CTD Intro Week 12

Responsive Design

Design Principles

- Create a plan which works across all target devices
  - Small phone, large phone, tablet, desktop
    - Select width breakpoints
  - Is print included?
- Document the plan in your code
  - HTML comments <!-- …-->
  - CSS comments /* …*/
- Examples
  - Mobile first
    - Very common, popular
  - Desktop first
    - Is phone use uncommon, impractical?
- Typical fundamental limits
  - 320px wide for the smallest device
  - Handle potential very large widths by using max-width and centering

Responsive Design

- Viewport meta tag in head
  - <meta name="viewport" content="width=device-width, initial-scale=1">
  - Avoids simulated larger screen
- Avoid fixed widths
  - Some exceptions significantly smaller than screen dimensions
- Natural responsiveness
  - Single column
  - Flexbox flex-wrap: wrap
    - Multiple lines if needed
  - Use flex-direction: column rather than row
  - grid minmax(250px, 1fr) – minimum of 250 px, or divide any extra space evenly
  - Grid auto-fit - create additional columns with extra space
    - `grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));`
- Mobile first
  - Start with single column
  - Add columns for wider displays

CSS Media Queries

- Adjusting css based on media details
- Example
  - @media only screen and (min-width: 80rem) {<any styling override> }
- Screen vs print
- Relative units vs pixels
  - 80rem – 80 root element font widths
- min-width – covers any greater width
- max-width – covers any lower width
- Breakpoint
  - Width range which uses a particular layout scheme
  - E.g. < 500px, 500px-1000px, > 1000px

CSS Grid

- 2D Layouts
- Styling applied to a grid container, often a <div>
  - Items in the grid are direct descendants of the container
  - display: grid;
- Grid templates – applied to the container
  - grid-template-rows: <height>;
  - grid-template-columns: <width>;
  - grid-template: <rows> / <columns>;
- Gaps – spacing between grid items
  - column-gap: <space-between-columns>;
  - row-gap: <space-between-rows>;
  - gap: <row-gap> <column-gap>;
- Implicit grid for extra items
  - grid-auto-rows: <height-for-implicit-rows>;
  - grid-auto-columns: <width-for-implicit-columns>;
- auto-fit
  - grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));

Debugging

- Browser developer tools
  - Right-click->inspect
  - Manual width adjustment
    - Interactive or width/height entry
  - Specific devices (Dimensions menu)
  - Media query breakpoints displayed at the top
- Some common problems
  - min-width vs max-width
  - Styling which forces a minimum absolute width
  - Specifying the entire screen width for an element
  - Handling images
    - object-fit (fill, contain, cover, scale-down…)
    - Different images at breakpoints

Questions?

Demo