

SAN DIEGO COMMUNITY COLLEGE DISTRICT  
CONTINUING EDUCATION  
COURSE OUTLINE

**SECTION I**

**SUBJECT AREA AND COURSE NUMBER**

COMM 672

**COURSE TITLE**

WEB PROGRAMMING: HTML AND CSS

**TYPE COURSE**

NON-FEE

VOCATIONAL

**CATALOG COURSE DESCRIPTION**

This course includes an overview of the internet, components of a website and introduces web programming using the Hypertext Markup Language (HTML) and Cascading Style Sheets (CSS). Topics include, current web technologies, website development tools, web programming skills and hand coding of a website. Students will learn to publish projects and sites to the web and utilize CSS for web page layout and formatting. Current industry standards, processes and techniques are also taught. (FT)

**LECTURE/LABORATORY HOURS**

180

**ADVISORIES**

COMM 670 INTRODUCTION TO WEB DEVELOPMENT  
COMM 671 SOFT SKILLS IN WEB DEVELOPMENT

**RECOMMENDED SKILL LEVEL**

Possess a 10th grade reading level; ability to communicate effectively in the English language.

### INSTITUTIONAL STUDENT LEARNING OUTCOMES

1. Social Responsibility  
SDCE students demonstrate interpersonal skills by learning and working cooperatively in a diverse environment.
2. Effective Communication  
SDCE students demonstrate effective communication skills.
3. Critical Thinking  
SDCE students critically process information, make decisions, and solve problems independently or cooperatively.
4. Personal and Professional Development  
SDCE students pursue short term and life-long learning goals, mastering necessary skills and using resource management and self-advocacy skills to cope with changing situations in their lives.

### COURSE GOALS

1. Introduce web programming using the Hypertext Markup Language (HTML)
2. Introduce Cascading Style Sheets (CSS) for web page structure and formatting
3. Learn to hand-code web pages
4. Explore how to make web pages adapt to varying view port sizes by implementing Responsive Web Design (RWD) techniques
5. Examine the process of designing web-based forms for web pages
6. Use third-party tools and frameworks for quick web page layout and formatting
7. Learn how to publish projects and sites to the Web and web-based file repositories
8. Understand current industry standards, processes and techniques

### COURSE OBJECTIVES

Upon successful completion of the course, the student will be able to:

1. Demonstrate an understanding of core web programming concepts and explain the components of a website
2. Demonstrate how to hand code HTML and CSS without the need for visual editors
3. Demonstrate how to use CSS to structure and format a web page
4. Design and build a website that is visually appealing on desktop, tablet, and mobile browsers
5. Integrate a form and form objects into a web page
6. Demonstrate ability to use third-party tools and frameworks within a web development project
7. Demonstrate how to publish a project to the web and to a web-based file repository
8. Explain current industry standards, processes and techniques

## **SECTION II**

### **COURSE CONTENT AND SCOPE**

1. An Overview of Web Development
  - 1.1. Web architecture
  - 1.2. The Hypertext Markup Language (HTML)
  - 1.3. Cascading Style Sheets (CSS)
  - 1.4. The role of JavaScript
  - 1.5. The World Wide Web Consortium (W3C)
  - 1.6. Semantic HTML
  - 1.7. Basic structure of a web page
  - 1.8. HTML syntax
  - 1.9. Browsers to download
  - 1.10. Using a code editor
  - 1.11. Creating a simple web page
  - 1.12. Validating your web page
2. Structure and Semantic Markup
  - 2.1. The structure of a tag
  - 2.2. Global attributes
  - 2.3. Head elements
  - 2.4. Section elements
  - 2.5. Headings
  - 2.6. Grouping elements
  - 2.7. Text-level semantic elements
3. Links, Lists, and Images
  - 3.1. Links
    - 3.1.1. Working with links
    - 3.1.2. Attributes for links
    - 3.1.3. Creating a link for text
    - 3.1.4. Creating a link for an image
    - 3.1.5. Creating a link to an email address and telephone number
    - 3.1.6. Creating a link to a placeholder
    - 3.1.7. Working with image maps
  - 3.2. Lists
    - 3.2.1. Working with lists
    - 3.2.2. Ordered lists
    - 3.2.3. Unordered lists
    - 3.2.4. Definition lists
  - 3.3. Images
    - 3.3.1. Working with images
    - 3.3.2. The difference between GIF, JPG, and PNG
    - 3.3.3. Using images in HTML
    - 3.3.4. Attributes for images
    - 3.3.5. Scalable Vector Graphics (SVG)
4. Introduction to CSS
  - 4.1. The role of CSS
    - 4.1.1. What CSS is
    - 4.1.2. Why you want to use CSS

COURSE CONTENT AND SCOPE (CONTINUED)

- 4.1.3. How CSS works
  - 4.1.4. Defining styles
  - 4.1.5. Three ways of defining styles
- 4.2. Selector Types
  - 4.2.1. CSS 1 Selector Types
  - 4.2.2. CSS 2 Selector Types
  - 4.2.3. CSS 3 Selector Types
  - 4.2.4. Beyond CSS 3: A module-based approach
- 4.3. Exploring the core CSS Properties
  - 4.3.1. Font properties
  - 4.3.2. Background properties
  - 4.3.3. Block properties
  - 4.3.4. Box properties
  - 4.3.5. Border properties
  - 4.3.6. List properties
  - 4.3.7. Positioning properties
- 5. CSS Grid Systems, Box Model Layouts, and Positioning
  - 5.1. An introduction to Grid Systems
    - 5.1.1. What a layout grid is
    - 5.1.2. History of grid-based layouts in design
    - 5.1.3. Why using grid-based layouts are helpful
    - 5.1.4. How grid systems work in web design
    - 5.1.5. The 960 Grid System
  - 5.2. How to Layout and Position Elements
    - 5.2.1. How to float and clear elements
    - 5.2.2. How to use floating in a 2-column, fixed-width layout
    - 5.2.3. How to use floating in a 3-column, fixed-width layout
    - 5.2.4. How to use floating to create a sidebar
    - 5.2.5. How to use clear to create a footer
    - 5.2.6. How to use floating in a 2-column, fluid layout
    - 5.2.7. How to use floating in a 3-column, fluid layout
    - 5.2.8. How to use CSS to create text columns
    - 5.2.9. How to position elements using static, relative, absolute, and fixed positioning
- 6. Advanced Layouts and Topics in CSS
  - 6.1. Valuable HTML5 and CSS Tools
  - 6.2. Exploring reset style sheets
  - 6.3. CSS Linters
  - 6.4. Tools that visually build CSS for you
  - 6.5. Tools that visually build CSS menus for you
  - 6.6. Advanced Structuring Concepts with HTML5 and CSS3
    - 6.6.1. An advanced web page layout that uses HTML5 section elements
    - 6.6.2. Making the markup backward compatible
    - 6.6.3. Designing a style sheet for printers
  - 6.7. Special Topics in CSS3
    - 6.7.1. CSS Sprites
    - 6.7.2. Multi-tier menus with CSS
    - 6.7.3. Accordion menus with CSS

COURSE CONTENT AND SCOPE (CONTINUED)

- 6.7.4. Tab menus with CSS
- 6.7.5. Collapsible panels with CSS
- 6.8. Web Fonts and Icons
  - 6.8.1. The CSS3 @font-face selector
  - 6.8.2. How to use Google Web Fonts
  - 6.8.3. How to use CSS icon libraries
- 7. CSS Flex Box Layout Module and Grid Layout Module
  - 7.1. The Flexible Box Layout Module
    - 7.1.1. The CSS Flexible Box layout module
    - 7.1.2. Flexible Box Layout terminology
    - 7.1.3. Flexible Box Layout properties
    - 7.1.4. A simple Flexible Box layout
    - 7.1.5. How to set flex item dimensions
    - 7.1.6. How to align flex items horizontally
    - 7.1.7. How to align flex items vertically
    - 7.1.8. How to wrap and align wrapped items
    - 7.1.9. How to change the order of flex items
  - 7.2. The Grid Layout Module
    - 7.2.1. Grid terminology
    - 7.2.2. Grid containers and grid items
    - 7.2.3. A simple Grid layout
    - 7.2.4. Defining rows and columns
    - 7.2.5. Adding gutters
    - 7.2.6. The fr unit
    - 7.2.7. Mixing units
    - 7.2.8. Positioning items
    - 7.2.9. Template areas
    - 7.2.10. Named lines
- 8. Responsive Web Design
  - 8.1. An introduction to Responsive Web Design
    - 8.1.1. Mobile matters
    - 8.1.2. UX of the modern web
    - 8.1.3. Flexible layouts (grids)
    - 8.1.4. Flexible type
    - 8.1.5. Flexible images
    - 8.1.6. Tables
    - 8.1.7. Flat design
    - 8.1.8. A new responsive grid system
    - 8.1.9. Responsive Web Design resources
    - 8.1.10. Responsive Web Design frameworks
    - 8.1.11. How to test a responsive design using a browser's developer tools
  - 8.2. CSS3 Media Queries
    - 8.2.1. Introduction to CSS3 Media Queries
    - 8.2.2. Media types, media features, and expressions
    - 8.2.3. Common media queries
    - 8.2.4. Media query support
    - 8.2.5. How to control the mobile viewport

COURSE CONTENT AND SCOPE (CONTINUED)

- 8.2.6. Revisiting the 960 Grid System
- 8.3. Third-Party RWD Frameworks and Tools
- 9. Web Forms
  - 9.1. Introduction to Web forms
  - 9.2. The form tag
  - 9.3. The collection of form tags
    - 9.3.1. The input tag (text and password)
    - 9.3.2. The input tag (checkbox and radio)
    - 9.3.3. The input tag (submit, reset, and button)
    - 9.3.4. The select, option, and optgroup tags
    - 9.3.5. The textarea tag
    - 9.3.6. The label tag
    - 9.3.7. New HTML5 tags
  - 9.4. HTML5 Data Validation Techniques
- 10. New HTML and CSS Modules
  - 10.1. New HTML Modules
    - 10.1.1. The classList API
    - 10.1.2. Working with audio in HTML
    - 10.1.3. Working with video in HTML
    - 10.1.4. The Web Storage API
    - 10.1.5. The Geolocation API
    - 10.1.6. The Canvas API
    - 10.1.7. The Drag and Drop API
    - 10.1.8. Additional Modules
  - 10.2. New CSS Modules
    - 10.2.1. Transitions
    - 10.2.2. Transforms
    - 10.2.3. Animations
    - 10.2.4. Filters
    - 10.2.5. Additional New CSS Modules
- 11. Introduction to Third-Party Web Development Frameworks
  - 11.1. Introduction to third-party web development frameworks
    - 11.1.1. Where to find and download third-party frameworks
    - 11.1.2. How to set up a third-party frameworks to work with your website
    - 11.1.3. How CSS, components, and JavaScript are handled with third-party frameworks
    - 11.1.4. How to use the starter templates
  - 11.2. Structuring a Web Page with a Third-Party Framework
    - 11.2.1. Introduction to the grid system
    - 11.2.2. How to create rows and columns
    - 11.2.3. How to adjust column widths based on the viewport size
    - 11.2.4. How to use grid components
  - 11.3. Formatting a Web Page in a Third-Party Framework using Components
    - 11.3.1. Exploring typographic styles
    - 11.3.2. How to use buttons and button styles
    - 11.3.3. How to add icons
    - 11.3.4. How to create and stylize a thumbnail gallery

### COURSE CONTENT AND SCOPE (CONTINUED)

- 11.3.5. How to build navigation menus
- 11.3.6. How to use tabs and pills for navigation
- 11.3.7. How to make the navigation menu responsive
- 11.3.8. How to add breadcrumbs to your web pages
- 11.3.9. How to add dropdown menus to the navigation menu
- 11.3.10. How to use create sticky navigation menus
- 11.3.11. How to build image gallery carousels
- 11.3.12. Additional components
- 12. Deploying a Web Site
  - 12.1. How to get a web host and domain name
  - 12.2. How to find a web host
  - 12.3. How to get a domain name
  - 12.4. How to use File Transfer Protocol (FTP)
    - 12.4.1. How to install an FTP client
    - 12.4.2. How to connect to a web site on a remote web server using FTP
    - 12.4.3. How to upload and download files using FTP
  - 12.5. More skills for deploying a web site
    - 12.5.1. How to test a web site that has been uploaded to the web server
    - 12.5.2. How to get your web site into search engines and directories
    - 12.5.3. How to control which pages are indexed and visited
    - 12.5.4. How to maintain a healthy web site

### APPROPRIATE READINGS

Reading assignments of appropriate reading level may include, but are not limited to, readings from the textbook, supplemental reading assignments, industry-related periodicals or magazines, manuals, online help pages, articles posted on the Internet, and information from Web sites, online libraries and databases. Topics should be related to basic web programming and include techniques for semantically structuring a web site using HTML and designing the layout and providing the format for a web page using CSS.

### WRITING ASSIGNMENTS

Writing assignments may include, but are not limited to, completing assigned reports, providing written answers to assigned questions, performing internet research and reporting on that research. An example would include a case study about the progression of CSS structuring techniques over the last 20 years moving from basic tables-based structuring to using CSS Flexible Box layout and/or Grid layout modules of today.

### OUTSIDE ASSIGNMENTS

Outside assignments may include, but are not limited to, appropriate internet research, reading, preparing reports and studying as needed to perform successfully in class. An appropriate assignment for instance, would include the creation of a web site that relies on third-party tools and frameworks.

### APPROPRIATE ASSIGNMENTS THAT DEMONSTRATE CRITICAL THINKING

Assignments which demonstrate critical thinking may include, but are not limited to outlining a web based project, use and creation of hand-coded HTML and CSS and publishing projects and sites to the web. Students may also be expected to participate in online class discussion posts, in-class discussions and project reviews.

### EVALUATION

Evaluation that a student has met the course competencies will include multiple measures of performance related to the course objectives. Evaluation methods may include, but are not limited to performance in a variety of activities and assignments, such as completing a research project individually or in a group, hands-on projects, demonstration of use of the internet, quizzes, class participation, written and practical tests, attendance and punctuality.

Upon successful completion of all courses in the program a Certificate of Program Completion will be issued.

### METHOD OF INSTRUCTION

Methods of instruction, may include, but are not limited to, lecture, in-class and online discussions, hands-on demonstrations, computer-assisted instruction, field trips, and laboratory assignments. This course, or sections of this course, may be offered through distance education.

### TEXTS AND SUPPLIES

*Murach's HTML5 and CSS3*, Zak Ruvalcaba and Anne Boehm, Murach Publishing, current edition

Web Resources:

Udemy: HTML5 and CSS3, <https://www.udemy.com/course/html5-and-css3-essential-training>

Recommended Supplies:

Pen, journal (composition book), notebook paper and a soft 3-ring binder, or a one-subject 110 sheet college ruled notebook, and appropriate storage media such as a USB Drive or external hard drive.

PREPARED BY Zak Ruvalcaba DATE November 3, 2021

REVISED BY \_\_\_\_\_ DATE \_\_\_\_\_

Instructors must meet all requirements stated in Policy 3100 (Student Rights, Responsibilities and Administrative Due Process), and the Attendance Policy set forth in the Continuing Education Catalog.

### REFERENCES:



San Diego Community College District Policy 3100  
California Community Colleges, Title 5, Section 55002  
Continuing Education Catalog