

## HTML, XHTML, and CSS: Introduction to Web Development (1611-101)

**Length:** 4 Days

### Description

In Section 1, students learn to use HTML/XHTML and CSS (Cascading Style Sheets) in tandem: HTML/XHTML to provide and identify the various elements of content on the web page (headings, paragraphs, images, etc.) and CSS to format and lay out the elements. Students produce sophisticated pages, including those that require the use of the CSS box model, floating, and positioning.

In Section 2, students refine their web pages with the features that users expect today. That includes:

- Adding Flash, audio, and video to your pages
- Formatting your web content for easy printing
- Adapting your pages to display on mobile devices
- Using tested JavaScript code to add features like image rollovers, image swaps, and slide shows (a great first step in learning JavaScript)

Section 3 rounds out students' skills with chapters on web design. Once you have a good grasp of how to develop web pages, you can understand the kinds of decisions you have to make in designing a site, and this sections shows you how to do that using today's best practices.

### Audience

This course is for anyone who wants to develop web pages the modern, professional way using HTML/XHTML and CSS. That includes new web designers and web designers who still use tables for page layout when they should be using CSS.

It is also for web programmers, whether they're working in JavaScript or using a server-side language like PHP, ASP.NET, or Java servlets and JSPs. Web programmers are often surprised at the intricacies involved in writing good HTML/XHTML/CSS code, yet understanding that piece makes them more effective members of the development team.

### Prerequisites

Familiar with browsers, programming experience is helpful, but not required.

### Topics

#### SECTION 1: HTML, XHTML, AND CSS BASICS

##### INTRODUCTION TO WEB DEVELOPMENT

The introduction shows how static and dynamic web pages work, and it gives a first

view of how HTML and CSS are intended to work together in a modern web site.

##### *How web applications work*

- The components of a web application
- How static web pages are processed

## HTML, XHTML, and CSS: Introduction to Web Development (1611-101)

- How dynamic web pages are processed
- How JavaScript fits into web development
- A survey of browsers, servers, and scripting languages

### *An introduction to HTML, XHTML, and CSS*

- The HTML for a web page
- The CSS for a web page
- The HTML, XHTML, and CSS standards

### **EDIT, TEST, AND VALIDATE A WEB PAGE**

This chapter presents the basic syntax for HTML, XHTML, and CSS, and shows how to test, debug, and validate the code.

#### *The HTML syntax*

- The basic structure of an HTML document
- Coding elements and tags
- How to code attributes
- Comments and whitespace

#### *The CSS syntax*

- Coding CSS rule sets and comments
- How to code basic selectors

### **USING HTML TO STRUCTURE A WEB PAGE**

This chapter presents a professional subset of HTML/XHTML. Here, you'll see that HTML and XHTML are used to present the web page content and its structure. The formatting and page layout are left to CSS.

#### *How to start an HTML document*

- The DOCTYPE declaration
- How quirks mode works

#### *How to code the head section*

- How to specify the title for a page
- How to identify a linked file

- How to include metadata

#### *How to code text elements*

- Headings and paragraphs
- Special blocks of text
- Inline elements for formatting and emphasizing text
- Character entities
- Div and span elements
- The core attributes

#### *How to code links, lists, and images*

- Absolute and relative URLs
- Links
- Lists
- How to include images

#### *A structured web page*

- The page layout
- The HTML file

### **USING CSS TO FORMAT THE ELEMENTS OF A WEB PAGE**

Today's professionals use CSS (Cascading Style Sheets) to format the HTML content in a web page. This chapter is a course in the CSS basics, which will be built upon in the following chapters.

#### *Basic Skills for using CSS*

- Three ways to include CSS in a web page
- Specifying measurements
- How to specify colors

#### *How to code selectors*

- Selectors for elements, ids, and classes
- Selectors for descendants, combinations, and siblings
- Other types of selectors
- How to code pseudo-class selectors
- How the cascade rules work

## HTML, XHTML, and CSS: Introduction to Web Development (1611-101)

### *How to work with text and links*

- Setting the font family and font size
- How to set the other properties for styling fonts
- Indent and align text
- How to transform and decorate text
- How to format links

### *A web page that uses an external style sheet*

- The page layout
- The HTML file
- The CSS file

### **USING THE CSS BOX MODEL FOR SPACING, BORDERS, AND BACKGROUNDS**

In this chapter, you'll learn how to use the CSS box model so you can set the margins, spacing, borders, and backgrounds for a web page just the way you want them.

#### *An introduction to the box model*

- How the box model works
- A web page that illustrates the box model

#### *How to size and space elements*

- How to set heights and widths
- How to set margins
- How to set padding

#### *A web page that illustrates sizing and spacing*

- The HTML for the web page
- The CSS for the web page
- Another way to implement spacing

#### *How to set borders and backgrounds*

- How to set borders
- How to set background colors and images

### *A web page that uses borders and backgrounds*

- A warning on the use of background colors and images
- The HTML for the web page
- The CSS for the web page

### **HOW TO USE CSS FOR PAGE LAYOUT**

This chapter shows you how to use CSS to create two- and three-column pages that have headers and footers. To do that, you'll learn how to float elements on a page, as well as how to use absolute and relative positioning.

#### *How to float elements*

- How floating works
- How to float an element
- How to clear a floated element

#### *A two-column layout that uses floating*

- The page layout
- The HTML
- The CSS for a fixed layout
- The CSS for a liquid layout

#### *A three-column layout that uses floating*

- The page layout
- The structure of the HTML
- The CSS

#### *How to position elements*

- Four ways to position an element
- How to use absolute positioning
- How to use relative positioning
- How to use fixed positioning

#### *A three-column layout that uses floating and positioning*

- The page layout
- The HTML
- The CSS

## HTML, XHTML, and CSS: Introduction to Web Development (1611-101)

### SECTION 2: OTHER HTML AND CSS SKILLS AS YOU NEED THEM

#### HOW TO WORK WITH LINKS AND LISTS

Previous chapters covered the basic skills for coding links and lists and for using CSS to change the appearance of links. Now, this chapter builds on those skills so that you'll be able to create links and lists that look and work just the way you want them to.

##### *How to code links*

- How to link to another page or start an email message
- How to use a link to open a new browser window
- How to create and use an anchor
- How to create a navigation bar

##### *How to code lists*

- How to code unordered lists
- How to code ordered lists
- How to code nested lists
- How to code definition lists

##### *How to format lists*

- Changing bullets for unordered lists
- How to change the numbering system for an ordered list
- Change the alignment of list items

#### HOW TO WORK WITH IMAGES

This chapter expands upon the skills you learned to add images to a web page and shows you how to use an image editor to work with the images for your web pages.

##### *Basic skills for working with images*

- Types of images for the Web
- How to include an image on a page
- How to resize an image
- How to align an image vertically

- How to float an image

##### *How to use an image editor*

- An introduction to Photoshop Elements
- How to size an image
- Change the image type and quality
- How to work with animated images
- Working with transparency and mattes

##### *Four more skills for working with images*

- How to get images and icons
- How to create and work with thumbnails
- How to create favicons
- How to create image maps

#### HOW TO WORK WITH TABLES

In the past, tables were widely used to format web pages, but today, formatting is done through CSS. This chapter focuses on using tables for what they were originally designed for: to present tabular data.

##### *How to code and format simple tables*

- An introduction to tables
- How to code a table
- How to use CSS to format a table
- How to add a caption
- How to add a header and footer

##### *Advanced skills for working with tables*

- How to merge cells in a column or row
- How to nest tables
- How to control wrapping
- How to provide for accessibility

#### HOW TO WORK WITH FORMS

Many web pages use forms for user interactions like buying a product, signing up for a newsletter, or asking for

## HTML, XHTML, and CSS: Introduction to Web Development (1611-101)

information. The entries in these forms are processed on the client or server side...or both! This chapter covers user-friendly forms, and later you'll use JavaScript to validate form entries on the client side.

### *How to code forms*

- How to create a form
- How to code buttons
- How to code labels and text fields
- How to code radio buttons and check boxes
- How to code drop-down lists
- How to code list boxes
- How to code text areas
- How to group controls

### *Other skills for working with forms*

- How to align controls
- How to format controls
- How to set the tab order for controls
- How to assign access keys
- How to add a search function to a web site

### **HOW TO ADD AUDIO, VIDEO, AND OTHER MEDIA TO YOUR WEB SITE**

Images are the most common type of media in web sites, and earlier chapters show how to work with them. Now, this chapter shows how to include other types of media, such as audio and video, in your web pages, as well as how to work with some common media players.

#### *An introduction to media*

- Media types and formats
- Media players
- How to link to a media file

#### *How to embed media players and files*

- The object and param elements

- How to use the embed element
- When and how to nest object and embed elements

#### *How to work with media players*

- How to work with Adobe Flash Player
  - How to work with Windows Media Player
  - How to work with RealPlayer
  - How to work with QuickTime Player
- An introduction to media

### **HOW TO WORK WITH PRINT MEDIA AND MOBILE DEVICES**

So far, we have learned how to create and format web pages for display on a computer screen. Now, this chapter shows how to create and use a style sheet for printing a web page. It also shows how to create and work with web pages that are displayed on mobile devices like cell phones and PDAs.

#### *How to define style sheets for different media types*

- How to identify the media type for a style sheet
- How to identify the media type for a rule set

#### *How to work with print media*

- Recommendations for formatting printed pages
- CSS properties for controlling printed pages
- The screen layout for a two-column web page
- The HTML for the web page
- The CSS for the screen layout
- The print layout for the web page
- The CSS for the print layout

## HTML, XHTML, and CSS: Introduction to Web Development (1611-101)

### *How to work with mobile devices*

- How to provide pages for mobile devices
- How to set viewport properties
- Guidelines for designing mobile web pages
- How to test mobile web pages
- The HTML for the mobile web page

### **HOW TO USE JAVASCRIPT TO ENHANCE YOUR WEB PAGES**

Although this chapter doesn't teach how to code JavaScript, it does provide a solid introduction to how JavaScript works. What's more, it shows you how to use tested JavaScript code so you can start enhancing your web pages with features like slide shows almost immediately. At the end, you'll be well-prepared to learn JavaScript in depth.

### *Introduction to JavaScript and DOM scripting 390*

- How JavaScript works
- How the DOM (Document Object Model) works
- How DOM scripting works
- How to include JavaScript in an HTML document

### *A standalone JavaScript application*

- The HTML for the Future Value application
- The JavaScript for the Future Value application

### *How to use JavaScript in your web pages*

- JavaScript for image rollovers
- JavaScript for image swaps
- JavaScript for slide shows
- JavaScript for tabbed data
- JavaScript for data validation

- Web sites for JavaScript code
- How to test and debug a JavaScript application

### **SECTION 3: DESIGNING A WEBSITE**

#### **HOW TO DESIGN A WEB SITE**

Once you know how to implement a web site, you're in a good position to understand the considerations that go into designing an effective web site. This chapter gives you procedures and guidelines for doing just that.

#### *Users, usability, and accessibility*

- What users want is usability
- The current conventions for usability
- The importance of accessibility

#### *Design guidelines*

- Use the home page to sell the site
- Let the users know where they are
- Make best use of web page space
- Write for the web
- Chunk long pages into shorter pages
- Basic principles of graphics design

#### *Design methods and procedures*

- The use of development teams
- Top-down design, stepwise refinement, and prototyping
- The lifecycle of a web site

#### *The design procedure applied to a small web site*

- Define the audience and set the goals
- Design the home page
- Design the primary content pages
- Design the other pages
- Plan the navigation
- Plan the directories for the site