

CSS – Cascading Style Sheets

Handling Styling, Selectors, Descriptions, Specificity, Priority, Pseudo-, CSS for ..., SCSS

CSS is not a markup language (unlike HTML5) used to represent content. CSS is not a programming language (unlike JavaScript/TypeScript), used to dynamically change content at run time. Rather, CSS is a styling language used to concisely represent styling information for web content.

CSS is used to declaratively describe the initial hierarchical set of styles that are to be applied to HTML elements which later may be manipulated by TypeScript/JavaScript code. CSS offers considerable flexibility in how styles are specified and how they can

be applied to elements. An important consideration is to maintain the same “look & feel” across multiple pages, Thus a site-wide “house style” is often defined. Also CSS allows the same content to be optimally rendered on output devices with differing capabilities (size, resolution, interactivity) and suitably for people with differing interaction needs.

This course looks at the role of CSS in the world of modern web development, provides a detailed tour of CSS features and explores how you can best deploy it for styling your own web solutions.

Contents of One-Day Training Course	
<p>Target Audience All developers wishing to gain a comprehensive understanding of the least known of the three pillars of web development (the other two being HTML and JavaScript/TypeScript).</p> <p>Prerequisites Knowledge of HTML is required along with some graphics background.</p>	<div style="display: flex; justify-content: space-between;"> <div style="width: 48%;"> <p>CSS Overview</p> <ul style="list-style-type: none"> Growing set of styling standards Defining CSS rules Style document layout Declaration block Internal CSS using <style> External CSS using <link> and .css file Could be used to “render” speech, but mostly used for visual representations <p>CSS 1 / CSS 2 / CSS 3 / CSS 4</p> <ul style="list-style-type: none"> Evolution of standards (CSS modules) Expanding set of layered standards What is [well less well] supported <p>Selector</p> <ul style="list-style-type: none"> Identifying categories of markup elements Element type Id Class Element hierarchy <p>Declaration Blocks</p> <ul style="list-style-type: none"> Collection of declarations in blocks Property and value (separated by colon) Representing values using custom syntax suitable for each value type <p>Inheritance</p> <ul style="list-style-type: none"> Importance of style ancestry Styling descendants <p>Pseudo-class/pseudo-element</p> <ul style="list-style-type: none"> Additional ways to select elements Styling after particular actions (e.g. click) <p>Combining selectors</p> <ul style="list-style-type: none"> Importance of selector order Priority </div> <div style="width: 48%;"> <p>CSS for Layout</p> <ul style="list-style-type: none"> Defining where elements appear on page Replacement for HTML tables (sometimes) Multi-columns and flexible layout <p>CSS for Color</p> <ul style="list-style-type: none"> Defining color scheme for elements Element color Background Ways to define color (e.g. gradient) <p>CSS for Font</p> <ul style="list-style-type: none"> Selecting font types for text Styling such fonts (italics, bold) Sizing fonts <p>CSS for Spacing</p> <ul style="list-style-type: none"> Margin Border Internal text spacing <p>CSS For Text</p> <ul style="list-style-type: none"> Alignment Paged media Translations <p>CSS For Content Handling</p> <ul style="list-style-type: none"> Web animation Transitions Generated content <p>SCSS</p> <ul style="list-style-type: none"> A CSS pre-processor that simplifies CSS syntax and extends its capabilities “allows you to use variables, nested rules, mixins, inline imports and more” with CSS <p>Project</p> <ul style="list-style-type: none"> How to use CSS in a large enterprise solution to provide flexible styling </div> </div>