



# CSS Grid

e-Portfolio by Friedrich Schmierer for  
Software Engineering Class, Ms. Berkling

# Prerequisite

- Have a Browser (Firefox recommended)
- Clone my repository from

<https://github.com/schmiefri/e-Portfolio-CSS-Grid>

# Content

- Layouts of the Past
- Clean Markup
- The Grid
- Practice

# Layouts of the Past

- 90's Table Layouts
- Div Soup
- HAVE YOU HEARD OF FLEXBOX??
  - Breaking Markup for Layouting

# Clean Markup – The Problem

- This is hard to maintain:

```
▼<div id="content-arrow" class="container">
  ::before
  ▶<div class="row">⋮</div>
  ▶<div class="row">⋮</div>
  ▼<div class="row">
    ::before
    ▼<div class="related-content related-content-subhome clear">
      ::before
      ▼<div class="col-sm-12 col-lg-8 related-content-tile
        clickable">⋮
        ▼<div class="col-sm-6 col-md-5 col-lg-6 pic">
          
          </div>
        ▼<div class="col-sm-6 col-md-7 col-lg-6 txt txt-ext">
          | ▶<h2>⋮</h2>
```

# Clean Markup – Motivation

- Divs for layouting get out of hand
- CSS Frameworks are easy, but confusing
- Accessibility becomes weird

# Clean Markup – Motivation

```
<!DOCTYPE html>
<html class="js" lang="en" dir="ltr" xml:lang="en" xmlns="http://www.w3.org/1999/
  <head>
  <body class="front not-logged-in page-home no-sidebars drupalorg-front" data-
    <div id="skip-links">
    <div id="header" class="clear-block">
    <div id="page" class="container-12 clear-block">
      <div id="page-inner">
        <div id="page-heading" class="grid-12"> </div>
        <div id="column-left" class="grid-12">
          <div id="content" class="clear-block">
            <div id="content-inner" class="clear-block">
              <div id="front-top-left" class="grid-4 alpha">
                <div id="front-top-left-inner">
                  <div class="block-content">
                    <h2>
                      <p>Use Drupal to build everything from person
                        imagine.</p>
                      <p>Drupal is free, flexible, robust and const
                    <p class="front-get-started">
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    <div id="front-top-middle" class="grid-4">
    <div id="front-top-right" class="grid-4 omega">
    <div id="front-middle" class="grid-12 alpha omega">
    <div id="front-bottom-left" class="grid-6 alpha">
    <div id="front-bottom-right" class="grid-6 omega">
```

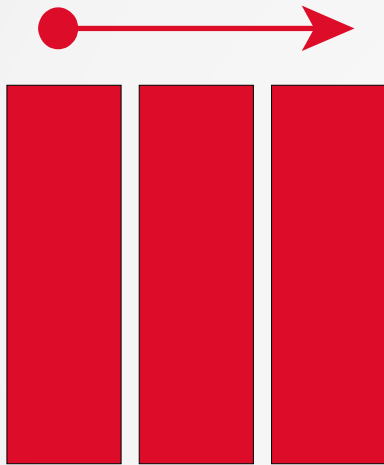
# Clean Markup – Better Ways

```
▶ <header role="banner">[...]</header>
▼ <main id="main" role="main">
  ▼ <section class="o-hero page-home">
    ▼ <div class="o-hero__content o-content-container o-two-columns v--inverse">
      ▼ <header class="o-hero__header">
        <h1 class="o-hero__header__headline">Where Work Happens</h1>
        ▶ <p class="o-hero__header__copy u-margin-left--flush">[...]</p>
        ▶ <div class="o-hero__header__cta c-cta c-cta--signup">[...]</div>
      </header>
      ▶ <div class="o-hero__illustration bg-home">[...]</div>
    </div>
  </section>
  ▶ <section class="o-section v--lighter-gray">[...]</section>
  ▶ <section class="o-section c-cta c-cta--signup u-hide-on-print">[...]</section>
</main>
```



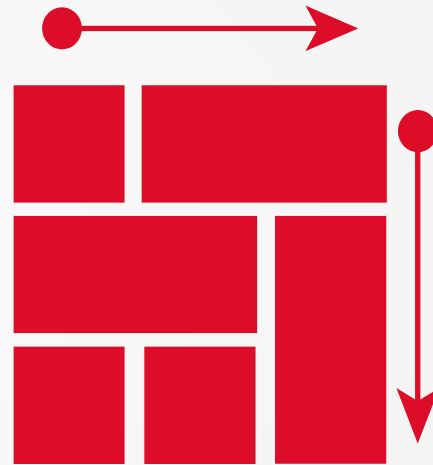
# The Grid

- Can do 2D!



Flexbox

**ONE DIMENSION**



CSS Grids

**TWO DIMENSIONS**

## The Grid – Consequences

You Don't Need To Mutilate Your Markup For Layouting Purposes Anymore. This Is Great!

# The Grid – Basics

- Affects direct descendants
- Can have relative dimensions
- Grids inside of grids inside of grids...
- Define columns, rows, and areas

# The Grid – How To Use

- Define parent as `display: grid;`

- Define columns with

```
grid-template-columns: foo(px|fr|%|em);
```

- Define rows with

```
grid-template-rows: bar(px|fr|%|em);
```

- Define Areas with

```
grid-template-area: "foo foo bar"
```

```
"foo foo bar"
```

```
"baz baz baz"
```

# The Grid – How To Use

- Assign children with

```
grid-row: foo/bar;
```

```
grid-column: foo/bar;
```

```
grid-area: baz;
```

# The Grid – Examples

- (Browser Session)

# Practice – Back to Bruker.com

- Go to [bruker.com](http://bruker.com) and replicate this area:



Bruker Introduces Additional Key Products for Diagnosis and Susceptibility Testing of Invasive Fungal Diseases (IFD)

[News](#)  
Oct 06, 2017

Bruker Introduces Software Package for Layer Analysis with XTrace Micro-XRF on Electron Microscopes

[Read full news article](#)  
Aug 07, 2017



Bruker Announces the New D8 DISCOVER Plus X-ray Diffraction (XRD) System with TXS-HE and ATLAS Goniometer



SCOUT - Automated sample changing system

[Watch Bruker Webinars - Live and On-Demand](#)

Bruker Launches the timsTOF™ Pro Mass Spectrometer to Enable the Revolutionary PASEF

# Extra Resources

Awesome videos explaining the why and how of the grid:

- <https://www.youtube.com/watch?v=7kVeCqQCxIk>
- <https://www.youtube.com/watch?v=jV8B24rSN5o>



# Sources

- Kittens:
  - <https://imgur.com/gallery/WBbKk>
- mor10 talk:
  - <https://www.youtube.com/watch?v=7kVeCqQCxIk>
- Traversy:
  - <https://www.youtube.com/watch?v=jV8B24rSN5o>
- Bruker:
  - <https://www.bruker.com/>
- MDN on Grid:
  - [https://developer.mozilla.org/de/docs/Web/CSS/CSS\\_Grid\\_Layout](https://developer.mozilla.org/de/docs/Web/CSS/CSS_Grid_Layout)
- Div Soup Example 1:
  - <https://perishablepress.com/power-of-html5-css3/>
- Div Soup Example 2:
  - <https://drupal.stackexchange.com/questions/2917/can-you-do-anything-about-the-div-soup>
- Grid vs Flex:
  - <http://maddesigns.de/wp-content/uploads/2017/03/grid-vs-flexbox.svg>