Guide2Code - React.js Roadmap ⚛️

🟢 Phase 1: Beginner Level

📚 Topics to Learn:

1. Introduction to React (What is React? SPA Concept, Advantages)
2. Setting Up the Environment (Node.js, npm, create-react-app, Vite)
3. JSX & Components (Functional vs Class Components)
4. Props & State (Passing Data, useState Hook)
5. Event Handling in React (onClick, onChange, Handling Forms)
6. Lists & Keys (Rendering Lists, Unique Keys)
7. Conditional Rendering (if-else, ternary operator, && operator)
8. Basic Styling in React (CSS, Styled Components, Tailwind CSS)
9. React Router Basics (

4o

You said:

remove horizontal lines

ChatGPT said:

🔥 Guide2Code - React.js Roadmap ⚛️

🟢 Phase 1: Beginner Level

📚 Topics to Learn:

1. Introduction to React (What is React? SPA Concept, Advantages)
2. Setting Up the Environment (Node.js, npm, create-react-app, Vite)
3. JSX & Components (Functional vs Class Components)
4. Props & State (Passing Data, useState Hook)
5. Event Handling in React (onClick, onChange, Handling Forms)
6. Lists & Keys (Rendering Lists, Unique Keys)
7. Conditional Rendering (if-else, ternary operator, && operator)
8. Basic Styling in React (CSS, Styled Components, Tailwind CSS)
9. React Router Basics (Routing, Link, useNavigate)

10. Simple API Calls (fetch, axios, useEffect for API requests)

🛠 Beginner Project Ideas:

- **To-Do List App** – Add, edit, and delete tasks
- **Weather App** – Fetch weather data using an API
- **Calculator App** – Perform basic arithmetic operations
- **Quote Generator** – Display random quotes using an API
- **Movie Search App** – Search for movies using an API

🟡 Phase 2: Intermediate Level

📚 Topics to Learn:

1. React Hooks (useEffect, useRef, useMemo, useCallback)
2. Context API & useReducer (Global State Management)
3. React Router Advanced (Dynamic Routes, Nested Routing, Private Routes)
4. Forms & Validations (React Hook Form, Yup Validation)
5. Component Lifecycle & Error Boundaries (Handling Errors in Components)
6. State Management with Redux Toolkit (Slices, Store, Dispatch, Selectors)
7. Authentication & Authorization (JWT, OAuth, Firebase Authentication)
8. Custom Hooks (Creating Reusable Hooks)
9. Performance Optimization (Lazy Loading, Code Splitting, Memoization)
10. Testing in React (Jest, React Testing Library, Cypress)

🛠 Intermediate Project Ideas:

- **E-commerce Product Page** – Display and filter products
- **Blog Website** – Fetch and display articles dynamically
- **Expense Tracker** – Track monthly expenses and incomes
- **Chat Application** – Real-time chat using Firebase
- **Portfolio Website** – Showcase projects and skills

🔴 Phase 3: Advanced Level

📚 Topics to Learn:

1. Server-Side Rendering (SSR) with Next.js (getServerSideProps, getStaticProps)

2. Progressive Web Apps (PWAs) in React (Service Workers, Manifest)
3. Advanced State Management (Recoil, Zustand, Jotai)
4. GraphQL with React (Apollo Client, Queries, Mutations)
5. WebSockets & Real-Time Apps (Socket.io Integration)
6. Micro Frontends with React (Module Federation, Single-SPA)
7. Animations in React (Framer Motion, GSAP)
8. React Native for Mobile Apps (Basic React Native Concepts)
9. Deploying React Apps (Vercel, Netlify, Docker)
10. Building a Full-Stack MERN App (MongoDB, Express, React, Node.js)

🛠 **Advanced Project Ideas:**

- **Social Media Dashboard** – Show real-time stats from multiple platforms
- **AI-Powered Chatbot** – Integrate OpenAI API for chat responses
- **Stock Market Dashboard** – Display real-time stock prices and trends
- **Task Management App** – Kanban board with drag & drop features
- **SaaS Dashboard** – Admin panel with user roles and analytics