

HTML5 and CSS3

ICT 580

Ali Khalil, Ph.D.

Computer Science / Software Engineering
University of Calgary - Continuing Education



Lesson 7

Responsive Design Part 2: Mobile First

Lesson Outcomes



By the end of this lesson, you will be able to:

- Understand that to optimize the performance of a website on smaller devices, the web site should be styled for mobile first and then desktop second
- Create a responsive image
- Create a responsive video
- Identify how to use the viewport metatag and the vw and vh units of measurement to create responsive text and margins



Review Question

What does Responsive Design mean?



Review Question

How would you create an alternate style for a smaller screen?

Style for mobile first



 To help load times for smaller screens you should design for the mobile device first and use a media query for the larger desktop screens

Why design for mobile first



- The browser will render CSS from top to bottom
- If mobile styles are listed first, they will be rendered before all others
- If the desktop styles are rendered as a media query, they won't be rendered at all for mobile devices



Mobile First Structure

```
grid-header {
      background-image: url("../img/beach-
      165213 banner-phone.jpg");
      background-repeat: no-repeat;
      background-position: center;
      grid-area: header;
@media only screen and (min-width: 1200px) {
      grid-header {
             background-image: url("../img/beach-
             165213 banner-desktop.jpg");
```



Media Queries -1200px and larger

- Specify images that are appropriate for that screen size
- Manage spacing/measurements for that specific size
- Alter the layout of tables so they span rather than stack many columns
- Only add selectors and properties that must respond

```
@media only screen and (min-width: 1200px) { Applies the styles only if the query in brackets is true.

p {
    Nested declaration font-size: 12px;
    }
}
```



Create a grid with the display property

- Grid is an alternative layout to positioning elements manually with position, overflow or float
- Defining the display property as a grid tells the browser to treat the div container as a grid

```
.container {
  display: grid;
}
```



Grid – grid-template-areas

 Defining the grid area using names can make it easier to easily see the pattern of your grid by laying out the pattern of the grid using named values



Grid – grid-area

- The grid-area defines what you want your grid item to be named
- Using these names, you can create a pattern for your div using the grid-template-areas property
 - grid-area: name;



Responsive images – using percent

 By defining the width as a percentage, the image will resize itself based on the size of the container holding it



Responsive images – background-image

- Using the background-image property allows you to:
 - Resize an image to match the available space of a container
 - Add content such as text or buttons inside the container



Responsive images – background properties

- background-image: url("myimage.jpg"); sets the url path to the containers background image
- background-image: linear-gradient(red, yellow, blue); adds a gradient to the container background
- background-image: linear-gradient(to bottom, rgba(0, 0, 0, 0), rgba(0, 0, 0, 0.6)) to bottom specifies the direction the gradient runs and using the rgba color value lets you add transparency to the gradient
- background-position: center; centers the image in the container
- background-repeat: no-repeat; disallows the image from repeating itself
- background-size: cover; resizes the background image to match the dimensions of the container; may cause cropping or stretching



Create a flexbox with the display property

 Defining the display property as a flex tells the browser to treat the div container as a flexbox

```
.container {
  display: flex;
}
```



Setting the Flexbox direction – row

- The flexbox-direction property sets the main axis or how the items will be laid out either horizontally or vertically
 - flex-direction: row; sets the main axis to be a single row of columns
 - flex-direction: row-reverse; sets the main axis to be a single row of columns that runs in the reverse direction



Flex – justify-content

 justify-content: center; – the items are centered inside the container separated by the grid-gap space



Flex – align-content

 align-content: center; – the vertical spacing of items are centered inside the container separated by the grid-gap space





Flex – flex

flex: 100%; – sets grow, shrink, basis properties to 100%



Viewport meta tag

 The viewport is the visible area of a website when browsing on a device or desktop



Viewport – HTML

- Include in the head tag of your webpage
- Sets the width of your webpage to be the width of the device

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



Viewport – CSS

- Use viewport units of measurement to specify width and height
 - vw 1vw = 1% of the viewport width
 (so in a 800 pixel wide viewport, a value of 1vw = 8px)
 - vh 1vh = 1% of the viewport height



Creating responsive video – video selector

- To make a video responsive set the value of the width to be 100% and the height to be auto
- If you do not want the video to get bigger than 100% use max-width: 100% instead



Creating responsive video – YouTube

You can make a youtube video responsive by using the positioning properties

This slide is for reference only!

https://avexdesigns.com/responsive-youtube-embed/





In this class we discussed:

- Designing for mobile first
- Using the viewport
- Making typography responsive
- Making an image responsive
- Making a video responsive



Review Question

Name one way you can define the pattern of a grid?



Questions?