WEEK 7 / LESSON 13:

RESPONSIVE BASICS

Leonie Dunnett

LEARNING OBJECTIVES

- Differentiate between fixed vs responsive layouts
- Apply media queries to achieve a responsive layout
- Apply CSS and JS to web sites to achieve a mobile site

TODAY'S SCHEDULE

- What is Responsive Design?
- Fixed vs Responsive Layout
- EM vs REM Units
- Layout Review `Code Along`
- Building Responsively `Code Along`
- Mobile Nav `Lab`

WHAT IS RESPONSIVE DESIGN?

RESPONSIVE WEB DESIGN

- Web design was once a static size (like print)
- With devices of varying size, that's no longer ideal
- We need to implement a design that:
  - Offers great user experience
  - Is simple to implement
  - Easy to update
  - Makes sense on each device
- Simplicity is key!

FIXED VS RESPONSIVE

FIXED LAYOUT

- Old School
- Relies on a container of fixed width
- Usually 960px or 980px
- Examples:
  - ups.com
  - google.com
  - theage.com.au
  - customink.com

RESPONSIVE LAYOUT

- Different styles for different screen widths
- Uses an Fluid/Elastic layout
  - Fluid - Sized in percentages
  - Elastic - Sized in ems/rems
  - More info via Smashing Mag

RESPONSIVE EXAMPLES

- thedesignfilesopenhouse.com
- lonelyplanet.com/partner
- Generalassemb.ly
- thankyou.co
- foodsense.is
- turo.com

EM VS REM

EM

Size based on the width of the letter "m"

- Same as percentages
- 1em = 100% font-size
- Child size is based on the font size of the parent

```
parent{font-size:16px;}
child{font-size:2em;}
```

Child's font size is 32px

SIZING TYPE WITH EM'S

- Sizing Type with Ems via Adam D Scott
- PX to EM conversion made simple via PXtoEM.com

REM ("ROOT" EM)

While em is relative to the font size of its direct or nearest parent, rem is only relative to the html (root) font-size.

I like to think of it as a reset.

If a style sheet is built in a modular fashion, then rem shouldn't be needed very often, but it can be handy at times.

Jeremy Church, on REM vs EM

FURTHER READING ON EM/REM'S

- Font Size Idea: px at the Root, rem for Components, em for Text Elements
- How we learned to leave default font-size alone and embrace the em

LAYOUT REVIEW

Floats and box model

Sizing for flexible layout

BUILDING RESPONSIVELY

- Media Queries
- Determining Breakpoints
- Mobile Display: Viewport
- Responsive Images

HELLO MEDIA QUERIES!

Media queries are a way to provide conditional CSS based on the type or device, width of device, or orientation.

```
/* Phones  ---------- */
@media screen and (max-width: 767px) {

}

/* Tablet/Desktop  ---------- */
@media screen and (min-width: 768px) {

}

// Standard media queries sizes
Small: up to 768px
Medium: 768-991px
Large: 992px+
```

MEDIA QUERY SYNTAX

```
// Separate multiple clauses with "and"

@media only screen and
(max-width: xPx)
(min-width: xPx)
(max-device-width:xPx)
(min-device-width:xPx)

For iPad
(orientation: portrait)
(orientation: landscape)
```

CSS Media Queries

Media Queries for Standard Devices

MEDIA QUERY EXAMPLE

```
/*float boxes into columns*/

.box{
    float:left;
}

@media only screen and (max-width:768px){
    /* stack floated boxes*/
    .box{
        float:none;
    }
}
```

If I put the media query before `.box{float:left;}` will this work as expected?

DETERMINING BREAKPOINTS

- Breakpoints determined by your design, not device (simplifies the process and you don't have to worry about catering for every new device)
- A simple, well thought out design will save you time and headaches in the long run
- Usability is key

MOBILE FIRST: DESIGN THINKING ✓

In terms of design thinking, it's good practice to think about why a user is coming to your site in simple form, ie:

- Determine your Minimum Value Product
- Sketch out what it would look like on mobile
- Then think about larger devices and add any extra content that would be useful/add value

MOBILE FIRST: BACKGROUND

- Jun 2011 - Ethan Marcotte's Responsive Web Design
  - Taking your existing desktop site/design and adding media queries to style for mobile
- Oct 2011 - Luke Wroblewski's Mobile First
  - Build site for mobile and then add media queries for desktop/larger device styles
  - Flipped the former approach!

Ethan's book is now in it's 2nd edition...

See: Responsive vs Mobile First Designs

MOBILE FIRST: MEDIA QUERIES ✗

- Can be helpful when starting out, but best practice:
  - Common styles in base style sheet (global styles)
  - Add media queries for mobile and larger device styles
  - Nothing has to override styles it doesn't need

MOBILE DISPLAY: VIEWPORT

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

WHY NECESSARY?

- Mobile browser assumption of fixed layout of 980px
- Optional: `user-scalable=no`

RESPONSIVE IMAGES

Set width and height in HTML (no pixels):

```
<img src="..." width="280" height="280">
```

In CSS:

```
max-width:100%;*
```

```
height:auto;
```

*Image won't scale larger actual size

More on flexible media via Shay Howe

LEGACY IE BROWSER SUPPORT

- Internet Explorer 8 and down don't understand **HTML5 elements** or **Media Queries**
- We overcome this by including:
  - HTML5 Polyfills
    - HTML5Shiv
    - Modenizr (HTML5Shiv built in)
    - respond.js (media query support, server only)
  - IE conditional statements in our HTML

MOBILE NAV

Hamburger example

HOMEWORK

Review: Slides/Resouces

Research: Media query transitions, Grid layouts

Code: Assignment 7 - Responsive Design of Relaxr Landing Page and Blog

RESOURCES

Responsive Web Design via Shay Howe