

Responsive and Adaptive Designs

- Confusing Terminology
- "**Responsive**" can be an umbrella term for both
 - Or treated as distinct concept
- Both about effective pages at different sizes
- Neither denies nor requires **gutters**
 - Some designs don't go full width
 - Essentially a max width on content
 - Has empty/background "gutters" on sides
 - <https://examplecat.com/> over 700px wide

Responsive Design

- A page that uses available size
- Content wraps as needed
- HTML is naturally responsive!
 - But we can mess it up with styling
 - Important to regularly confirm you didn't

Adaptive Design

- Changing layout at certain **breakpoints**
 - Layout = Arrangement of elements
 - Ex: Change number of blocks in a row
 - Ex: Change row of items to column
 - Ex: Menu becomes "Hamburger Menu"
 - See MDN as example
- Not automatic
 - Dev has to code the rules to apply this
- Allows for good use of space
 - Not just wrapped use of space

How?

TL;DR: Usually **Media queries**

- CSS operates off of 2+ options for layout
 - Based on **viewport** dimensions
 - Can reorder elements in layout
 - Can collapse menus/sections
- May target:
 - Mobile/Desktop
 - Mobile/tablet/desktop
 - Landscape/portrait(s)/desktop/widescreen
 - Or just where content fits better

Grey Area (Gray Area?)

Is this Responsive or Adaptive?

- Media queries to how many items shown
- Flexbox/Grid to vary items/row w/o media query

Read multiple search results: "Responsive vs Adaptive"

- Will get different answers
- Doesn't really matter
 - What we call it doesn't impact users
- I'll call these "Adaptive"

Simple example

Page:

- header
- menu
- content
- footer

Desktop size:

- Vertical to left of main content

Mobile size:

- Horizontal menu across top of main content

Sample HTML

```
<header class="header"> <h1>Kitten Co.</h1> </header>
<nav class="primary-nav">
  <ul class="menu">
    <li><a href="/">Play</a></li>
    <li><a href="/">Eat</a></li>
    <li><a href="/">Sleep</a></li>
    <li><a href="/">Blep</a></li>
  </ul>
</nav>
<main class="main">
  <p>
    Ipsum nemo iste vel lorem officiis dicta!
    Tenetur amet dolor ratione quas excepturi. Libero
    recusandae excepturi perspiciatis
  </p>
  <p>Lorem ipsum repellendus magni soluta libero.
    Outtaus spaceum soa incorrectus indentingum</p>
</main>
<footer class="footer">
  Gifts of catnip are accepted
</footer>
```

Media Queries

```
@media (min-width: 1000px) { /* at least this width */
    /* CSS Rules here */
}

@media (max-width: 1000px) { /* below this width */
}

@media (min-width: 600px) and (max-width: 1000px) {
    /* a middle size */
}

@media (600px <= width <= 1000px ) {
    /* Newer range syntax (confirm age on caniuse.com) */
}

@media (width > 760px) {
    /* Also newer range syntax – as of May 2023 for Safari*/
}
```


But how do we test mobile?

Not perfect, but helpful

DevTools has many preset mobile options

- Why doesn't the mobile work?!
- Desktop resizing does!

Mobile Cheats

- Layout viewport
 - Media query checks!
- Visual viewport
 - Allows for zoom and pan

Most mobile devices render as if a larger size

- then zoom out

Override with

```
<meta name="viewport" content="width=device-width">
```

Don't stay in mobile mode!

I'm not sure why many students do this

- Use mobile mode to test mobile sizes
- Use normal (desktop) mode normally
- Turn the mode on and off
 - Not just "on" once and forever

Summary - Responsive

- HTML is naturally **responsive**
 - Content uses available space
 - Height as needed
 - Wraps as needed
- Devs can mess it up with fixed sizes
 - Including text size assumptions
- Remember your specific screen isn't "normal"
 - There is no "normal" screen size
- Remember users may have different text sizes

Summary - Adaptive

- **Adaptive** sites change **layout**
 - At different viewport **breakpoints**
 - Often shift horizontal layout to vertical
 - Often collapse menus/sections
- Implement with **media queries**
 - Often the min-width/max-width

Summary - Setting Meta viewport

- Mobile often render with different viewport size
 - Then visually zooms in
- Can cause disconnect with media queries
- Override with:

```
<meta name="viewport" content="width=device-width">
```

- Easy to forget!
- Esp. if you don't use mobile view in DevTools
- Remember to do on assignments!