION SECTION - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- Continue to use flexbox, including `flex-wrap` to build a multi-column list;
- How and why to use CSS masks with `mask-image` and `mask-size`.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE DESCRIPTION SECTION - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE USER REVIEWS SECTION

# WHAT YOU WILL LEARN IN THIS LECTURE

- Continue using and practicing flexbox.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

BUILDING THE CTA SECTION

# WHAT YOU WILL LEARN IN THIS LECTURE

- Yet another creative and modern hover effect 😍

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

WRITING MEDIA QUERIES - PART 1

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

WRITING MEDIA QUERIES - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

TRILLO PROJECT - MASTER FLEXBOX!

## LECTURE

WRAPPING UP THE TRILLO PROJECT: FINAL CONSIDERATIONS

# CHALLENGES, ANYONE?

1. Display some kind of user menu when hovering over the username in `.user-nav`;
2. Display a message menu when hovering over the chat icon in `.user-nav` (maybe like facebook);
3. Display a box with search suggestions as soon as the user starts typing in the search field;
4. Create a caption for the `.gallery__item` with a nice hover effect;
5. Make the page 100% responsive even for viewport sizes below 500px, maybe even responsive images.

# SECTION 8 – CSS GRID INTRO

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

SECTION INTRO

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

WHY CSS GRID: A WHOLE NEW MINDSET

# A WHOLE NEW MINDSET

## CSS GRID

- CSS Grid Layout is a brand new module that brings a two-dimensional grid system to CSS for the first time;
- CSS Grid replaces float layouts, using less, and more readable and logical CSS and HTML;
- CSS Grid works perfectly together with Flexbox, which is best to handle one-dimensional components and layouts;
- CSS Grid completely changes the way that we envision and build two-dimensional layouts.

# A WHOLE NEW MINDSET

### World's best luxury homes

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Tenetur distinctio necessitatibus pariatur voluptatibus.

### Only the best properties

Voluptatum mollitia quae. Vero ipsum sapiente molestias accusamus rerum sed a eligendi aut quia.

### All homes in top locations

Tenetur distinctio necessitatibus pariatur voluptatibus quidem consequatur harum.

### New home in one week

Vero ipsum sapiente molestias accusamus rerum. Lorem, ipsum dolor sit amet consectetur adipisicing elit.

### Top 1% realtors

Quidem consequatur harum, voluptatum mollitia quae. Tenetur distinctio necessitatibus pariatur voluptatibus.

### Secure payments on XXX

Pariatur voluptatibus quidem consequatur harum, voluptatum mollitia quae.

# CSS GRID TERMINOLOGY

```
display: grid
```

(display: grid-inline)

# CSS GRID TERMINOLOGY

# CSS GRID PROPERTIES OVERVIEW

## CONTAINER

1. grid-template-rows
   grid-template-columns
   grid-template-areas
   → grid-template

2. grid-row-gap
   grid-column-gap
   → grid-gap

3. justify-items
   align-items
   justify-content
   align-content

4. grid-auto-rows
   grid-auto-columns
   grid-auto-flow

## ITEM

1. grid-row-start
   grid-row-end
   → grid-row
   grid-column-start
   grid-column-end
   → grid-column
   → grid-area

2. justify-self
   align-self

3. order

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

QUICK SETUP FOR THIS SECTION

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

CREATING OUR FIRST GRID

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

GETTING FAMILIAR WITH THE FR UNIT

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

POSITIONING GRID ITEMS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

SPANNING GRID ITEMS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

GRID CHALLENGE

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

GRID CHALLENGE: A BASIC SOLUTION

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

NAMING GRID LINES

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

NAMING GRID AREAS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

IMPLICIT GRIDS VS. EXPLICIT GRIDS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

ALIGNING GRID ITEMS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

ALIGNING TRACKS

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

USING MIN-CONTENT, MAX-CONTENT AND THE MINMAX() FUNCTION

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

A QUICK INTRODUCTION TO CSS GRID LAYOUTS

## LECTURE

RESPONSIVE LAYOUTS WITH AUTO-FIT AND AUTO-FILL

# SECTION 9 – NEXTER PROJECT – MASTER CSS GRID LAYOUTS!

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

PROJECT OVERVIEW & SETUP

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE OVERALL LAYOUT - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to build a complex and modern layout using advanced CSS Grid techniques;
- How to choose different row and column track sizes for different types of content.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE OVERALL LAYOUT - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE FEATURES SECTION - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How and why to create grids inside of grids;
- How to create a responsive component without media queries;
- How to build a small component using CSS Grid.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE FEATURES SECTION - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE STORY SECTION - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to deal with overlapping grid items;
- Why images are special and behave differently than other grid items;
- How to decide if flexbox is a better tool in certain situations.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE STORY SECTION - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE HOMES SECTION - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to build a rather complex component using a mix of CSS Grid properties, overlapping and flexbox.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE HOMES SECTION - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE GALLERY - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to create a complex grid-looking gallery;
- Using `object-fit` together with images for grid items.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE GALLERY - PART 2

@JONASSCHMEDTMAN

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE FOOTER

# WHAT YOU WILL LEARN IN THIS LECTURE

- Apply the concepts you already learned.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE SIDEBAR

# WHAT YOU WILL LEARN IN THIS LECTURE

- Apply the concepts you already learned.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE HEADER - PART 1

# WHAT YOU WILL LEARN IN THIS LECTURE

- How to manage vertical spacing in a responsive layout using CSS Grid techniques;
- How to use `::before` and `::after` as grid items.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE HEADER - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT – MASTER CSS GRID LAYOUTS!

## LECTURE

BUILDING THE REALTORS SECTION

# WHAT YOU WILL LEARN IN THIS LECTURE

- Apply the concepts you already learned.

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

WRITING MEDIA QUERIES - PART 1

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT - MASTER CSS GRID LAYOUTS!

## LECTURE

WRITING MEDIA QUERIES - PART 2

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

NEXTER PROJECT – MASTER CSS GRID LAYOUTS!

## LECTURE

BROWSER SUPPORT FOR CSS GRID

# SECTION 10 – THAT'S IT, EVERYONE!

JONAS.IO
SCHMEDTMANN

# ADVANCED CSS AND SASS

## TAKE YOUR CSS TO THE NEXT LEVEL!

@JONASSCHMEDTMAN

## SECTION

THAT'S IT, EVERYONE!

## LECTURE

SEE YOU NEXT TIME, CSS MASTER!

# END