✅ ReactJS vs NextJS – Key Differences :  
  
🔹 Type:  
ReactJS: UI Library  
NextJS: Full-stack Framework built on React  
  
🔹 Routing:  
ReactJS: Manual setup using react-router-dom  
NextJS: File-based routing out of the box (pages/ folder)  
  
🔹 Rendering Methods:  
ReactJS: Client-side rendering (CSR) by default; SSR needs manual setup  
NextJS: Supports CSR, SSR (Server-side), SSG (Static), ISR (Incremental)  
  
🔹 Backend/API Support:  
ReactJS: Needs a separate backend (e.g., Node/Express)  
NextJS: Built-in API routes (pages/api/)  
  
🔹 SEO (Search Engine Optimization):  
ReactJS: Not SEO-friendly by default  
NextJS: Great SEO with server-rendered pages  
  
🔹 Performance:  
ReactJS: Depends on how you optimize it  
NextJS: Optimized automatically (code-splitting, caching, images)  
  
🔹 Deployment:  
ReactJS: Deployed via services like Netlify, Vercel, etc.  
NextJS: Seamless integration with Vercel (the team behind Next.js)  
  
🔹 Learning Curve:  
ReactJS: Beginner-friendly for UI development  
NextJS: Slightly steeper, but powerful for full-stack apps  
  
🔹 Ideal Use Cases:  
ReactJS: SPAs, widgets, dashboards  
NextJS: Full-stack apps, SEO-friendly websites, blogs, eCommerce