

WEEKLY

­­

**WEEK 2**

|  |  |  |
| --- | --- | --- |
| Day | Learning Objective | Topics Covered |
| **Day 1** | JavaScript (core concepts) | * Data types * Scopes * Functional Programming * OOP * This * Array methods * Imperative vs Declarative style coding |
| JavaScript (advanced syntax & apis) | * New syntax * Arrow functions * Spread & Rest operators * Modules * Build tools * UI application |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **Day-2** | React.js (core concepts ) | * State & props (recap) * Component life-cycle methods |
| React.js (hands-on) | * Simple-online-exam user-interface with react.js using props & state |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Day-3** | React.js ( multiple components ) | * Inter-component communication * Context * When to use context |
| React.js (component common logic) | * Container components * Inheritance vs Composition * Higher-Order-progamming * Higher-Order-components |
|  |  |  |
| **Day-4** | React.js ( handling Forms) | * Controlled & uncontrolled form components * Validation * Dynamic Forms |
|  |  |  |
| **Day-5** | React.js ( routing ) | * SPA (Single Page Application) challenges * Understanding browser history api * React-router * Live-demo with routing features |