

# React Using Typescript

By Dr. Vishwanath Rao

Day 1

Introduction to ES6

ES6 New Features

## Typescript Deep Dive-Type Script Architecture

- Basic types
- Variable declarations
- Interfaces
- Classes
- Functions
- Generics
- Enums
- Type inference
- Type compatibility
- Advanced types
- Symbols
- Iterators
- Modules
- Namespaces
- Namespaces vs modules
- Module resolution declaration merging
- Decorators

### • Introduction

- ♣ What is ReactJS?
- ♣ Who is using ReactJS?
- ♣ Concept of Virtual DOM
- ♣ Concept of React Component and Component Tree
- ♣ Unidirectional Data Flow
- • **Environment Setup**
- ♣ Running React directly in Browser (Client Side Processing)

Server Side Rendering using NodeJS, npm packages, babel & web-pack

## REACT ESSENTIAL FEATURES AND SYNTAX

- React App Project Directory Structure

- Overview of Webpack, Babel
- React Component Basic
- Create React Component
- Understanding TSX
- Limitations of TSX
- Working with Components and Reusing Components
- Helloworld app in React

## Multiple Components

- ♣ Nesting Components within components
- ♣ Breaking down wireframe into granular components

## Day 2

- **State Management**
  - ♣ Difference between UI State and Application State (Data)
  - ♣ Difference between Props & State
  - ♣ Unidirectional Data-Flow from Parent to Child
  - ♣ Defining State for a component
  - ♣ Passing Props (Data Properties) to Child Component from its Parent component
  - ♣ Passing Event Handler (of Parent) to Child Components to handle event raised by Child Component in Parent Component
  - ♣ Using the Props (Data Properties) in Child Component
  - ♣ Using the Props (Event Handlers) in Child Component
  - ♣ Concept of "Lifting State Up"
- **Conditional Logic**
  - ♣ Showing / Hiding Child Components based on condition
  - ♣ Adding / Removing Child Components based on condition
  - ♣ Repeating multiple Child Components
- **Styling Components**
  - ♣ Defining CSS ClassName attribute
  - ♣ Defining Style attribute and JavaScript Object
  - ♣ Applying conditional formatting using Style attribute

- ♣ Using Sass with React Components

- **Forms & Input Elements**

- ♣ Using the Ref callback attribute

- ♣ Uncontrolled Forms

- ♣ Controlled Forms

- **Component Lifecycle**

- ♣ Methods invoked while loading a component

- ♣ Methods invoked on re-rendering a component

Day 3

Methods invoked while unloading a component

- **Routing**

- ♣ Implementing the concept of SPA using React Router

- **HTTP Communication**

- ♣ Using Fetch to achieve HTTP based client-server communication

- ♣ Working with REST API

- **Application Architecture for React**

- ♣ Flux

- ♣ Essence of Flux

- ♣ Redux

- ♣ Overview of Three Principles of Redux

- ♣ Single Immutable State Tree

- ♣ Pure v/s Impure Functions

- ♣ Avoiding Object Mutation

- ♣ Describing State changes with Actions

- ♣ Reducers

- ♣ Stores Methods

- ♣ Redux v/s Flux

- **Testing & Quality**

- ♣ Working with jest and enzyme



## Advanced Redux Side-Effects

Redux Thunk vs. Redux Sagas

- ♣ JavaScript Generators
- ♣ Saga Helpers
- ♣ Declarative Effects
- ♣ Error Handling
- ♣ Sequencing Sagas
- ♣ Composing Sagas
- ♣ Testing Sagas
- **Reactive Programming with React/Redux**
- ♣ Overview of Reactive Programming
- ♣ Redux-Observable and Epics