What is Responsive Design?

What is Responsive Web Design?

- It is designing your sites with multiple screen sizes/resolutions in mind.

- Sites should “work” under any platform, any browser size, any orientation. The user should have the power.

Adapting to user needs and device capabilities

- A small screen should NOT mean less content.
- People are doing more on their phones than ever before
    - watching videos, filling out applications, coding, ….
- Never assume the user won’t need access to a functionality

“Responsive” options

Responsive Web Design (RWD) – fluid measurements, flexible grids, and varying CSS rules

Adaptive Design (dynamic serving) – returns one of multiple versions of a page based on the type of device

Separate Mobile Site (.m)- a separate page URL for the mobile site

RWD

- Is it responsive? If the server is sending back the same code regardless of the device, you are using RWD.
- This can be detected automatically, by looking for meta name = “viewport”

Adaptive Design

- Server returns different code (HTML and CSS) depending on the device requesting the page.
- The same URL is used.
- May get messed up if the wrong device type is detected.

Separate URL

- Separate URLs serve different code to desktop and mobile devices (and perhaps even tablets), and on different URLs.
- You can relate the URLS with a <link> tag and rel="canonical" and rel="alternate" elements.

Why RWD?

- Easier to share your data with a single URL
- Easier for search engines (Google) to index the page
- Fewer files = less maintenance
- Less redirection = lower load time

Fluid Measurements

Static/Relative measurements

- px, mm, cm, in, pt, pc
- %, em, rem, vw, vh, vmax, vmin

Breakpoints

What does “trigger” mean

- People may use the phrase “breakpoints trigger changes”.
- To be honest, most people don’t resize their window.
- So I will use breakpoints to refer to which rules are applied.

Determining the Breakpoints

- Breakpoints should correspond to:
  - devices and/or
  - content

Screen sizes

- iPhone 4 (320px, 480px)
- iPhone 5 (320px, 568px)
- iPhone 6 (375px, 667px)
- iPhone 6+ (414px, 736px)
- Galaxy S3/S4 (320px, 640px)
- Galaxy S5 (360px, 640px)

Media Queries

Media Queries

- Media queries are a process that allow the style to depend upon the media type

- CSS 2.1 used media types

```
<link rel="stylesheet”…href="style.css" media="screen" />
<link rel="stylesheet”…href="print.css" media="print" />
```

CSS3

- CSS3 increased the capabilities. Style can depend on many features
  - width, height, orientation, resolution, …
- Boolean operators can also be applied to increase power

The two query components

1. A media type
   - screen, print, aural, braille, all, …

2. The actual query of a media feature and width
   - width, height, orientation, resolution, …

screen and (min-device-width: 680px) and (resolution: 163dpi)

Step 1: Grab information

- The meta viewport tag tells mobile browser’s viewport how to behave.

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

- Disallow zooming:

```
<meta name = ‘viewport’ content=‘width=device-width, initial-scale=1, maximum-scale = 1’>
```

Step 2: Fluid layout

- If you use breakpoints, some absolute measurements are not unusual
- percentages vs ems
  - ems are measurement of typography. 1em is width of one letter M in current typeface.
- paddings and margins affected by width, not height

Step 3: Use Media queries

- Fluid layout that is triggered by certain sizes.

screen and (min-device-width: 680px) and (resolution: 163dpi)

```
 p.desc {
          display: block;
          font-size:
    150%; }
    }
@media screen and (min-width:700px){
      p.desc {
                display: inline-block;
               width: 50%; }
   }
  @media screen and (min-width:1200px){
        p.desc {
                width: 35%;}
     }
```

Ordering your rules

```
@media only screen and (min-width: 1024px) {
   body{ background: blue; }
}

@media only screen and (min-width: 780px) {
   body{ background: yellow; }
}

body{ background: green; }
```

RD: Media Queries Part 2

Stop and Code:

RD: Fluid Measurements and Media Queries

Can you change the CSS for single column on small screen?

Concept Check

- Should your media queries be at the top or bottom of the page?
- Trick question - depends upon if you are using max-width or min-width
- Assuming min-width, put the rules on the bottom.

Frameworks

Frameworks

- The term “web framework” can mean many things, it depends upon who you ask.
- Agreement, a framework should make your coding job easier by providing code and structure.

What does “framework” mean here?

- Front-end developers
  - CSS, JavaScript, jQuery
- Back-end developers
  - Routing, resources, security

Popular front-end frameworks

- Bootstrap (2011)
  - Its popularity makes it.... popular
- Foundation by ZURB (2011)
- Semantic UI (2013)
- Pure by Yahoo! (2013)
- UIkit by YOOtheme(2013)

sitepoint.com

What it means for you

- Many people build their sites directly from templates.
- Others use their own code and add functionality.
- Even “pure” coders should know the basics of one popular framework

Bootstrap benefits

- 12-column grid system
  - Helps with spacing issues
  - Built-in responsive design
- Common jQuery functionalities
  - Accordion, Drop-down menus, Carousel
- Familiar “look and feel”
  - Many sites use Bootstrap
  - Makes your forms look “legitimate”

Why you should use it

- Fast development
- Platform Independent
- Responsive by default
- Customizable

Why you shouldn’t use it….

- Doesn’t follow best practices
  - Content and layout are intertwined
  - http://blog.nocturnalmonkey.com/css-frameworks-and-semantics/
- Can be resource-heavy
- The look is somewhat generic
  - This can be good or bad.

What we will cover

- There are two ways to use Bootstrap
    - As a supplement to your style
    - As a theme that you expand upon
- We will talk about some of the basics, but there are still many parts left untouched.
- It is important to test often when using code that isn’t yours.

Bootstrap Navigation

These slides are just for exposure, not for memorization

Navigation Bars

- One of the components that gives Bootstrap its familiar “look and feel” is the navigation options.
- The nav class is combined with other classes to create each style

Making a navigation bar

- Decide what type of links you want:
  - –nav-tabs vs nav-pills
    - **RD:nav-tabs**
    - **RD: nav-pills**
- Decide on layout (horizontal, stack, justified, etc.)
  - nav-stacked, nav-justified

Drop downs

- To add dropdown menus you need to include the Bootstrap js files AND a link to the jQuery.
- Bootstrap example: http://getbootstrap.com/components/#pills-with-dropdowns

RD:dropdowns

navbar class

- **The navbar class** serves as a navigation header for your application or site.

- Positioning includes:
  - navbar-static-top
  - navbar-fixed-top
  - navbar-fixed-bottom

RD:navbar

Collapsible Navigation

RD: Advanced Navigation

Accessibility

- Using a “nav” class does not convey semantics.
- Use the <nav> tag or ARIA attribute role=“navigation”.

Responsive Images

Making Your Images Responsive

- So many webpages are pictorial, it only makes sense to talk about responsive images.
- Two approaches:
  - Write your own code
  - Use Bootstrap

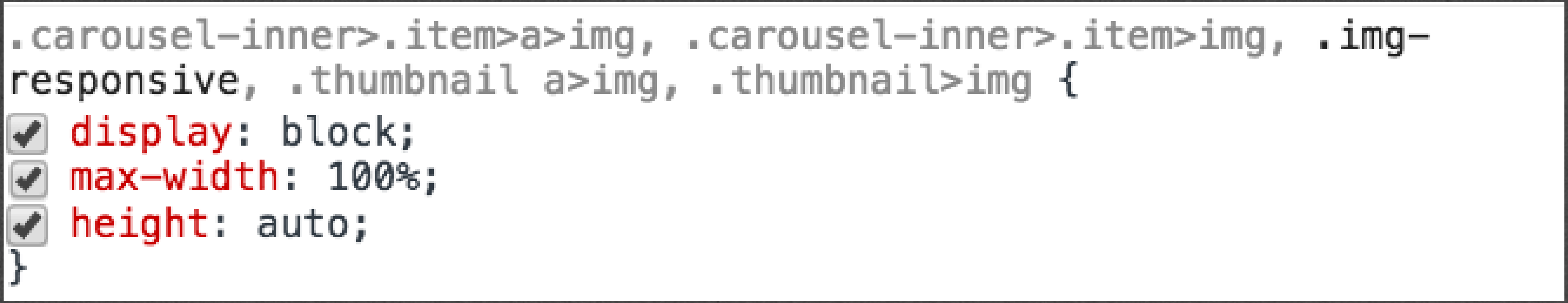
Your Own Code

- In your own CSS you will want to use fluid measurements
- width: 100%
- max-width:750px;
- mid-width:200px;
- Set height to auto

Using Bootstrap

- Bootstrap provides a number of image classes.
    - img-responsive
    - img-rounded
    - img-circle
    - img-thumbnail

img-responsive



```
.carousel-inner>.item>a>img, .carousel-inner>.item>img, .img-
responsive, .thumbnail a>img, .thumbnail>img {
☑ display: block;
☑ max-width: 100%;
☑ height: auto;
}
```

RD: Responsive Images

Gallery

RD: Responsive Image Thumbnails

Getting Started

- How do you actually use Bootstrap?
- You need access to the CSS code and JS code.
- Options:
    - Download copy
    - Use absolute reference

Saving your own copy of Bootstrap

- You can download a copy of Bootstrap at http://getbootstrap.com

- Use the default values or select http://getbootstrap.com/customize/

- Make sure you know where you saved your files!

Using a CDN

- A CDN is a content delivery network
- Provides a way to connect to the Bootstrap code using an absolute reference
  - http://getbootstrap.com/getting-started/

Template

- It is common to start with a template file.

http://getbootstrap.com/getting-started/#examples

- A modified version from getbootstrap.com can be found here:

RD:Bootstrap Template

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1”>

    <title>Bootstrap 101 Template</title>

   <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css"
    integrity="sha384-
1q8mTJOASx8j1Au+a5WDVnPi2lkFfwwEAa8hDDdjZlpLegxhjVME1fgjWPGmkzs7"
    crossorigin="anonymous">

     <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"
     integrity="sha384-
0mSbJDEHialfmuBBQP6A4Qrprq5OVfW37PRR3j5ELqxss1yVqOtnepnHVP9aJ7xS"
    crossorigin="anonymous"></script>
  </head>
  <body>
```

How do you know it is working?

- For such a simple file, it may be hard at first to see if it is working.
- To test:
  - Comment out link to css
  - Use Inspect Element
    - This method is definitely preferred, particularly as you start to add your own styles.

Don’t forget…

- To use element inspector.
- To include the JavaScript at the bottom.
- That you can customize bootstrap defaults.
- That you can overwrite bootstrap defaults.

That’s It!

- Even if you don’t understand HTML, CSS, of JavaScript it is possible to create a great site using a Bootstrap template.
- But it is ***soooo*** much better if you have enough knowledge to change things!