

Web Design

Responsive Design

Considering Screen Sizes

- Not all users will view your page with the same screen size
 - some may have large monitors, while others have small phones
- We need to design our sites to be able to handle being viewed at any reasonable width and still look good

Using Relative Units

- One way to account for varying screen sizes is by using relative units
- Content will dynamically resize relative to the window/container size
- Units include: %, vw, vh

Media Queries

- Conditionally apply CSS styles depending on the device type and features
- Media-type specifies how the user is getting your content. Includes: all, print, screen, speech
- Media features let us specify feature requirements for the style. Can require a certain screen size or that hovering is available (for example)
- Can use the media-type and media features (along with logical operators) to set the conditional rules

```
.element {  
  width: 300px;  
  background-color: red;  
}  
  
@media all and (min-width: 600px) {  
  .element {  
    width: 400px;  
  }  
}  
  
@media print {  
  .element {  
    background-color: transparent;  
  }  
}  
  
@media all and (aspect-ratio: 11/6) {  
  .element {  
    height: 10px;  
  }  
}
```

Mobile-First Design

- We should design our content for mobile by default
- Then use media queries to add styling for larger screen sizes

```
/* Mobile-First: */

.element {
  width: 100%;
}

@media all and (min-width: 700px) {
  .element {
    width: 50%;
  }
}

@media all and (min-width: 1000px) {
  .element {
    width: 33%;
  }
}
```

```
/* Not Mobile-First: */

.element {
  width: 33%;
}

@media all and (max-width: 700px) {
  .element {
    width: 50%;
  }
}

@media all and (max-width: 1000px) {
  .element {
    width: 100%;
  }
}
```