

## COURSE OVERVIEW

# Programming in HTML5 with JavaScript and CSS3 & Developing ASP.NET MVC 4 Web Applications Bootcamp

*Combined (2 courses for 1)*

*5 day Fast Track exam – 70-480 and 70-486*

*Duration: 5 days – 8:30am to 18:30pm*

## Overview

*This course covers how to use HTML5, CSS3, JavaScript and ASP.NET MVC 4.5 to build modern Web Applications. The course format will be in the form of a bootcamp with revision in the evenings. The objective of the bootcamp is to prepare students for the following exams:*

- *70-480 Programming in HTML5 with JavaScript and CSS3 and*
- *70-486 Developing ASP.NET MVC 4 Web Applications*

## Audience

*Professional developers with 6-12 months of experience developing Web Applications with the following tools and technologies:*

- *Writing simple HTML and JavaScript*
- *Experience with the .Net Framework and C#*
- *ASP.NET*
- *Visual Studio 2010/2012 or 2013*

## Course Outline

*The following areas will be covered using the standard Microsoft Official Curriculum Courseware:*

### HTML5

- *Semantic mark-up*
- *HTML5 Canvas and Scalable Vector Graphics*

## CSS3

- *Styling text and HTML*
  - *Web Open Font Format*
  - *@font-face*
  - *Size, colour, alignment and drop shadows*
- *HTML Box styling*
  - *Border, outline, corners, padding and margins*
  - *Graphic effects, transparency, shadows and position*
- *Flexible box model, multi-column and grid-alignment layouts*
- *Media queries, adaptive UI, CSS transitions and transformations (3D and 2D),*
- *CSS selectors and pseudo-classes/elements*
- *CSS inheritance and overrides*

## JavaScript

- *JavaScript variables*
- *JavaScript program statements and logic*
- *Prototypes, functions and call-backs*
- *JavaScript and DOM events*
- *Exception handling*
- *jQuery AJAX calls and the XMLHttpRequest*
- *HTML5 WebSocket API*
- *Web Workers*

## HTML Forms

- *Input Validation, HTML5 attributes and types*
- *JavaScript Input Validation*
- *Processing JSON and XML data, parsing and encoding*
- *Serializing and Form submit*

## MVC Architecture

- *Application layers, separation of concerns, models, views and controllers*
- *Client-side and server-side choices*
- *Windows Azure Service lifecycle events*
- *State management and scalability – cookies and session state*
- *Implementing and using WebSocket connections*
- *Synchronous and asynchronous HTTP modules and handlers*
- *Controllers and Actions*
  - *Authorisation attributes and global filters*

- *Action behaviours and results*
- *Routes, URL patterns, parameters and constraints*
- *MVC extensibility*
  - *Filters and Controller Factories*
  - *View engines, model binders and route handlers*

## *MVC User Interface Design*

- *Dynamic HTML page content*
- *Partials views, Razor templates, design layouts and master/application pages*
- *Adaptive User Interfaces and Mobile Views*
- *Search engine optimisation*
- *User Interface and JavaScript resources for multilingual applications*
- *Minifying CSS and JavaScript, gzip/deflate data and Content Delivery Networks*

## *Debugging*

- *Tracing, logging, Code Contracts and IntelliTrace*
- *Exception handling across layers and custom error pages*
- *Unit tests and mocks*
- *Windows Azure diagnostics API and log types*

## *Security*

- *Authentication settings, sessions and membership providers*
- *Role and WCF Service Authorisation and role providers*
- *Claims-based authentication, Windows Azure Access Control Service, SAML, SWT and JWT tokens*
- *Encryption and signing of application data and configuration settings*
- *Preventing cross-site scripting attacks, SQL injection attacks, cross-site request forgeries and deferred validation*