COMP 307
Principles of Web Development

Contents

Elements
Environments

MCGILL UNIVERSITY

COMP 307
Principles of Web Development

Lecture 10

Unit 3 – Frontend Design

Dynamic Website (Part A)

COMP 307
Principles of Web Development

Class Outline

- Elements of responsive design
- Responsive vs Liquid
- DIY Methods

COMP 307
Principles of Web Development

Readings

- Full Stack Developer
  - Chapters 4, 6 and 8
- Internet Resources
  - https://www.w3schools.com/html/html_responsive.asp
  - https://www.toptal.com/designers/responsive/responsive-design-best-practices

COMP 307
Principles
of Web
Development

What are Dynamic Websites?

- Two Forms
  - <u>Responsive Design</u>
    How should the layout of the website change for different types of screens or for windows that vary in size?
    (Part A)
  - <u>Generated Content Design</u>
    Software the automatically creates webpages programmatically.
    (Part B)

This lecture focuses on Responsive Design.

COMP 307
Principles of Web Development

Elements of responsive design

Dynamic Website (Part A)

COMP 307
Principles of Web Development

What is Responsive Design?

- Using HTML, CSS and JS to automatically resize and reformat a website to look good on any device and window size
  - PC
  - Mobile
  - Tablet
  - Other specialty device

This is different from "interactive web pages", which respond to events, like mouseover.

COMP 307
Principles of Web Development

What is Responsive Design?

- Responsive Design comes in two forms:
  - Responsive Design Proper
    Using the concept of a **viewport** to classify different screen sizes and attach each classification to a custom layout. **Benefits**: Exactly what you want. **Drawbacks**: More coding.
  - Liquid Design
    Using the **natural** reorganization flow of the browser (or library) to automatically cascade the webpage elements following a strategy. **Benefits**: Very little coding. **Drawbacks**: May not result in what I wanted.

Liquid Layout

COMP 307
Principles of Web Development

Auto Scaling

Using percentages in CSS:

```
<style>
       body {
           font-family: sans-serif; line-height: 1.5; padding: 0 16px;
       }
       article {
           width: 66%;
           float: left;
       }
       aside {
           width: 33%;
           float: right;
       }
</style>
```

When in a row, important to sum to 100%

COMP 307
Principles of Web Development

Using percentages

In HTML the % value permits the tag to auto adjust its size:

- `<img src="img_girl.jpg" style="width:100%;">`
  - Infinite scaling in both width and height based on viewport limits
- `<img src="img_girl.jpg" style="max-width:100%; height:auto;“>`
  - Height maxes out at image’s pixel height

Auto Scaling

See Example 1 & 2 Liquid

Let’s break this down

COMP 307
Principles
of Web
Development

Basics of Responsive

Define the “viewport”

Replaces browser default behavior with device specifications:

```
<meta name     = "viewport"
      content = "width=device-width, initial-scale = 1.0">
```

- width = readjusts max width to be device screen resolution
- initial-scale = zoom

COMP 307 Principles of Web Development

“vw” = Viewport Width

Define sizes based on the viewport:

```
<h1 style="font-size:10vw">Hello World</h1>
```

Automatically scales

1vw = 1% viewport width

COMP 307 Principles of Web Development

The Media Query

(Entire webpage)

Define multiple style sheets depending on the screen size:

```
<style>
  .left, .right {
     float: left;
     width: 20%; /* The width is 20%, by default */
  }

  .main {
     float: left;
     width: 60%; /* The width is 60%, by default */
  }

  /* Use a media query to add a breakpoint at 800px: */
  @media screen and (max-width: 800px) {
     .left, .main, .right {
        width: 100%;
      /* The width is 100%, when the viewport is 800px or smaller */
     }
  }
</style>
```

default

COMP 307
Principles of Web Development

<picture> tag media control

(Micro Control)

Permits you to define image sizes for different screen resolutions:

```
<html><body>
  <picture>

    <source media="(min-width:800px)" srcset="img_pink_flowers.jpg">

    <source media="(min-width:600px)" srcset="img_white_flowers.jpg">

    <img src="img_orange_flowers.jpg" alt="Flowers" style="width:auto;">

  </picture>
</body></html>
```

- Organge flowers below 600px
- White flowers below 800px
- Pink flower above 800px
- Can specify max-width as well

COMP 307
Principles of Web Development

<video> tag media control

(Micro Control)

Permits you to define video sizes for different screen resolutions:

```
<html><head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
  video {
    max-width: 100%; // compare with width: 100%;
    height: auto;
  }
</style>
</head>
<body>
  <video width="400" controls>
    <source src="mov_bbb.mp4" type="video/mp4">
    <source src="mov_bbb.ogg" type="video/ogg">
    Your browser does not support HTML5 video.
  </video>
</body></html>
```

https://developer.mozilla.org/en-US/docs/Web/HTML/Element/video

COMP 307
Principles of Web Development

Examples

https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_media_query3

https://www.w3schools.com/howto/tryit.asp?filename=tryhow_css_sidebar_responsive

Let’s break this down

Other useful examples

- https://www.w3schools.com/css/tryit.asp?filename=tryresponsive_mediaquery_breakpoints
  Exploring Media Query Sizes
- https://www.w3schools.com/css/tryit.asp?filename=trycss_mediaqueries_hide
  Hiding elements on smaller screens
- https://www.w3schools.com/css/tryit.asp?filename=trycss_mediaqueries_fontsize
  Changing the font size on smaller screens

JS Nav Bar Example

https://www.w3schools.com/howto/tryit.asp?filename=tryhow_js_topnav

Let's break this down

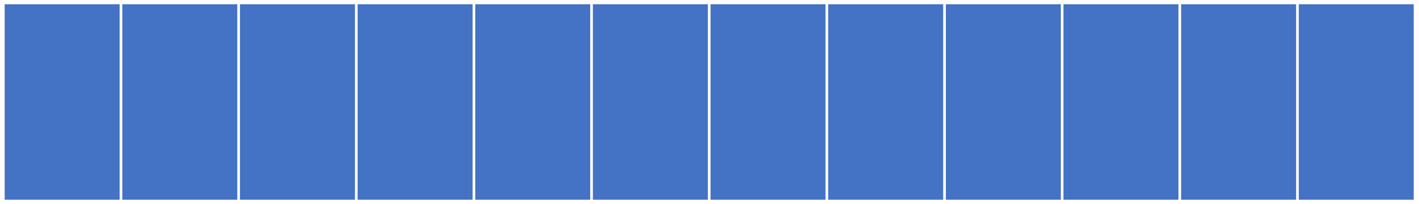
COMP 307
Principles
of Web
Development

CSS Grid View

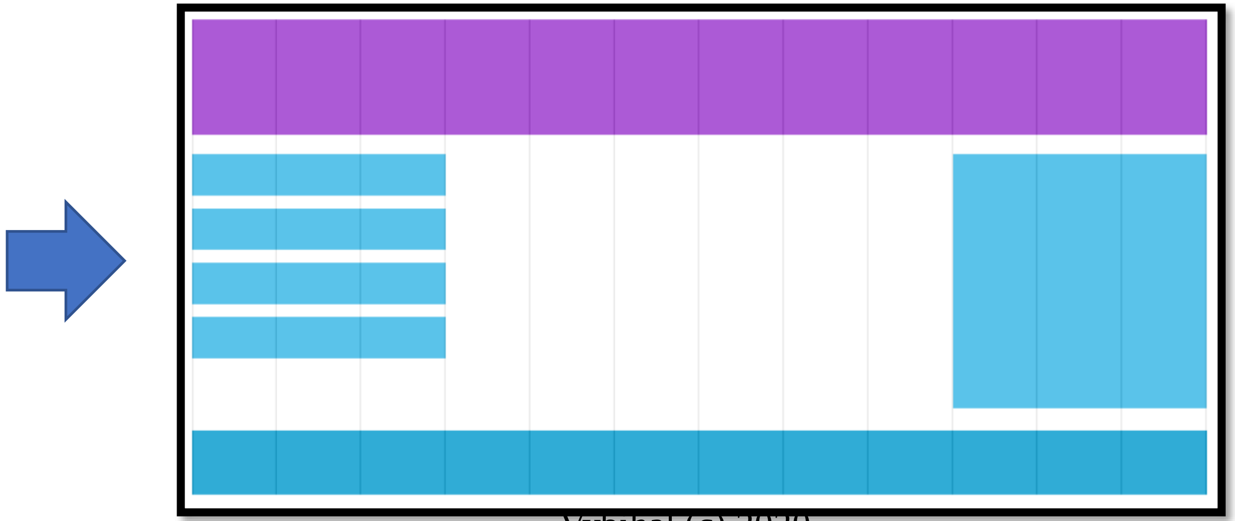
Dynamic Website (Part A)

What is Grid View

- A technique of laying out your webpage in a way that is flexible in relation to screen sizes
  - Design your page using the 12-column technique



Notice there are 12 columns in this page.



COMP 307
Principles of Web Development

Steps to create a Grid View

1. Make sure all measurements included

```
* { box-sizing: border-box; }
```

2. Define the 12 columns (100% / 12 = ?)

Classes:

```
.col-1 {width: 8.33%;}
.col-2 {width: 16.66%;}
.col-3 {width: 25%;}
.col-4 {width: 33.33%;}
.col-5 {width: 41.66%;}
.col-6 {width: 50%;}
.col-7 {width: 58.33%;}
.col-8 {width: 66.66%;}
.col-9 {width: 75%;}
.col-10 {width: 83.33%;}
.col-11 {width: 91.66%;}
.col-12 {width: 100%;}
```

Wrap each row in a <div class="">

3. The 12 columns need to float freely

```
[class*="col-"] {float: left; padding: 15px; border: 1px solid red;}
```

4. Since floating left can overlap…

```
.row::after {content: ""; clear: both; display: table;}
```

Example

https://www.w3schools.com/css/tryit.asp?filename=tryresponsive_col-s

Let’s break this down

COMP 307
Principles of Web Development

Question

Using Grid View, create a webpage with:

- A header
- 4 columns for desktop
- 2 columns for tablet
- 1 column for phone
- A footer

COMP 307
Principles of Web Development

Prepare for Next Class

- Assignments
  - Mini 5 given
  - Mini 4 due tonight
- Lab B
  - Lab B out Thursday, but TA will cover it after the break
- Do on your own
  - Try to build a simple page with CSS Grid from scratch