

ReactJS/NextJS - Important Topics

1. ReactJS

Basics

1. **JSX Syntax**
2. **React Components (Function vs. Class)**
3. **Props and State**
4. **Event Handling**
5. **Conditional Rendering**
6. **Lists and Keys**
7. **Basic Forms Handling**
8. **Lifting State Up**
9. **Component Composition**

Intermediate

1. Hooks

- `useState`
- `useEffect`
- `useRef`
- `useMemo`
- `useCallback`
- `useContext`

2. Custom Hooks

3. Context API for Global State

4. React Router (v6+)

5. Forms with Libraries (e.g., Formik, React Hook Form)

6. Code Splitting and Lazy Loading

7. Error Boundaries

8. Portal API

9. Refs and DOM Manipulation

10. Controlled vs Uncontrolled Components

Advanced

1. State Management Libraries

- Redux (with Toolkit)
- Zustand
- Recoil
- Jotai

2. Server-Side Rendering (SSR) and Static Site Generation (SSG)

- Using **Next.js**

3. Performance Optimization

- Memoization
- Avoiding unnecessary re-renders
- React Profiler

4. Concurrent Features (React 18+)

- `useTransition`
- `useDeferredValue`
- Suspense for data fetching

5. React Query / TanStack Query

6. Testing

- Jest + React Testing Library
- E2E (e.g., Cypress, Playwright)

7. Animations

- Framer Motion
- React Spring

8. Accessibility (a11y)

9. Internationalization (i18n)

Expert-Level / Architecture

1. Clean Architecture and Component Design Patterns

- Container/Presentational
- Compound Components
- Render Props
- Higher-Order Components

2. Custom DevTools and Debuggers

Tooling & Ecosystem

- Vite / CRA / Next.js / Remix
- ESLint + Prettier setup
- Storybook for Component Documentation
- Bundlers: Webpack, Rollup, Parcel
- CI/CD with GitHub Actions or similar
- Using TypeScript with React (optional)

Reference

✓ Official Documentation :: <https://react.dev/learn>

2. NextJS

Basics

1. What is Next.js & Why Use It
2. File-based Routing (App Router)
 - `app/` directory structure
 - Layouts (`layout.tsx`)
 - Pages (`page.tsx`)
 - Templates (`template.tsx`)
3. Linking Between Pages (`next/link`)
4. Navigation (`useRouter`, `usePathname`)
5. Static Assets (`public/`)
6. CSS & Styling
 - CSS Modules
 - Tailwind CSS integration

Intermediate

1. Data Fetching

- `fetch` in Server Components
- `useEffect` in Client Components
- `generateStaticParams`, `generateMetadata`

2. Server Components vs Client Components

- When and why to use `'use client'`

3. Dynamic Routes & Catch-All Routes

4. API Routes (in `app/api/`)

5. Middleware

- Authentication, redirects, etc.

6. Metadata API (for SEO)

7. Image Optimization with `next/image`

8. Layouts and Nested Routing

9. Custom Fonts (`next/font`)

Advanced

1. Server Actions (New in Next.js 15)
 - Form handling with server-side logic
 - Mutations in Server Components
2. Edge & Serverless Functions
 - Edge runtime via middleware or API routes
3. Streaming with Suspense
4. Internationalization (i18n)
5. Auth Handling
 - NextAuth.js
 - Middleware-based token checks
6. Error Handling
 - `error.tsx`, `not-found.tsx`
7. Prefetching & Performance Tuning
8. Loading UI with `loading.tsx`
9. Caching & Revalidation (`fetch` cache options)
10. Dynamic Metadata (e.g. title per product)

Tooling / Ecosystem

- ESLint + Prettier + TypeScript setup
- Tailwind CSS with DaisyUI or shadcn/ui
- Storybook for components
- Vercel Analytics / Monitoring
- Using Environment Variables

Reference

✓ Official Documentation : <https://nextjs.org/docs>

🔍 App Router Overview: <https://nextjs.org/learn/dashboard-app>
