

Advanced Angular 13 Programming (3 days)

By Vishwanath Rao

Chapter 1. Advanced Component Concepts

- Detecting Change to @Input Binding
- Example Child and Parent Components
- Using a Custom Setter
- Using ngOnChanges
- Advanced Inter Component Communication
- Direct Access to Child
- Using a Template Local Variable
- Using the @ViewChild Decorator
- Static and Dynamic Children with @ViewChild
- More About @ViewChild
- The @ViewChildren Decorator
- Live Monitoring of Children
- Direct Access to the Parent Component
- Communication Using Subject API
- Creating a Subject
- Publishing a Message in a Subject
- Subscribing to the Subject
- Problem With Ordering
- Content Projection
- Setup Projection Using ng-content
- Supplying Template for ng-content
- The Host Element
- Static Styling of the Host Element
- Setting DOM Properties of the Host
- Dynamically Loading a Component
- Dynamic Loading Example
- HostComponent Code
- Setting entryComponents
- Optimizing Change Detection
- Example Excessive Template Execution
- Using OnPush Change Detection Strategy
- Properly Changing @Input Variables
- Additional Notes on OnPush
- Summary

Chapter 2. Advanced RxJS

- Observable Creator Functions
- Subscribing to DOM Events

- The interval Function
- forkJoin and zip Function
- The concat Function
- The merge Function
- Recap of Operators
- The debounceTime Operator
- The distinct Operator
- The delay Operator
- The reduce Operator
- The mergeMap Operator
- The switchMap Operator
- The retryWhen Operator
- Basic Example of retryWhen
- Retry for a Finite Time
- Creating a Custom Operator
- Advanced Example Custom Operator
- Intercepting All HTTP Calls
- Example Interceptor Service
- Registering the Interceptor
- Summary

Chapter 3. Custom Attribute Directive

- Recapping Attribute Directives
- Ways to Manipulate the Host Element
- Binding to the DOM Properties of the Host
- Listening for Host Events
- Obtaining the DOM Event Object
- Accessing the Host Component
- Adding @Input to a Directive
- Summary

Chapter 4. Custom Structural Directive

- How Do Structural Directives Work?
- Commonly Used API
- A Very Simple Conditional Directive
- Using the Sample Directive
- Passing Context Data to the Template
- Example Looping Directive
- Using the Looping Directive
- Summary

Chapter 5. Custom Validators

- Validator for Reactive Forms
- Example Validator Function
- Using the Validator
- Passing Data to Validator Function

- Validator for Template Driven Forms
- Example Validator Directive
- Using the Validator Directive
- Asynchronous Validator
- Example Asynchronous Validator Function
- Using the Validation Function
- Asynchronous Validator Directive
- Optimizing Asynchronous Validation
- Summary

Chapter 6. Distributing Angular Libraries

- Introduction
- About Workspace and Project
- Creating a Library Project
- Anatomy of a Library Project
- Create a Component in the Library
- Create a Service in the Library
- Building the Library
- Distributing the Library
- Using the Library from an Application
- Using the Library Artifacts
- Writing a Sample Application
- Summary

Chapter 7. Internationalization (I18N)

- Introduction to I18N
- What Angular Provides
- Add @angular/localize Package
- Translatable Text
- Extract Translatable Text
- Example XLIFF File
- Providing Translation
- Configure the Build System
- Building the Application
- Deploying the Application
- Custom Translation Unit ID
- Getting the Current Locale
- Translatable Attribute Text
- Summary

Chapter 8. Security in Angular Applications

- Overview of Authentication and Authorization
- About Identity Token
- Exchanging the Identity Token
- Example Login Code in Angular Service
- Options for Saving the Identity Token

- Sending the Token to Server
- Obtaining User Information
- Cross Site Scripting (XSS) Attack
- Angular and XSS
- Sanitizing Content
- Trusting HTML Content
- Cross-Site Request Forgery (XSRF)
- Preventing XSRF Using a Token
- Summary