Responsive Web Design

Girl Develop It! RDU

What you need

- Slides: http://bit.ly/gdi-rwd-slides
- Browser: Google Chrome or Mozilla Firefox
- Text editor: Sublime Text, Text Wrangler, Atom, etc.
- Class files: https://github.com/amwelles/gdi-rwd

Welcome!

Girl Develop It is here to provide affordable and accessible programs to learn software through mentorship and hands-on instruction.

Some “rules”

- We are here for you!
- Every question is important
- Help each other
- Have fun!

Class 1

Introduction to RWD

Autumn Welles

- Tinkered with websites before I could drive
- In the business since 2007
- Drank the RWD Koolaid in 2012
- Involved with GDI since 2013

@amwelles on Twitter
amwelles@gmail.com

Tell us about yourself

- Who are you?
- What do you do (or want to do)?
- What do you hope to get out of this class?
- What’s your spirit animal?

Tips

- Don’t copy and paste… no matter how tempting!
- Raise your hand for questions and clarification.

What is Responsive Web Design?

And why you should care

Who decided this was a good idea?

A short history

What are the alternatives?

And why RWD is better

Where do we start?

Understanding media queries

Media Queries

```
@media screen and (min-width: 768px) {
    /* window >= 768 pixels wide */
}

@media screen and (max-width: 768px) {
    /* window <= 768 pixels wide */
}
```

https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Media_queries

Metatag

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

https://developer.mozilla.org/en-US/docs/Mozilla/Mobile/Viewport_meta_tag

How do I handle …?

Solutions to some of the most common problems

Images

```
img {
    max-width: 100%;
}
```

Sidebars

```
/* utility class */
.clearfix:after {
    content: "";
    clear: both;
    display: block;
}

.widebar,
.sidebar {
    float: left;
}
```

Sidebar on the right

```
.widebar {
    width: (width/container)%; /* e.g. 740/960 */
    max-width: (width); /* e.g. 740px */
}

.sidebar {
    width: (width/container)%; /* e.g. 220/960 */
    max-width: (width); /* e.g. 220px */
}
```

Sidebar on the left

```
.widebar {
    [...]
    margin-left: (sidebar-width/container)%;
}

.sidebar {
    [...]
    margin-left: -100%;
}
```

Navigation

- Float list items left
- Leave them in a row for narrow columns

Bonus:

- Use a bit of jQuery to make the menu slide up/down

Tables

- Create a wrapper element
- Make your table scrollable (on the x-axis)
- Make it obvious that there’s more to the table
- Remember to remove those styles on larger screens

http://codepen.io/anon/pen/GJRVoZ

Q&A

- What questions do you have?
- Do you need further clarification?
- Have anything you'd like to show the class?
- What do you want to learn next week?

Class 2

Email templates, start to finish

Media query review

```
@media screen and (min-width: X) {
    /* greater than or equal to X */
}

@media screen and (max-width: X) {
    /* less than or equal to X */
}
```

Email templating basics

- 500-600 pixels wide
- Design for desktop first (vs. mobile first for websites)
- Keep your important info up top
- Single column layouts are easiest
- Should work well on most phones/browsers/clients

https://www.campaignmonitor.com/guides/mobile/design/

Let's develop it! (pt. 1 of 3)

- Open up class-2-email-mockup.jpg
- With pen (or pencil) and paper, let's block out our elements for desktop view
- Write the markup, but don't worry about adding class names
- Try to ignore the content/design elements and think about your blocks (divs, headings, paragraphs, lists, etc.)

Mockup from http://freepsdfiles.net/email-marketing-templates/newsletter-psd-template-set-1

Let's develop it! (pt. 2 of 3)

- Now let’s write some CSS to design it up!
- We won’t worry about making it look exactly like the image -- just get it as close as you can
- Add class names as you need them; try to reuse when possible!

Let's develop it! (pt. 3 of 3)

- Now that our desktop version is looking good, let’s style it for mobile
- What should move around?
- What should be hidden?
- How can we make it more usable on mobile?

Exercises

- Pick out 5 templates from Freebies Bug
- Use pen & paper to:
  - block out the elements on the page
  - use those same blocks to do a mobile wireframe
- Take one of the templates and create it using HTML/CSS (mobile first!)

Q&A

- What questions do you have?
- Do you need further clarification?
- Have anything you’d like to show the class?
- What do you want to learn next week?

Class 3

Frameworks and flexbox

Why bother with frameworks?

- Quicker to set up
- Many elements already defined (navigation, grid, etc.)
- Already following best practices

Skeleton

- Written in CSS
- Super simple
- Less styles to start with
- Easier to override

http://getskeleton.com/

Foundation

- Written in Sass
- Customizable & modular
- Personal favorite ♥

http://foundation.zurb.com/

Bootstrap

- Written in Less
- Customizable & modular
- Extremely popular (which means more resources)

http://getbootstrap.com/

Let's develop it!

1. Download Foundation or Bootstrap as CSS.
2. Add a second sidebar on the left. Hint: You'll want to add the markup just above the original sidebar.
3. Take our existing markup and change the classes to fit the new framework.
4. Extra credit: Make your header navigation responsive work on mobile. Hint: You'll need to call some JavaScript.

Flexbox

- Makes it easy to reorder content
- No more sidebar/widebar background woes
- IE10+, Safari, Chrome, & Firefox

https://css-tricks.com/snippets/css/a-guide-to-flexbox/

http://caniuse.com/#feat=flexbox

Markup structure

```
<div class="container">
    <div class="item widebar">Widebar</div>
    <div class="item sidebar">Sidebar<br>with a second line</div>
</div>
```

Flexbox CSS

```
.container {
     display: flex;
     max-width: 960px;
     margin: auto;
}

.widebar {
     flex: 2; /* 2 out of 3 */
}

.sidebar {
     flex: 1; /* 1 out of 3 */
}
```

Margins? Padding? No problem!

```
.item {
    padding: 2em;
}

.widebar {
    background: #eee;
    margin-right: 3em;
}

.sidebar {
    background: #ccc;
}
```

Tables (headers on side)

- Make our headers span the full width
- Use flexbox to line up the table data cells

http://codepen.io/anon/pen/YXzmWP

Q&A

- What questions do you have?
- Do you need further clarification?
- Have anything you'd like to show the class?

Class 4

Flexbox review and converting an old site

Flexbox review

- Used instead of floated divs
- Can reorder content with ease
- Makes equal-height columns a breeze
- Finally, an easy way to vertically-align content!

Let’s develop it! (choose your adventure)

Project 1

- Make a simple 6-column grid layout using flexbox.
- Make sure the grid never gets larger than 960 pixels wide.
- One row should have 6 items (1 column wide).
- Next row should have 3 items (2 columns wide, 3 columns wide, 1 column wide).
- Last row should have 2 items (4 columns wide, 2 columns wide).

Project 2

- Create an old-school splash page
- Center your text (and images) in the middle of the page using flexbox
- Make sure it looks good on mobile, too!

What about my existing site?

Updating a desktop site for mobile

Making a fixed-width site responsive

Basic process

- Resize your browser until it breaks
- Create a breakpoint there
- Style away!

Tips

- Use "max-width" when applicable
- Remember to use percentages
- Make sure your images don't break out of their containers

Let’s develop it!

- Open class-4.html & class-4.css
- Start resizing the browser, figuring out where it breaks, and adding in max-width media queries

Q&A

- What questions do you have?
- Do you need further clarification?
- Have anything you'd like to show the class?
- What's next? JavaScript, jQuery, Sass, WordPress, PHP, Intermediate HTML & CSS