MVC 4.0, Part 03 of 11: Views and Models

page 1

Meet the expert: Philip Ledgerwood has been a software developer for fifteen years. He currently works primarily in .NET technologies producing custom software for organizations of all sizes. He has also done extensive training for those same organizations in both technical and business process topics. Philip is a strong advocate of Lean and agile software development and spends most of his time helping companies interested in the value those practices can bring to their development efforts. He does this through a combination of training and working "in the trenches" as a developer on these teams, keeping a hand in the academic side of emerging technology and practices while also directly applying it in real projects to bring real business value.

Prerequisites: This course assumes that you are familiar and experienced with Microsoft's .NET Framework and ASP.NET development tools. You should be familiar with Web development and understand how HTTP and HTML work to produce Web pages for the user. You should have strong experience with .NET Framework 4.0 or later programming. You should have experience with Visual Studio 2012. Experience with building database applications using these tools will be helpful, although not strictly necessary.

Runtime: 01:51:34

Course description: You don't have to be an artist to know that the way an application looks to a user is one of the most important things you can deliver. An application that is easy to use, performs quickly, and is easy to change can make or break you in the market or mean thousands of dollars of productivity gained or lost.This course will cover how Views can affect your application, and then you will see how Typed Views can bind to Models. Next you will be introduced to a view engine called Razor. To close the course you will see how to use Layouts to take advantage of the MVC framework's features that allows you to reuse views and parts of views.

Course outline:

Views
- Introduction
- What a View Is
- Demo: Views
- Finding the Right View
- Demo: Assigning Views
- ViewData and ViewBag
- Demo: ViewData and ViewBag
- Summary

Typed Views
- Introduction
- Typed Views
- Demo: Typed Views
- ViewModels
- Demo: Creating a ViewModel
- Demo: Using a ViewModel
- Scaffolding
- Demo: Adding a Scaffolded View
- Demo: Other Scaffolded Views
- Summary

View Engines
- Introduction
- View Engines
- Code Nuggets
- Demo: Code Nuggets
- HTML Encoding
- Demo: HTML Encoding
- Code Blocks
- Demo: Code Blocks
- Summary

Layouts
- Introduction
- Shared Layouts
- Demo: Shared Layouts
- Sections
- Demo: Sections
- Partial Views
- Demo: Partial Views
- Summary

What are Models
- Introduction
- What is a Model
- Demo: Models
- Demo: Model a Bookstore
- Demo: Model Class
- Demo: New Class
- Summary

Scaffolding
- Introduction
- Crud
- Scaffolding
- Demo: Scaffolding
- Demo: Run Application
- Demo: Add Controller
- Demo: Rerun Application
- Summary

Model Data
- Introduction
- Model Data
- Entity Framework
- Demo: Entity Framework
- Demo: Add Controller
- Demo: Data Context
- Demo: Run Application
- Demo: Adding More Controllers
- Demo: Build Application
- Summary