



## COURSE DESCRIPTION

In this training you will learn how to use CSS for more complex page layouts. You will gain a deeper understanding of how block and inline elements work. Master the positioning of elements (relative, absolute, fixed) and target elements with advanced CSS selectors. Learn about newer CSS3 transitions, transforms, and animations as well as create parallax scrolling effects using only CSS. **Live face-to-face instructor. No minimum class size - all classes guaranteed to run!**

## DETAILED COURSE OUTLINE

### THE BOX MODEL

This is the first in a series of three exercises in which you will style a simple blog using foundational CSS styling techniques.

- ➔ Adding the Viewport Meta Tag
- ➔ Using normalize.css
- ➔ Linking our CSS File and setting page defaults
- ➔ Scaling down hi-res images to fit the browser width
- ➔ Constraining the width of content
- ➔ The box model: adding padding, margins, & borders
- ➔ Visualizing the box model in Chrome's DevTools
- ➔ Fixing spacing issues around images

### BACKGROUND IMAGES AND CSS SHORTHAND

In this exercise we will take a deep dive into CSS background images and learn how to streamline the CSS with shorthand.

- ➔ Modifying the H1 Text Style
- ➔ Adding a background image to the h1
- ➔ Using shorthand for the Background Property

### REDEFINING CSS TYPOGRAPHY

In this exercise we will refine the look of our basic blog by fine-tuning the typography. We will examine font-style, font-weight, and improve legibility with line-height.

- ➔ Differentiating Text with Font-Style
- ➔ Unitless Line-Height
- ➔ Grouping Selectors Using a Comma Separator
- ➔ Numeric Font-Weight: Beyond Normal & Bold

### CREATING COLUMNS WITH FLOAT

CSS lets you "float" content next to other elements. The float property specifies whether or not an element should float. The clear property is used to control the behavior of floating elements.

- ➔ Creating a 2-column layout with float
- ➔ Solution #1: using the clear property
- ➔ Adding a border between the columns
- ➔ Solution #2: setting the overflow property to hidden

### INTRODUCTION TO MEDIA QUERIES

In this exercise, we will introduce you to media queries which are useful overrides for alternative layouts such as phones. We will use media query to switch our design from a 1-column layout to a 2-column layout.

- ➔ Finding an appropriate breakpoint
- ➔ Anatomy of a media query
- ➔ Using a media query to create an alternate layout
- ➔ Max-width media queries
- ➔ Sizing down the headings on smaller screens

### THE DISPLAY PROPERTY: A DEEP DIVE

In this exercise, we will style the header and footer links using the elements' display properties making our pages totally responsive. You will use inline-block as a hassle-free alternative to floats.

- ➔ Display types: block, inline, & inline-block
- ➔ Styling the header & footer navigation the DRY way
- ➔ Changing the display property to increase tappable area
- ➔ Displaying the navigation side-by-side on wider screens
- ➔ Using inline-block as an alternative to float

### THE POSITION PROPERTY: THE KEY TO COMPLEX LAYOUTS

In this exercise we will explore the layout options that the position property affords you. The position property is your best defense against a boring layout. Mastering this property and getting to know its relative, absolute, and fixed values opens up a world of "outside the box" possibilities.

- ➔ The static value & the normal document flow
- ➔ A nostalgic wanderer: the relative value
- ➔ The absolute value
- ➔ The dynamic duo: relative parent, absolute child
- ➔ The fixed value

### CREATING A FIXED HEADER

In this exercise, we will utilize the position property to make a website header more interesting. In the layout for wider screens, we will use fixed positioning to make a fixed header.

- ➔ Moving the Contact list item to the navbar's far right
- ➔ Creating a fixed header on wider screens
- ➔ Positioning a background image next to the FAQ

### IMAGE REPLACEMENT

In this exercise, you will place a background image behind a heading, hiding the text, and create a fluid image container that scales well with all devices.

- ➔ Replacing HTML text with CSS background images
- ➔ Removing text with negative text-indent value
- ➔ Using overflow: hidden;
- ➔ Removing text with a zero height
- ➔ Creating a fluid image container by using proportional top padding



## DETAILED COURSE OUTLINE (CONT.)

### CSS SPRITES

In this exercise, you will learn how to create a navbar using CSS sprites. The sprite technique allows you to create a search engine-friendly and accessible text-based navbar with the added luxury of using well-designed images.

- What are Sprites?
- Creating Sprites
- Image Replacement

### BACKGROUND GRADIENTS & TRANSPARENT COLORS

In this exercise, we will add content to the Tahoe Adventure Club page, including a background image. The image is a black & white photo that we will colorize using semi-transparent background gradients.

- Specifying colors are RGB
- Adding transparency (alpha) to colored backgrounds with RGBA
- CSS background gradients

### MULTI-COLUMN LAYOUT

In this exercise, we will continue adding to the Tahoe Adventure Club website. We will add three articles and investigate the ease of creating columns using inline-block instead of floats.

- 3-column layout using inline-block
- Nested CSS Selectors

### ADVANCED CSS SELECTORS

In this exercise, we will finish styling the Tahoe Adventure Club page. Instead of adding a bunch of classes or IDs to our markup, we are going to rely on the relationships between elements in order to target them cleanly and efficiently with CSS selectors.

- first-of-type
- first-child
- last-child
- last-of-type
- nth-child
- child selector
- :before and :after

### CSS ATTRIBUTE SELECTORS

In this exercise, we will make the links at the bottom of each article functional. We will focus on using attribute selectors to add nifty little icons next to each link to let users know what kind of resources they are linking to.

- Attribute selector syntax
- Caret (^) Operator
- Dollar (\$) Operator
- Asterisk (\*) Operator

### FORM STYLING

In this exercise, we will use attribute selectors to style forms, which is a more common and useful real-world scenario.

- Using an attribute selector to target inputs
- Overriding default form element styling

### CLEARING FLOATS

In this exercise, we layout content for narrow screens, which converts to a two column layout on large screens. We will run up against a new issue with clearing floats that will require us to learn how to use the CSS "clearfix" method.

- The CSS Clearfix class

### SIZING TYPOGRAPHY

In this exercise, we will look at how we can customize font sizes for different screen sizes. We will go over the difference between fixed pixel font sizes, em units, and rem units.

- Working with REM
- REM versus EM

### CSS3 SHADOWS

A well-placed drop shadow can add depth and dimension to any design. In this exercise, you will use thebox-shadow and text-shadow properties to elements on our page. You will also learn how to adjust an element's stacking order on the page using the z-index property.

- Box shadow
- Text shadow
- z-index

### HIDING & SHOWING ELEMENTS

In this exercise, we will investigate various ways to hide and show elements: setting the display property to none, the visibility to hidden, and setting opacity to 0. Many of these settings are designed to make content accessible to the visually impaired.

- display: none;
- Visibility
- Opacity

### CSS TRANSITIONS

In this exercise, we will add CSS3 transitions that are triggered by users. We will add two transitions: a basic fade-in that animates the color and text-shadow in the links in the header.

- Transition-Property
- Transition-Duration
- Transition Shorthand & the Transition Stack



## DETAILED COURSE OUTLINE (CONT.)

### CSS TRANSFORMS

In this exercise, we will add 2D scale and translate transforms onto our preexisting transitions to make our page even more engaging to our users.

- ⌕ The Scale Transform
- ⌕ The Rotation Transform
- ⌕ Transitioning Transforms

### YOU ARE HERE INDICATOR

In this exercise, we will work on a local city guide website that features travel info about a few U.S. cities. Our goal is to add an essential user experience feature: an indicator in the navigation that shows users the current page.

- ⌕ Navigation page marker
- ⌕ CSS Shapes

### CSS SPECIFICITY

In this exercise, we will demystify CSS' specificity hierarchy so you can override styles with confidence.

- ⌕ Advanced CSS Specificity
- ⌕ The Specificity Hierarchy
- ⌕ How Attribute Selectors & Pseudo-Classes Are Weighted
- ⌕ Inline Styles: The Highest Priority

Our outlines are a guide to the content covered on a typical class. We may change or alter the course topics to meet the objectives of a particular class.