

React Course - From Zero to Pro (5 Years Experience Ready)

Week 1: React Basics + Core Concepts

Day 1: Intro to React + Setup

- What is React? Why React?
- SPA vs MPA
- Install Node.js + create app using:
npx create-react-app react-journey
cd react-journey
npm start
- Folder structure breakdown
- JSX Intro

Task: Create a component that prints your name and a button saying "Hello".

Day 2: JSX + Components

- JSX syntax (camelCase, expressions)
- Functional vs Class components
- Props introduction
- Reusable components

Task: Create a Card component that accepts name, role, image via props.

Day 3: State & Event Handling

- useState hook
- Button click handlers
- State change and re-render

Task: Counter App with Increment/Decrement and Reset.

Day 4: Conditional Rendering & Lists

- if/else, &&, ternary inside JSX
- Array.map() to render lists
- Key prop usage

Task: List of todos with "completed" status in green/red

Day 5: Forms + Controlled Components

- onChange, onSubmit
- Form state using useState
- Validation basics

Task: Login form with validation (no empty fields)

Day 6: useEffect + Lifecycle

- useEffect intro
- Side effects (API call, timer)
- Dependency array logic

Task: Show current time updating every second.

Day 7: Project Day + Git

- Mini Project: User Info Card (Form to input name, image, shows live preview)
- Git:
git init
git add .
git commit -m "Day 7: User Info Card App"

Week 2: Intermediate React + API + Routing

Day 8: Fetch API + Axios

- fetch() & axios.get()
- JSON parsing

- Display API data

Task: Fetch and display 5 GitHub users with their profile pic and link

Day 9: React Router v6

- Installing and setting up:
npm install react-router-dom
- Routes, Link, useNavigate
- Nested routes

Task: Create a 3-page app (Home, About, Contact)

Day 10: Lifting State Up + Props Drilling

- Passing data parent → child
- Lifting data child → parent
- Controlled components

Task: Create Parent component with two children that share state (e.g., like/dislike counter)

Day 11: useContext Hook

- Why useContext?
- Creating and consuming context

Task: Theme switcher (light/dark mode using context)

Day 12: Custom Hooks

- Why custom hooks?
- Create useFetch() and useTimer() as custom hooks

Task: Use useTimer() to show time passed since page load.

Day 13: Forms with Validation

- Reusable form components
- Regex validation
- Errors and messages

Task: SignUp form with validations for email/password

Day 14: Git + Mini Project

- Mini Project: Blog Reader (fetch posts, display, routing)
- Git branching:
git checkout -b blog-project

Week 3: Advanced Concepts + State Management

Day 15: useReducer Hook

- Why useReducer vs useState
- Actions, reducers, dispatch

Task: Build a basic todo app with add/remove functionality

Day 16: React + Local Storage

- Read/write to localStorage
- Persist data on reload

Task: Todo app with local storage

Day 17: React Portals & Fragments

- What are Portals?
- Modals using portal
- Fragments usage

Task: Build a modal popup component with backdrop

Day 18: Error Boundaries

- Catching errors in components
- Fallback UI

Task: Create a component that breaks and show fallback UI

Day 19: Performance Optimizations

- React.memo
- useCallback, useMemo

Task: Optimize a slow-rendering list with memoization

Day 20: Context + Reducer Combo

- Combine useContext + useReducer
- Global state handling

Task: Create Auth context (login/logout)

Day 21: Project Day + Git

- Project: Dashboard UI (Sidebar + routing + cards)
- Git tag:
git tag -a v1.0 -m "Dashboard project done"

Week 4: React + Integration + Deployment

Day 22: React + Backend (Node/Express Intro)

- Brief intro to REST API
- axios.post(), axios.delete()

Task: Connect form to backend endpoint

Day 23: Deploy React App

- Vite + React comparison
- Build:
npm run build
- Deployment options: Netlify, Vercel

Day 24–28: Final Project

Project: Expense Tracker with Login

- Pages: Login, Dashboard, Add Expense, History
- Features:
 - Protected Routes
 - CRUD
 - Local storage
 - Mobile responsive
- Git Setup:
git init
git commit -m "Start Expense Tracker"
git push origin main

Bonus Topics for Job-Ready:

- Testing with React Testing Library (npm install @testing-library/react)
- Unit vs Integration tests
- TypeScript in React
- Redux Toolkit
- Recoil / Zustand
- TailwindCSS

Project Ideas for Resume:

- Movie Search App (OMDb API)
- Ecommerce Cart App
- Chat App (Socket.io)
- Weather App
- Portfolio site in React