# Responsive design

Responsive design covers topics that range from fixed layouts to fully responsive websites. Responsive design addresses accessibility challenges and how developers can use flexible units to overcome them. On top of being accessible, fully responsive websites are a joy to use and push web development forward into the future.

**At the end of this module, you should be able to:**

- craft pixel perfect designs that are also fully responsive
- create a fully responsive layout that fits nicely on phones, tablets, and desktops
- demonstrate knowledge of media queries by correctly implementing them in a project.
- describe the differences in fixed, fluid, adaptive, and responsive layouts.

## Prepare

☑ **Tip:** Review every preclass resource before class.

- **Responsive Web Design for CS8 w/ Josh Knell (https://youtu.be/0C5SqH1M1i0)**
  Responsive Web Design
- Optional Related Instruction Recordings
  - Responsive Web Design - I for CS11 w/ Josh Knell (https://youtu.be/_mr6UVCX6AA)
    A lesson on Responsive Web Design and Media Queries

    - CC Reverse String for CS11 w/Patrick Kennedy (https://youtu.be/1u9kVkqiDPI)
      Reverse String code challenge explanation
    - CC Reverse Number for CS11 w/Patrick Kennedy (https://youtu.be/AqQpkY0H1nw)
      Reverse Number code challenge explanation
    - Responsive Web Design - II for CS11 w/ Josh Knell (https://youtu.be/ea2gK9S43B0)
      Part two of the lesson on RWD and Media Queries
- Chrome Extension: Window Resizer (https://chrome.google.com/webstore/detail/window-resizer/kkelicaakdanhinjdeammmilcgefonfh?hl=en)
  This is not required to apply the techniques discussed in the video but it will make your life much easier.
- Chrome Extension: Perfect Pixel (https://chrome.google.com/webstore/detail/perfectpixel-by-welldonec/dkaagdgjmgdmbnecmcefdhjekcoceebi?hl=en)
  This tool will change your front end career by allowing you to get as close to perfect as possible.
- Percent based layout techniques (https://www.youtube.com/watch?v=epa5QFFpGHI&feature=youtu.be)
  Knowing how to navigate % based layouts is key to a fully responsive website. This short tutorial complete with example code examples. Please see the description in the video for the exercise files.
- A guide to em, rem, and px units (https://engageinteractive.co.uk/blog/em-vs-rem-vs-px)
  This quick read takes you though some modern considerations for responsive units.
- Rem font-size study (https://snook.ca/archives/html_and_css/font-size-with-rem)
  This is one of the first articles on the internet to suggest making the root font-size match the users settings. It's old but still very relevant.
- Fixed vs Fluid vs Adaptive vs Responsive (https://www.markupbox.com/blog/fixed-vs-fluid-vs-adaptive-vs-responsive-layout/)

# Project

## Responsive Web Design Project 1

Follow the link to the repo and follow the instructions.

Begin project (https://github.com/LambdaSchool/responsive-web-design)

# Learn

## Learn to craft pixel perfect designs that are also fully responsive

It’s one skill to create your own designs from scratch. It’s a totally different and required skill to be able to craft someone else’s designs. Learning the techniques and patterns on how to approach a design and create a responsive layout will set you apart from other developers.

### *Preclass Links*

- Chrome Extension: Window Resizer (https://chrome.google.com/webstore/detail/window-resizer/kkelicaakdanhinjdeammmilcgefonfh?hl=en)
  This is not required to apply the techniques discussed in the video but it will make your life much easier.
- Chrome Extension: Perfect Pixel (https://chrome.google.com/webstore/detail/perfectpixel-by-welldonec/dkaagdgjmgdmbnecmcefdhjekcoceebi?hl=en)
  This tool will change your front end career by allowing you to get as close to perfect as possible.

## Learn to create a fully responsive layout that fits nicely on phones, tablets, and desktops

Every web developer needs to have the ability to transform a fixed width website into a fully responsive website. Responsive websites have become the norm instead of the exception. Knowing how to navigate breakpoints, percent based layouts, and devices is essential for any full stack web developer.

### *Preclass Links*

- Percent based layout techniques (https://www.youtube.com/watch?v=epa5QFFpGHI&feature=youtu.be)
  Knowing how to navigate % based layouts is key to a fully responsive website. This short tutorial complete with example code examples. Please see the description in the video for the exercise files.
- A guide to em, rem, and px units (https://engageinteractive.co.uk/blog/em-vs-rem-vs-px)
  This quick read takes you though some modern considerations for responsive units.
- Rem font-size study (https://snook.ca/archives/html_and_css/font-size-with-rem)
  This is one of the first articles on the internet to suggest making the root font-size match the users settings. It's old but still very relevant.

---

## Learn to demonstrate knowledge of media queries by correctly implementing them in a project.

Media queries are an amazing tool for developers to build many types of responsive websites. Having a grasp of how they work and what you can do with them increases your ability to code for thousands of devices.

### *Overview*

Max-width and min-width are the keys to the car when it comes to media queries. Understanding that “mobile first design” directly translates into min-width is a concept that many new developers struggle with.

This simple example demonstrates the differences in max and min width:

https://codepen.io/bigknell/pen/ddYKVq

Media queries use max and min width to determine where breakpoints are introduced so it’s a big deal to make sure we understand how each works in relation to the viewport.

### *Follow Along*

Lets put media queries to work by working on a desktop-down approach.

Go to this codepen: https://codepen.io/bigknell/pen/GQjGgr?editors=1100

Base colors:
box1: aqua
box2: azure
box3: cadetblue

At 1000px and below:

- Change the width to 80% for ALL boxes

- box1: blue
- box2: green
- box3: red

At 768px and below:

- Change the width to 60% for ALL boxes
- box1: teal
- box2: gold
- box3: pink

At 400px and below:

- Change the width to 60% for ALL boxes
- box1: purple
- box2: black
- box3: gray

### *Challenge*

Keep the color and width requirements we just did as a group, but now lets take a mobile first approach by using min-width.

That means the base colors of: box1: aqua box2: azure box3: cadetblue

Need to show up AFTER 1000px and up. This also means the boxes should go back to 100px at 1000px and up as well.

This is a solo challenge, please don't share code until the solution is presented!

https://codepen.io/bigknell/pen/GQjGgr?editors=1100

### *Dig Deeper*

- Media Query Deep Dive: 100% Correct Way to do CSS Breakpoints (https://medium.freecodecamp.org/the-100-correct-way-to-do-css-breakpoints-88d6a5ba1862)
  This article is opinionated but I agree with the sentiments. Worth a read as you learn the "why" around media queries.

---

## Learn to describe the differences in fixed, fluid, adaptive, and responsive layouts.

Understanding the different approaches in responsive web design will help you make informed decisions on future projects you build.

*Fixed: Widths are fixed and sites don't budge. Very rigid.
*Fluid: 100% based on the viewport but can have strange effects at different sizes
*Adaptive: Breakpoint based on viewport but also very rigid.
*Responsive: Fluid + Adaptive = all bases covered from phone to desktop.

### *Preclass Links*

- Fixed vs Fluid vs Adaptive vs Responsive (https://www.markupbox.com/blog/fixed-vs-fluid-vs-adaptive-vs-responsive-layout/)

---

## Code Review

To demonstrate mastery of this module, you need to complete and pass a code review on each of the following:

- Code Challenge: EqualSides
- Code Challenge: CombineTwoStrings
- Objective challenges
- Project: Responsive Web Design Project 1



Site last generated: Jun 14, 2018