CSS-PRE - CSS e Preprocessor: Tecniche Moderne di Styling con LESS, SASS e Stylus

ITCore Sviluppo - HTML - Sviluppo

Durata:	Lingue:	Certificazione:
2 Giorni	**Italiano**	**-**

Descrizione del corso

l corso **"CSS e Preprocessor"** è progettato per web designer e sviluppatori che vogliono padroneggiare le tecniche avanzate di **styling** utilizzando **CSS** e i principali **preprocessor**: **LESS**, **SASS** e **Stylus**. L'obiettivo è migliorare l'efficienza, la manutenibilità e la scalabilità dei fogli di stile nella realizzazione di applicazioni web moderne.
Obiettivi del Corso:
Al termine del corso, i partecipanti saranno in grado di:
• Scrivere fogli di stile modulari e scalabili utilizzando **CSS** moderno.
• Sfruttare al massimo i **preprocessor** LESS, SASS e Stylus per migliorare la produttività e la qualità del codice.
• Integrare strumenti avanzati di sviluppo nel proprio workflow.
• Applicare best practices per la gestione dello styling in progetti di grandi dimensioni.
Destinatari:
• Web designer e sviluppatori front-end
• Professionisti che vogliono migliorare l'efficienza nella scrittura di CSS
• Chi desidera aggiornarsi con gli strumenti più moderni per lo sviluppo web
Prerequisiti: Conoscenza base di HTML.
Un corso pratico e completo per creare fogli di stile **ottimizzati**, **modulari** e **manutenibili** sfruttando la potenza dei **preprocessor** moderni come **LESS**, **SASS** e **Stylus**.

Programma

- **Introduzione a CSS**
  - Regole di base: selettori, proprietà, unità di misura
  - Tecniche avanzate: pseudo-classi, pseudo-elementi, combinatori

- Layout moderni: Flexbox e CSS Grid
- Animazioni e transizioni

- **Limiti di CSS e Introduzione ai Preprocessor**
  - Vantaggi dei preprocessor: modularità, riutilizzo del codice e variabili
- **LESS (Leaner CSS)**
  - Installazione e configurazione
  - Variabili, mixin e funzioni
  - Nesting e importazione di file
- **SASS (Syntactically Awesome Style Sheets)**
  - Differenza tra **SASS** e **SCSS**
  - Nesting, variabili e operatori
  - Partials e modularità con @import
  - Mixins e funzioni avanzate
  - Utilizzo di **@extend** e **placeholders**
- **Stylus**
  - Sintassi minimale e flessibile
  - Uso di variabili, mixins e nesting
  - Funzioni personalizzate e manipolazione dinamica
- **Strumenti e Workflow**
  - Integrazione di preprocessor nei moderni ambienti di sviluppo
  - Utilizzo di **Task Runners** (Gulp, Webpack) per automatizzare la compilazione
- **Best Practices**
  - Organizzazione del codice CSS con metodologie come **BEM** (Block, Element, Modifier)
  - Strategie per scrivere fogli di stile manutenibili e performanti