Angular 2, Part 1 of 3: Introduction and Components

page 1

Meet the expert: Gregor Dzierzon, MCSD, MCDBA has over 15 years of experience architecting and developing solutions using C#, VB.NET, SQL Server, Reporting Services, and Analysis Services, but most of his work has been with web development using ASP.NET, C#, JQuery, and AngularJS. He is a Senior Software Engineer and Software Architect who has owned a Software Consulting Company. He has 12 years of training experience and travels nationally as well as internationally to Canada and the UK to deliver training.

Prerequisites: This course assumes that students have some HTML and JavaScript programming background. As with any such course, the more experience you bring to the course, the more you’ll get out of it. This course moves quickly through a broad range of Angular2 topics, but it does not require any prior AngularJS skills since Angular2 has been completely re-written and is not simply an upgrade to AngularJS 1.x. The course also assumed that you are well familiar with how to use the operating system that you are running. For example, the course might say simply “Open Notepad” without explaining how to do that. You should also be able to navigate the folder hierarchy using Windows Explorer (Windows) or Finder (Mac) and with Command-Prompt (Windows) or Terminal (Mac).

Runtime: 02:14:29

Course description: In this course, software engineer and architect Gregor Dzierzon walks you down the path to Angular 2 software development. You will learn how to choose a code editor as well as the Angular approach to modular applications. You will also get an introduction to and learn the benefits of using the popular programming language TypeScript to build a simple application for Angular 2.

Course outline:

Prerequisites and IDE
- Introduction
- Pre-Requisites
- JavaScript Object Notation
- Setting up the Environment
- What is Node.js?
- Verify the Install
- What Is nmp?
- Text Editors
- IDE
- How to Choose an Editor?
- Demo: Installation
- Demo: Creating a Project
- Demo: Adding Components
- Summary

Angular MVC
- Introduction
- Angular2 Uses MV*
- What Is MVC
- Intro to MVC
- Multiple MVC Architecture
- Advanced Hotel Application
- How Does Angular2 Fit In?
- Angular2 Modules
- Demo: Simple Component
- Summary

TypeScript
- Introduction
- Angular2 and TypeScript
- Downsides to TypeScript
- Introduction to TypeScript
- Typed Function Parameters
- Creating Interfaces
- Creating Classes
- Transpiling TypeScript
- Using TypeScript with Angular2
- Demo: Using TypeScript
- Demo: Interface
- Demo: Class
- Demo: JSON
- Summary

Building Angular Components
- Introduction
- Angular2 Components
- Component = Controller
- Building Components
- @Component Decorator
- Component Code
- Component Life Cycle
- Bootstrapping
- Demo: Create Component
- Demo: Lifecycle Events
- Summary

Templates
- Introduction
- Angular2 Templates
- Template = View
- Define Custom Selector
- Hosting Templates
- Building Templates
- Adding Styles
- Demo: Templates
- Demo: Styling Templates
- Summary

Databinding
- Introduction
- Why Databinding?
- Data = Model
- Where Is the Model?
- Binding to the Model
- String Interpolation
- Creating a ViewModel
- Interpolation in HTML Tags
- Property/Attribute Binding
- Two-Way Binding
- Demo: Data Binding
- Demo: Event Binding
- Demo: Two-Way Binding
- Summary

Forms
- Introduction
- Building Forms with Angular
- Define User Input
- Design HTML Form
- Improve User Experience
- Add Bootstrap to the Project
- Demo: Create a Form
- Demo: Add Bootstrap
- Process User Input
- HTML 5 Validation with Angular
- Validate User Input
- Using ngControl for Validation
- Define CSS Styles
- Add Validation Messages
- Demo: Add Validation
- Demo: Validation Cont.
- Create Forms with FormBuilder
- Build the HTML Template
- Build Form in the Controller
- Add Logic to Submit Event
- Why Use FormBuilder?
- Demo: FormBuilder
- Demo: FormComponent
- Summary