1. Responsive website development - media queries

1.1 Responsive website: what are media queries?

- **What is the problem with non-responsive websites?**
  - As people are using websites in different devices, a single design might not look best in all device sizes. Therefore, the ideal solution is to have different designs for different device sizes. As an example, you can look at the Puppy Lover's Page you developed. When you open your Puppy Lover's Page on the browser and resize your screen, you can see that your page does not display the same design as you change the size of your screen.
- **Responsive website design: definition**
  - It is the process of designing websites that will respond to the various device they are viewed on. Meaning, if a website is responsive, page elements reshuffle/adjust as the screen size grows or shrinks. It keeps images from being larger than the screen width and prevents visitors on mobile devices from needing to do extra work to read your content.
- **How can we implement a responsive website, meaning, how can we implement different designs for different screen sizes using CSS?**
  - Using CSS media queries. Media queries are CSS techniques introduced in CSS3. They use the @media rule to include a block of CSS properties only if certain conditions are true.
  - Example: https://www.w3schools.com/css/css_rwd_mediaqueries.asp

```
@media (max-width: 600px) {

body {

    background-color: lightblue;

 }

 }
```

  - The above media query code means that the CSS code inside only applies up until the width of the screen is 600px

1.2 Breakpoints

- A break point is where the website content/layout responds/changes according to the device width. CSS breakpoints are also called media query breakpoints, as they are used with media query.
- Here are the most common device breakpoints

```
Very narrow screens - below 540px
  @media only screen and (max-width: 540px) {...}
Smartphones - below 720px
  @media only screen and (max-width: 720px) {...}
Tablets - below 960px
  @media only screen and (max-width: 960px) {...}
Laptops - below 1140px
  @media only screen and (max-width: 1140px) {...}
Large screens - above 1140px
   @media only screen and (min-width: 1140px) {...}
```

1.3 Defining breakpoints for screen sizes

- You can choose your own numbers for screen size limitation, however, these are the most commonly used breakpoints
- Example: change the background color of the body tag based on screen width

```
@media (max-width: 576px) {

    /* Where it changes */

 /* This is where you put the CSS you want to take effect
only in this breakpoint   */

.container{

   max-width: 100%;  /*Actual width of the container */
```

```
}
}
@media (min-width: 576px) and (max-width: 767px) {
.container{
    max-width: 540px; /* Actual width of the container*/
}
}


@media (min-width: 768px) {
  .container {
    max-width: 720px;
}
}
@media (min-width: 992px) {
  .container {
     max-width: 960px;
    }
}
@media (min-width: 1200px) {
    .container {
        max-width: 1140px;
    }
}
```

1.4 The mobile first development approach

- This just means, you write all your CSS code in a way it works for mobile devices by default, without the need to use media queries

- This can be done by sketching the design for the smallest screen first and gradually working up to larger screen sizes.
- Refer this website for more explanation: https://www.mightyminnow.com/2013/11/what-is-mobile-first-css-and-why-does-it-rock/

1.5 Building the puppy page using media query: class demo

- Watch the video