Content Reference
Wireframes
“Back in my day ...”

- In the early days of web design, wireframes – sometimes called Schematics – were simple drawings with boxes indicating where page components would go on a page.
- Nowadays, wireframes are often exceedingly detailed.
  - The often contain actual content
  - Many look like the finished web page
  - The layout is already fixed
  - Decisions already made about the final content and placement
Sample Wireframes

Default / Browser

Tablet Portrait

Smartphone

©2012 AcmeWidgets. All Rights Reserved. Support@AcmeWidgets.com
Intricate Wireframes
The Problems with Intricate Designs

- Clients are not going to see the grid placement, they are see what looks like final copy
  - The comments will be about the actual copy text and images not layout
- If you create a Mock Layout they require a lot of create so you want the Mock Layout to be as close to the final layout as possible, otherwise you have a lot of rework
But Wireframe Alone Are not the Solution

- Content reference wireframes are simply old fashioned wireframes
- The body of the wireframe are
  - Minimal layouts
  - Referencing content rather than depicting it
- But wireframes built in graphics tools are static depictions,
  - They need to evolve and help the client perception of what they want grow with their knowledge
- We should create Low-fi Web Wireframes that we can evolve
Sample Problem

- Consider a website for a Book
Base Markup

- See `wireframe-0.html`
- This is a simple plain HTML skeleton
Base Markup

- Next look at the content inventory and prioritize the content
  1. Book title
  2. Synopsis
  3. Purchase
  4. Resources
  5. Errata

- Seems simple and content appears to be in the correct order
Base Markup

- See wireframe-01.html
- The `id` attribute is optional, but as the wireframe evolves into a more detailed prototype, it can make reading your own code easier
- If you open wireframe-01.html you will not see anything since we have not added any textual content
Base Markup

- **See** wireframe-02.html
- **If you open** wireframe-02.html you will not see anything since we have not added any textual content
Base Markup

- Now we need to add some wireframes styles
- To keep things clean, for now we will add a special class **attribute to the body element**
Setting up base Styles

- **See** base-03.css **used in** wireframe-03.html
  - We give the **body** a background colour and font family
  - For wireframe sections we set a colour and a border for the sections
  - We add a link to the stylesheet
Setting up base styles

- We can use the style files to improve the aesthetics
- See `base-04.css` used in `wireframe-04.html`
- The process of setting up the base styles should take no more than a few minutes
Adjusting the wireframe to be Mobile Friendly

- Resizing the windows shows that the current wireframe is not very mobile friendly
- The height of each block is determined by the size of the heading within it
- Some changes
  - Add navigation links to menu
  - On a small screen it is common to provide a link at the top to navigate to the bottom
- See `base-05.css` used in `wireframe-05.html`
- This still gives us a linear navigation
Let Look at some Larger Screens

- Imagine for the book site example that we want a layout that will work well on most smartphones and tablets, and most desktops
  - We are not concerned with specifics yet
- We are not going to start looking up the sizes of specific devices
  - We want the content to dictate where the layout changes
A Tale of Two Viewports

Useful web paper to read

- A Tale of Two Viewports — Part One
- A Tale of Two Viewports — Part Two
A Tale of Two Viewports

- When considering a webpage you have to consider two distinct viewports
  - Physical or Visual Viewport – the screen size
  - The Layout Viewport – the size of the layout
A Tale of Two Viewports

Physical Viewport

Layout Viewport

virtual viewport = layout viewport
A Tale of Two Viewports

- For Responsive Design we are interested in the width of the screen on smaller devices, and the window on devices and in browsers that supports them.
- We need to tell the browser that when we say “min-width”, the “width” of the page should be the width of the device (or the window).
- See the meta element in wireframe-06.html for viewport
  - Sets min-width
  - Sets the zoom to 100%
Support for multiple layouts

- We can tailor the formatting based on device width
- See wireframe-06.html
Exercise

- Read 2\textsuperscript{nd} half of Chapter 2 – Fluid Layouts
- The code is available as a Netbean project in the Supplemental Materials Section
- Pay Attention to
  - How they used the Grid method to design and create a web page
  - Combined fixed an fluid components
  - Created a Fluid page