React + TypeScript Roadmap (Beginner → Advanced)

Estimated Duration: 10–14 Weeks (2–3 Months)

PHASE 1 — Fundamentals (Week 1–2)

- What is React? SPA? Virtual DOM?
- JSX
- Functional components
- Props & State
- useState, useEffect
- Rendering & events
- Controlled components

Project: Todo App / Notes App

PHASE 2 — Real-World React (Week 3)

- Fetching data
- Axios vs Fetch
- Loading & Error states
- Lifting state up
- Component composition
- useContext

Project: User Management CRUD

PHASE 3 — TypeScript Essentials (Week 4)

- Typing props
- Typing state
- Typing events
- Typing API responses
- Interfaces vs types
- Generics in React
- Utility types (Partial, Pick, Record, Readonly)

Project: Convert CRUD to TypeScript

PHASE 4 — Intermediate React (Week 5–6)

- Routing (React Router)

- useCallback, useMemo, memo
- useRef
- Custom hooks
- Error boundaries
- File uploads

Project: Dashboard with Protected Routes

PHASE 5 — State Management (Week 7)

- Zustand (recommended)
- Redux Toolkit (interview friendly)
- Global state concepts

Project: Cart / Favorites with global store

PHASE 6 — Advanced Patterns (Week 8–9)

- React 18 concurrency
- Suspense & lazy loading
- Code splitting
- Pagination & infinite scroll
- React Hook Form
- Modal patterns
- Compound components

Project: Ticketing or Booking UI

PHASE 7 — React + Backend Integration (Week 10–11)

- Axios interceptors
- Refresh tokens
- Error handling middleware
- Upload images & PDFs
- Role-based UI

Project: Full-stack Ticketing App

PHASE 8 — Deployment (Week 12–13)

- Vercel / Netlify
- Production builds

- Environment variables
- Basic CI/CD

Project: Deploy your full project

Timing Summary:

- 2 hours per day → 12–14 weeks
- 3–4 hours per day → 8–10 weeks