



# Introduction to Vue.js

Course ID #: 7000-472-ZZ-Z

Hours: 14

## Course Content

### Course Description:

This 2-day course is designed to get you started building small to intermediate sized front end applications for browsers. The course is structured to match the philosophy of Vue, that is, adopt Vue incrementally. You start by adding small functionalities to an existing application. Later you'll learn to build more ambitious Single Page Application (SPA). With SPA, Vue controls the entire frontend which is completely decoupled from the backend software.

### Prerequisites:

- Good knowledge of JavaScript.
- Knowledge of the usual web frontend technologies like HTML, CSS and AJAX.

### Target Student:

Developers responsible for building web-based frontend UI applications.

### Topics:

#### Chapter 1. Introduction to Vue.js

- What is Vue.js?
- What Makes Vue.js Unique?
- Progressive Adoption of Vue.js
- Using Vue.js
- Hello World
- The Vue Instance
- Data Binding Basics
- Binding to an Attribute
- Computed Property
- A Note About ES6 Object Literal
- Event Handling
- More About Events
- Introduction to Components
- Example Component
- Differences Between Vue Instance and Component
- Summary

#### Chapter 2. Template Syntax

- What is a Template?
- Showing Data
- Using Expressions
- Rendering HTML Text
- Setting Element Attributes
- Conditionally Showing Elements
- v-else-if and v-else
- v-show
- Rendering a List
- Using v-for
- Getting the Loop Index
- Identifying Each Item
- Event Handling Inside a Loop
- Grouping Sibling Elements
- Registering Event Handlers
- Event Modifiers
- Conditional CSS Styling
- Supply Class Array
- Inline Styles



# Introduction to Vue.js

Course ID #: 7000-472-ZZ-Z

Hours: 14

## Chapter 3. Working With Forms

- Two Way Data Binding
- Example Two Way Data Binding
- Checkbox
- Bind a Non-boolean Property
- Radio Button
- Radio Buttons from a Dynamic List
- Select Dropdown
- Binding an Object to Select
- Working With Numbers
- Handle Form Submission
- Handling Input Change Event
- Summary

## Chapter 4. Vue Components

- Introduction to Components
- Declaring a Component
- The Vue.component() Method
- Using a Component
- Component State
- The Component Tree
- Single Root Element
- Passing Data to a Child Component
- Supplying the Input
- Raising Events
- Example: Raising an Event
- Handling the Event from the Parent
- Obtaining Reference to a Child Component
- Summary

## Chapter 5. Lifecycle Hooks

- Lifecycle
- Common Lifecycle Hooks
- Example: Component Lifecycle Hooks
- Use the Component
- Example: Vue Instance Lifecycle
- Other Lifecycle Hooks
- Summary

## Chapter 6. Single File Component

- Shortcomings of Regular Components
- What is Single File Component (SFC)?
- Example Single File Component
- The .vue File Syntax
- Using a SFC
- Steps to Use a SFC
- Vue Command Line Interface (CLI)
- Installing Vue CLI
- Developing SFC
- Editor Extensions
- Summary

## Chapter 7. Single Page Application (SPA)

- What is a Single Page Application (SPA)
- Example SPA Workflow
- Traditional Web Application Capabilities
- Single Page Application Advantages
- SPA Challenges
- Overview of SPA Using Vue Router
- Create a New Application Project
- Anatomy of the Project
- Creating View Components
- Setting Up the Route Table
- Adding Links to Views
- Programmatic Navigation
- Accepting Path Parameters
- Supplying Path Parameters
- Test, Build and Deploy
- Summary

## Chapter 8. HTTP Client Development

- Ways to Write HTTP Client
- Using Axios in Single File Component
- Using Axios in a Conventional Component
- The axios Object Methods
- The Configuration Object
- Response Object Properties
- Making a POST Call
- Error Handling
- Summary



# Introduction to Vue.js

Course ID #: 7000-472-ZZ-Z

Hours: 14

## Lab Exercises

- Lab 1. Getting Started with Vue.js
- Lab 2. Basic Event Handling
- Lab 3. Basic Template Design
- Lab 4. Form Input Handling
- Lab 5. Basic Component Development
- Lab 6. Componentize an Application
- Lab 7. Single File Component
- Lab 8. Basic Single Page Application (SPA)
- Lab 9. HTTP Client Development
- Lab 10. Advanced Routing